

# Optimierungsprobleme in der mathematischen Modellierung

Grundlegende Aspekte und Chancen aus Sicht der  
Mathematikdidaktik – herausgestellt an aktuellen Problemen aus  
der Forschung zu künstlicher Intelligenz und erneuerbaren  
Energien

Zur Erlangung des akademischen Grades eines

## DOKTORS DER NATURWISSENSCHAFTEN

von der KIT-Fakultät für Mathematik des  
Karlsruher Instituts für Technologie (KIT)  
genehmigte

## DISSERTATION

von

Sarah Schönbrodt, M. Ed.

---

Tag der mündlichen Prüfung:	23.02.2022
Erstgutachter:	Prof. Dr. Martin Frank
Zweitgutachter:	Prof. Dr. Oliver Stein
Drittgutachter:	Univ.-Prof. Mag. Dr. Hans Humenberger



Ich versichere wahrheitsgemäß, die Arbeit selbstständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde.

.....  
Sarah Schönbrodt  
Heidelberg, 13.01.2022



# Danke!

Zuallererst möchte ich meinem Betreuer Martin Frank danken, der mich immerzu unterstützt, motiviert und mir mit fachlichem Rat zur Seite gestanden hat. Mit Martin zusammenzuarbeiten und über Ideen zu diskutieren, die die Bildung im Bereich der mathematischen Modellierung vorantreiben, empfand ich als äußerst inspirierend. Martin hat mich stets darin bestärkt eigene Ideen und Wege zu verfolgen und wusste mir mit den richtigen Fragen und Denkanstößen in Phasen Orientierung zu bieten, in denen ich mal im Dunkeln tappte. Danke!

Zudem möchte ich Prof. Hans Humenberger und Prof. Oliver Stein für ihre zahlreichen, konstruktiven Hinweise zu dieser Arbeit danken. Der fachliche bzw. fachdidaktische Rat war überaus hilfreich und hat wesentlich dazu beigetragen die Qualität dieser Arbeit zu erhöhen.

Ein ganz besonderer Dank gilt meinen jetzigen und ehemaligen Kolleginnen und Kollegen, die mich sowohl in meiner Zeit in Karlsruhe als auch zu Beginn in Aachen durch fachlichen Austausch und anregende Diskussionen unterstützt haben. Die Promotion ist sicherlich nicht immer ein geradliniger Weg. Gerade in Phasen, in denen es steinig wurde, waren aufmunternde Worte und erfahrener Rat Gold wert. Besonders möchte ich Thomas Camminady, Jonas Kusch und Jannick Wolters für fachlichen Rat und Unterstützung bei programmiertechnischen Fragen danken. Auch danke ich Thomas Camminady, Isabell Helbing, Stephanie Hofmann, Jonas Kusch, Marcel Marnitz, Tobias Roglic, Kirsten Wohak und Maike Wolff für anregende Gespräche, Rückmeldungen zum entwickelten Lernmaterial oder das Korrekturlesen meiner Arbeit. Danke auch an Pascal Richter für seine ständige Bereitschaft als Experte im Bereich der Solarenergieforschung mit mir zu diskutieren. Auch Mark Schmitz möchte ich für den Austausch rund um die Solarthermie herzlich danken. Zudem danke ich den abgeordneten Lehrkräften des Teams von Simulierte Welten für ihr didaktisches Feedback zum entwickelten Lernmaterial und für die Möglichkeiten dieses in ihrem Unterricht zu erproben. Ein weiterer Dank gilt dem gesamten CAMMP Team. Es hat mir stets viel Freude bereitet in einem motivierten Team mitzuarbeiten, das begeistert daran arbeitet jungen Menschen authentische Erfahrungen im mathematischen Modellieren zu ermöglichen.

Ganz besonders möchte ich meiner Familie danken, die seitdem ich denken kann für mich da ist und mich auch in diesem Lebensabschnitt stets unterstützt und bestärkt hat. Zu guter Letzt danke ich meinem Freund Martin – für seinen Rat, für Austausch und fachliche Diskussionen, für fortwährende Unterstützung und Rückhalt.



# Vorwort

Viele Anwendungen und Technologien aus unserem Alltag wären ohne Mathematik nicht denkbar. Sei es die Bestimmung unserer aktuellen Position auf der Erde (basierend auf GPS), die Verwendung der Suchmaschine von Google oder das Entsperren des Smartphones per Gesichtserkennung. Die enorme Relevanz von Mathematik für Technik, Alltag und Gesellschaft erfahrbar zu machen ist eine zentrale Aufgabe des Mathematikunterrichts. So forderte bereits [Winter \(1995, S. 17\)](#), dass der Mathematikunterricht Lernenden ermöglichen sollte „Erscheinungen der Welt um uns, die uns alle angehen oder angehen sollten, aus Natur, Gesellschaft und Kultur, in einer spezifischen Art wahrzunehmen und zu verstehen.“

Eine Möglichkeit die Bedeutung von Mathematik für unsere Gesellschaft aktiv erfahrbar zu machen ist der Einbezug mathematischer Modellierungsaktivitäten in den Mathematikunterricht. Modellieren, was in Kürze als Prozess des Hin- und Herwechsels zwischen Mathematik und Realität beschrieben werden kann, ist mittlerweile auch in den Bildungsstandards der Kultusministerkonferenz fest verankert. Dort wird das mathematische Modellieren als eine von sechs allgemeinen mathematischen Kompetenzen, die Lernende im Mathematikunterricht erwerben sollen, ausgewiesen (vgl. [Kultusministerkonferenz, 2003](#); [Kultusministerkonferenz, 2012](#)).

Die Einbindung authentischer Problemstellungen in den Mathematikunterricht und das Lösen dieser mithilfe (computergestützter) mathematischer Modellierung hat unter anderem das Potential Interesse und Motivation am Mathematiktreiben zu fördern, ein ganzheitliches Bild von Mathematik zu vermitteln und die Bedeutung von Mathematik zu erleben (vgl. [Greefrath et al., 2013, S. 20](#)).

Ogleich zahlreiche Gründe für mehr Gelegenheiten zum Modellieren im Mathematikunterricht sprechen, ist der Anteil, den das Modellieren im Unterricht einnimmt, oft als gering anzusehen. Unter anderem die COACTIV-Studie, in deren Rahmen ca. 45000 Aufgaben untersucht wurden, die im Schuljahr 2003/2004 in der 9. und 10. Klasse im Mathematikunterricht gestellt wurden, unterstreicht, dass nur wenige Aufgaben tatsächlich außer- oder innermathematische Bezüge im Sinne des Modellierens herstellen (vgl. [Jordan et al., 2008, S. 103](#)).

[Blum \(2007\)](#) diskutiert in seinem Beitrag *Mathematisches Modellieren – zu schwer für Schüler und Lehrer?* verschiedene Gründe für die Lücke zwischen Wunsch und Wirklichkeit in Bezug auf das Modellieren im Mathematikunterricht:

„Weshalb ist die Situation im Unterricht so? Das liegt wesentlich daran, dass Modellieren auch für Lehrer schwer ist, der Unterricht wird hierdurch komplexer und weniger vorhersagbar, und es ist außermathematisches Sachwissen nötig (Freudenthal 1973, Pollak 1979, Blum 1996, Burkhardt 2004)“ ([Blum, 2007, S. 6](#)).

Wenn Modellieren so schwierig ist und insbesondere Wissen aus anderen Disziplinen notwendig ist, warum sollte es dann überhaupt im Mathematikunterricht gemacht werden? Diese Frage wurde auch von [Blum et al. \(2007, S. 4–5\)](#) aufgegriffen:

„At various times in the history of mathematics teaching and learning, it has been debated whether some forms of mathematical applications (or, in more recent times, modelling) should have a place in different sorts of mathematics curricula, or whether the extra-mathematical utilisation of mathematics should be the responsibility of those subjects that utilise the applications and the modelling. [...] The very fact that there are, from time to time, such debates about the possible place and role of applications and modelling in the teaching and learning of mathematics suggests that there are issues to consider and think about.“

In ihrem Beitrag basieren [Blum et al. \(2007\)](#) weiterführende Diskussionen rund um Anwendungen und Modellierung auf der Annahme, dass das Modellieren einen Platz in der mathematischen Ausbildung haben sollte. Wir bleiben hier jedoch bei der vorgelagerten Frage, *ob* das Modellieren einen Platz in den Curricula des Fachs Mathematik haben sollte. Würde es nicht vollkommen ausreichen, den Ingenieurinnen und Ingenieuren, den Physikerinnen und Physikern oder den Informatikerinnen und Informatikern das Modellieren zu überlassen? Letztlich den Personen, in deren Anwendungsfeld das Problem auftritt? Könnten Anwendungen und Modellierung nicht genauso gut (oder vielleicht noch besser) einen Platz in anderen Disziplinen und Schulfächern einnehmen? Oder sind gegebenenfalls doch **fundierte innermathematische Überlegungen im Hinblick auf das Wechselspiel zwischen mathematischen Methoden und Anwendungsfragen sinnvoll oder gar notwendig**? In dieser Arbeit werden mögliche Antworten auf diese Fragen anhand einer zentralen Problemklasse, den **Optimierungsproblemen**, geliefert. Probleme im Bereich des Optimierens, das als eine fundamentale Idee der Mathematik bezeichnet wird, sind höchst divers und facettenreich (vgl. [Vogel, 2010a, S. 6](#); [Humenberger, 2015b, S. 2](#)). Zum einen durchdringen sie verschiedenste Bereiche der Mathematik (sowohl der diskreten als auch der kontinuierlichen) und greifen zahlreiche elementar-mathematische Grundlagen und Konzepte auf. Zum anderen verbergen sich solche Probleme hinter diversen Anwendungen und Technologien.

### **Ziele dieser Arbeit**

Mit dieser Arbeit beziehungsweise der ihr zugrundeliegenden Forschung werden drei Hauptziele verfolgt:

Das **erste, fachliche Ziel** bezieht sich auf das Herausarbeiten des mathematischen Verständnisses und der mathematischen Begriffe, die notwendig sind, um Optimierungsprobleme zu bearbeiten. Dazu werden verschiedene reale Probleme aus unterschiedlichen Blickrichtungen (diskret, kontinuierlich, Einbezug von Unsicherheiten etc.) angegangen. Dabei wird insbesondere herausgestellt, welche innermathematischen Überlegungen und Modellentscheidungen zentral sind, um Optimierungsprobleme so zu formulieren, dass sich mathematisch wohlgestellte Probleme ergeben. Das heißt solche Probleme, bei denen Existenz und Eindeutigkeit eines Optimalpunktes gegeben ist und die auch in der Praxis lösbar sind. Losgelöst von einzelnen

---

Fallbeispielen wird diskutiert, wie sich mathematische Methoden und Anwendungsfragen wechselseitig beeinflussen (können). Etwas informeller gesprochen soll der Frage nachgegangen werden, inwieweit es Mathematikerinnen und Mathematiker braucht, um aus Anwendungsfragen resultierende Optimierungsprobleme geeignet zu lösen. Dies nämlich ließe sich als weiteres Argument dafür ansehen, das Modellieren im Mathematikunterricht zu verankern – und nicht ausschließlich in anderen Fächern und Disziplinen.

Das **zweite Hauptziel** dieser Arbeit ist es, aus den Erkenntnissen des mathematisch-fachlichen Teils didaktische Konsequenzen für das Modellieren mit Schülerinnen und Schülern abzuleiten. Konkret wurde innovatives, digitales Lernmaterial für Lernende der Mittel- und Oberstufe zu realen Problemstellungen, die auf Optimierungsfragen hinauslaufen, entwickelt und erprobt. Das Lernmaterial verdeutlicht exemplarisch, wie die facettenreiche Diskussion von Optimierungsproblemen im Rahmen mathematischer Modellierungsaktivitäten zu authentischen Fragestellungen realisiert werden kann. Die Entwicklung und Erprobung des Lernmaterials fand entlang von vier aufeinander aufbauenden Schritten statt: Im *ersten Schritt* wurden zwei ausgewählte Problemstellungen mathematisch modelliert und wesentliche Aspekte beim Formulieren und Lösen der auftretenden Optimierungsprobleme herausgearbeitet, die im *zweiten Schritt* didaktisch reduziert wurden. Basierend auf den Erkenntnissen der ersten beiden Schritte wurde im *dritten Schritt* Lehr- und Lernmaterial entwickelt. Der *vierte Schritt* stellt schließlich die Erprobung des Materials dar. Dazu wurde dieses im Rahmen verschiedener Modellierungsveranstaltungen des Computational and Mathematical Modeling Programs (CAMMP),<sup>1</sup> mit Standorten am Karlsruher Institut für Technologie (KIT) und an der RWTH Aachen, eingesetzt und evaluiert. Die vorliegende Arbeit ist in der Stoffdidaktik zu verorten. Die Evaluationen, die im Rahmen von Fragebögen für Schülerinnen und Schüler sowie basierend auf teilnehmenden Beobachtungen durchgeführt wurden, sollten allgemeine Tendenzen und Erfahrungswerte zusammentragen. Auf Grundlage dieser wurden die Materialien iterativ überarbeitet und erneut erprobt.

Für die Entwicklung des Unterrichtsmaterials wurden Problemstellungen aus zwei Kontexten gewählt, die zum einen höchst relevant für Gesellschaft, Technik und Forschung sind, die aber insbesondere im Alltag vieler Schülerinnen und Schüler eine bedeutende Rolle spielen. Das erste Problem stammt aus dem Bereich der erneuerbaren Energien, genauer der Solarenergieforschung. Es geht um die Frage, wie ein solarthermisches Kraftwerk mit planaren Fresnelspiegeln optimal gebaut werden kann. Diese Fragestellung ist auch aus fachlich-methodischer Sicht in der aktuellen Forschung vertreten. Dort werden neue Methoden entwickelt, um die Effizienz der Kraftwerke zu erhöhen.

Auch der zweite Kontext ist in Forschung und Technik äußerst präsent und in den Medien vielfach vertreten: künstliche Intelligenz (KI) beziehungsweise das maschinelle Lernen. Von Gesichtserkennung über personalisierte Werbung, von automatischer Krankheitsdiagnostik bis hin zum autonomen Fahren findet KI in mehr und mehr Bereichen Anwendung. Konkret geht es bei der zweiten Problemstellung um die Frage, wie Nutzerinnen und Nutzern von Streaming-Plattformen individuell zugeschnittene

---

<sup>1</sup>Für mehr Informationen siehe Kapitel 3 oder [www.cammp.online](http://www.cammp.online), letzter Aufruf: 13.11.2021.

Empfehlungen für neue Filme gegeben werden können. Doch was verbirgt sich dahinter? Auf welchen mathematischen Konzepten basieren maschinelle Lernmethoden? Es lässt sich an dieser Stelle bereits vorwegnehmen, dass einer der Grundpfeiler zahlreicher maschineller Lernmethoden die mathematische Optimierung ist. Dem Formulieren von Optimierungsproblemen und dem Entwickeln effizienter Algorithmen kommt damit auch in der Forschung zu KI eine essentielle Rolle zu (vgl. [Bottou et al., 2018](#), S. 225; [Sra et al., 2011](#), S. 1). Die Optimierung schlägt im Rahmen dieser Dissertation somit die Brücke zwischen den gewählten Problemstellungen aus den Bereichen erneuerbare Energien und KI.

Das **dritte Hauptziel** ist es, Lehrkräften ein ganzheitliches Bild verschiedener Ansätze beim Formulieren und Lösen von Optimierungsproblemen und möglicher didaktischer Reduktionen zu vermitteln. Dazu wird ein kompakter Überblick zu grundlegenden Überlegungen beim Formulieren und Lösen von Optimierungsproblemen in der Anwendung geliefert. Diese Arbeit soll Lehrkräften Orientierung bieten, wenn Modellierungsprojekte zu realen Problemen, die auf Optimierungsprobleme hinauslaufen, möglichst unangeleitet von Lernenden bearbeitet werden. Denkbare Lösungsansätze (auch seitens der Lernenden) und zentrale innermathematische Überlegungen werden in dieser Arbeit zusammengestellt.

### **Zielgruppen und Aufbau**

Diese Arbeit richtet sich in erster Linie an *Lehramtsstudierende, Referendarinnen und Referendare* und *Mathematiklehrkräfte*, die Modellierungsprojekte zu realen Problemstellungen in ihren Unterricht integrieren möchten. Dazu erhalten (angehende) Lehrkräfte mit dieser Arbeit nicht nur das notwendige Hintergrundwissen zu ausgewählten, authentischen Problemstellungen, sondern darüber hinaus konkrete, exemplarisch erprobte Unterrichtsmaterialien an die Hand. Da nicht davon auszugehen ist, dass (angehende) Lehrkräfte über Grundvorlesungen zur Analysis und Numerik hinausgehende Fachvorlesungen zur Optimierung besucht haben, wird ein Überblick über wesentliche Eigenschaften, Unterscheidungen und Formulierungen von Optimierungsproblemen gegeben.

Diese Arbeit richtet sich zudem an *Mathematikdidaktikerinnen und -didaktiker* sowie didaktisch interessierte *Mathematikerinnen und Mathematiker*, die authentische Anwendungen in ihre Hochschullehre einbringen und Studierenden Erfahrungen im Modellieren realer Probleme (mit Fokus auf der Optimierung) ermöglichen wollen. Diese Arbeit liefert dazu zahlreiche Anknüpfungspunkte und ein Konzept, wie Modellierungsveranstaltungen kollaborativ und online durchgeführt werden können.

Die vorliegende Arbeit ist in fünf Teile gegliedert:

- Teil **I** dieser Arbeit liefert das notwendige fachliche Hintergrundwissen zur Modellierung und Optimierung. Dazu wird in Kapitel 1 die aktuelle fachdidaktische Diskussion um mathematisches Modellieren im Mathematikunterricht aufgegriffen. Es wird beschrieben, was Modellierung ist, wie sie betrieben werden kann, welche Ziele mit dem Modellieren im Mathematikunterricht verfolgt werden und welche Rolle dem Computer beziehungsweise digitalen Werkzeugen dabei zukommt. Auch werden Kriterien für *gute* Modellierungsprobleme

---

benannt. Anschließend wird in Kapitel 2 die gesamte mathematische Problemklasse in den Blick genommen: Optimierungsprobleme. Es wird dargestellt, welche Arten von Optimierungsproblemen es gibt und wie diese angegangen werden (können). Dabei wird insbesondere herausgearbeitet, welches Wissen notwendig ist, um Optimierungsprobleme so zu formulieren, dass sie möglichst *gewünschte* Eigenschaften haben. Es wird diskutiert, was – im Hinblick auf Anwendungsfragen – überhaupt gewünschte Eigenschaften derartiger Probleme sind. Zudem werden exemplarische Strategien zum Lösen der Probleme beschrieben. Abschließend wird herausgestellt, inwieweit die mathematische Optimierung bereits im Mathematikunterricht verankert ist und eine Übersicht über bestehende fachdidaktische Literatur zur Optimierung gegeben.

- Ein Teilziel dieser Arbeit war die Entwicklung von innovativem, digitalem Lernmaterial. In Teil II werden die Ziele, die mit dem Material verfolgt werden, beschrieben. Zudem werden dessen Aufbau und Einsatzmöglichkeiten sowie didaktisch-methodische Entscheidungen erläutert. Der organisatorische Rahmen, unter dem das Lernmaterial erprobt wurde, wird vorgestellt. Zudem werden die digitalen Werkzeuge beschrieben, die zur Realisierung des Lernmaterials wie auch für die unterrichtspraktische Erprobung in kollaborativen (online durchgeführten) Modellierungsprojekten eingesetzt wurden.
- Im Fokus von Teil III steht die Optimierung verschiedener Parameter eines solarthermischen Kraftwerks. Beim Formulieren und Lösen der Optimierungsprobleme wird die gesamte Bandbreite an denkbaren Formulierungen ausgeschöpft. Es wird detailliert aufgeschlüsselt, welche unterschiedlichen Ansätze beim Formulieren von Optimierungsproblemen auch seitens der Lernenden denkbar sind. Zudem werden Vor- und Nachteile der Formulierungen diskutiert. Anschließend werden Ziele, Aufbau und Inhalte des entwickelten Lernmaterials sowie Erfahrungen aus den Erprobungen mit Lernenden beschrieben.
- Teil IV liefert zunächst einen Einblick in das maschinelle Lernen und Chancen den Mathematikunterricht durch aktuelle maschinelle Lernmethoden zu bereichern. Anschließend wird die Modellierung eines Empfehlungssystems für Filme beschrieben. Es wird diskutiert, welche Optimierungsprobleme auftreten und wie diese mit Lernenden bearbeitet werden können. Die Optimierungsprobleme werden dabei weniger kleinschrittig diskutiert als bei der Modellierung und Optimierung des Solarkraftwerks. Abschließend wird das entwickelte Lernmaterial für stärker angeleitete Modellierungsprojekte und die Erfahrungen aus diversen Erprobungen mit Schülerinnen und Schülern beschrieben.
- Basierend auf den Erfahrungen aus den Modellierungsprojekten und den Erkenntnissen aus der Modellierung realer Problemstellungen, fasst Teil V zusammen, inwieweit und an welchen Stellen sich das Optimieren für die Gestaltung möglichst offener, unangeleiteter Modellierungsprojekte eignet. Dabei wird einmal mehr aufgegriffen, welche Überlegungen im Hinblick auf die wechselseitige Beeinflussung von Anwendungsfragen und innermathematischen Entscheidungen beim Formulieren und Lösen von Optimierungsproblemen relevant sind.

Diese Dissertation basiert zum Teil auf Arbeiten, die bereits veröffentlicht oder zur Veröffentlichung angenommen wurden:

In dem Beitrag [Frank et al. \(im Druck\)](#) zu computergestützter mathematischer Modellierung habe ich insbesondere zu den Inhalten rund um Kriterien für *gute* Problemstellungen und den Erfahrungen aus Schülerworkshops beigetragen. Auszüge daraus finden sich in Abschnitt 1.3 dieser Arbeit wieder.

Das ganzheitliche Konzept für die online Durchführung von kollaborativen mathematischen Modellierungsprojekten, welches in Teil II dieser Arbeit beschrieben wird, wurde in [Schönbrodt et al. \(im Druck\)](#) zur Veröffentlichung angenommen. Bis auf die Diskussion der Erfahrungswerte und den Vergleich der beschriebenen Modellierungsprojekte mit Freys Projektmethode, die von Kirsten Wohak erarbeitet wurde, wurde diese Veröffentlichung eigenständig von mir verfasst. Zudem finden sich einige Beschreibungen der Elemente des digitalen Lernmaterials sowie didaktisch-methodische Begründungen für deren Einsatz und Umsetzung in [Gerhard et al. \(im Druck\)](#) wieder. Dort habe ich in besonderem Maße bei der Diskussion der einzelnen Elemente des Lernmaterials und der damit verbundenen Ziele für die Gestaltung mathematischer Modellierungsaktivitäten mitgewirkt.

Zur Modellierung eines Solarkraftwerks wurde von [Krahforst \(2016\)](#) bereits Lernmaterial entwickelt und mit Schülerinnen und Schülern erprobt. Das bestehende Lernmaterial von [Krahforst \(2016\)](#) befasste sich ausschließlich mit der Modellierung der Spiegelausrichtung und der durch das Kraftwerk erzielten Leistung. Auch [Hattebuhr et al. \(2015\)](#), [Roeckerath \(2012\)](#) und [Frank et al. \(2017\)](#) haben bereits in verschiedenen Veröffentlichungen diskutiert, inwieweit sich die Modellierung von Solarkraftwerken für authentische Modellierungsprojekte eignet. Die von den benannten Autorinnen und Autoren beschriebene Modellierung der Leistung und des Energieumsatzes eines solarthermischen Kraftwerks werden zum besseren Verständnis in Teil III dieser Arbeit aufgegriffen und entsprechend gekennzeichnet. Bei der Entwicklung des Unterrichtsmaterials wurde das Material von [Krahforst \(2016\)](#) aufgegriffen, adaptiert, mit anderen digitalen Werkzeugen umgesetzt und um das in dieser Arbeit zentrale Lernmaterial zur Optimierung erweitert. Teile der Beschreibung des Lernmaterials und des inner- und außermathematischen Hintergrunds zur Modellierung und Optimierung eines solarthermischen Kraftwerks erscheinen in [Schönbrodt \(im Druck\)](#). Eine kurze Zusammenfassung des Lernmaterials wurde zudem in [Schönbrodt und Frank \(2020\)](#) veröffentlicht.

Ein Teil der Beschreibung des mathematischen Modells für Empfehlungssysteme in Teil IV dieser Arbeit sowie denkbare didaktische Reduktionen und Umsetzungen im Lernmaterial wurden bereits in [Schönbrodt et al. \(2021\)](#) sowie [Schönbrodt und Frank \(2021\)](#) veröffentlicht und wurden eigenständig von mir verfasst.

Im Lernmaterial im Anhang und an verschiedenen Stellen in dieser Arbeit wurden Icons der Fontawesome-Schriftart<sup>2</sup> von Dave Gandy und Xavier Danaux verwendet. Diese sind unter SIL Open Font License (OFL) verfügbar.

---

<sup>2</sup>Weitere Informationen unter: <https://fontawesome.com>, letzter Aufruf: 23.12.2021.

---

### Zugriff auf das entwickelte Lernmaterial

Um den Leserinnen und Lesern einen ersten Eindruck davon zu vermitteln, wie das entwickelte digitale Lernmaterial aussieht, wird empfohlen bei der Lektüre der Teile **III** und **IV** einen Blick auf das Material zu werfen. Dieses befindet sich auf einer online zugänglichen Workshop-Plattform.<sup>3</sup> Der Zugriff wird in einem kurzen Video unter [www.cammp.online/214.php](http://www.cammp.online/214.php) beschrieben. Das Passwort, welches benötigt wird, um die Lösungsversionen des Lernmaterials zu erhalten, lautet: **CAMMP=MatheFun!**

---

<sup>3</sup>Die Registrierung auf der Workshop-Plattform erfolgt über <https://jupyterhub.cammp.online/hub/login>, letzter Aufruf: 05.01.2022.



# Inhaltsverzeichnis

<b>Vorwort</b>	<b>v</b>
<b>Abbildungsverzeichnis</b>	<b>xix</b>
<b>Tabellenverzeichnis</b>	<b>xxiii</b>
<b>I Fundamentales zu Modellierung und Optimierung</b>	<b>1</b>
<b>1 Mathematische Modellierung in der Schule</b>	<b>3</b>
1.1 Was ist mathematische Modellierung? . . . . .	3
1.2 Warum mathematische Modellierung im Mathematikunterricht? . . .	5
1.3 Kriterien für gute Modellierungsprobleme . . . . .	8
1.4 Mathematisches Modellieren mit digitalen Werkzeugen . . . . .	9
<b>2 Optimierung</b>	<b>12</b>
2.1 Grundlagen und zentrale Begriffe . . . . .	12
2.1.1 Existenz und Eindeutigkeit globaler Minimalpunkte . . . . .	15
2.1.2 Welche Arten von Optimierungsproblemen gibt es? . . . . .	21
2.2 Wie werden Optimierungsprobleme gelöst? . . . . .	24
2.2.1 Abstiegsverfahren . . . . .	26
2.2.2 Lösen von restringierten Optimierungsproblemen . . . . .	27
2.3 Warum Optimierung in der Schule? . . . . .	28
<b>II Realisierung von kollaborativen Modellierungsprojekten online</b>	<b>33</b>
<b>3 Einleitung</b>	<b>35</b>
<b>4 Die Unterrichtsformate</b>	<b>38</b>
4.1 Modellierungstage – Projekte mit kurzem Zeitrahmen . . . . .	38
4.2 Modellierungswochen – Projekte mit großzügigem Zeitrahmen . . . . .	40
4.3 Herausforderungen für die Online-Durchführung . . . . .	43
<b>5 Eingesetzte digitale Werkzeuge</b>	<b>45</b>
5.1 Auswahl einer Kommunikationsplattform . . . . .	45
5.2 Jupyter Notebook . . . . .	47
5.3 Bereitstellung des digitalen Lernmaterials . . . . .	48

<b>6</b>	<b>Ziele, Aufbau und Elemente des digitalen Lehr- und Lernmaterials</b>	<b>50</b>
6.1	Übergeordnete Zielsetzungen und didaktisch-methodische Entscheidungen . . . . .	50
6.2	Elemente des Lehr- und Lernmaterials für angeleitetes Modellieren . .	51
6.2.1	Präsentationsfolien . . . . .	51
6.2.2	Arbeitsblätter . . . . .	52
6.2.3	Erklärvideos . . . . .	54
6.2.4	Gestufte Hilfen . . . . .	54
6.2.5	Vertiefendes Zusatz- und Informationsmaterial . . . . .	54
6.2.6	Antwort- und Dokumentationsblätter . . . . .	55
6.3	Elemente des Lernmaterials für unangeleitetes Modellieren . . . . .	55
<b>7</b>	<b>Beispielhafter Ablauf eines Online-Modellierungstages</b>	<b>57</b>
<b>III</b>	<b>Unterrichtsmaterial zur Optimierung eines solarthermischen Kraftwerks</b>	<b>61</b>
<b>8</b>	<b>Einleitung</b>	<b>63</b>
<b>9</b>	<b>Basiswissen zu solarthermischen Kraftwerken</b>	<b>64</b>
9.1	Die Sonnenstrahlung – Datengrundlage . . . . .	65
<b>10</b>	<b>Modellierung eines Fresnelkraftwerks</b>	<b>69</b>
10.1	Ausrichtung der Spiegel . . . . .	71
10.2	Modellierung von Leistung und umgesetzter Energie . . . . .	71
10.2.1	Leistung am Absorberrohr . . . . .	71
10.2.2	Energieumsatz eines Tages . . . . .	74
10.2.3	Energie am Rohr erzeugt durch mehrere Spiegel . . . . .	75
10.3	Modellierung von Effekten zwischen Spiegeln . . . . .	75
<b>11</b>	<b>Optimierung verschiedener Kraftwerkparameter</b>	<b>80</b>
11.1	Optimierung der Absorberrohrhöhe . . . . .	80
11.1.1	Optimierung mit einfachem Modell . . . . .	81
11.1.2	Modellerweiterung: atmosphärische Abschwächung . . . . .	82
11.1.3	Modellerweiterung: Berücksichtigung von Unsicherheiten . . .	84
11.1.4	Kombination beider Modellerweiterungen . . . . .	90
11.1.5	Monte-Carlo-Methode . . . . .	90
11.2	Optimierung des Layouts eines Spiegelfeldes . . . . .	92
11.2.1	Ein Blick in die Forschung . . . . .	93
11.2.2	Kontinuierliche Formulierungen . . . . .	95
11.2.3	Diskrete Formulierungen . . . . .	102
11.2.4	Evaluation der Ergebnisse . . . . .	106
11.3	Zwischenfazit und Konsequenzen für die Modellierung mit Lernenden	107
<b>12</b>	<b>Umsetzung im Lernmaterial</b>	<b>108</b>
12.1	Voraussetzungen und Ziele . . . . .	108

12.2	Darstellung des Lernmaterials . . . . .	112
12.2.1	Workshopübergreifendes Lernmaterial . . . . .	112
12.2.2	Lernmaterial zum Workshop I . . . . .	112
12.2.3	Lernmaterial zum Workshop II . . . . .	118
12.2.4	Lernmaterial zum unangeleiteten Modellieren . . . . .	124
12.3	Flexibler Einsatz des Lernmaterials . . . . .	127
12.4	Lehrplananbindung . . . . .	127
<b>13</b>	<b>Erfahrungswerte und didaktische Reflexion</b>	<b>133</b>
13.1	Durchgeführte Modellierungsveranstaltungen . . . . .	133
13.2	Evaluation . . . . .	133
13.3	Didaktische Reflexion . . . . .	144
13.4	Ausblick . . . . .	147
<b>IV</b>	<b>Unterrichtsmaterial zu Empfehlungssystemen und dem Net-</b>	
	<b>flix Preis</b>	<b>149</b>
<b>14</b>	<b>Einleitung</b>	<b>151</b>
14.1	Data Science, maschinelles Lernen und Optimierung . . . . .	152
<b>15</b>	<b>Basiswissen zu Empfehlungssystemen</b>	<b>158</b>
15.1	Arten von Empfehlungssystemen . . . . .	158
15.1.1	Methoden des kollaborativen Filterns . . . . .	159
15.1.1.1	Arten von Bewertungsdaten . . . . .	160
15.1.1.2	Spärlichkeit der Daten und das Kaltstartproblem . . . . .	161
15.1.2	Weitere Methoden für Empfehlungssysteme . . . . .	161
<b>16</b>	<b>Entwicklung eines Empfehlungssystems</b>	<b>164</b>
16.1	Der Netflix Preis und der zugrundeliegende Datensatz . . . . .	164
16.2	Matrix-Faktorisierungen als Modell zur Vorhersage von Bewertungen	165
16.2.1	Zusammenhänge in den Daten . . . . .	167
16.2.2	Wahl einer Fehlerfunktion . . . . .	169
16.2.3	Gewichtete Niedrigrang-Matrixapproximation . . . . .	170
16.2.4	Alternierende Methode der kleinsten Fehlerquadrate . . . . .	171
16.2.5	Modellerweiterung: Regularisierung . . . . .	175
16.2.6	Anwendung auf Testdaten . . . . .	179
16.2.7	Modellerweiterung: Verzerrungen in den Daten . . . . .	180
16.3	Nachbarschaftsmethoden – Ähnlichkeiten modellieren . . . . .	181
16.4	Zwischenfazit und Konsequenzen für die Modellierung mit Lernenden	182
<b>17</b>	<b>Umsetzung im Lernmaterial</b>	<b>184</b>
17.1	Voraussetzungen und Ziele . . . . .	184
17.2	Darstellung des Lernmaterials . . . . .	186
17.2.1	Lernmaterial für angeleitete Modellierungsprojekte . . . . .	186
17.2.2	Lernmaterial für Modellierungswochen . . . . .	193
17.3	Lehrplananbindung . . . . .	194

<b>18 Erfahrungswerte und didaktische Reflexion</b>	<b>198</b>
18.1 Durchgeführte Modellierungsveranstaltungen . . . . .	198
18.2 Evaluation . . . . .	198
18.2.1 Ergebnisse der angeleiteten Modellierungsprojekte . . . . .	199
18.2.2 Ergebnisse der Modellierungswoche . . . . .	207
18.3 Didaktische Reflexion . . . . .	209
18.4 Ausblick . . . . .	211
<b>V Abschließende Kommentare</b>	<b>215</b>
<b>19 Optimierungsprobleme in Modellierungsprojekten – wenig bis nicht angeleitet?</b>	<b>217</b>
19.1 Individuelle Modellentscheidungen rund um Optimierungsprobleme .	218
19.2 Optimierungsprobleme im Rahmen mathematischer Modellierungs- wochen . . . . .	224
19.3 Leitfragen beim Formulieren und Lösen von Optimierungsproblemen .	227
<b>20 Zusammenfassung und Ausblick</b>	<b>230</b>
<b>Literaturverzeichnis</b>	<b>235</b>
<b>Anhang</b>	
<b>A Ergänzungen zum Basiswissen rund um Solarkraftwerke</b>	<b>251</b>
A.1 Modellierung von Verschattungs- und Blockierungseffekten . . . . .	251
<b>B Lernmaterial zum Thema Solarenergie – Workshop I</b>	<b>253</b>
B.1 Ablauf des Workshops . . . . .	253
B.2 Arbeitsblätter zum Modell für Leistung und Energie . . . . .	253
B.2.1 Arbeitsblatt 1: Ausrichtung eines Spiegels . . . . .	255
B.2.2 Arbeitsblatt 2: Leistung am Spiegel . . . . .	257
B.2.3 Arbeitsblatt 3: Leistung am Rohr . . . . .	259
B.2.4 Arbeitsblatt 4: Horizontal verschobene Spiegel . . . . .	261
B.2.5 Arbeitsblatt 4 - Kurzversion: Horizontal verschobene Spiegel .	264
B.2.6 Zusatzblatt 1: mehrere Spiegel . . . . .	265
B.2.7 Zusatzblatt 2: Energie über einen Tag . . . . .	266
B.3 Arbeitsblätter zur Optimierung der Rohrhöhe . . . . .	267
B.3.1 Arbeitsblatt 5: Optimale Rohrhöhe . . . . .	267
B.3.2 Arbeitsblatt 6: Modellverbesserung 1 – atmosphärische Effekte	270
B.3.3 Arbeitsblatt 7: Modellverbesserung 2 – Fehlerberücksichtigung	272
B.3.4 Arbeitsblatt 8: Kombination von Modellverbesserung 1 & 2 . .	276
B.3.5 Zusatzblatt: Optimierung mit verbessertem Modell . . . . .	277
B.3.6 Infoblatt: Optimierung unter variablem Fehler . . . . .	279
B.4 Antwort- und Dokumentationsblätter . . . . .	280
B.4.1 Dokumentationsblatt zu Arbeitsblättern 1 – 4 . . . . .	280
B.4.2 Dokumentationsblatt zu Arbeitsblättern 6 – 7 . . . . .	281

B.4.3	Antwortblätter zu Arbeitsblättern 5 – 8 . . . . .	281
B.5	Präsentationsfolien . . . . .	283
B.5.1	Folien für den Einstiegsvortrag . . . . .	283
B.5.2	Folien für die Zwischen- und Abschlussdiskussionen . . . . .	285
<b>C</b>	<b>Lernmaterial zum Thema Solarenergie – Workshop II</b>	<b>287</b>
C.1	Ablauf des Workshops . . . . .	287
C.2	Arbeitsblatt 1: Ausrichtung eines Spiegels . . . . .	289
C.3	Arbeitsblatt 2: Energie über einen Tag . . . . .	292
C.4	Arbeitsblatt 3: Optimierung der Spiegelpositionen . . . . .	294
C.5	Übersichtsblatt zu Algorithmen . . . . .	297
C.6	Infoblatt zur Optimierung . . . . .	298
C.7	Arbeitsblatt zur Brute-Force-Methode . . . . .	299
C.8	Arbeitsblatt zum Greedy-Algorithmus . . . . .	301
C.9	Arbeitsblatt zur musterbasierten Optimierung . . . . .	303
C.10	Arbeitsblatt zur freien Variablen Optimierung . . . . .	305
C.11	Arbeitsblatt 5: Verschattungseffekte . . . . .	307
C.12	Arbeitsblatt 6: Optimierung unter Verschattungseffekten . . . . .	312
C.13	Antwortblätter . . . . .	313
C.13.1	Antwortblatt zu Arbeitsblatt 3 . . . . .	313
C.13.2	Antwortblatt zu Arbeitsblatt 4 . . . . .	314
C.14	Präsentationsfolien . . . . .	315
C.14.1	Folien für die Zwischen- und Abschlussdiskussionen . . . . .	315
<b>D</b>	<b>Arbeitsblätter zu offenen Modellierungsproblemen – Solarenergie</b>	<b>321</b>
D.1	Übersichtsblatt . . . . .	321
D.2	Problem 1: Optimierungsverfahren implementieren . . . . .	323
D.3	Problem 2: Verschattung modellieren . . . . .	324
D.4	Problem 3: Beliebige Kraftwerkparameter optimieren . . . . .	326
<b>E</b>	<b>Ergänzungen zum Basiswissen rund um Empfehlungssysteme</b>	<b>327</b>
E.1	Das stochastische Gradientenverfahren . . . . .	327
E.2	ALS- und SGD-Verfahren für die Optimierung mit Bias-Termen . . . . .	328
E.3	Numerische Experimente . . . . .	330
E.3.1	Auswahl eines Teildatensatzes . . . . .	331
E.3.2	Ergebnisse basierend auf simplen Vorhersageregeln . . . . .	331
E.3.3	Ergebnisse mit dem ALS-Verfahren . . . . .	332
E.3.4	Ergebnisse unter Berücksichtigung von Bias-Termen . . . . .	333
E.3.5	Abschließende Kommentare . . . . .	335
<b>F</b>	<b>Lernmaterial zum Thema Empfehlungssysteme</b>	<b>337</b>
F.1	Ablauf des Workshops . . . . .	337
F.2	Arbeitsblatt 1: Datenanalyse . . . . .	339
F.3	Arbeitsblatt 2: Entwicklung eines Modells . . . . .	341
F.4	Arbeitsblatt 3: Die Fehlerfunktion . . . . .	345
F.5	Arbeitsblatt 4: Das Optimierungsverfahren . . . . .	348
F.5.1	Kurzversion . . . . .	348

F.5.2	Langversion . . . . .	351
F.6	Arbeitsblatt 5: Anwendung auf den Netflixdatensatz . . . . .	356
F.7	Zusatzmaterial . . . . .	359
F.7.1	Zusatzblatt: Skalarprodukt . . . . .	359
F.7.2	Zusatzblatt: Die Wahl eines Fehlermaßes . . . . .	360
F.7.3	Zusatzblatt: Regularisierung . . . . .	362
F.7.4	Zusatzblatt: Eigene Empfehlungen . . . . .	363
F.8	Antwortblätter . . . . .	364
F.9	Präsentationsfolien . . . . .	369
F.9.1	Folien für den Einstiegsvortrag . . . . .	369
F.9.2	Folien für die Zwischen- und Abschlussdiskussionen . . . . .	370
F.10	Material für Modellierungswochen . . . . .	377
F.10.1	Problembeschreibung . . . . .	377
F.10.2	Jupyter Notebooks . . . . .	378
<b>G</b>	<b>Evaluationen der Durchführungen</b>	<b>381</b>
G.1	Evaluationsergebnisse zum Thema Solarenergie . . . . .	381
G.2	Evaluationsergebnisse zum Thema Empfehlungssysteme . . . . .	393
G.2.1	Angleitete Modellierungsprojekte . . . . .	393
G.2.2	Modellierungswoche . . . . .	401
<b>H</b>	<b>Offene Problemstellungen aus Modellierungswochen</b>	<b>407</b>
H.1	Optimale Energieversorgung basierend auf drei erneuerbaren Energie- quellen . . . . .	407
H.2	Parkplatzwahl auf Automobilterminals . . . . .	408
H.3	Optimierung eines Windparks . . . . .	409
H.4	Optimierung eines Fresnelkraftwerks . . . . .	409

# Abbildungsverzeichnis

1.1	Modellierungskreislauf angelehnt an Blum (1985) . . . . .	3
1.2	Computergestützte Modellierungsspirale adaptiert von der <i>Solution Helix of Math</i> der Initiative Computer-Based Math (o. D.) . . . . .	4
1.3	Siebenschrittiger Modellierungskreislauf von Blum und Leiß (2005) entnommen aus Greefrath et al. (2013, S. 18) . . . . .	5
2.1	Lokale und globale Minimalpunkte . . . . .	14
2.2	Kompakte und nichtkompakte Niveaumengen . . . . .	16
2.3	Konvexe und nichtkonvexe Funktionen . . . . .	18
4.1	Beispielhaftes Programm einer Modellierungswoche . . . . .	41
5.1	Die Kommunikationsplattform Mattermost . . . . .	46
5.2	Screenshot eines digitalen Arbeitsblatts (Jupyter Notebook) aus dem Lernmaterial zu Solarkraftwerken . . . . .	48
6.1	Screenshot der Rückmeldung auf einem digitalen Arbeitsblatt aus dem Lernmaterial zu Solarkraftwerken. Eine Lösung der Aufgabe wurde bereits eingetragen. . . . .	53
8.1	Fresnelkraftwerk Puerto Erado 2 in Südspanien, Quelle: EBL . . . . .	63
9.1	Schematischer Aufbau eines Fresnelkraftwerks . . . . .	64
9.2	Schematische Reflexion von Sonnenstrahlen in einem Fresnelkraftwerk dargestellt im Querschnitt . . . . .	66
9.3	Approximierte Bestrahlungsstärke der direkten Sonnenstrahlung am 21. Juni in Daggett . . . . .	67
9.4	Azimut $\gamma_s$ und Altitude $\alpha_s$ . . . . .	67
9.5	Beschreibung der einfallenden Sonnenstrahlen über den transversalen Sonneneinfallswinkel $\alpha_t$ . . . . .	68
10.1	Ein um $x$ horizontal verschobener Spiegel der Länge $s$ . . . . .	69
10.2	Visualisierung des Kosinuseffekts am Beispiel des Schattenwurfs. Links fallen die Sonnenstrahlen senkrecht auf den Schirm, rechts treffen die Sonnenstrahlen schräg auf . . . . .	72
10.3	Darstellung der Hilfsstrecke mit Länge $e$ , die senkrecht auf den Sonnenstrahlen steht . . . . .	72
10.4	Kosinuseffekt am Sekundärreflektor . . . . .	73
10.5	Leistung am Rohr im Tagesverlauf generiert durch einen Spiegel unter dem Rohr ( $b = 0.8$ m und $s = 1.1$ m) . . . . .	74

10.6	Skizze des Schattenwurfs für zwei Spiegel auf den Positionen $x_1$ und $x_2$	76
10.7	Energie in Abhängigkeit der Position $x_1$ eines Spiegels. Zwei weitere Spiegel sind fest auf den Positionen $x_2 = 1$ m und $x_3 = 4$ m positioniert ( $b = 0.8$ m, $h = 6$ m $s = 1.1$ m, $\Delta t = 0.05$ h).	79
11.1	Leistung in Abhängigkeit von der Höhe des Absorberrohrs ( $t = 11$ h, $b = 0.8$ m, $s = 1.1$ m, $x = -2$ m)	81
11.2	Leistung in Abhängigkeit von der Höhe des Absorberrohrs mit (gestrichelte Linie) und ohne (durchgezogene Linie) atmosphärischer Abschwächung ( $t = 11$ h, $b = 0.8$ m, $s = 1.1$ m, $x = -2$ m)	83
11.3	Skizze der Reflexion bei fehlerhafter Spiegeleinstellung	85
11.4	Länge des bestrahlten Bereichs des Sekundärreflektors $l$ bei fehlerhafter Spiegeleinstellung	86
11.5	Sechs Fälle, die unter Fehlereinfluss am Sekundärreflektor auftreten können	86
11.6	Leistung am Rohr in Abhängigkeit von der Rohrhöhe mit (gestrichelte Linie) und ohne (durchgezogene Linie) Berücksichtigung von Einstellungsfehlern ( $\epsilon = 0.5^\circ$ , $t = 11$ h, $b = 0.8$ m, $s = 1.1$ m, $x = -2$ m)	87
11.7	Monte-Carlo-Simulation der erwarteten Leistung am Rohr für unterschiedliche Werte für $n_{\text{error}}$ mit $t = 11$ h, $b = 0.8$ m, $s = 1.1$ m, $\mathbf{x} = (-5.5, -4, -2.5, -1, 1, 2.5, 4, 5.5)$	92
11.8	Energieumsatz gemäß dem Modell ohne Verschattungseffekte ( $E_{\text{tube}}$ ) und gemäß dem Modell mit Verschattungseffekten ( $E_{\text{tubeEffects}}$ ) in Abhängigkeit von der Position $x$ eines Spiegels ( $b = 0.8$ m, $h = 6$ m, $s = 1.1$ m, $\Delta t = 0.05$ h)	95
11.9	Energieumsatz gemäß dem Modell ohne Verschattungseffekte ( $E_{\text{tube}}$ ) in Abhängigkeit von den Positionen $x_1$ und $x_2$ zweier Spiegel	96
11.10	Energieumsatz gemäß dem Modell mit Verschattungseffekten ( $E_{\text{tubeEffects}}$ ) in Abhängigkeit von den Positionen $x_1$ und $x_2$ zweier Spiegel	97
11.11	Diskrete Betrachtung des Layoutoptimierungsproblems	103
12.1	Übersicht über die Inhalte des Workshops für die Mittelstufe (Workshop I) und für die Oberstufe (Workshop II)	109
12.2	Näherungsweise Berechnung der umgesetzten Energie im Zeitraum von $t_1 = 11.0$ h bis $t_2 = 11.5$ h	119
13.1	Digitales Whiteboard (bereitgestellt von awwapp), das von einer Lerngruppe während eines Modellierungstages gestaltet wurde	140
13.2	Seitenansicht eines Solarturmkraftwerks in Jülich, Quelle: DLR	148
13.3	Luftaufnahme eines Solarturmkraftwerks in Jülich, Quelle: DLR	148
14.1	CRISP-DM Modell angelehnt an Chapman et al. (2000, S. 13)	153
16.1	Darstellung von drei User- und drei Movievektoren im Eigenschaftsraum. Eigenschaften könnten beispielsweise Genres wie Action oder Comedy sein.	168
16.2	Darstellung der Zielfunktion (16.13). Es existiert kein eindeutiger globaler Minimalpunkt.	173

16.3	Darstellung der Zielfunktion (16.15) basierend auf der Wahl der Summe der absoluten Abweichungen . . . . .	173
16.4	Darstellung der Zielfunktion (16.13) erweitert um einen Regularisierungsterm $f(u_{11}, u_{12}) + \lambda \cdot (u_{11}^2 + u_{12}^2)$ mit $\lambda = 2$ . Es existiert ein eindeutiger globaler Minimalpunkt. . . . .	177
16.5	Fehler bezüglich der Trainingsdaten für unterschiedliche Werte von $\lambda$ . Die Anzahl der Eigenschaften beträgt in allen Fällen $n_f = 2$ . . . . .	178
16.6	Ergebnisse basierend auf dem regularisierten Problem (16.21) für verschiedene Wahlen der Parameter $n_f$ und $\lambda$ . . . . .	179
17.1	Interaktive Visualisierung der Daten auf dem ersten Arbeitsblatt (Streudiagramm) . . . . .	187
17.2	Tabellarische Darstellung der Filme aus dem Netflixdatensatz. Die Lernenden können die Tabelle auf dem Arbeitsblatt nach beliebigen Spalten sortieren. . . . .	188
17.3	Unterteilung der bekannten Daten in Trainings- und Testdaten . . . . .	192
18.1	Gradientenverfahren visualisiert im zwei- und dreidimensionalen Fall . . . . .	212
19.1	Arithmetisches Mittel und Median als Minimalpunkt(e) von Optimierungsproblemen. Die Summe der orientierten Abstände eignet sich nicht als Fehlermaß. . . . .	219
19.2	Skizze verschiedener Ansätze für die Anpassung von Messdaten durch Ausgleichsgeraden . . . . .	220
19.3	Quadratische Zielfunktion eines Optimierungsproblems unter der Nebenbedingung $ x - 2  \geq 2$ . Die zulässige Menge ist nicht konvex. . . . .	223
A.1	Skizze zur Berechnung der Projektion auf Rohrhöhe $p$ . . . . .	252
B.1	Übersicht über Inhalte und Arbeitsblätter des Workshops für die Mittelstufe . . . . .	254
C.1	Übersicht über Inhalte und Arbeitsblätter des Workshops für die Oberstufe . . . . .	288
E.1	Fehler bezüglich der Trainingsdaten für unterschiedliche Werte von $\lambda$ . In allen Fällen beträgt die Anzahl der Eigenschaften $n_f = 2$ und die Lernrate $\alpha = 0.01$ . . . . .	328
E.2	Histogramm der Bewertungen im Trainings- und Testdatensatz . . . . .	330
E.4	Ergebnisse basierend auf dem ungewichteten, regularisierten Problem (16.21) (R-ALS) für verschiedene Wahlen der Parameter $n_f$ und $\lambda$ . Die Ergebnisse werden ab der ersten Iteration (d. h. nach dem ersten Update von $U$ und $M$ ) dargestellt. . . . .	333
E.5	Ergebnisse basierend auf dem gewichteten regularisierten Problem (16.24) (WR-ALS) für verschiedene Wahlen der Parameter $n_f$ und $\lambda$ . Zum Vergleich wird das Ergebnis des ungewichteten regularisierten Problems (R-ALS) für ein Szenario gezeigt. . . . .	334

E.6	Ergebnisse basierend auf dem regularisierten Problem ohne Bias-Terme (16.24) (R-ALS) und dem Problem mit Bias-Termen (16.30) (R-ALS-Bias) für verschiedene Wahlen der Parameter $n_f$ und $\lambda$ . Zum Vergleich wird das Ergebnis des gewichteten regularisierten Problems mit Bias-Termen (WR-ALS-Bias) für ein Szenario gezeigt. . . . .	335
E.7	Fehler bezüglich der Test- und Trainingsdaten für $\alpha = 0.001$ und $\lambda = 0.1$ mit dem SGD-Verfahren . . . . .	336
F.1	Übersicht über Inhalte und Arbeitsblätter des Workshops zu Empfehlungssystemen . . . . .	338

# Tabellenverzeichnis

9.1	Übersicht über vorliegende Daten zur Bestrahlungsstärke der direkten Sonnenstrahlung in Daggett basierend auf einem MRM-Modell . . . . .	67
10.1	Variablen und Funktionen bei der Modellierung des Fresnelkraftwerks	70
11.1	Ergebnisse der freien Variablen Optimierung . . . . .	100
11.2	Ergebnisse der musterbasierten Optimierung . . . . .	102
11.3	Ergebnisse der Optimierung mit einer Brute-Force-Methode . . . . .	105
11.4	Ergebnisse der Optimierung mit einem Greedy-Algorithmus . . . . .	106
13.1	Durchgeführte Modellierungsveranstaltungen zum Thema Solarenergie	134
15.1	Klassifizierung von Empfehlungssystemen angelehnt an Aggarwal (2016, S. 16) . . . . .	163
16.1	Variablen und Operatoren bei der Modellierung eines Empfehlungssystems . . . . .	166
17.1	Übersicht über das Lernmaterial zu Empfehlungssystemen . . . . .	185
18.1	Durchgeführte Modellierungsveranstaltungen zum Thema Empfehlungssysteme . . . . .	199
20.1	Durchgeführte Modellierungsprojekte im Rahmen von Lehramtsseminaren. Die mit * markierte Veranstaltung hat in Präsenz stattgefunden. Thema Solar I $\cong$ Optimierung der Rohrhöhe, Thema Solar II $\cong$ Optimierung der Spiegelpositionen . . . . .	232
E.1	RMSE bezüglich der Trainings- und Testdaten für verschiedene simple Vorhersageregeln . . . . .	332
E.2	RMSE nach 10 Iterationen mit dem ALS-Verfahren für das regulierte ungewichtete Problem (16.21) (R-ALS) . . . . .	334
E.3	RMSE nach 10 Iterationen mit dem ALS-Verfahren für das regulierte gewichtete Problem (16.24) (WR-ALS) . . . . .	334



# **Teil I**

## **Fundamentales zu Modellierung und Optimierung**



# 1 Mathematische Modellierung in der Schule

In diesem Kapitel wird diskutiert, was unter mathematischer Modellierung verstanden wird, wie diese betrieben werden kann und welche Rolle sie im Mathematikunterricht der Schule einnimmt beziehungsweise einnehmen sollte. Es werden Kriterien für gute Modellierungsprobleme benannt und die Bedeutung digitaler Werkzeuge für das Modellieren beschrieben.

## 1.1 Was ist mathematische Modellierung?

Der Begriff *Modellieren* wurde insbesondere von Pollak (1979) geprägt, der das Modellieren als einen Teilaspekt der angewandten Mathematik hervorhebt und dabei das (wiederholte) Hin- und Herwechseln zwischen der Mathematik und dem „Rest der Welt außerhalb der Mathematik“ in den Fokus rückt (Pollak, 1979, S. 255 f., zitiert nach Greefrath et al., 2013, S. 11).

Der Prozess des Modellierens wird vielfach idealisiert in Form eines Kreislaufs dargestellt. Einer der wohl bekanntesten ist der vierschrittige Modellierungskreislauf von Blum (1985) (vgl. Abb. 1.1). Ausgangspunkt der Modellierung ist stets eine reale Problemstellung. Da reale Probleme oft viel zu komplex sind, um sie direkt mathematisch zu beschreiben, werden sie meist *strukturiert*, *vereinfacht* und *idealisiert*. Zentral ist dabei, dass das Problem verstanden wurde. Die Vereinfachungen und Strukturierungen liefern ein vereinfachtes Problem,<sup>1</sup> welches dann in *mathematischen Ausdrücken* beschrieben werden kann. Ein mathematisches Modell wird entwickelt. Anschließend werden *mathematische Methoden angewandt*, um zu

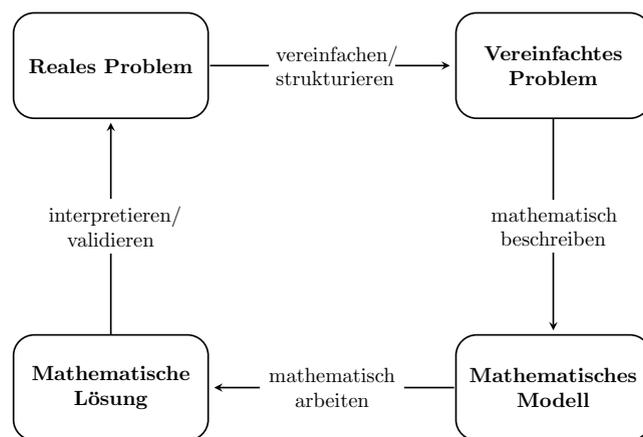


Abb. 1.1: Modellierungskreislauf angelehnt an Blum (1985)

mathematischen Ausdrücken beschrieben werden kann. Ein mathematisches Modell wird entwickelt. Anschließend werden *mathematische Methoden angewandt*, um zu

<sup>1</sup>Blum (1985, S. 200) nutzt anstelle des Begriffs *vereinfachtes Problem* die Bezeichnung *reales Modell* oder *Realmodell*.

einer mathematischen Lösung zu gelangen. Dabei wird heutzutage oft auf digitale Hilfsmittel (insbesondere Computer) zurückgegriffen. Die erhaltene mathematische Lösung muss schließlich in Bezug zur Realität gesetzt werden. Dazu wird die Lösung im Hinblick auf das reale Problem *interpretiert und validiert* (vgl. Blum et al., 2007, S. 9–10; Blum, 2007, S. 6; Blum, 1985, S. 200–205).

Die beschriebenen Schritte werden keineswegs stets in der angegebenen Reihenfolge durchlaufen. Beispielsweise ist es denkbar, dass beim Mathematisieren festgestellt wird, dass die notwendigen mathematischen Kompetenzen beziehungsweise Werkzeuge fehlen, um das vereinfachte Problem mathematisch darzustellen. Dann werden gegebenenfalls weitere Vereinfachungen vorgenommen und anschließend ein neuer Versuch unternommen ein mathematisches Modell zu entwickeln (vgl. Maaß, 2007, S. 13). Vielfach liefert das einmalige Durchlaufen des Modellierungskreislaufs noch keine zufriedenstellende Lösung, weshalb alle oder einzelne Schritte erneut absolviert werden. Modellieren kann damit als iterativer Prozess verstanden werden.

Der vierschriftige Modellierungskreislauf eignet sich aufgrund seiner Einfachheit für den Einsatz als metakognitives, strategisches Element, welches die Reflexion des gesamten Modellierungsprozesses durch die Lernenden unterstützen und sogleich den Aufbau mathematischer Modellierungskompetenz erleichtern soll (vgl. Hinrichs, 2008, S. 55). Der vierschriftige Kreislauf wird in dem Unterrichtsmaterial, das im Rahmen dieser Arbeit entwickelt wurde, wiederholt aufgegriffen (vgl. Kap. 6).

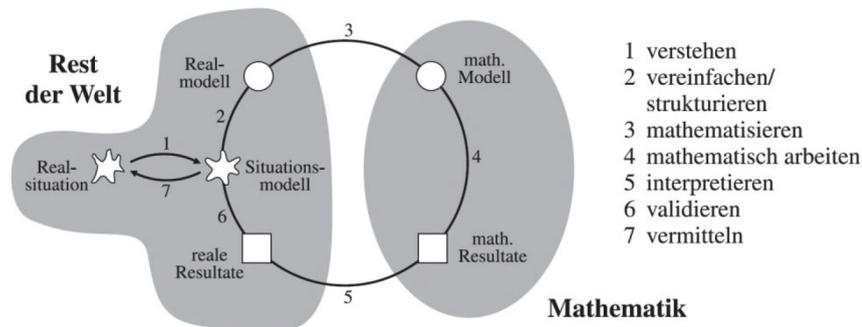


**Abb. 1.2:** Computergestützte Modellierungsspirale adaptiert von der *Solution Helix of Math* der Initiative *Computer-Based Math* (o. D.)

Erweitern wir den vierschriftigen Kreislauf von Blum (1985) um eine weitere Dimension, erhalten wir die Darstellung als Modellierungsspirale, die zusätzlich den Lösungsfortschritt verdeutlicht (vgl. Abb. 1.2).

Darüber hinaus existieren verschiedene weitere idealisierte Darstellungen des Modellierungsprozesses, die zu unterschiedlichen Zwecken entwickelt wurden und die die einzelnen Teilprozesse beim Modellieren unterschiedlich detailliert analysieren beziehungsweise aufschlüsseln (vgl. Greefrath et al., 2013, S. 18). An dieser Stelle sei auf

den von [Blum und Leiß \(2005\)](#) entwickelten siebenschrittigen Kreislauf hingewiesen (vgl. [Abb. 1.3](#)). Dieser eignet sich insbesondere für diagnostische Zwecke, da kognitive Aspekte des Modellierens detailliert dargestellt werden (vgl. [Niss und Blum, 2020, S. 25](#)). Denken wir in Richtung Beurteilung und Bewertung von Modellierungspro-



**Abb. 1.3:** Siebenschrittiger Modellierungskreislauf von [Blum und Leiß \(2005\)](#) entnommen aus [Greefrath et al. \(2013, S. 18\)](#)

zessen der Lernenden, eine in der Forschung viel diskutierte und in der Praxis noch offene Herausforderung, so wäre ein denkbarer Ansatz die Modellierungsschritte der Lernenden den Schritten des siebenschrittigen Kreislaufs zuzuordnen und diese dann separat zu bewerten (vgl. [Niss und Blum, 2020, S. 99 f.](#)).

Zwei Abgrenzungen im Hinblick auf die Begrifflichkeit des Modellierens scheinen an dieser Stelle angebracht: Unter Modellieren wird in dieser Arbeit [Blum et al. \(2007, S. 9\)](#) folgend nicht nur die Übersetzung eines realen Problems in ein mathematisches Modell bezeichnet, sondern der gesamte Prozess des (wiederholten) Hin- und Herwechsels zwischen Mathematik und Realität. Zudem sei auf eine zentrale Unterscheidung zwischen dem Prozess des Modellierens und einem mathematischen Modell hingewiesen. Mathematische Modelle sind lediglich ein (wenn auch integraler) Teil des gesamten Prozesses, machen diesen jedoch keinesfalls in Gänze aus. Ein mathematisches Modell kann beispielsweise ein Gleichungssystem oder eine Differentialgleichung zur Beschreibung einer außermathematischen Situation (z. B. eines physikalischen Prozesses) sein ([Frank et al., im Druck](#)).

## 1.2 Warum mathematische Modellierung im Mathematikunterricht?

Die Frage *Warum mathematische Modellierung im Mathematikunterricht?* lässt sich insbesondere im Hinblick auf die von [Winter \(1995, S. 37\)](#) beschriebenen Grunderfahrungen beantworten. Diese lauten

„Der Mathematikunterricht sollte anstreben, die folgenden drei Grunderfahrungen, die vielfältig miteinander verknüpft sind, zu ermöglichen:

- (1) Erscheinungen der Welt um uns, die uns alle angehen oder angehen sollten, aus Natur, Gesellschaft und Kultur, in einer spezifischen Art wahrzunehmen und zu verstehen,
- (2) mathematische Gegenstände und Sachverhalte, repräsentiert in Sprache, Symbolen, Bildern und Formeln, als geistige Schöpfungen, als eine deduktiv geordnete Welt eigener Art kennen zu lernen und zu begreifen,
- (3) in der Auseinandersetzung mit Aufgaben Problemlösefähigkeiten, die über die Mathematik hinaus gehen, (heuristische Fähigkeiten) zu erwerben.

Das Wort Erfahrung soll zum Ausdruck bringen, dass das Lernen von Mathematik weit mehr sein muss als eine Entgegennahme und Abspeicherung von Information, dass Mathematik erlebt (möglicherweise auch erlitten) werden muss.“

Echtes Modellieren und die aktive Auseinandersetzung mit Problemen aus dem wirklichen Leben, die somit über eingekleidete Aufgabenstellungen hinausgehen, haben das Potential, Lernenden insbesondere die erste Grunderfahrung Winters zu ermöglichen. Dies wird auch aus folgenden Ausführungen Winters zu seiner ersten Grunderfahrung erkennbar:

„In (1) ist die Mathematik als nützliche, brauchbare Disziplin angesprochen, und tatsächlich ist sie in dieser Hinsicht von schier universeller Reichweite. Dies allein impliziert noch nicht eine Bedeutung für Allgemeinbildung; den Gebrauch von Mathematik, der über Alltagsrechnen hinausgeht, könnte man ja der Berufsausbildung zuweisen. Interessant und wirklich unentbehrlich für Allgemeinbildung sind Anwendungen der Mathematik erst, wenn in Beispielen aus dem gelebten Leben erfahren wird, wie mathematische Modellbildung funktioniert und welche Art von Aufklärung durch sie zustande kommen kann, und Aufklärung ist Bürgerrecht und Bürgerpflicht (und wird durchaus nicht in den Schoß geworfen)“ (Winter, 1995, S. 38).

Die Frage nach dem „Warum?“ lässt sich zudem über die von der Kultusministerkonferenz festgelegten Bildungsstandards (die ebenso auf Winter fußen) beantworten. In diesen wird das mathematische Modellieren als eine der allgemeinen mathematischen Kompetenzen benannt, die Lernende im Fach Mathematik erwerben sollen. Auch die oben beschriebenen Teilschritte des Modellierungsprozesses werden dort explizit ausgewiesen (vgl. Kultusministerkonferenz, 2012, S. 15; Kultusministerkonferenz, 2003, S. 8). Zur Kompetenz „Mathematisch modellieren“ heißt es:

„Hier geht es um den Wechsel zwischen Realsituationen und mathematischen Begriffen, Resultaten oder Methoden. Hierzu gehört sowohl das Konstruieren passender mathematischer Modelle als auch das Verstehen oder Bewerten vorgegebener Modelle. Typische Teilschritte des Modellierens sind das Strukturieren und Vereinfachen gegebener Realsituationen, das Übersetzen realer Gegebenheiten in mathematische Modelle, das

Interpretieren mathematischer Ergebnisse in Bezug auf Realsituationen und das Überprüfen von Ergebnissen im Hinblick auf Stimmigkeit und Angemessenheit bezogen auf die Realsituation. Das Spektrum reicht von Standardmodellen (z. B. bei linearen Zusammenhängen) bis zu komplexen Modellierungen“ (Kultusministerkonferenz, 2012, S. 15).

Vereinfacht gesagt kann die **Modellierungskompetenz** als die Fähigkeit bezeichnet werden die notwendigen Prozessschritte beim Hin- und Herwechseln zwischen Mathematik und dem „Rest der Welt außerhalb der Mathematik“ problemadäquat auszuführen (vgl. Blum, 2007, S. 6; Greefrath et al., 2013, S. 18). Die oben beschriebenen Teilschritte des Modellierungsprozesses können dabei als Teilkompetenzen des Modellierens betrachtet werden (vgl. Kaiser, 2007).

Die **Ziele, die mit dem Modellieren im Mathematikunterricht** verfolgt werden, sind vielfältig und können als weitere Argumente, die für eine stärkere Einbindung von Anwendungsbezügen und Modellieren in den Mathematikunterricht sprechen, aufgefasst werden (vgl. Blum und Niss, 1991, S. 42–44; Greefrath et al., 2013, S. 20):

- Modellierung und Anwendungsorientierung soll zu einem *tieferen Verständnis* und einem besseren Behalten mathematischer Inhalte beitragen. Zudem sollen *Interesse und Motivation* an der Beschäftigung mit Mathematik vermittelt oder gefördert werden. Im Hinblick auf die dritte Grunderfahrung Winters können Modellierung und Realitätsbezüge dazu beitragen andere allgemeine mathematische Kompetenzen, wie etwa *Problemlösefähigkeiten, heuristische Strategien* und *Argumentations- sowie Kommunikationsfähigkeiten*, zu fördern und einen  *kreativen Umgang mit Mathematik* zu ermöglichen (vgl. Greefrath et al., 2013, S. 19; Maaß, 2007, S. 15 f.).
- Erneut bezugnehmend auf Winters erste Grunderfahrung soll Modellieren dazu beitragen, dass Schülerinnen und Schüler lernen ihre Umwelt durch die Beschäftigung mit realen Problemen in einer spezifischen Art wahrzunehmen und sich diese mit mathematischen Mitteln zu erschließen (vgl. Greefrath et al., 2013, S. 20). Auch soll Modellieren helfen ein *ausgewogenes Bild von Mathematik als Wissenschaft* zu vermitteln. Die Bedeutung von Mathematik für Kultur und Gesellschaft soll erfahrbar werden. Durch die aktive Beschäftigung mit realen Problemen sollen Nutzen und Sinn von Mathematik für die Lebenswelt der Lernenden deutlich werden.
- Die Lernenden sollen „Bezüge zwischen Mathematik und Realität erkennen, Kenntnisse über den Gebrauch und Missbrauch von Mathematik erwerben und die Grenzen der Mathematisierbarkeit erfahren“ (Maaß, 2007, S. 15).
- Zudem sollen soziale Kompetenzen der Lernenden durch die gemeinsame Bearbeitung von Modellierungsproblemen gefördert werden (vgl. Greefrath et al., 2013, S. 20).

## 1.3 Kriterien für gute Modellierungsprobleme

Die Auswahl geeigneter Problemstellungen nimmt eine übergeordnete Stellung bei der Entwicklung von Lehr- und Lernmaterialien und der Gestaltung von mathematischem Modellierungsunterricht ein. Es existieren verschiedene Kriterien, die berücksichtigt werden können, um eine Problemstellung als mehr oder weniger geeignet einzuordnen. Die Kriterien, die in dieser Arbeit bei der Auswahl der Problemstellungen im Vordergrund stehen, sind **Authentizität**, **Relevanz** und **Reichhaltigkeit**.

Einer Definition von Niss (1992) folgend wird ein außermathematisches Problem in dieser Arbeit als **authentisch** bezeichnet, wenn es für Expertinnen und Experten in diesem Gebiet relevant ist und von diesen als ein Problem angesehen wird, mit dem sie sich in ihrer Arbeit auseinandersetzen würden (vgl. Niss, 1992, zitiert nach Kaiser und Schwarz, 2010, S. 53; Maaß, 2010, S. 292). Diese Definition kann um den Aspekt erweitert werden, dass beim Lösen der Probleme in Workshops / im (Projekt-)Unterricht die Mathematik thematisiert wird und die mathematischen Modelle entwickelt werden, die in der Praxis (gegebenenfalls in komplexerer Form) tatsächlich zum Lösen der Probleme verwendet werden (vgl. Greefrath et al., 2017, S. 934; Frank et al., im Druck). Die Problemstellung soll zudem real sein und damit nicht für das Einüben eines mathematischen Inhalts konstruiert worden sein (Stichwort: eingekleidete Aufgaben). Tatsächlich greift diese durchaus strikte Definition von Authentizität bei den in dieser Arbeit diskutierten Modellierungsproblemen. Darüber hinaus existieren verschiedene weitere (weniger strikte) Definitionen und gänzlich andere Auffassungen des Authentizitätsbegriffs (vgl. Niss und Blum, 2020, S. 103; Vos, 2011, S. 68). Für eine umfassende Diskussion dessen sei auf Vos (2011) oder Palm (2002) verwiesen. In der vorliegenden Arbeit wird der Authentizitätsbegriff überdies auf die Nutzung digitaler Werkzeuge übertragen, was in Abschnitt 1.4 detaillierter ausgeführt wird.

Als **relevant** wird ein Problem bezeichnet, wenn es aus der Erfahrungswelt der Lernenden stammt oder sie zumindest eine Beziehung zu dem Problem aufbauen können (vgl. Maaß, 2010, S. 292). Ob eine Schülerin oder ein Schüler ein Problem als relevant ansieht oder Interesse für dieses entwickeln kann, hängt jedoch von der individuellen Lernsituation der/des Lernenden ab (vgl. Maaß, 2010, S. 292; Greefrath et al., 2013, S. 26).

Der Begriff der **Reichhaltigkeit** einer Problemstellung ist in der didaktischen Literatur nicht klar definiert. In dieser Arbeit wird eine Problemstellung als reichhaltig bezeichnet, wenn sie möglichst viele der folgenden Eigenschaften aufweist, die unter anderem bei Heiliö und Pohjolainen (2016) benannt werden und bei Frank et al. (im Druck), Wohak et al. (2021) und Sube (2019) aufgegriffen wurden:

- Das Problem ist sowohl innermathematisch als auch außermathematisch interdisziplinär. Zur Lösung sind verschiedene Gebiete der Mathematik sowie weiteres außermathematisches Wissen notwendig (vgl. Heiliö und Pohjolainen, 2016, S. 4).
- Das Problem ist offen und lässt verschiedene Lösungsstrategien zu. Die Offenheit eines Problems verstehen wir dabei insbesondere im Sinne von Wiegand

und Blum (1999). Die Autoren beschreiben ein Problem als eine geistige Anforderung, bei der ein gewisser Anfangszustand (A) über ein gewisse Transformation (T) in einen gewünschten Zielzustand (Z) überführt werden soll. Als offen bezeichnen sie ein Problem beziehungsweise eine Aufgabe wenn mindestens einer dieser drei Bestandteile (A, T oder Z) unklar oder mehrdeutig ist (vgl. Blum und Wiegand, 2000, S. 52; Wiegand und Blum, 1999, S. 590–591).

- Das Problem erlaubt es zu erkennen, dass eine vollständige und optimale Lösung nicht immer existiert und auch die Definition von „gelöst“ divers sein kann (vgl. Heiliö und Pohjolainen, 2016, S. 4).
- Das Problem lässt einfache und auch komplexe Modellierung zu und ermöglicht den Lernenden einen kreativen Einsatz von Mathematik (vgl. Heiliö und Pohjolainen, 2016, S. 4; Sube, 2019, S. 11).
- Die Arbeit an dem Problem verdeutlicht den Lernenden die Bedeutung von Teamarbeit für das Lösen realer Problemstellungen (vgl. Heiliö und Pohjolainen, 2016, S. 4).

Durchaus muss die Ausprägung der verschiedenen Aspekte an die gegebenen Rahmenbedingungen angepasst werden. Findet die Modellierungsaktivität in einer längerfristigen Projektzeit statt, so können die Lernenden Problemstellungen durchaus sehr offen, unangeleitet und selbstbestimmt bearbeiten. Steht hingegen weniger Zeit zur Verfügung, wie beispielsweise im Rahmen einer Doppelstunde im Mathematikunterricht, so ist es notwendig die Arbeitsaufträge kleinschrittiger und enger zu fassen (vgl. Frank et al., im Druck). Dies gilt insbesondere, wenn garantiert werden soll, dass bei der Bearbeitung komplexer Probleme alle Schritte des Modellierungskreislaufs von den Lernenden durchlaufen werden.

Die im Rahmen dieser Arbeit entwickelten Unterrichtsmaterialien wurden für die Durchführung unter verschiedenen Rahmenbedingungen und damit auch mit variierendem Offenheitsgrad entwickelt und erprobt. Auf die konkrete Umsetzung wird in Teil II eingegangen.

Die hier beschriebenen Kriterien für gute Problemstellungen werden in den Abschnitten 13.3 und 18.3 wieder aufgegriffen. Dort wird diskutiert, inwieweit die gewählten Probleme aus den Bereichen Solarenergie und KI diesen Kriterien genügen.

## 1.4 Mathematisches Modellieren mit digitalen Werkzeugen

Die Erprobung der entwickelten Lehr- und Lernmaterialien hat bedingt durch die COVID-19-Pandemie fast ausschließlich im Rahmen von online Modellierungsprojekten stattgefunden. Digitale Werkzeuge haben somit eine grundlegende Rolle gespielt. Der Einsatz von digitalen Werkzeugen in mathematischen Modellierungsaktivitäten ist jedoch keineswegs nur sinnvoll, wenn diese online durchgeführt werden, sondern kann auch die Präsenzdurchführungen bereichern. Verschiedene Ziele und Vorteile des Einsatzes von Technologien für mathematische Modellierungsaktivitäten, die

unter anderem bereits von Greefrath und Siller (2018, 2010) hervorgehoben wurden, umfassen folgende Punkte:

- Technologien erleichtern oder ermöglichen es, sich mit realen Problemstellungen auseinanderzusetzen. Reale Probleme sind vielfach komplex, erfordern die Arbeit mit großen Datenmengen oder das Ausführen rechenaufwendiger Operationen (vgl. Siller und Greefrath, 2010, S. 2138; Greefrath und Siller, 2018, S. 5). Der Einsatz von Technologien bietet die Chance die Vielfalt an Problemen zu erweitern, die mit Schülerinnen und Schülern behandelt werden können. Dies ist insbesondere in der heutigen Zeit relevant, in der zahlreiche Probleme und Anwendungen, die den Lernenden tagtäglich begegnen, auf riesigen Datenmengen basieren. Beispiele dafür sind Apps zur Musikererkennung, Fitnessstracker oder die automatische Gesichtserkennung. Alle diese Anwendungen fußen auf mathematischen Modellen, die computergestützt und in elementarer Form auch Schülerinnen und Schülern zugänglich gemacht werden können (vgl. Wohak et al., 2021, S. 41–43).
- Durch den Einsatz von Technologien kann der Fokus auf den Modellierungsprozess gelegt werden, indem repetitive oder aufwendige Rechnungen dem Computer überlassen werden (vgl. Greefrath und Siller, 2018, S. 5). Dies erlaubt es zudem zeiteffizient auch experimentell zu arbeiten, beispielsweise in Form von Parameterstudien beziehungsweise systematischen Variationen.
- Ein aktiver, gestaltender, aber auch kritisch-reflektierter Einsatz von digitalen Werkzeugen ist in der heutigen Zeit, in der wir im Alltag mit sich rasant wandelnden Technologien und digitalen Medien konfrontiert werden, eine wesentliche Kompetenz (vgl. Kultusministerkonferenz, 2017, S. 11).
- In Wissenschaft, Technik und Forschung werden reale Probleme mithilfe von digitalen Werkzeugen angegangen und gelöst. Damit kann der Einsatz digitaler Werkzeuge in Problemlöseprozessen auch im Hinblick auf die angestrebte Authentizität von Modellierungsunterricht legitimiert werden.
- Der Einsatz von Technologien erlaubt die Darbietung verschiedener Visualisierungen im Modellierungsprozess (vgl. Siller und Greefrath, 2010, S. 2139).

Auch in den Bildungsstandards wird ein lernfördernder Einsatz digitaler Werkzeuge hervorgehoben:

„Die Entwicklung mathematischer Kompetenzen wird durch den sinnvollen Einsatz digitaler Mathematikwerkzeuge unterstützt. Das Potenzial dieser Werkzeuge entfaltet sich im Mathematikunterricht

- beim **Entdecken** mathematischer Zusammenhänge, insbesondere durch interaktive Erkundungen beim Modellieren und Problemlösen,
- durch **Verständnisförderung** für mathematische Zusammenhänge, nicht zuletzt mittels vielfältiger Darstellungsmöglichkeiten,
- mit der **Reduktion** schematischer Abläufe und der **Verarbeitung größerer Datenmengen**,

- durch die Unterstützung individueller Präferenzen und Zugänge beim Bearbeiten von Aufgaben einschließlich der reflektierten Nutzung von **Kontrollmöglichkeiten**“ ([Kultusministerkonferenz, 2012](#), S. 13).

Sämtliche der hier genannten Potentiale digitaler Werkzeuge wurden in unterschiedlicher Ausprägung im entwickelten computergestützten Lernmaterial aufgegriffen und umgesetzt.

Durchaus kann der Einsatz digitaler Werkzeuge jedoch auch zu neuen Herausforderungen führen. So muss – neben der womöglich bereits anspruchsvollen Bearbeitung eines realen Problems – sogleich die Verwendung des digitalen Werkzeugs erlernt und beherrscht werden. Damit es seitens der Lernenden nicht zu Überforderung kommt, sollte die Einführung, die Gestaltung und der Einsatz (neuer) digitaler Werkzeuge durchdacht erfolgen (vgl. [Niss und Blum, 2020](#), S. 106).

## 2 Optimierung

Das Optimieren spielt in der Mathematik bereits seit der Antike eine Rolle; damals insbesondere „im Zusammenhang mit der Analyse geometrischer Konstruktionen“ (Schupp, 1992, S. 53). Eine Übersicht über die Entwicklung der Optimierung und zentraler Ausrichtungen von der Antike bis in die 90er Jahre liefert Schupp (1992, S. 53 ff.) in seinem Buch „Optimieren“. Dabei greift er unter anderem die Entwicklung des Kalküls der Differentialrechnung im 17. Jahrhundert und damit einhergehend die Bestimmung von Extremwerten differenzierbarer Funktionen auf. Diese wurde in besonderem Maße durch Arbeiten von Leibniz und Euler geprägt. Zudem diskutiert er, wie sich die im 20. Jahrhundert aufstrebenden Zweige der Spieltheorie, der kombinatorischen Optimierung und die Theorie zu linearer sowie nichtlinearer Optimierung entwickelt haben.

Die mathematische Optimierung findet sich in der Literatur vielfach auch unter dem Begriff der *mathematischen Programmierung* wieder. Dieser geht laut Nocedal und Wright (2006, S. 9) auf die 1940er Jahre zurück. Damals war die Bezeichnung „Programmierung“ im Kontext der Optimierung weniger mit der Informatik und der Entwicklung von Computersoftware verknüpft (wie es heute der Fall ist) als vielmehr mit dem Design und der Entwicklung von Algorithmen.

In diesem Kapitel wird zunächst ein Überblick über die mathematische Optimierung gegeben. Zentrale Begriffe werden geklärt und die theoretischen Grundlagen geliefert, die für die spätere Diskussion der Optimierungsprobleme im entwickelten Unterrichtsmaterial wesentlich sind. Zudem werden unterschiedliche Arten von Optimierungsproblemen einander gegenübergestellt. Abschließend wird die bestehende didaktische Literatur und Forschung rund um Optimierungsprobleme beleuchtet und Konsequenzen für die stoffdidaktische Forschung dieser Arbeit aufgezeigt.

### 2.1 Grundlagen und zentrale Begriffe

Ausgangspunkt der Optimierung ist das Ziel aus einer Menge an möglichen Alternativen diejenigen auszuwählen, die eine Größe beziehungsweise ein Kriterium maximieren oder minimieren. Die Menge der betrachteten Alternativen wird auch als **zulässige Menge**  $\mathcal{F}$  bezeichnet (vgl. Stein, 2021a, S. 2). In der Mathematik wird das Kriterium mit dem die Optimalität gemessen werden kann, üblicherweise durch eine Funktion<sup>1</sup> beschrieben. Diese Funktion wird auch als **Zielfunktion** oder

<sup>1</sup>In den meisten Fällen handelt es sich um ein Funktional. In diesem Kapitel wird jedoch auch die multikriterielle Optimierung und damit vektorwertige Funktionen angerissen, für die diese Beschreibung nicht zutrifft. Deswegen wird hier durchgängig von einer Funktion gesprochen.

**Bewertungsfunktion** bezeichnet. Sie weist jeder Alternative  $\mathbf{x} \in \mathcal{F}$  einen Wert zu. Kern der Optimierung ist es, die Zielfunktion auf der zulässigen Menge zu minimieren oder zu maximieren. Formal lassen sich Optimierungsprobleme schreiben als

$$\min_{\mathbf{x}} f(\mathbf{x}) \text{ u. d. Nb. } \mathbf{x} \in \mathcal{F} \quad (2.1)$$

mit Zielfunktion  $f : \mathcal{F} \rightarrow \mathbb{R}$ . Die Abkürzung *u. d. Nb.* steht für *unter der Nebenbedingung* oder *unter den Nebenbedingungen*. In den folgenden Abschnitten werden zentrale Aussagen für Probleme diskutiert für die  $\mathcal{F} \subseteq \mathbb{R}^n$  gilt. In diesem Fall wird nicht mehr von Alternativen, sondern von Punkten gesprochen. Ein Punkt  $\mathbf{x} \in \mathbb{R}^n$ , für den  $\mathbf{x} \in \mathcal{F}$  gilt, wird auch **zulässiger Punkt** genannt.

Durchaus gibt es Optimierungsprobleme bei denen die zulässige Menge  $\mathcal{F}$  nicht als Teilmenge des  $\mathbb{R}^n$  beschrieben werden kann. Beispielsweise Probleme bei denen  $\mathcal{F}$  Teilmenge eines (unendlichdimensionalen) Funktionenraums oder eine Menge von Permutationen ist.

### Maximierungs- oder Minimierungsprobleme?

In der mathematischen Optimierung ist es üblich Minimierungsprobleme zu formulieren. Viele der computergestützten Optimierungsverfahren, die entwickelt und implementiert werden, sind für das Lösen von Minimierungsproblemen ausgelegt. Maximierungsprobleme können als Minimierungsprobleme formuliert werden, indem die negative Zielfunktion betrachtet wird (d. h.  $\max f = \min(-f)$ ). In diesem Kapitel werden die theoretischen Grundlagen anhand von Minimierungsproblemen eingeführt.

### Lokale und globale Minimalpunkte

Ziel der globalen Optimierung ist es, Punkte  $\mathbf{x}^*$  aus der zulässigen Menge zu finden, die unter allen Punkten in der zulässigen Menge den kleinsten Funktionswert besitzen. Derartige Punkte werden auch als globale Minimalpunkte bezeichnet. Formal definieren wir diese wie folgt:

**Definition 1** (globaler Minimalpunkt).

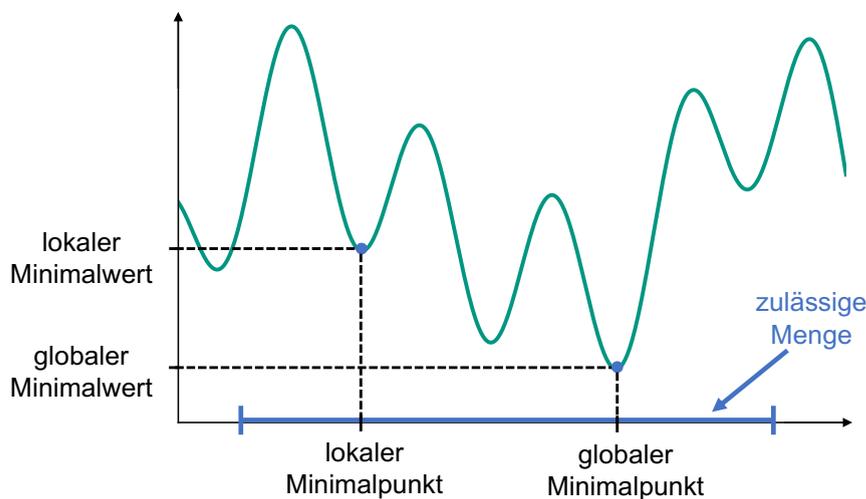
*Seien eine zulässige Menge  $\mathcal{F} \subseteq \mathbb{R}^n$  und eine Funktion  $f : \mathcal{F} \rightarrow \mathbb{R}$  gegeben. Ein Punkt  $\mathbf{x}^* \in \mathcal{F}$  wird globaler Minimalpunkt oder globale Lösung von  $f$  auf  $\mathcal{F}$  genannt, wenn  $f(\mathbf{x}^*) \leq f(\mathbf{z})$  für alle  $\mathbf{z} \in \mathcal{F}$  gilt. Ist die strikte Ungleichung  $f(\mathbf{x}^*) < f(\mathbf{z})$  für  $\mathbf{z} \neq \mathbf{x}^*$  erfüllt, so spricht man von einem strikten globalen Minimalpunkt. Der Funktionswert  $f(\mathbf{x}^*)$  zum globalen Minimalpunkt wird globaler Minimalwert genannt.*

Existiert ein globaler Minimalpunkt, so spricht man davon, dass das Optimierungsproblem *lösbar* ist. Erzielt ein Punkt  $\mathbf{x}$  lediglich in einer Umgebung  $\mathcal{N}$  von  $\mathbf{x}$  den kleinsten Zielfunktionswert, so liegt ein lokaler Minimalpunkt vor (vgl. Abb. 2.1). Formal halten wir fest:

**Definition 2** (lokaler Minimalpunkt).

Seien eine zulässige Menge  $\mathcal{F} \subseteq \mathbb{R}^n$  und eine Funktion  $f : \mathcal{F} \rightarrow \mathbb{R}$  gegeben. Ein Punkt  $\mathbf{x}^* \in \mathcal{F}$  wird lokaler Minimalpunkt oder lokale Lösung von  $f$  auf  $\mathcal{F}$  genannt, wenn eine Umgebung  $\mathcal{N}$  von  $\mathbf{x}^*$  existiert, sodass  $f(\mathbf{x}^*) \leq f(\mathbf{z})$  für alle  $\mathbf{z} \in \mathcal{F} \cap \mathcal{N}$  gilt. Ist die strikte Ungleichung  $f(\mathbf{x}^*) < f(\mathbf{z})$  für  $\mathbf{z} \neq \mathbf{x}^*$  erfüllt, so spricht man von einem strikten lokalen Minimalpunkt. Der Funktionswert  $f(\mathbf{x}^*)$  zum lokalen Minimalpunkt wird lokaler Minimalwert genannt.

Der Begriff der **Lösung** eines Optimierungsproblems wird in der Literatur nicht immer einheitlich verwendet. Zum Teil wird darunter nur der Optimalpunkt verstanden (vgl. Alt, 2011, S. 3), zum Teil umfasst der Begriff den Optimalpunkt **und** den Optimalwert (vgl. Stein, 2021a, S. 3). Hier wird der Begriff auf den Optimalpunkt bezogen. Der Grund dafür ist didaktischer Natur: Im Mathematikunterricht wird üblicherweise zwischen Extremstelle, Extremwert und Extrempunkt unterschieden (vgl. Brandt et al., 2014, S. 84). Der Begriff Extrempunkt bezieht sich dabei auf die Angabe des optimalen Wertes für die Variable  $x \in \mathbb{R}$  (Extremstelle) **und** des zugehörigen Zielfunktionswertes  $f(x)$  (Extremwert). Um Verwechslungen zwischen den mit unterschiedlicher Bedeutung belegten Begriffen eines Extrempunktes ( $x$  und  $f(x)$ ) und eines Optimalpunktes (nur  $\mathbf{x}$ ) zu vermeiden, wird im entwickelten Lernmaterial der Begriff einer optimalen Lösung anstelle des Begriffs eines Optimalpunktes verwendet.



**Abb. 2.1:** Lokale und globale Minimalpunkte

In der Regel ist man daran interessiert globale Minimalpunkte zu finden. Dies ist jedoch oft keine leichte Aufgabe. Insbesondere bei der Modellierung von realen Problemen sind die Zielfunktionen nicht selten so komplex, dass es nicht möglich ist sich ein globales Bild der Funktion  $f$  zu verschaffen. Stattdessen liegen nur lokale Informationen über die Funktion vor (vgl. Nocedal und Wright, 2006, S. 12).

Beim Formulieren und Lösen von Optimierungsproblemen sind insbesondere im Hinblick auf Anwendungsprobleme verschiedene Fragestellungen zentral:

1. Welche **Eigenschaften** haben Zielfunktion, Nebenbedingungen und Variablen des Optimierungsproblems?
2. **Existenz:** Existieren überhaupt globale Minimalpunkte? Mit anderen Worten, ist das Optimierungsproblem lösbar?
3. **Eindeutigkeit:** Falls ein globaler Minimalpunkt existiert, ist dieser eindeutig?
4. **Auswahl eines Verfahrens:** Welche Strategien beziehungsweise Verfahren eignen sich zum Lösen des Optimierungsproblems?
5. **Laufzeit des Verfahrens:** Wie lange dauert es, um mit einem gewählten Verfahren einen Minimalpunkt zu bestimmen?
6. **Güte:** Wie nah liegt die mit einem Verfahren gefundene Lösung an dem tatsächlichen Minimalpunkt, sofern dieser existiert?

Die Fragen 1 bis 3 beziehen sich auf die Analyse und Untersuchung von Eigenschaften und Struktur eines Optimierungsproblems. Je nach Eigenschaften können die Probleme in Klassen unterteilt werden. Auf verschiedene Klassen von Optimierungsproblemen wird in Abschnitt 2.1.2 eingegangen. Die Fragen 4 bis 6 zielen darauf ab konkrete Verfahren / Algorithmen zu entwickeln, mit denen ein Minimalpunkt des Optimierungsproblems gefunden werden kann. Für die Entwicklung oder Auswahl und die Evaluierung eines Optimierungsverfahrens sind die Antworten auf Fragen 1 bis 3 zentral.

### 2.1.1 Existenz und Eindeutigkeit globaler Minimalpunkte

Wir beschäftigen uns zunächst mit der Existenz eines globalen Minimalpunktes beziehungsweise der Lösbarkeit eines Optimierungsproblems. Auch hier gehen wir nur auf wenige zentrale Sätze ein die in der Modellierung der ausgewählten Problemstellungen in Teil III und IV aufgegriffen werden, um die Existenz von Minimalpunkten bei den betrachteten Optimierungsproblemen zu zeigen.

Eine hinreichende (wenngleich nicht notwendige) Bedingung, die die Existenz globaler Minimalpunkte garantiert, liefert ein zentraler Satz der Analysis: der **Satz von Weierstraß** (vgl. Stein, 2021a, S. 19–20).

**Satz 1** (Satz von Weierstraß).

*Die Funktion  $f : \mathcal{F} \rightarrow \mathbb{R}$  sei stetig und die Menge  $\mathcal{F} \subseteq \mathbb{R}^n$  sei nichtleer und kompakt. Dann besitzt die Funktion  $f$  auf  $\mathcal{F}$  mindestens einen globalen Minimalpunkt und einen globalen Maximalpunkt.*

**Beweis.** Der Beweis kann bei Stein (2021a, S. 20) eingesehen werden.

Damit der Satz von Weierstraß anwendbar ist, muss die zulässige Menge des Optimierungsproblems kompakt, das heißt abgeschlossen und beschränkt sein; was in vielen Fällen nicht gegeben ist. Beispielsweise bei der Minimierung der quadratischen Funktion  $f : \mathbb{R} \rightarrow \mathbb{R}$  mit  $f(x) = x^2$  ist der Satz von Weierstraß nicht direkt

anwendbar. Um dennoch hinreichende Bedingungen für die Existenz globaler Minimalpunkte angeben zu können, werden wir eine Version des Satzes von Weierstraß ausnutzen, bei der die Voraussetzungen der Beschränktheit und Abgeschlossenheit abgeschwächt werden (vgl. Stein, 2021b, S. 14). Dazu benötigen wir zunächst die Definition von unteren Niveaumengen:

**Definition 3** (Untere Niveaumenge).

Sei  $f : D \rightarrow \mathbb{R}$  eine Funktion mit  $D \subseteq \mathbb{R}^n$ . Für ein  $a \in \mathbb{R}$  heißt die Menge

$$\text{Niv}(f, a) = \{\mathbf{x} \in D \mid f(\mathbf{x}) \leq a\}$$

untere Niveaumenge von  $f$  auf  $D$  zum Niveau  $a$  (vgl. Stein, 2021b, S. 15).

Der verschärfte Satz von Weierstraß nutzt die Definition unterer Niveaumengen, um ein hinreichendes Kriterium für die Existenz globaler Minimalpunkte abzuleiten. Er wird in der nichtlinearen Optimierung vielfach genutzt, um die Existenz globaler Minimalpunkte zu zeigen (vgl. Alt, 2011, S. 5).

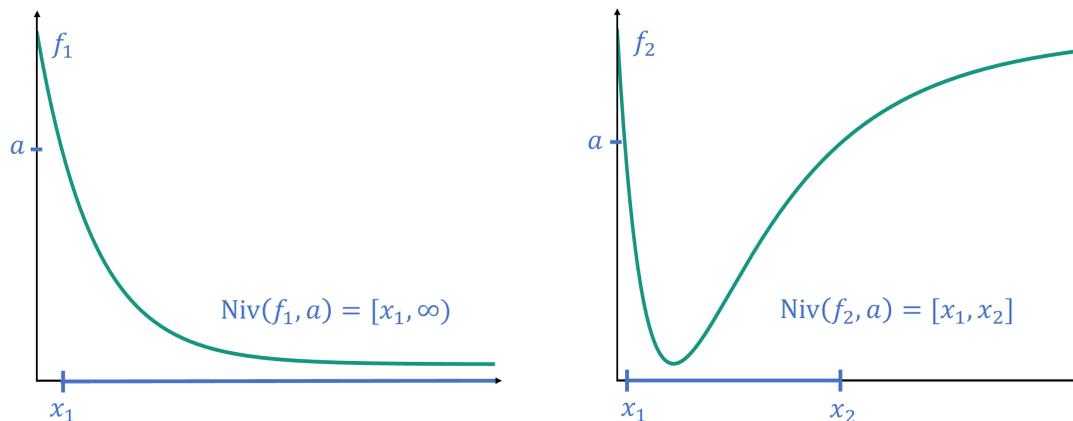
**Satz 2** (Verschärfter Satz von Weierstraß).

Ist die Funktion  $f : \mathcal{F} \rightarrow \mathbb{R}$  des Problems (2.1) mit  $\mathcal{F} \subseteq \mathbb{R}^n$  stetig und ist für ein  $a \in \mathbb{R}$  die untere Niveaumenge  $\text{Niv}(f, a)$  nichtleer und kompakt, dann existiert mindestens ein globaler Minimalpunkt des Optimierungsproblems.

**Beweis.** Der Beweis kann bei Alt (2011, S. 5) eingesehen werden.

In Abbildung 2.2 sind Beispiele für Funktionen mit kompakter und nichtkompakter unterer Niveaumenge dargestellt.

Der verschärfte Satz von Weierstraß lässt sich mit  $D = \mathbb{R}^n$  nun auch auf Probleme ohne Nebenbedingungen (sog. unrestringierte Probleme) anwenden.



(a) Nichtkompakte Niveaumenge

(b) Kompakte Niveaumenge

**Abb. 2.2:** Kompakte und nichtkompakte Niveaumengen

Um nicht stets ein explizites Niveau  $a$  angeben zu müssen, für das die Bedingungen aus Satz 2 erfüllt sind, lässt sich die sogenannte **Koerzivität** von Funktionen ausnutzen, um die Existenz globaler Minimalpunkte zu garantieren (vgl. Stein, 2021a, S. 26).

**Definition 4** (Koerzivitat).

Sei  $f : D \rightarrow \mathbb{R}$  eine Funktion mit  $D \subseteq \mathbb{R}^n$ . Falls fur alle Folgen  $(\mathbf{x}^k)_{k \in \mathbb{N}} \subseteq D$  mit  $\lim_k \|\mathbf{x}^k\| \rightarrow +\infty$  und alle konvergenten Folgen  $(\mathbf{x}^k)_{k \in \mathbb{N}} \subseteq D$  mit  $\lim_k \mathbf{x}^k \notin D$

$$\lim_k f(\mathbf{x}^k) = +\infty$$

gilt, dann heist  $f$  koerziv auf  $D$  (vgl. [Stein, 2021a](#), S. 33).

Ein Beispiel fur eine koerzive Funktion auf  $\mathbb{R}$  ist  $f$  mit  $f(x) = x^2$ . Hingegen ist die Funktion  $f$  mit  $f(x) = e^x$  aufgrund des asymptotischen Verhaltens fur  $x \rightarrow -\infty$  nicht koerziv auf  $\mathbb{R}$ , jedoch auf  $\mathbb{R}_{\geq 0}$ .

Die Koerzivitat einer Funktion konnen wir nun ausnutzen, um die Kompaktheit unterer Niveaumengen fur beliebige Niveaus  $a$  zu garantieren:

**Satz 3.** Sei  $f : D \rightarrow \mathbb{R}$  eine stetige und auf  $D \subseteq \mathbb{R}^n$  koerzive Funktion. Dann ist die untere Niveaumenge  $\text{Niv}(f, a)$  fur jedes Niveau  $a \in \mathbb{R}$  kompakt.

**Beweis.** Der Beweis ist bei [Stein \(2021a, S. 33\)](#) zu finden.

Fur die Existenz globaler Minimalpunkte genugt es gema dem folgenden Lemma somit Stetigkeit und Koerzivitat der Zielfunktion  $f$  auf der zulassigen Menge  $\mathcal{F}$  zu fordern:

**Lemma 1.** Die zulassige Menge  $\mathcal{F} \subseteq \mathbb{R}^n$  des Problems (2.1) sei nichtleer und die Zielfunktion  $f : \mathcal{F} \rightarrow \mathbb{R}$  sei stetig und koerziv auf  $\mathcal{F}$ . Dann existiert (mindestens) ein globaler Minimalpunkt des Optimierungsproblems.

**Beweis.** Der Beweis kann bei [Stein \(2021a, S. 34\)](#) eingesehen werden.

Nun haben wir bereits verschiedene hinreichende Bedingungen kennengelernt, die die Existenz globaler Minimalpunkte garantieren. Vielfach ist man jedoch nicht nur daran interessiert, ob ein globaler Minimalpunkt existiert, sondern ob dieser zudem eindeutig ist. In diesem Fall ist eine andere Eigenschaft von Funktionen auerst nutzlich: **Konvexitat**. Bevor zentrale Aussagen zu konvexen Optimierungsproblemen benannt werden, sei zunachst an wichtige Definitionen erinnert:

**Definition 5** (Konvexe Mengen und Funktionen).

- Eine Menge  $D \subseteq \mathbb{R}^n$  heist konvex, wenn fur alle  $\mathbf{x}, \mathbf{y} \in D$  und  $\lambda \in (0, 1)$

$$(1 - \lambda) \mathbf{x} + \lambda \mathbf{y} \in D$$

gilt. Das bedeutet anschaulich, dass die Verbindungsstrecke zweier Punkte aus  $D$  stets vollstandig innerhalb der Menge  $D$  liegt.

- Sei  $D \subseteq \mathbb{R}^n$  konvex. Die Funktion  $f : D \rightarrow \mathbb{R}$  heißt konvex auf  $D$ , wenn für alle  $\mathbf{x}, \mathbf{y} \in D$  und  $\lambda \in (0, 1)$

$$f((1 - \lambda)\mathbf{x} + \lambda\mathbf{y}) \leq (1 - \lambda)f(\mathbf{x}) + \lambda f(\mathbf{y})$$

gilt. Das bedeutet anschaulich, dass jede Verbindungslinie von zwei Punkten auf dem Funktionsgraphen von  $f$  oberhalb des Graphen selbst liegt (vgl. Abb. 2.3). Falls für  $\mathbf{x} \neq \mathbf{y}$  sogar die strikte Ungleichung gilt, heißt  $f$  auch strikt konvex.

- Sei  $D \subseteq \mathbb{R}^n$  konvex. Die Funktion  $f : D \rightarrow \mathbb{R}^n$  heißt (strikt) konkav, wenn die Funktion  $-f$  (strikt) konvex ist.

Beispiele für konvexe und nichtkonvexe Funktionen sind in Abbildung 2.3 dargestellt.

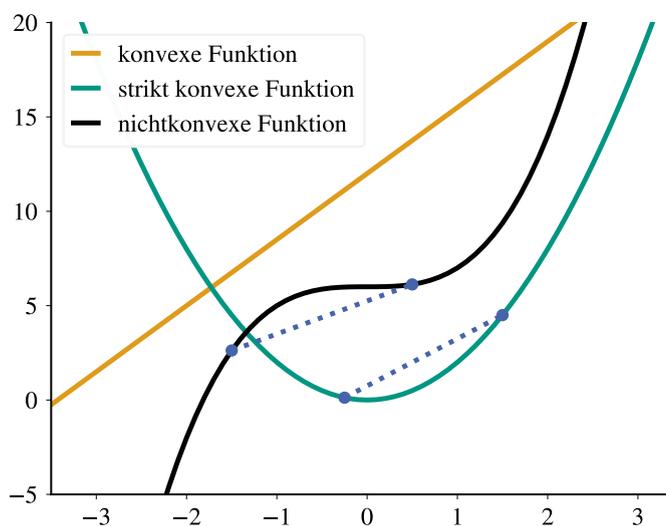


Abb. 2.3: Konvexe und nichtkonvexe Funktionen

**Definition 6** (Konvexes Optimierungsproblem).

Ein Optimierungsproblem der Form (2.1) heißt konvex, wenn die zulässige Menge  $\mathcal{F}$  konvex ist und zudem die Zielfunktion  $f$  konvex auf  $\mathcal{F}$  ist.

Eine wesentliche Eigenschaft, die konvexe Optimierungsprobleme zu „besonders gern gesehenen“ Problemen macht, liefert der folgende Satz:

**Satz 4.** Jeder lokale Minimalpunkt eines konvexen Optimierungsproblems ist auch globaler Minimalpunkt.

**Beweis.** Der Beweis ist beispielsweise bei Stein (2021a, S. 43) zu finden.

Bei konvexen Problemen genügt es damit, einen lokalen Minimalpunkt zu suchen. Der folgende Satz liefert schließlich eine Bedingung, die die Eindeutigkeit von Lösungen garantiert:

**Satz 5.** Das Optimierungsproblem (2.1) sei konvex. Falls die Zielfunktion  $f$  strikt konvex auf  $\mathcal{F}$  ist, dann existiert höchstens ein globaler Minimalpunkt.

**Beweis.** Der Beweis ist bei [Stein \(2021a, S. 52\)](#) zu finden.

Für konvexe Optimierungsprobleme mit nichtleerer zulässiger Menge  $\mathcal{F}$  und einer auf  $\mathcal{F}$  strikt konvexen, stetigen und koerziven Zielfunktion  $f$  existiert damit nach Lemma 1 ein globaler Minimalpunkt. Nach Satz 5 ist dieser eindeutig. Das Optimierungsproblem ist in diesem Fall eindeutig lösbar. Derartige Optimierungsprobleme sind beim Lösen realer Problemstellungen gern gesehen, da man sich keine Gedanken darüber machen muss, ob mit einem Verfahren gegebenenfalls *nur* ein lokaler Minimalpunkt (und womöglich einer mit *schlechtem* Zielfunktionswert) gefunden wird.

Für ein- und zweimal stetig differenzierbare Funktionen können nützliche Charakterisierungen von beziehungsweise Bedingungen für (strikte) Konvexität angegeben werden, die vielfach leichter zu überprüfen sind als die in Definition 5 gegebenen Eigenschaften.

**Satz 6** ( $C^1$ -Charakterisierung von Konvexität).

Die Funktion  $f : D \rightarrow \mathbb{R}$  sei mindestens einmal stetig differenzierbar und die Menge  $D \subseteq \mathbb{R}^n$  sei konvex. Es gilt:

a) Die Funktion  $f$  ist genau dann konvex, wenn für alle  $\mathbf{x}, \mathbf{y} \in D$  gilt

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x}) \cdot (\mathbf{y} - \mathbf{x}).$$

Hier bezeichnet  $\nabla f(\mathbf{x})$  den Gradienten von  $f$  an  $\mathbf{x}$ .

b) Die Funktion  $f$  ist genau dann strikt konvex, wenn für alle  $\mathbf{x}, \mathbf{y} \in D, \mathbf{x} \neq \mathbf{y}$  gilt

$$f(\mathbf{y}) > f(\mathbf{x}) + \nabla f(\mathbf{x}) \cdot (\mathbf{y} - \mathbf{x}).$$

**Beweis.** Der Beweis ist bei [Boyd und Vandenberghe \(2004, S. 70\)](#) zu finden.

**Satz 7** (Hinreichende Bedingung für Konvexität zweiter Ordnung).

Die Funktion  $f : D \rightarrow \mathbb{R}$  sei mindestens zweimal stetig differenzierbar und die Menge  $D \subseteq \mathbb{R}^n$  sei konvex.

a) Falls die Hesse-Matrix  $\nabla^2 f(\mathbf{x})$  für alle  $\mathbf{x} \in D$  positiv semidefinit<sup>2</sup> ist, dann ist  $f$  konvex auf  $D$ .

b) Falls die Hesse-Matrix  $\nabla^2 f(\mathbf{x})$  für alle  $\mathbf{x} \in D$  positiv definit ist, dann ist  $f$  strikt konvex auf  $D$ .

**Beweis.** Der Beweis von Aussage a) kann bei [Stein \(2021a, S. 58 f.\)](#) nachgelesen werden. Der Beweis von Aussage b) folgt analog zum Beweis von Aussage a) unter Verwendung des Satzes von Taylor und der  $C^1$ -Charakterisierung von strikter Konvexität.

<sup>2</sup>Zur Erinnerung: Eine symmetrische Matrix  $A \in \mathbb{R}^{n \times n}$  heißt positiv semidefinit, falls  $\mathbf{x}^\top A \mathbf{x} \geq 0$  für alle  $\mathbf{x} \in \mathbb{R}^n$ . Ist diese Ungleichung für alle  $\mathbf{x} \neq 0$  sogar strikt erfüllt, so heißt  $A$  positiv definit.

**Optimalitätsbedingungen**

In der Regel ist man insbesondere im Hinblick auf Anwendungsfragen nicht nur daran interessiert, theoretische Aussagen über die Existenz und Eindeutigkeit von Minimalpunkten zu machen, sondern diese auch zu identifizieren. Dazu können Optimalitätsbedingungen genutzt werden, die notwendige oder hinreichende Bedingungen für die Optimalität eines Punktes  $\mathbf{x} \in \mathcal{F}$  liefern. Optimalitätsbedingungen für unrestringierte Probleme, die auf Eigenschaften der ersten oder zweiten Ableitung beruhen, sind bereits in der Schulmathematik der Oberstufe (für Probleme mit  $f : \mathbb{R} \rightarrow \mathbb{R}$ ) fest verankert.

Wir beschränken uns in diesem Kapitel auf Optimalitätsbedingungen für unrestringierte Probleme. Zwar tauchen im Laufe dieser Arbeit auch restringierte Probleme auf, da wir diese jedoch meist durch Umformulieren in unrestringierte Probleme lösen werden (vgl. Abschn. 11.2.2), spielen Optimalitätsbedingungen für restringierte Probleme nicht direkt eine Rolle.

Voraussetzung für die Anwendung der folgenden Optimalitätsbedingungen ist, dass die betrachteten Zielfunktionen (mehrfach) stetig differenzierbar sind.

**Satz 8** (Notwendige Bedingung erster Ordnung).

*Es sei  $\mathbf{x}^* \in \mathbb{R}^n$  ein lokaler Minimalpunkt der differenzierbaren Funktion  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ . Dann gilt  $\nabla f(\mathbf{x}^*) = 0$ .*

**Beweis.** *Der Beweis ist bei [Stein \(2021b, S. 31 f.\)](#) zu finden.*

Punkte  $\mathbf{x}$ , für die  $\nabla f(\mathbf{x}) = 0$  gilt, werden auch als *kritische Punkte* bezeichnet. Im eindimensionalen Fall ( $f : \mathbb{R} \rightarrow \mathbb{R}$ ) entspricht dies gerade dem in der Schule vielfach eingesetzten Kriterium  $f'(x) = 0$ .

**Satz 9** (Notwendige Bedingung zweiter Ordnung).

*Es sei  $\mathbf{x}^* \in \mathbb{R}^n$  ein lokaler Minimalpunkt der Funktion  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  und  $f$  sei zweimal differenzierbar in  $\mathbf{x}^*$ . Dann gilt  $\nabla f(\mathbf{x}^*) = 0$  und die Hesse-Matrix  $\nabla^2 f(\mathbf{x}^*)$  ist positiv semidefinit.*

**Beweis.** *Der Beweis ist bei [Alt \(2011, S. 43\)](#) zu finden.*

Diese Optimalitätsbedingungen sind zentrale Grundlage für zahlreiche Algorithmen, die zur Lösung von unrestringierten Problemen entwickelt wurden. Auf verschiedene Weise wird versucht kritische Punkte zu finden (vgl. [Boyd und Vandenberghe, 2004, S. 16](#)). Ein Algorithmus, der auf dieser Grundidee basiert, wird in Abschnitt 2.2 aufgegriffen.

Abschließend geben wir noch ein hinreichendes Kriterium für lokale Minimalität an.

**Satz 10** (Hinreichende Bedingung zweiter Ordnung).

*Es sei  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  in  $\mathbf{x}^*$  zweimal differenzierbar. Es gelte  $\nabla f(\mathbf{x}^*) = 0$  und die Hesse-Matrix  $\nabla^2 f(\mathbf{x}^*)$  sei positiv definit. Dann ist  $\mathbf{x}^*$  strikt lokaler Minimalpunkt von  $f$ .*

**Beweis.** *Der Beweis kann bei Stein (2021b, S. 44 f.) eingesehen werden.*

Auch für restringierte Optimierungsprobleme können Optimalitätsbedingungen hergeleitet werden. Die Bedingungen für restringierte Probleme beruhen im wesentlichen auf den sogenannten *Karush-Kuhn-Tucker (KKT)-Bedingungen*. Für weiterführende Informationen seien interessierte Leserinnen und Leser auf Stein (2021a, S. 78 ff.) oder Nocedal und Wright (2006, S. 320 ff.) verwiesen.

### 2.1.2 Welche Arten von Optimierungsproblemen gibt es?

Optimierungsprobleme lassen sich je nach Eigenschaften von Zielfunktion, Entscheidungsvariablen und zulässiger Menge zu Klassen zusammenfassen.

Wirft man einen Blick in Vorlesungsverzeichnisse und Fachliteratur zur Optimierung, so wird schnell ersichtlich, dass die mathematische Optimierung ein enorm breites Feld mit zahlreichen Spezialisierungen und Unterklassen ist. Diese reichen von einer groben Unterscheidung in diskrete und kontinuierliche Optimierungsprobleme bis hin zu einer feinen Unterscheidung in lineare, quadratische, konvexe, nichtlineare, gemischt-ganzzahlige etc. Optimierungsprobleme. Mit dieser Arbeit wird nicht das Ziel verfolgt, sämtliche Teilaspekte und Unterklassen von Optimierungsproblemen detailliert aufzuzeigen und abzugrenzen. Stattdessen werden die Unterscheidungen beschrieben, die zum einen im entwickelten Unterrichtsmaterial eine Rolle spielen und die zum anderen (angehenden) Lehrkräften bei der Betreuung von Modellierungsaktivitäten rund um Optimierungsprobleme Orientierung bieten sollen.

#### Restringierte und unrestringierte Optimierung

Eine grundlegende Unterscheidung kann zwischen Optimierungsproblemen mit Nebenbedingungen (restringierte Probleme) und solchen ohne Nebenbedingungen (unrestringierte Probleme) vorgenommen werden. In Abschnitt 2.1.1 wurde bereits deutlich, dass diese Unterscheidung im Hinblick auf die Optimalitätsbedingungen durchaus zentral ist. Für eindimensionale Probleme würde es, wie in der Schule üblich, sogar ausreichen, die Bedingungen für unrestringierte Probleme anzuwenden und zusätzlich die Randpunkte der zulässigen Menge zu betrachten. Da die Menge der Randpunkte jedoch bei mehrdimensionalen Problemen unendlich sein kann, ist dies nicht mehr praktikabel (vgl. Stein, 2021a, S. 64).

Unrestringierte Optimierungsprobleme treten in vielen Kontexten auf. Probleme ohne Nebenbedingungen treten zum Teil auch beim Lösen von restringierten Problemen auf. Diese werden in unrestringierte Probleme umformuliert, indem die Nebenbedingungen beispielsweise als Strafterme in die Zielfunktion integriert werden. Darauf wird in Abschnitt 2.2 erneut eingegangen. Zudem ist es teilweise möglich Nebenbedingungen wegzulassen, wenn klar ist, dass diese keinen Einfluss auf die Optimalpunkte haben (vgl. Nocedal und Wright, 2006, S. 6).

## Kontinuierliche vs. diskrete Optimierung

Eine wichtige Unterscheidung von Optimierungsproblemen kann hinsichtlich der zulässigen Menge getroffen werden. Ist die Menge der zulässigen Werte endlich (bzw. abzählbar unendlich) so spricht man von einem **diskreten Optimierungsproblem**. Ist die Menge der zulässigen Werte hingegen ein Kontinuum, so spricht man von einem **kontinuierlichen Optimierungsproblem**.

Bei verschiedenen Anwendungsproblemen kann es sinnvoll oder gar notwendig sein, alle oder einzelne Variablen auf ganzzahlige Werte einzuschränken. Diese speziellen diskreten Probleme werden auch als **ganzzahlige Optimierungsprobleme** (engl. *Integer Programs*) bezeichnet. Die Forderung nach Ganzzahligkeit der Variablen kann dabei durch Nebenbedingungen ausgedrückt werden (z. B.  $\mathbf{x} \in \mathbb{Z}$  oder  $\mathbf{x} \in \mathbb{N}$ ). Probleme, die als ganzzahlige Optimierungsprobleme formuliert werden, sind die Standortplanung, beispielsweise von Fabriken oder medizinischen Einrichtungen. Ebenso können sogenannte Zuordnungsprobleme oder Matching-Probleme, die unter anderem bei der Verteilung von Studierenden auf verfügbare Seminarkurse oder bei der Zuweisung von Arbeitskräften zu gewissen Aufgaben auftreten, als ganzzahlige Optimierungsprobleme formuliert werden. Die genannten Beispiele werden auch als **kombinatorische Optimierungsprobleme** bezeichnet. Die kombinatorische Optimierung ist eng verwoben mit der Graphentheorie, der Kombinatorik, dem Operations Research (OR) und der theoretischen Informatik<sup>3</sup> (vgl. [Korte und Vygen, 2018](#), S. XXI).

Einen leicht verständlichen Zugang zu Problemen der diskreten (kombinatorischen) Optimierung und deren Einsatzmöglichkeiten für einen authentischen Mathematikunterricht sind bei [Hußmann und Lutz-Westphal \(2007\)](#) zu finden. Optimierungsprobleme rund um die Standortplanung und deren potentielle Einbindung in den mathematischen Modellierungsunterricht werden bei [Kreckler \(2015\)](#) diskutiert.

In anderen Fällen ist es hingegen sinnvoll, dass die Variablen alle Werte eines Kontinuums annehmen können. Dies trifft beispielsweise auf Probleme zu, bei denen gegebene Messwerte (z. B. eines physikalischen Experiments) bestmöglich durch eine Funktion modelliert werden sollen. Kontinuierliche Optimierungsprobleme werden vielfach in folgender (Standard-)Form geschrieben:

$$\begin{array}{lll} \min_{\mathbf{x}} & f(\mathbf{x}) & \\ \text{u. d. Nb.} & h_i(\mathbf{x}) = 0 & \text{für } i = 1, \dots, p \\ & g_j(\mathbf{x}) \leq 0 & \text{für } j = 1, \dots, q. \end{array} \quad (2.2)$$

Hierbei bezeichnet  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  die Zielfunktion,  $h_i(\mathbf{x}) = 0$  Gleichungsnebenbedingungen und  $g_j(\mathbf{x}) \leq 0$  Ungleichungsnebenbedingungen mit  $h_i : \mathbb{R}^n \rightarrow \mathbb{R}$  und  $g_j : \mathbb{R}^n \rightarrow \mathbb{R}$ . Die zulässige Menge ist gegeben durch  $\mathcal{F} := \{\mathbf{x} \in \mathbb{R}^n \mid h_i(\mathbf{x}) = 0, g_j(\mathbf{x}) \leq 0, \forall i = 1, \dots, p, j = 1, \dots, q\}$  (vgl. [Boyd und Vandenberghe, 2004](#), S. 127).

<sup>3</sup>Die Forschungsbereiche OR und theoretische Informatik beschäftigen sich durchaus intensiv mit Problemstellungen aus Graphentheorie bzw. kombinatorischer Optimierung, jedoch keineswegs ausschließlich. So spielen im OR auch Probleme der kontinuierlichen Optimierung eine zentrale Rolle.

## Lineare, quadratische und nichtlineare Optimierung

Optimierungsprobleme, bei denen sowohl die Zielfunktion als auch sämtliche Nebenbedingungen lineare Funktionen sind, werden als **lineare Optimierungsprobleme** oder als lineare Programme bezeichnet. Ein zentraler Vorteil von linearen Optimierungsproblemen ist, dass diese konvex sind, womit jeder lokale Minimalpunkt einem globalen Minimalpunkt entspricht.

**Nichtlineare Optimierungsprobleme** zeichnen sich dadurch aus, dass mindestens eine der Nebenbedingungen oder die Zielfunktion selbst nichtlineare Funktionen sind. Insbesondere bei ingenieurwissenschaftlichen Fragestellungen und auch in den Naturwissenschaften treten derartige Probleme vermehrt auf. Beispielsweise bei der Simulation und Optimierung solarthermischer Kraftwerke (vgl. Teil III).

Als ein in der Praxis meist gut zu lösender Spezialfall der nichtlinearen Probleme sei die Klasse der konvexen nichtlinearen Optimierungsprobleme benannt. Eine weitere Unterklasse nichtlinearer Probleme, die meist gesondert betrachtet wird, umfasst die **quadratischen Optimierungsprobleme**. Bei diesen ist die Zielfunktion eine quadratische Funktion. Zudem werden sämtliche Nebenbedingungen durch lineare Funktionen induziert (vgl. Nocedal und Wright, 2006, S. 448). Ein wichtiges Beispiel, welches auf einem quadratischen Optimierungsproblem beruht, ist die Methode der kleinsten Quadrate, die vielfach im Bereich der Ausgleichsrechnung zur Anpassung von Messdaten Anwendung findet. Dies wird in Abschnitt 19.1 erneut aufgegriffen.

## Deterministische vs. stochastische Optimierung

In der Realität sind Daten vielfach mit gewissen Unsicherheiten behaftet. Beispielsweise, wenn diese aus physikalischen Experimenten stammen oder wenn die Daten zum Zeitpunkt der Modellierung lediglich geschätzt werden können, weil sie sich auf zukünftige Entwicklungen beziehen. Als Beispiel sei die Produktionsplanung genannt, die insbesondere von dem *zukünftigen Interesse / Bedarf* der Kundschaft abhängt.

Es gibt unterschiedliche Ansätze, um Unsicherheiten bei der Optimierung zu berücksichtigen. Im Bereich der **stochastischen Optimierung** wird versucht, die Unsicherheit zu quantifizieren. Diese Quantifizierungen werden dann genutzt, um einen Punkt zu finden, der den unter Unsicherheiten zu *erwartenden* Ausgang eines Modells optimiert (vgl. Nocedal und Wright, 2006, S. 7). Ein weiteres Forschungsgebiet, welches sich intensiv mit dem Lösen von Optimierungsproblemen unter unsicheren Daten beschäftigt, ist die robuste Optimierung.

Der stochastischen Optimierung steht die **deterministische Optimierung** gegenüber. Bei dieser wird angenommen, dass das Modell vollständig bekannt ist und die Daten keinen Unsicherheiten unterliegen (vgl. Nocedal und Wright, 2006, S. 7). Der Fokus dieser Arbeit liegt auf der deterministischen Optimierung, wengleich in Abschnitt 11.1.5 ein kurzer Exkurs in die stochastische Optimierung vollzogen wird.

## Einkriterielle vs. multikriterielle Optimierung

Bisher haben wir Probleme betrachtet, bei denen *ein* Kriterium zu minimieren war. Diese Probleme werden auch als **einkriterielle** Probleme bezeichnet. In der Praxis treten durchaus Probleme auf bei denen mehrere (meist konkurrierende) Ziele verfolgt werden. Derartige Probleme sind dem Bereich der **multikriteriellen** Optimierung, auch Pareto Optimierung oder Vektoroptimierung genannt, zuzuordnen. Formal kann ein multikriterielles Optimierungsproblem geschrieben werden als

$$\min_{\mathbf{x}} (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x}))^\top \quad \text{u. d. Nb. } \mathbf{x} \in \mathcal{F}, \quad (2.3)$$

wobei Zielfunktionen  $f_l : \mathcal{F} \rightarrow \mathbb{R}$  mit  $l = 1, \dots, m$  und  $m \geq 2$  betrachtet werden. Das  $\min$  in (2.3) bezieht sich dabei auf alle  $m$  Funktionen, die möglichst gleichzeitig minimiert werden sollen. Auch hier muss die Menge  $\mathcal{F}$  nicht notwendigerweise Teilmenge des  $\mathbb{R}^n$  sein.

Oftmals existiert bei multikriteriellen Problemen kein Punkt  $\mathbf{x}$ , der Minimalpunkt jeder einzelnen Zielfunktion ist. Stattdessen muss ein geeigneter Kompromiss gefunden werden. Eine Möglichkeit, um multikriterielle Probleme zu lösen, ist diese über einen Skalarisierungsansatz in ein einkriterielles Problem zu überführen (vgl. [Miettinen, 1998](#), S. 1).

## 2.2 Wie werden Optimierungsprobleme gelöst?

In der Schule, insbesondere in der Oberstufe, werden Optimierungsprobleme (Stichworte: Extremwertaufgaben und Kurvendiskussion) zumeist **analytisch** gelöst. Dazu werden Optimalitätsbedingungen erster und zweiter Ordnung ausgenutzt. Aufgrund der Komplexität von Optimierungsproblemen, die in der Praxis auftreten, ist es jedoch vielfach nicht praktikabel oder möglich Optimierungsprobleme analytisch anzugehen. Deswegen werden Optimierungsverfahren entwickelt mit denen versucht wird eine gute oder gar optimale Lösung zu berechnen.

Den meisten Verfahren ist Folgendes gemein: ausgehend von einem Startpunkt für die Entscheidungsvariable  $\mathbf{x}$  (der entweder von uns gewählt oder aus vorherigen Simulationen und Modellen generiert wird) wird versucht, iterativ neue Werte der Variablen mit verbesserten Zielfunktionswerten zu finden bis das Verfahren aufgrund eines gewählten Abbruchkriteriums endet (vgl. [Nocedal und Wright, 2006](#), S. 9). Bei der Entwicklung von Optimierungsverfahren wird in der Regel versucht folgenden Kriterien zu genügen:

- **Robustheit:** Das Verfahren kann in einem breiten Spektrum von Optimierungsproblemen der gleichen Art angewendet werden und funktioniert für verschiedene (sinnvolle) Wahlen eines Startpunktes zuverlässig (vgl. [Nocedal und Wright, 2006](#), S. 9; [Antoniou und Lu, 2007](#), S. 10).
- **Effizienz:** Die Rechenleistung und der Speicherplatz, die zur Ausführung des Verfahrens benötigt werden, sollten nicht zu groß sein (vgl. [Nocedal und Wright, 2006](#), S. 9).

- **Genauigkeit:** Das Verfahren sollte mit hoher Genauigkeit einen Optimalpunkt finden ohne dabei zu sensitiv auf Fehler zu reagieren; seien es Fehler in den Daten oder Rundungsfehler, die bei der Verarbeitung / Darstellung von Daten oder bei der Umsetzung des Algorithmus mithilfe des Computers entstehen (vgl. Nocedal und Wright, 2006, S. 9).

Durchaus können diese Kriterien miteinander im Konflikt stehen.

Eine Unterscheidung von Optimierungsverfahren lässt sich hinsichtlich dessen vornehmen, welche Informationen über die Zielfunktion und deren erste und zweite Ableitung (sofern diese existieren) ausgenutzt werden:

- **Verfahren erster Ordnung** nutzen den Gradienten der Zielfunktion, um einen Minimalpunkt zu berechnen. Da der Gradient lediglich lokale Eigenschaften der Zielfunktion abbildet, sind die meisten gradientenbasierten Verfahren insbesondere für die Bestimmung von lokalen Minimalpunkten beziehungsweise für konvexe Probleme geeignet. Ein Beispiel für ein Verfahren erster Ordnung ist das Gradientenverfahren, bei dem iterativ in die Richtung des steilsten Abstiegs vorangeschritten wird (vgl. Abschn. 2.2.1).
- **Verfahren zweiter Ordnung** nutzen neben dem Gradienten zusätzlich die Hesse-Matrix. Ein Beispiel ist das Newton-Verfahren, welches eingesetzt werden kann, um die Nullstellen des Gradienten (und damit die kritischen Punkte) zu berechnen.
- **Verfahren nullter Ordnung** nutzen keine Informationen über den Gradienten oder die Hesse-Matrix. Diese Verfahren werden auch als **gradientenfreie Verfahren** bezeichnet. Sie werden unter anderem bei Problemen eingesetzt, bei denen keine analytische Form der Zielfunktion vorliegt, beispielsweise wenn die Auswertung der Zielfunktion auf komplexen Simulationen beruht (vgl. Nocedal und Wright, 2006, S. 221). Selbst wenn eine analytische Form der Zielfunktion vorliegt, ist es nicht immer praktikabel oder möglich einen Gradienten zu berechnen, beispielsweise weil dessen Bestimmung äußerst rechenintensiv wäre oder weil die Zielfunktion nicht überall differenzierbar ist. Gradientenfreie Verfahren werden somit auch zum Lösen von Problemen mit nichtdifferenzierbarer Zielfunktion eingesetzt. Beispiele für solche Verfahren sind das Nelder-Mead-Verfahren, das Simulated Annealing oder genetische Algorithmen. Die gradientenfreien Verfahren sind weniger weit entwickelt als die gradientenbasierten Verfahren und lassen sich vielfach nur bei Problemen mit simplen Nebenbedingungen (bspw. Box-Restriktionen der Art  $l_b \leq x \leq u_b$ ) einsetzen (vgl. Nocedal und Wright, 2006, S. 221).

Je nach mathematischen Eigenschaften des gegebenen Optimierungsproblems eignen sich unterschiedliche Verfahren.

Im Folgenden wird kurz die Grundidee und Funktionsweise von sogenannte Abstiegsverfahren beschrieben. Bei der Beschreibung dieser werden exemplarisch wesentliche Merkmale vieler numerischer Optimierungsverfahren (Startpunkt, Schrittweite, Suchrichtung etc.) deutlich.

## 2.2.1 Abstiegsverfahren

Abstiegsverfahren finden vielfach zur Lösung unrestringierter nichtlinearer Optimierungsprobleme mit stetig differenzierbarer Zielfunktion Anwendung. Die Grundidee ist, ausgehend von einem festzulegenden Startpunkt, gemäß einer festzulegenden Schrittweite, in Richtung kleinerer Zielfunktionswerte voranzuschreiten. Dies wird solange durchgeführt bis ein Abbruchkriterium erfüllt ist. Im Detail werden folgende Schritte ausgeführt (vgl. Alt, 2011, S. 69):

1. Wähle einen Startpunkt  $\mathbf{x}^{(0)}$ , eine Abbruchtoleranz  $\epsilon$ , eine maximale Anzahl an Iterationen  $k_{max}$  und setze  $k := 0$ .
2. Falls  $\|\nabla f(\mathbf{x}^{(0)})\| < \epsilon$  oder  $k > k_{max}$  stoppe das Verfahren.
3. Berechne eine Abstiegsrichtung  $\mathbf{d}^{(k)}$  und eine Schrittweite  $\alpha_k$ , sodass

$$f(\mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}) < f(\mathbf{x}^{(k)})$$

und setze  $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}$

4. Setze  $k := k + 1$  und gehe zu Schritt 2.

Für die Festlegung der Schrittweiten  $\alpha_k$  in den einzelnen Iterationen existieren verschiedene Strategien, wie die inexakte, jedoch effiziente Armijoregel (vgl. Alt, 2011, S. 98). Ein Spezialfall des allgemeinen Abstiegsverfahrens ist das sogenannte **Gradientenverfahren**. Bei diesem wird als Suchrichtung der negative Gradient<sup>4</sup> in der aktuellen Iterierten gewählt:  $\mathbf{d}^{(k)} = -\nabla f(\mathbf{x}^{(k)})$ . Das Gradientenverfahren ist auch als *Verfahren des steilsten Abstiegs* bekannt.

Bei Anwendung des Verfahrens auf ein Problem mit nichtkonvexer Zielfunktion mit vielen lokalen Minimalpunkten, ist es möglich, dass bei ungünstiger Wahl des Startpunktes nur ein lokaler Minimalpunkt (ggf. mit schlechtem Zielfunktionswert) gefunden wird. Wird zudem die Schrittweite bei einer derartigen Funktion ungünstig gewählt, so kann es passieren, dass Minimalpunkte übersprungen werden. Abstiegsverfahren und Probleme, die je nach Gestalt der Zielfunktion auftreten können, scheinen durchaus geeignet, um im ein- oder zweidimensionalen Fall auf anschaulicher Ebene auch mit Schülerinnen und Schülern diskutiert zu werden.

Bei der Modellierung eines solarthermischen Kraftwerkes wurden verschiedene Optimierungsprobleme basierend auf dem Gradientenverfahren gelöst (vgl. Abschn. 11.1). Bei der Modellierung eines Empfehlungssystems für Filme wird in Anhang E.1 zudem eine im Bereich des maschinellen Lernens oft genutzte Erweiterung, das stochastische Gradientenverfahren, aufgegriffen.

<sup>4</sup>Der Gradient in einem Punkt  $\mathbf{x}$  zeigt in die Richtung des steilsten Anstiegs. Deswegen wird beim Lösen von Minimierungsproblemen mit dem Gradientenverfahren der negative Gradient (die Richtung des steilsten Abstiegs) gewählt.

### 2.2.2 Lösen von restringierten Optimierungsproblemen

Zum Lösen von Optimierungsproblemen mit Nebenbedingungen existieren verschiedene Ansätze. Zentral sind ...

- Ansätze, bei denen das restringierte Problem in ein oder sukzessiv in mehrere unrestringierte Probleme überführt wird, welche dann gelöst werden. Verfahren, die auf diesem Ansatz beruhen, sind Straftermverfahren, Barrieremethoden oder augmentierte Lagrangeverfahren. Im Folgenden wird die Idee der Modellierung als unrestringiertes Problem detaillierter ausgeführt, da dies bei der Optimierung von Solarkraftwerken relevant sein wird (vgl. Abschn. 11.2).
- Ansätze, bei denen die Optimalitätsbedingungen für restringierte Probleme (konkret die KKT-Bedingungen) ausgenutzt werden. Dazu gehören beispielsweise Verfahren des *Sequential Quadratic Programming* (vgl. Nocedal und Wright, 2006, S. 423; Stein, 2021b, S. 189).

Die Grundidee der Modellierung als unrestringiertes Problem ist, die Zulässigkeit eines Punktes als weiteres Ziel zu formulieren. Dazu können sogenannte Straftermfunktionen  $\phi : \mathbb{R}^n \rightarrow \mathbb{R}$  definiert werden, für die

- $\phi(\mathbf{x}) = 0$ , falls  $\mathbf{x} \in \mathcal{F}$
- $\phi(\mathbf{x}) > 0$ , falls  $\mathbf{x} \notin \mathcal{F}$ .

gelten soll. Die Forderung nach Zulässigkeit kann als Minimierung der Straftermfunktion  $\phi$  betrachtet werden. Es wird nun versucht, sowohl die ursprüngliche Zielfunktion  $f$  als auch die Straftermfunktion  $\phi$  zu minimieren. Damit liegt ein unrestringiertes multikriterielles Optimierungsproblem der Form

$$\min_{\mathbf{x}} \begin{pmatrix} f(\mathbf{x}) \\ \phi(\mathbf{x}) \end{pmatrix} \quad (2.4)$$

vor. Anstatt das Problem über Methoden der multikriteriellen Optimierung anzugehen, kann dieses über einen Skalarisierungsansatz in ein unrestringiertes einkriterielles Problem mit Strafparameter  $\mu > 0$  überführt werden (vgl. Stein, 2021b, S. 190), etwa

$$\min_{\mathbf{x}} f(\mathbf{x}) + \mu \cdot \phi(\mathbf{x}). \quad (2.5)$$

Nun bleibt die Frage, wie die Straftermfunktion gewählt werden sollte. Wünschenswert wäre, wenn die Minimalpunkte des neuen unrestringierten Problems für ausreichend große Strafparameter  $\mu$  den Minimalpunkten des ursprünglichen restringierten Problems entsprächen. Zwei gängige Straftermfunktionen sind ...

- die  $\ell_1$ -Straftermfunktion

$$\phi_1(x) = \sum_{j=1}^q \max\{0, g_j(\mathbf{x})\} + \sum_{i=1}^p |h_i(\mathbf{x})|.$$

Diese hat den Vorteil, dass sich die Optimalpunkte des restringierten Problems unter gewissen Voraussetzungen und für genügend großen Strafparameter  $\mu$  durch Lösen des unrestringierten Problems ergeben. Deshalb wird diese

Funktion auch als *exakte Straftermfunktion* bezeichnet. Nachteil ist, dass die  $\ell_1$ -Funktion nicht überall differenzierbar ist. Sie wird deswegen oft eingesetzt, wenn bereits die Zielfunktion des ursprünglichen Problems nicht überall differenzierbar ist (vgl. [Nocedal und Wright, 2006](#), S. 507; [Stein, 2021b](#), S. 195).

- die  $\ell_2$ -Straftermfunktion

$$\phi_2(x) = \sum_{j=1}^q (\max\{0, g_j(\mathbf{x})\})^2 + \sum_{i=1}^p (h_i(\mathbf{x}))^2.$$

Diese ist zwar überall differenzierbar, jedoch nicht exakt.

Die meisten Straftermverfahren, die sich der Ankopplung einer Straftermfunktion an die ursprüngliche Zielfunktion zunutze machen, basieren darauf, eine Folge unrestringierter Probleme mit wachsendem Strafparameter zu lösen, und zwar so lange, bis die Verletzung der Nebenbedingungen ausreichend klein ist (vgl. [Nocedal und Wright, 2006](#), S. 511).

## 2.3 Warum Optimierung in der Schule?

Die Leitfrage dieses Abschnitts soll im Hinblick auf die einschlägige fachdidaktische Literatur rund um Optimierungsprobleme diskutiert werden. Es wird deutlich, dass nicht nur im Hinblick auf die mathematische Modellierung, sondern auch darüber hinaus vielfach das Potential hervorgehoben wurde, welches die Optimierung für den Mathematikunterricht bereithält:

**Optimierung als Grundbedürfnis des Menschen:** „Wir wollen das Beste aus dem Tag machen, die Zeit optimal nutzen, das Geld einträglich und sicher anlegen [...] Kurz: Wir wollen maximalen Nutzen mit minimalen Einsatz erzielen und das in möglichst vielen Lebensbereichen und Lebenslagen“ ([Vogel, 2010a](#), S. 4). Die Optimierung liegt in der Natur des Menschen und birgt vermutlich deswegen ein enormes Motivationspotential – auch für den Mathematikunterricht. Die Optimierung unterstreicht damit in besonderem Maße, was [Winter \(1975, S. 59\)](#) bei der Diskussion allgemeiner Lernziele für den Mathematikunterricht zu betonen wusste: Mathematik und Mensch sind sich in ihrem Wesen in verschiedener Hinsicht durchaus ähnlich.

**Optimierung als fundamentale Idee der Mathematik:** Unter anderem [Schupp \(1997\)](#) hebt das Optimieren als *fundamentale Idee* der Mathematik im Sinne des amerikanischen Psychologen Jerome Bruner hervor. [Bruner \(1960\)](#) hatte mit seinem Buch „The Process of Education“ erstmals den Begriff fundamentaler Ideen<sup>5</sup> im Hinblick auf deren Bedeutung für die Curriculumsgestaltung und den Unterricht geprägt. Bruner fordert, den Unterricht eines Faches an dessen fundamentalen Ideen auszurichten und diese spiralförmig auf unterschiedlichem Niveau wiederkehren zu lassen (vgl. [Humenberger und Reichel, 1995](#), S. 2). Aufbauend auf Bruner haben sich verschiedene Mathematikdidaktikerinnen und -didaktiker der Entwicklung von

---

<sup>5</sup>In der fachdidaktischen Literatur werden vielfach auch Synonyme, wie *zentrale*, *universelle*, *bedeutsamen* Ideen, verwendet.

Kriterien und Katalogen für fundamentale Ideen der Mathematik gewidmet. In diversen dieser Kataloge, taucht das *Optimieren*, die *Optimierung* oder *Optimalität* als eine fundamentale Idee explizit auf. Hier sei exemplarisch auf die Ideenkataloge von [Schreiber \(1983\)](#) und [Tietze et al. \(1997\)](#) verwiesen. Eine Übersicht über die Entwicklung fundamentaler Ideen in der mathematikdidaktischen Diskussion ist etwa in [Humenberger und Reichel \(1995\)](#) zu finden. Eben diese Autoren haben zudem einen Katalog von *fundamentalen Ideen für die angewandte Mathematik* erarbeitet, dem auch das Optimieren angehört (vgl. [Humenberger und Reichel, 1995](#), S. 30).

Ebenfalls im Hinblick auf fundamentale Ideen hat [Von der Bank \(2012\)](#) das Vernetzungspotential der Optimierung an verschiedenen Beispielen herausgearbeitet. Dazu hat sie einen sogenannten Vernetzungspentagraph entwickelt, der dabei helfen soll „Unterrichtsinhalte zu analysieren und so einen Beitrag leisten [kann], im Unterricht Aspekte, die für Mathematik ganz wesentlich sind, nicht auszublenden“ ([Von der Bank, 2012](#), S. 37). Die Autorin stellt verschiedene Optimierungsaufgaben vor und veranschaulicht mithilfe des Vernetzungspentagraphen, inwieweit bei der Bearbeitung dieser Aufgaben Inhalte aus Analysis, Algebra, Geometrie, Stochastik und diskreter Mathematik vernetzt werden können.

**Optimieren bietet zahlreiche Anlässe für mathematisches Modellieren:** Viele Anwendungen und Technologien aus unserem Alltag beruhen letztlich auf dem Formulieren und Lösen von Optimierungsproblemen. Sei es, wenn die optimale Bestrahlung von Tumoren schon vor der eigentlichen Strahlentherapie simuliert und optimiert wird, wenn die Flügel eines Flugzeuges mit Blick auf die Aerodynamik optimal designt werden oder Gesichter von Menschen in der automatischen Gesichtserkennung bestmöglich erkannt werden sollen. Optimierungsprobleme haben eine enorme Relevanz für Fragestellungen aus Alltag und Forschung und stellen damit einen vielversprechenden Kandidaten für die mathematische Modellierung mit Schülerinnen und Schülern dar. Die Chancen, die die Optimierung für den stärkeren Einbezug von Anwendungsfragen und damit das mathematische Modellieren bietet, wurde bereits von verschiedenen Autorinnen und Autoren hervorgehoben (vgl. [Kreckler, 2015](#); [Lutz-Westphal, 2006](#); [Oldenburg, 2006](#); [Vogel, 2010a](#); [Humenberger und Reichel, 1995](#)).

Nicht nur die Anlässe für das Modellieren, sondern auch die Vielfalt an Möglichkeiten Optimierungsprobleme zu formulieren und Strategien um diese zu lösen, sprechen für eine stärkere Einbindung von Optimierungsfragen in den Unterricht. Die verschiedenen Stellen, an denen das Formulieren und Lösen von Optimierungsproblemen Raum bietet mathematisch kreativ an das Lösen realer Probleme heranzugehen, werden in Abschnitt [19.1](#) diskutiert.

[Vogel \(2010a\)](#) nennt als weitere Vorteile, die die Optimierung für den Mathematikunterricht haben kann, dass sie a) Möglichkeiten zur Förderung **algorithmischen Denkens** bietet und b) den **Weg zu tragfähigen Begriffen** weisen kann. Er bezieht sich dabei auf den Median und das arithmetische Mittel, die aus der Lösung von Optimierungsproblemen resultieren. Hierauf wird in Abschnitt [19.1](#) erneut eingegangen.

### Wunsch und Wirklichkeit

Obgleich die Optimierung als fundamentale Idee der (angewandten) Mathematik angesehen wird, findet sie in der Schule zumeist nur im Bereich Analysis (unter anderem bei quadratischen Funktionen / Parabeln) und in der Oberstufe vielfach reduziert auf das Lösen von Extremwertaufgaben sowie das mitunter rezeptartige Anwenden des Kriteriums *Ableitung gleich Null* statt. Dabei bietet die Optimierung deutlich mehr Möglichkeiten den Mathematikunterricht zu bereichern und wahrhaftig das Potential auszuschöpfen, welches ihr als fundamentale Idee innewohnt (Schupp, 1997, S. 4).

Ebenso Oldenburg (2006) diskutiert, dass das Einschränken der Optimierung auf die Bestimmung der Ableitung und das Nullsetzen dieser, den Anwendungsbereich der Optimierung enorm reduziert. Denn dies verlangt, dass die betrachtete Zielfunktion analytisch darstellbar und mindestens einmal differenzierbar ist. Dies ist in der Realität und auch bei den in dieser Arbeit betrachteten Problemen nicht immer der Fall. Er schlägt vor, Optimierungsprobleme mithilfe numerischer Verfahren mit den Lernenden anzugehen und nennt zugleich verschiedene Beispielprobleme und Verfahren (bspw. das Newton-Verfahren oder die Nutzung von Straftermverfahren für restringierte Probleme), die sich für den Unterricht eignen könnten. Er richtet sich in seinen Ausführungen an Lehrkräfte und diskutiert verschiedene Probleme, die für den Unterricht denkbar wären. Vorschläge für konkrete Unterrichtsmaterialien und deren Erprobung sind nicht Bestandteil des Artikels. Auch wird die diskrete Optimierung ausgeklammert. Zudem wird nicht detailliert auf zentrale Eigenschaften von Optimierungsproblemen, der damit verbundenen Existenz und Eindeutigkeit von Optimalpunkten sowie das Wechselspiel zwischen innermathematischen Überlegungen und Anwendungsfragen eingegangen.

Unter anderem Schupp (1997, S. 4) und Humenberger (2010, S. 44) fordern das Optimieren schon in der Sekundarstufe I stärker in den Mathematikunterricht einzubinden: „Daraus ergibt sich: die Begegnung mit dem Optimieren als zentraler und aktueller mathematischer Aktivität erfolgt in unseren Schulen zu spät (für viele Schüler überhaupt nicht) und wenig repräsentativ“ (Schupp, 1997, S. 4). Schupp liefert neben dieser Kritik zugleich inhaltliche, didaktische und unterrichtspraktische Anregungen, um das benannte Defizit abzubauen. Als Leitlinien für das verstärkte Einbinden der Optimierung in den Mathematikunterricht benennt er das Optimieren durch Finitisieren (lineare Optimierung, Approximation der Zielfunktion durch Verwendung von Wertetabellen, Ablesen von Extrema am Funktionsgraphen), das Optimieren in endlichen Mengen (bspw. bereits in der Primarstufe, bei der Bestimmung des ggT / kgV), die Verwendung von Mitteln aus Graphentheorie / Kombinatorik, Analysis oder linearer Algebra, die Optimierung über die Verwendung von Niveaulinien oder den Rückbezug auf Basisungleichungen (vgl. Schupp, 1997, S. 8–10). Auch Danckwerts und Vogel (2001), Vogel (2010a) und Humenberger (2015a) heben die Möglichkeiten des Optimierens mithilfe elementarer Methoden an verschiedenen Beispielen hervor und unterstreichen, dass es nicht der Differentialrechnung bedarf, um Optimierungsprobleme zu verschiedenen Zeitpunkten der schulmathematischen Laufbahn zu verankern.

Im Hinblick auf die diskrete Optimierung und Potentiale anhand dieser den Mathematikunterricht zu bereichern, sei auf [Hußmann und Lutz-Westphal \(2007\)](#) und [Lutz-Westphal \(2006\)](#) hingewiesen, die zahlreiche Impulse liefern, wie die kombinatorische Optimierung im Mathematikunterricht verankert werden kann. [Lutz-Westphal \(2006\)](#) hebt dabei insbesondere die Chancen für die authentische Gestaltung von Mathematikunterricht hervor. Sie diskutiert, dass sich kombinatorische Optimierungsprobleme durch eine besonders leichte Zugänglichkeit auszeichnen und einen modularen Einsatz des Stoffes im Unterricht erlauben:

„Im Falle der kombinatorischen Optimierung ist besonders die Modularität des Stoffes hervorzuheben. Der Stoff ist nicht hierarchisch gegliedert, sondern bietet die Möglichkeit in verschiedene Richtungen ausgekundschaftet zu werden. [...] Diese Offenheit im Erkundungsprozess ermöglicht es, über mehrere Schulstunden hinweg freies Forschen zuzulassen, um dann die Erkundungswege zusammenzutragen, zu vergleichen und jeweils noch nicht Bearbeitetes von anderen zu lernen“ ([Lutz-Westphal, 2006, S. 69](#)).

Zudem diskutiert [Fest \(2007\)](#), dass sich die kombinatorische Optimierung besonders dafür eignet, greifbare Probleme aus dem Alltag der Lernenden im Mathematikunterricht zu behandeln.

Im Hinblick auf die Umsetzung mathematischer Modellierungsprojekte, die über das *Nullsetzen der Ableitung* hinausgehen, sei überdies auf die Arbeit von [Kreckler \(2015\)](#) hingewiesen. Die Autorin arbeitete in ihrer Dissertation umfänglich heraus, wie Standortprobleme im mathematischen Modellierungsunterricht mit Schülerinnen und Schülern basierend auf Methoden der diskreten oder geometrischen Optimierung angegangen werden können.

### **Konsequenzen für diese Arbeit**

Die Forderung, Optimierung im Schulunterricht früher, häufiger und inhaltlich breiter zu behandeln, ist, wie die oben zitierten Arbeiten zeigen, nicht neu. Es existieren bereits diverse Publikationen, die sich dem Thema Optimierung und deren Chancen für den Mathematikunterricht nähern. Die Mehrheit der Publikationen fokussiert einen ausgewählten innermathematischen Bereich der Optimierung oder eine beziehungsweise mehrere ausgewählte Methoden und beschreibt Möglichkeiten diesen Aspekt / diese Methode im Mathematikunterricht zu verankern.

In der vorliegenden Arbeit wird hingegen nicht nur eine ausgewählte Klasse von Optimierungsproblemen in den Blick genommen. Stattdessen werden ausgehend von realen Problemstellungen wesentliche Aspekte von Optimierungsproblemen mathematisch untersucht. Dabei werden übergeordnete Leitfragen beim Formulieren und Lösen von Optimierungsproblemen diskutiert, die sich nicht auf eine Art von Optimierungsproblemen oder eine ausgewählte Methode beschränken. Vielmehr steht das Wechselspiel zwischen Anwendungsfragen und innermathematischen Überlegungen im Hinblick auf Eigenschaften der Optimierungsprobleme im Vordergrund. Der Facettenreichtum der Optimierung und diverser denkbarer Modellentscheidungen (auch seitens der Lernenden) werden in dieser Arbeit umfänglich ausgeschöpft. So

wird beispielsweise ein- und dieselbe Optimierungsfrage im Lernmaterial zu den Solarkraftwerken über drei verschiedene Formulierungen (sowohl diskrete als auch kontinuierliche) angegangen und gelöst.

In dieser Arbeit werden Optimierungsprobleme in den Blick genommen, die aufgrund ihrer Komplexität den Einsatz von numerischen Methoden zum Lösen der Probleme erfordern. Damit wird sich in natürlicher Weise von dem in der Schule gängigen Vorgehen gelöst – und damit sogleich der von [Schupp \(1997, S. 12\)](#) zurecht angemerkten Kritik entgegengewirkt:

„Die von unseren Analysisbüchern vermittelte Exaktheit des Infinitesimalkalküls ist eine Schimäre. Sie lebt davon, dass man sich auf wenige Funktionsklassen beschränkt, bei denen  $f'(x) = 0$  exakt lösbar ist. Hinzu kommt, dass für den Anwender die Exaktheit nicht unbedingt erforderlich ist [...].“

## **Teil II**

# **Realisierung von kollaborativen Modellierungsprojekten online**



## 3 Einleitung

Der in diesem Kapitel vorgestellte ganzheitliche Ansatz für die Online-Durchführung von kollaborativen Modellierungsaktivitäten wurde im Rahmen des Computational and Mathematical Modeling Programs (CAMMP)<sup>1</sup> entwickelt. CAMMP ist ein Schülerlabor an der RWTH Aachen und ein Programm am Karlsruher Institut für Technologie, welches seit 2011 Modellierungsaktivitäten für Schülerinnen und Schüler der Klassenstufen 8 bis 13 durchführt. Die in dieser Arbeit entwickelten Lernmaterialien wurden im Rahmen zweier Arten von Modellierungsaktivitäten erprobt: a) solche mit kurzem Zeitrahmen, die Modellierungstage,<sup>2</sup> und b) solche mit großzügigem Zeitrahmen, die Modellierungswochen.<sup>3</sup> Das Konzept und Lernmaterial für die Modellierungstage kann und wurde bereits für die Durchführung von Unterrichtsreihen angepasst.

Neben den Modellierungstagen und -wochen, die im Zuge von CAMMP durchgeführt werden, gibt es in Deutschland eine Reihe weiterer, von Universitäten organisierter Modellierungsaktivitäten. Zwei Beispiele sind die Hamburger Modellierungstage und die im Rahmen des Projekts KOMMS in Kaiserslautern durchgeführten Modellierungstage und -wochen (vgl. [Vorhoelter und Freiwald, im Druck](#); [Bock und Bracke, 2015](#)).

### Wieso online?

Meist findet der größte Teil des Lernens und Lehrens von Mathematik in der Schule statt oder, wie im Rahmen dieser Arbeit, in Modellierungsveranstaltungen für Schülerinnen und Schüler, die von wissenschaftlichen Mitarbeitenden oder studentischen Hilfskräften organisiert und durchgeführt werden – beispielsweise im Rahmen eines Schülerlabors. Gerade in Zeiten von Distanzunterricht und Homeschooling ist es wichtig geworden, Lernenden die Möglichkeit zu geben, die in Abschnitt 1.2 erwähnten Modellierungskompetenz(en) durch die Arbeit an Problemen *von zu Hause aus* zu erwerben. Daher wurde ein ganzheitliches und anpassungsfähiges Konzept erprobt, um mathematisches Modellieren online zu ermöglichen. Der Anlass für die Entwicklung eines Konzepts zur Durchführung mathematischer Modellierungsaktivitäten online war die COVID-19-Pandemie. Gleichwohl wurde das entwickelte Lernmaterial bereits vor der Pandemie für den Einsatz in Präsenzworkshops *digital* umgesetzt. Dies liegt in den Vorteilen digitaler Werkzeuge für das Modellieren begründet (vgl. Abschn. 1.4). Die Potentiale, die digitale Lernumgebungen als Ergänzung zum traditionellen Präsenzunterricht mit sich bringen, hebt auch die [Kultusministerkonferenz \(2017, S. 13–14\)](#) hervor:

<sup>1</sup>Weitere Informationen unter: [www.cammp.online](http://www.cammp.online), letzter Aufruf: 31.10.2021.

<sup>2</sup>CAMMP days, siehe [www.cammp.online/116.php](http://www.cammp.online/116.php), letzter Aufruf: 26.11.2021.

<sup>3</sup>CAMMP week, siehe [www.cammp.online/21.php](http://www.cammp.online/21.php), letzter Aufruf: 26.11.2021.

„Diese digitalen Lernumgebungen helfen Schülerinnen und Schülern, sich im Team zu organisieren, gemeinsam Lösungen zu entwickeln, selbstständig Hilfen heranzuziehen und ermöglichen unmittelbare Rückmeldungen. Sie vereinfachen die Organisation und Kommunikation von Arbeitsprozessen und helfen dabei, dass Arbeitsmaterialien und Zwischenstände jederzeit dokumentiert und verfügbar sind. Zusätzlich zum regulären Lernen im Klassenverband kann der virtuelle Lern- und Arbeitsraum aufgrund seiner Unabhängigkeit von festgesetzter Zeittaktung und physischer Anwesenheit Lernsituationen zwischen verschiedenen Lerngruppen innerhalb einer Schule oder auch zwischen verschiedenen Schulen sowie in außerunterrichtlichen Kontexten vereinfacht ermöglichen. Insgesamt bietet sich die Chance, den Schülerinnen und Schülern mehr Verantwortung für die Gestaltung des eigenen Lernens zu übertragen und damit ihre Selbstständigkeit zu fördern.“

#### **Wieso kollaborativ?**

Aufgrund der komplexen Natur realer Probleme aus unserem Alltag spielt Teamwork typischerweise eine wichtige Rolle bei Problemlösungsprozessen in der realen Welt. Daher ist Teamarbeit und damit kollaboratives<sup>4</sup> Modellieren auch bei dem vorgestellten Konzept für Online-Modellierungsaktivitäten zentral. Die Nähe zu realen Problemlösungsprozessen ist natürlich nicht der einzige Vorteil von kollaborativem oder kooperativem Lernen.<sup>5</sup> Es kann auch dazu beitragen, das Selbstwertgefühl der Lernenden zu stärken und sie zu motivieren, sich aktiv am Lernprozess zu beteiligen (vgl. Panitz, 1999b, S. 59). Darüber hinaus hat „kollaboratives Lernen als Hauptmerkmal eine Struktur, die das Gespräch der Schülerinnen und Schüler ermöglicht: Die Schülerinnen und Schüler sollen miteinander reden und es ist in diesem Gespräch, in dem ein Großteil des Lernens stattfindet“ (Golub, 1988, S. 1, eigene Übersetzung). Wie im Folgenden ersichtlich wird, ist Kommunikation zwischen den Lernenden auch in den hier beschriebenen Modellierungsaktivitäten zentral. In Panitz (1999b) und Johnson und Johnson (1989) werden einige weitere Vorteile des kooperativen und kollaborativen Lernens aufgezeigt.

Im nächsten Kapitel werden zunächst die beiden Unterrichtsformate vorgestellt:

- Projekte mit kürzerem Zeitrahmen (Modellierungstage / Unterrichtsreihen), bei denen die Lernenden stärker angeleitet an didaktisch aufbereiteten Lernmaterialien arbeiten, und

---

<sup>4</sup>Der Begriff kollaborativ bzw. Kollaboration wird im Bereich Lernmethoden / Lernpsychologie wertfrei verwendet. Hier wird sich klar von der negativen Besetzung des Begriffs „Zusammenarbeit mit dem Feind“, der insbesondere durch die NS-Zeit geprägt wurde, abgegrenzt.

<sup>5</sup>Die Begriffe kollaboratives und kooperatives Lernen werden oft austauschbar verwendet. Panitz (1999a) und Bruffee (1995) unterscheiden jedoch zwischen beiden Begriffen. Nach Bruffee (1995) besteht ein Unterschied darin, dass kollaboratives Lernen eher schülerzentriert ist, während kooperatives Lernen eher lehrerzentriert ist und somit stärker von der Lehrkraft kontrolliert wird (vgl. Panitz, 1999a, S. 5). Bei der Verwendung der Begriffe kollaboratives und kooperatives Lernen wird sich in dieser Arbeit auf die Definition von Bruffee (1995) bezogen.

- 
- Projekte mit längerem Zeitrahmen (Modellierungswochen), in denen die Lernenden offene Probleme aus Forschung und Industrie selbstbestimmt und unangeleitet bearbeiten.

Es wird beschrieben, wie die beiden Unterrichtsformate in Präsenz gestaltet und durchgeführt werden. Anschließend werden Herausforderungen für die Online-Realisierung eben dieser Unterrichtsformate benannt und Anforderungen an digitale Werkzeuge herausgestellt. Es wird begründet, welche digitalen Werkzeuge für die Gestaltung und Durchführung der Modellierungsaktivitäten eingesetzt wurden. Anschließend werden das entwickelte digitale Lernmaterial und dessen Ziele, Aufbau und wesentlichen Elemente beschrieben.

# 4 Die Unterrichtsformate

## 4.1 Modellierungstage – Projekte mit kurzem Zeitrahmen

Im Rahmen eines Modellierungstages beschäftigen sich die Lernenden mit der Lösung von realen Problemen aus Alltag, Technik, Wirtschaft oder Wissenschaft (vgl. Frank et al., 2018, S. 138; Wohak et al., 2021). Drei von vielen Problemstellungen, die im Rahmen der Modellierungstage von CAMMP zum Einsatz kommen, sind:

- Wie können Gesichter oder andere Objekte auf Bildern automatisch klassifiziert werden? Die Lernenden entwickeln ein Klassifikationsmodell auf der Grundlage eines Datensatzes mit zwei- oder dreidimensionalen Daten. Anschließend wenden sie dieses auf höherdimensionale Daten an, um handgeschriebene Ziffern oder Gesichter von Menschen zu klassifizieren (vgl. Schmidt, 2019; Schönbrodt, 2019).
- Wie können wir Nutzerinnen und Nutzern einer Streaming-Plattform bestmögliche Empfehlungen für neue Filme, Serien etc. aussprechen (vgl. Teil IV)?
- Wie sollte ein solarthermisches Kraftwerk gebaut werden, damit der Energieumsatz maximal wird (vgl. Teil III)?

Während eines Modellierungstages wird der Modellierungsprozess durch digitales Lernmaterial mit Arbeitsblättern und verschiedenen Differenzierungsmaterialien angeleitet. In Kapitel 6 wird der Aufbau des digitalen Materials im Detail beschrieben. Die Modellierungstage von CAMMP finden (bei Durchführung in Präsenz) meist mit ganzen Mathematikkursen und damit durchaus heterogenen Lerngruppen ab Klasse 8 statt (vgl. Frank et al., 2018; Wohak et al., 2021; Gerhard et al., im Druck).

### Hauptelemente bei der Durchführung in Präsenz

Bei der Durchführung der Modellierungstage in Präsenz befinden sich Lehrkraft<sup>1</sup> und Lernende im selben Raum. Zu Beginn gibt die Lehrkraft mithilfe von **Präsentationsfolien** eine Einführung in das Problem. Dabei werden die Lernenden durch **Diskussionsfragen** einbezogen. Nach der Einführung beginnen die Lernenden in Zweier- oder Dreiergruppen mithilfe eines Computers / Laptops / Tablets und vorbereiteten **digitalen Arbeitsblättern** an dem Problem zu arbeiten. Während der

<sup>1</sup>Die meisten Modellierungsaktivitäten wurden von wissenschaftlichen Mitarbeitenden angeleitet. Eine erste Durchführung durch eine Gymnasiallehrkraft fand 2020 statt.

Gruppenarbeit findet normalerweise ein reger **Austausch** zwischen den Lernenden statt. Die digitalen Arbeitsblätter führen die Lernenden durch den Modellierungsprozess. Je nach Problemstellung spielen **Skizzen**, die die Lernenden zur (gegenseitigen) Erklärung anfertigen, eine wichtige Rolle. Kommunikation, aber auch gemeinsame schriftliche Dokumentationen sind in den Arbeitsphasen zentral.

Kommen die Lernenden nicht weiter, können sie auf **gestufte Hilfekarten** zugreifen, die im digitalen Lernmaterial bereitgestellt werden. Wenn sie trotz Hilfekarten Schwierigkeiten haben, können sie die **Lehrkraft um Rat fragen**. Die Schwierigkeiten der Lernenden sind in der Regel durch kleine Hinweise, eine kurze Diskussion oder eine Skizze der Lehrkraft lösbar. Stellt die Lehrkraft fest, dass ein Problem in mehreren Gruppen auftritt, kann er/sie dieses spontan im Plenum besprechen. Dadurch, dass sich alle Lernenden im selben Raum befinden, kann die Lehrkraft den individuellen Fortschritt der Gruppen jederzeit einsehen und sich einen Überblick über Arbeitsstand und Motivation der Lernenden verschaffen. Bei beobachteten Schwierigkeiten kann die Lehrkraft von sich aus intervenieren und Unterstützung anbieten. Die Lehrkraft verfolgt bei den Interventionen das **Prinzip der minimale Hilfe** im Sinne von [Aebli \(2006\)](#).<sup>2</sup> Um der Heterogenität innerhalb einer Lerngruppe gerecht zu werden, können schnellere Lernende an vertiefenden Aufgaben oder offenen Teilproblemen arbeiten (vgl. Abschn. 6.3). Zudem werden im Lernmaterial unterschiedliche Arten von Lernhilfen gemäß der Taxonomie von [Zech \(1998\)](#) aufgegriffen, die auch als Weiterentwicklung des Prinzips von Aebli angesehen werden kann (vgl. [Stender, 2018](#), S. 104). Zech unterscheidet in seiner Taxonomie „Motivationshilfen“, „Rückmeldungshilfen“, „allgemein-strategische Hilfen“, „inhaltsorientierte strategische Hilfen“ und „inhaltliche Hilfen“ ([Zech, 1998](#), S. 315–319), die hierarchisch aufeinander aufbauen. Im Lernmaterial werden insbesondere die Rückmeldungshilfen realisiert. Zudem greifen die gestuften Hilfekarten unterschiedliche Ebenen von Zechs Taxonomie auf. Die Elemente, die im Lernmaterial umgesetzt werden, um die Heterogenität innerhalb einer Lerngruppe zu berücksichtigen (z. B. gestufte Hilfen, automatische Lösungskontrolle), werden in Abschnitt 6.2 ausführlich diskutiert.

Zu verschiedenen Zeitpunkten des Modellierungstages werden **Zwischendiskussionen im Plenum** abgehalten, um bisherige Ergebnisse zu vergleichen und zu validieren oder um weitere Fragen und Ideen der Lernenden zu diskutieren. In den Plenumsdiskussionen verwendet die Lehrkraft erneut Präsentationsfolien. Zudem kann die **Tafel** als weiteres Medium hilfreich sein, damit die Lernenden Ideen visualisieren und mit ihren Mitlernenden teilen können. Eine **Plenumsdiskussion**, die von der Lehrkraft moderiert wird, beendet den Modellierungstag. Der gesamte Modellierungsprozess wird zusammengefasst, die erzielten Ergebnisse werden interpretiert

---

<sup>2</sup>[Aebli \(2006, S. 300\)](#) versteht unter dem Prinzip der minimalen Hilfe, dass die Lehrkraft „dem selbstständigen Nachdenken der Schüler solange seinen Lauf läßt, als sie auf dem Wege der Lösung des Problems weiterkommen. Aber auch wenn sie Hilfe brauchen, interveniert er nicht sofort auf massive Weise.“ Die Lehrkraft bietet stattdessen die minimale Unterstützung und Anleitung, die die Lernenden zur Lösung einer Aufgabe benötigen. Aebli begründet dies dadurch, dass es unbefriedigend ist mehr Hilfe zu erhalten, als man eigentlich braucht. „Der Schüler findet, man nehme ihn nicht für voll [...] und leistet daher weniger, als er könnte“ ([Aebli, 2006, S. 300](#)).

/ validiert und weitere Modellverbesserungen diskutiert (vgl. Wohak et al., 2021; Schönbrodt et al., im Druck).

### 4.2 Modellierungswochen – Projekte mit großzügigem Zeitrahmen

Während einer Modellierungswoche arbeiten die Lernenden in kleinen Teams (5–6 Lernende) an realen, offenen Problemen aus Forschung, Wirtschaft oder Alltag. Jede Lerngruppe wird von einem/einer wissenschaftlichen Mitarbeitenden betreut. Die Probleme sind insofern offen beziehungsweise ungelöst, dass Personen aus dem jeweiligen Anwendungsfeld selbst noch an diesem Problem arbeiten. Außer einer Problembeschreibung und gegebenenfalls einem (aufbereiteten) Datensatz erhalten die Schülerinnen und Schüler keine Lernmaterialien. Sie strukturieren und gestalten den Modellierungsprozess selbständig. Zwei Beispiele für Probleme, die die Lernenden bereits während der halbjährlich im Rahmen von CAMMP durchgeführten Modellierungswochen bearbeitet haben, sind:

- Wie sollten Windturbinen in einem Offshore-Windpark positioniert werden, damit sie sich nicht gegenseitig abschatten, aber gleichzeitig die verfügbare Fläche optimal ausgenutzt wird (vgl. Abschn. 19.2)?
- Wie kann die Patienten-Raum-Zuteilung in Krankenhausstationen optimiert werden?

Die Modellierungswochen finden mit Lernenden statt, die sich explizit für die Teilnahme bewerben. Damit kann die teilnehmende Schülerschaft durchaus als mathematisch interessiert beziehungsweise begabt eingeschätzt werden.

#### Hauptelemente bei der Durchführung in Präsenz

Ein beispielhafter Ablauf einer Modellierungswoche ist in Abbildung 4.1 dargestellt. Zu Beginn der Woche hält die Betreuungsperson eine kurze **Präsentation** zu den noch ungelösten Problemen. Danach beginnt die Arbeitsphase in Gruppen. Dazu hat jede Gruppe ihren **eigenen Gruppenraum**<sup>3</sup> und es stehen mehrere **Laptops** pro Gruppe zur Verfügung. Während der Gruppenarbeit visualisieren die Lernenden ihre Ideen und strukturieren anstehende Aufgaben auf **Flipcharts**. In der Regel sitzen mehrere Lernende um einen Laptop zusammen und **implementieren gemeinsam** ein Programm. Während der Arbeitsphasen ist oft zu beobachten, dass sich die Lernenden in **kleinere Untergruppen** aufteilen, um an verschiedenen Teilproblemen zu arbeiten. Immer wieder gibt es Phasen, in denen sie sich in der Gesamtgruppe sammeln, um den aktuellen Fortschritt zu besprechen.

Die betreuende Person der Gruppe begleitet den gesamten Arbeitsprozess und hilft

---

<sup>3</sup>Die Modellierungswochen finden meist in einer Jugendherberge oder in Seminarräumen der Universität (KIT oder RWTH Aachen) statt.

## Programm

### Sonntag

bis 17:30	Ankunft an der Jugendherberge
17:45	Willkommen & Kennenlernspiel
18:30 - 19:30	Abendessen
19:30	Vortrag zur mathematischen Modellierung

### Montag

08:00 - 09:00	Frühstück
09:00 - 10:00	Vorstellung der Problemstellungen
10:30 - 12:30	Gruppenarbeit
12:30 - 14:00	Mittagessen
14:00 - 18:00	Gruppenarbeit
18:30 - 19:30	Abendessen
19:30 Uhr	MATLAB - Crash-Kurs (optional)

### Dienstag

08:00 - 09:00	Frühstück
09:00 - 12:30	Gruppenarbeit
12:30 - 14:00	Mittagessen
14:00 - 18:00	Gruppenarbeit
18:30 - 19:30	Abendessen
19:30 Uhr	Betreuer:innentreffen

### Mittwoch

08:00 - 09:00	Frühstück
09:00 - 12:30	Gruppenarbeit
12:30 - 13:45	Mittagessen
14:00 - 14:45	Studieninfo
14:45 - 17:45	Sports Game
17:45	Grillen
19:00	Gäste besuchen die Gruppen

### Donnerstag

08:00 - 09:00	Frühstück
09:00 - 12:30	Gruppenarbeit
12:30 - 14:00	Mittagessen
14:00 - 15:00	Gruppenarbeit
15:00 - 18:00	Probenvorträge & Deadline für den Gruppenbericht
18:30 - 19:30 Uhr	Abendessen
19:30 Uhr	Betreuer:innentreffen

### Freitag

08:00 - 09:00	Frühstück
09:00 - 09:30	Evaluation
11:15 - 16:30	Präsentationen der Schüler:innen & Abschied

Abb. 4.1: Beispielhaftes Programm einer Modellierungswoche

der Gruppe, falls nötig, ihren Arbeitsfortschritt zu strukturieren, Ergebnisse zu diskutieren und die Zeit im Auge zu behalten. Da Programmierkenntnisse keine Voraussetzung für die Teilnahme an einer Modellierungswoche sind, hilft die betreuende Person bei Bedarf mit Hinweisen zur Entwicklung eines Programms. Vielfach wurde während der Modellierungswochen die Programmierumgebung Matlab verwendet. Prinzipiell ist jedoch jede andere Sprache mit ähnlich einfacher Syntax (bspw. Julia, Python oder R) denkbar. Insgesamt wird bei der Betreuung das **Prinzip der minimalen Hilfe** im Sinne von Aebli (2006) verfolgt. Am Ende der Modellierungswoche **präsentieren** die Lernenden ihre Ergebnisse vor Publikum (wissenschaftliches Personal, Problemstellende, Eltern, Lehrkräfte etc.) und geben einen **Bericht** ab, der ihren Modellierungsprozess dokumentiert.

Wie aus den vorherigen Ausführungen ersichtlich wurde, durchlaufen die Lernenden während der Modellierungswoche verschiedene Phasen. Diese lassen sich den von Gudjons (2014, S. 79–92) beschriebenen Schritten und Merkmalen eines handlungsorientierten Projektunterrichts zuordnen (vgl. Roeckerath, 2012, S. 13–14):

### 1. Schritt: Auswahl einer geeigneten, problemhaltigen Sachlage.

Die Sachlage soll so ausgewählt werden, dass sie nicht zu weit von bisherigen Erfahrungen der Lernenden entfernt ist, aber dennoch ein echtes Problem / eine echte Herausforderung für diese darstellt (Situationsbezug). Zudem soll sie Bezug zum realen Leben haben, sich an den Interessen der Lernenden orientieren und gesellschaftliche Praxisrelevanz aufweisen (vgl. Gudjons, 2014, S. 79–82). Die Probleme der Modellierungswochen stammen aus der aktuellen Forschung von Firmen oder Instituten, die an praxisrelevanten Fragestellungen arbeiten. Die Auswahl der Probleme erfolgt durch die Organisierenden der Modellierungswoche unter anderem im

Hinblick auf Relevanz (auch für die Lernenden) und inner- sowie außermathematische Zugänglichkeit und Reichhaltigkeit.

### **2. Schritt: Gemeinsame Entwicklung eines Plans zur Problemlösung.**

Zielgerichtet werden einzelne Arbeitsschritte, die Verteilung von Aufgaben, der zeitliche Verlauf, die Erstellung der Endprodukte (im Falle der Modellierungswoche der Bericht und die Präsentation) sowie die Auswertung geplant (vgl. Gudjons, 2014, S. 83 f.). Zentral ist dabei die Selbstorganisation und -verantwortung der Lernenden. Dies wird im Rahmen der Modellierungswochen insbesondere durch das Prinzip der minimalen Hilfe realisiert.

### **3. Schritt: Handlungsorientierte Auseinandersetzung mit dem Problem.**

In diesem Schritt sind soziale Lernprozesse, im Sinne eines kommunikativen, rücksichtsvollen und interagierenden Arbeitens und eines Lernens von- und miteinander, zentral (vgl. Gudjons, 2014, S. 85 f.). Dies wird in den Modellierungswochen in besonderem Maße ermöglicht. Auch ist in diesem Schritt der Einbezug vieler Sinne wünschenswert, was in den Modellierungswochen nicht umgesetzt wird.

### **4. Schritt: Überprüfung der erarbeiteten Problemlösung an der Realität.**

Merkmal dieses Schrittes ist die verstärkte Produktorientierung. Die Erstellung eines Produkts bietet die Möglichkeit, die erarbeitete Lösung an der Wirklichkeit zu überprüfen und zu reflektieren, inwieweit initiale Ziele erreicht wurden (vgl. Gudjons, 2014, S. 86 f.). Wesentlich dabei ist, dass die Ergebnisse auch Personen außerhalb der Lerngruppe zugänglich gemacht werden. Im Rahmen der Modellierungswochen wird die Produktorientierung durch den Bericht, der Erkenntnisse und Ergebnisse dokumentiert, sowie durch die Präsentationen, die vor Öffentlichkeit stattfinden, realisiert.

Zudem betont Gudjons die **Interdisziplinarität** bei der Durchführung von Projektunterricht. Diese ist bei den für die Modellierungswochen ausgewählten Problemen gegeben, insbesondere im Hinblick auf Mathematik und Informatik und meist auch darüber hinaus.

Des Weiteren ließe sich der Verlauf und die Gestaltung der Phasen einer Modellierungswoche den Komponenten der Projektmethode nach Frey (2012, S. 158–159) zuordnen. Eine Diskussion dazu findet sich bei Schönbrodt et al. (im Druck).

### **Kurze Einordnung der Unterrichtsformate – Holistisch oder atomistisch?**

In der Fachliteratur werden zwei Extremformen der Gestaltung mathematischer Modellierungsaktivitäten zur Förderung mathematischer Modellierungskompetenz(en) unterschieden. Inwieweit sich die beiden vorgestellten Unterrichtsformate diesen Extremformen zuordnen lassen, wird hier kurz beschrieben. Eine ausführliche Diskussion der beiden Extremformen findet sich unter anderem bei Blomhøj und Jensen (2003, S. 128 f.) und Niss und Blum (2020, S. 83 f.).

Die erste Extremform sind sogenannte **holistische Ansätze**. Beim holistischen Ansatz werden die Modellierungsaktivitäten so konzipiert, dass die Lernenden bei der Bearbeitung einer Modellierungsaufgabe / eines Problems alle Schritte des Modellierungsprozesses vollständig durchlaufen. Demgegenüber steht die zweite Extremform,

sogenannte **atomistische Ansätze**. Grundidee dieser Ansätze ist, die Teilkompetenzen des Modellierens separat zu fördern. Dazu bearbeiten die Lernenden mehrere (einzelne) Aufgaben, die jeweils auf die Förderung von ein bis zwei Teilkompetenzen ausgerichtet sind (vgl. [Niss und Blum, 2020, S. 83](#)).

Mit den mathematischen Modellierungswochen wird ein holistischer Ansatz verfolgt. Die Lernenden durchlaufen den Modellierungsprozess eigenständig und in Gänze. Das Problem bzw. der Modellierungsprozess wird nicht in kleinere Teile zergliedert oder vorstrukturiert. Dies ist wiederum der Fall bei den Modellierungstagen, bei denen die Zuordnung zu den beiden Extremformen weniger eindeutig ist. Zwar spielen im Laufe eines Modellierungstages durchaus alle zentralen Schritte von Modellierungsprozessen eine Rolle, womit der Grundgedanke ganzheitlicher Natur ist. Jedoch durchlaufen die Lernenden den Modellierungsprozess nicht komplett eigenständig. Stattdessen werden sie durch kleinere Aufgabenstellungen angeleitet. Diese strukturieren das Problem und den Modellierungsprozess vor. Betrachtet man diese stärker angeleitete Detailebene der Modellierungstage, so lassen sich diese eher dem atomistischen Ansatz zuordnen.

## 4.3 Herausforderungen für die Online-Durchführung

Wie aus den obigen Ausführungen zu Modellierungstagen und -wochen ersichtlich wurde, sind Teamarbeit und Kommunikation bei beiden Unterrichtsformaten unerlässlich. Um beides während der Online-Durchführung zu ermöglichen, sind die folgenden Fragen zentral:

- Wie kann die Kommunikation zwischen den Lernenden und mit der Lehrkraft online realisiert werden?
- Wie kann die Betreuung virtuell organisiert werden? Wie beziehungsweise wann kann die Lehrkraft bei Problemen kontaktiert werden und Unterstützung leisten?
- Wie kann die Lehrkraft den Fortschritt der Lernenden verfolgen?
- Welche Werkzeuge eignen sich für die Realisierung von Skizzen und reichhaltigen Visualisierungen, die einen wichtigen Teil der Modellierungsaktivitäten ausmachen?
- Plenumsdiskussionen finden häufig während der Modellierungsaktivitäten statt. Wie können diese virtuell und spontan realisiert werden? Wie können Plenumsphasen so gestaltet werden, dass die Lernenden aktiv einbezogen werden?
- Die Lernenden arbeiten bereits in den Präsenzveranstaltungen mit computergestütztem, digitalem Lernmaterial – ein guter Ausgangspunkt für die Online-Realisierung. Wie kann jedoch das digitale Lernmaterial ohne die Installation von Software oder Programmen auf den Geräten der Lernenden (Computer, Laptop, Tablet) bearbeitet werden?

In den nachfolgenden Kapiteln werden alle genannten Fragen aufgegriffen, wobei auf einige detaillierter eingegangen wird als auf andere. Die Fragen sind eng verknüpft mit dem Forschungsfeld *Computer-Supported Collaborative Learning* (CSCL), in dem zahlreiche Publikationen auf die Chancen und Herausforderungen hinweisen, die sich bei der Implementierung von kollaborativem Lernen in einer virtuellen Umgebung ergeben (vgl. Roberts, 2004; Stahl et al., 2006). CSCL „ist ein aufstrebender Zweig der Lernwissenschaften, der sich mit der Frage beschäftigt, wie Menschen mithilfe von Computern gemeinsam lernen können“ (Stahl et al., 2006, S. 1, eigene Übersetzung). Johnson und Johnson (2014, S. 3, eigene Übersetzung) heben hervor, dass „Technologie Schülerinnen und Schüler nicht isolieren und trennen muss. Wenn sie effektiv eingesetzt wird, kann sie die Schülerinnen und Schüler in kooperativen Bestrebungen / Projekten zusammenbringen und die Erfahrungen der Lernenden verbessern.“ Dieser Grundgedanke war auch bei der Konzipierung eines ganzheitlichen Lern- und Lehrkonzepts für kollaborative Online-Modellierungsaktivitäten leitend.

Im Folgenden werden zunächst die digitalen Werkzeuge vorgestellt, die zur Beantwortung der in diesem Abschnitt aufgeworfenen Fragen eingesetzt und erprobt wurden.

# 5 Eingesetzte digitale Werkzeuge

## 5.1 Auswahl einer Kommunikationsplattform

Wie bereits erwähnt, spielt Teamarbeit bei der Bearbeitung von (komplexen) Modellierungsproblemen eine wesentliche Rolle. In Präsenzkursen können die Lernenden leicht zusammenkommen und ihre Probleme und Ideen diskutieren, die Lehrkraft beantwortet Fragen und motiviert oder ermutigt die Lernenden. Um die Kommunikation in einer digitalen Lernumgebung zu ersetzen, wird eine virtuelle Kommunikationsplattform benötigt, die möglichst alle der folgenden Anforderungen erfüllen sollte:

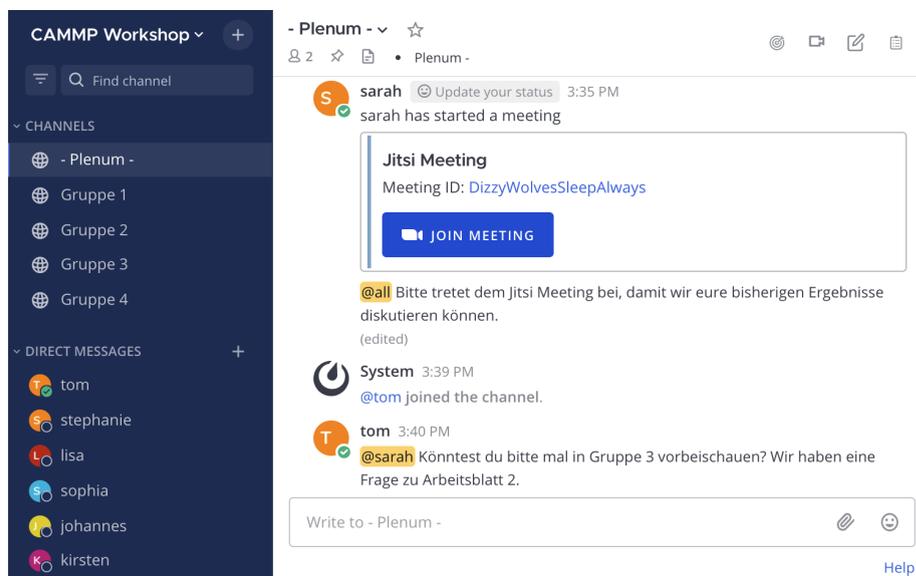
- Die Plattform sollte Open Source sein, damit keine Datenschutzprobleme beim Umgang mit personenbezogenen Daten auftreten.
- Sowohl mündlicher als auch schriftlicher Austausch sollte innerhalb einer Gruppe in gängiger Klassenstärke möglich sein.
- Es werden virtuelle Gruppenräume benötigt, in denen sich die Lernenden in kleinen Teams austauschen können.
- Eine Option zur Präsentation von Folien und zum Teilen von Bildschirmen ist zwingend erforderlich.
- Die Plattform sollte (idealerweise) ein Whiteboard zur Verfügung stellen, auf dem sich Lernende und Lehrkraft über Ideen durch Zeichnungen austauschen können.
- Der Austausch von Dateien sollte möglich sein.

Alle diese Funktionen sollten übersichtlich, einfach und intuitiv zu bedienen sein. Aufgrund langsamer und instabiler Internetverbindungen in vielen Gegenden sollte die Plattform möglichst robust sein, das heißt einen reibungslosen Verbindungsaufbau ermöglichen.

Ein Werkzeug, das die genannten Anforderungen erfüllt, ist *Mattermost*<sup>1</sup>. Mattermost kann sowohl von Lehrkräften als auch von Lernenden selbstständig und flexibel genutzt werden. Innerhalb von Mattermost können alle Benutzerinnen und Benutzer (Lernende und Lehrkräfte) bei Bedarf verschiedene Kanäle, zu verstehen wie ein Gruppenraum, erstellen (vgl. Abb. 5.1). In diesen Kanälen können die Benutzerinnen

---

<sup>1</sup>Weitere Informationen unter: <https://mattermost.com>, letzter Aufruf: 19.11.2021.



**Abb. 5.1:** Die Kommunikationsplattform Mattermost

und Benutzer Textnachrichten versenden, Umfragen erstellen und Dokumente hochladen. Zudem können Videokonferenzen über einen Jitsi<sup>2</sup>-Plugin gestartet werden. Der Erfahrung nach ist es hilfreich, einen Kanal für Plenumsphasen und einen Kanal für jede Gruppe von Lernenden einzurichten. Weitere Vorteile von Mattermost für die Realisierung von Online-Modellierungsaktivitäten sind:

- Die Lernenden können selbstständig Videokonferenzen starten und zwischen verschiedenen Konferenzen wechseln. Die Lehrkraft muss die Lernenden nicht manuell zu Untergruppen (Kanälen) zuweisen.
- Die Lernenden können ihren Bildschirm in Videokonferenzen freigeben, ohne Erlaubnis einholen zu müssen. Dies erleichtert die Zusammenarbeit in den Gruppen.
- Der Chatverlauf in den Kanälen der Schülergruppen bleibt erhalten. Dies ist besonders für mehrtägige Modellierungsprojekte nützlich, wenn auf ältere Nachrichten, Links oder Screenshots zugegriffen werden muss.
- Die Lehrkraft kann alle Lernenden durch eine einzige Nachricht im Chat des Plenumskanals benachrichtigen (vgl. Abb. 5.1). Dies ist zum Beispiel sinnvoll, um einen Hinweis zu einer Aufgabe zu geben, die alle betrifft, oder um Diskussionen im Plenumskanal anzukündigen. Umgekehrt können die Lernenden die Lehrkraft jederzeit kontaktieren und bitten, an der Videokonferenz ihrer Gruppe teilzunehmen.

Neben einem Werkzeug zur Kommunikation wird zudem ein digitales Tool für die Realisierung der Lernmaterialien benötigt. Ein solches wird im nächsten Abschnitt vorgestellt.

<sup>2</sup>Weitere Informationen unter: <https://meet.jit.si>, letzter Aufruf: 12.11.2021.

## 5.2 Jupyter Notebook

Ein Werkzeug, welches sich für die Entwicklung von interaktivem digitalen Lernmaterial eignet, ist *Jupyter Notebook*.<sup>3</sup> Vereinfacht gesagt sind Jupyter Notebooks digitale Dokumente, die es sowohl erlauben Texte zu schreiben als auch Code auszuführen und damit mathematische Ausdrücke zu berechnen. Die Notebooks bieten gegenüber reinen Text-Dateien und reinen Code-Dateien den Vorteil, dass Aufgabenstellungen (narrativer Text), erläuternde Abbildungen oder Videos und kleine Code-Abschnitte übersichtlich in einem einzelnen Dokument dargestellt werden können (vgl. Abb. 5.2). Des Weiteren können Hilfen und zusätzliche Informationen leicht integriert und in entsprechenden Aufgabenstellungen verlinkt werden. Zusätzlich können alle Elemente beliebig kombiniert und angeordnet werden.

Jupyter Notebooks sind webbasiert, Open Source und werden aktiv weiterentwickelt und kontinuierlich verbessert. Zudem finden sie in zahlreichen Branchen von Industrie, Forschung und Wirtschaft Einsatz (vgl. Pimentel et al., 2021, S. 1). Damit liefern nicht nur die eingesetzten Modellierungsprobleme, sondern zudem das verwendete digitale Werkzeug einen authentischen Einblick in aktuelle Problemlösestrategien und technische Umsetzungen in der angewandten Mathematik und in MINT-Berufen. Die Vorteile, die Jupyter Notebooks für die Gestaltung von Unterrichtsmaterial speziell für den Informatikunterricht liefern, wurden unter anderem von Kapp und Schulte (2019) aufgezeigt. Möglichkeiten, Jupyter Notebooks im Rahmen von mathematischen Modellierungsaktivitäten einzusetzen, werden bei Wohak et al. (2021), Gerhard et al. (im Druck) und Wohak (2021) hervorgehoben.

Jupyter Notebooks bestehen grundlegend aus zwei Elementen (Zellen), die flexibel angeordnet werden können. Die beiden Zelltypen sind sogenannte Text- und Codezellen. Beide Zelltypen werden in Abbildung 5.2 hervorgehoben. Je nachdem, wie viel Anleitung oder (visuelle) Unterstützung die Lehrkraft geben möchte, können die Zellen mehr oder weniger Inhalt enthalten. Zwei extreme Formen der Unterstützung sind:

- a) Das Entwerfen und Strukturieren von Lernmaterial mit verschiedenen Erklärungen, kleineren Aufgaben, Unterstützungsmaterial und Abbildungen (geeignet für stärker angeleitete Modellierungsprojekte, wie die oben genannten Modellierungstage)
- b) Offene Projekte, wie die Modellierungswochen, in denen die Lernenden ermutigt werden, Jupyter Notebooks selbstständig zu gestalten.

Im Folgenden werden die Text- und Codezellen sowie deren Gestaltungsmöglichkeiten kurz beschrieben.

### Textzellen

Wie der Name andeutet, besteht die Hauptverwendung von Textfeldern darin, erzählenden Text bereitzustellen. Dies ist beispielsweise in Form von längeren Erklärungen oder kleinen Notizen möglich, die das Problem erläutern und die die Lernenden

<sup>3</sup>Weitere Informationen unter: [www.jupyter.org](http://www.jupyter.org), letzter Aufruf: 19.07.2021.

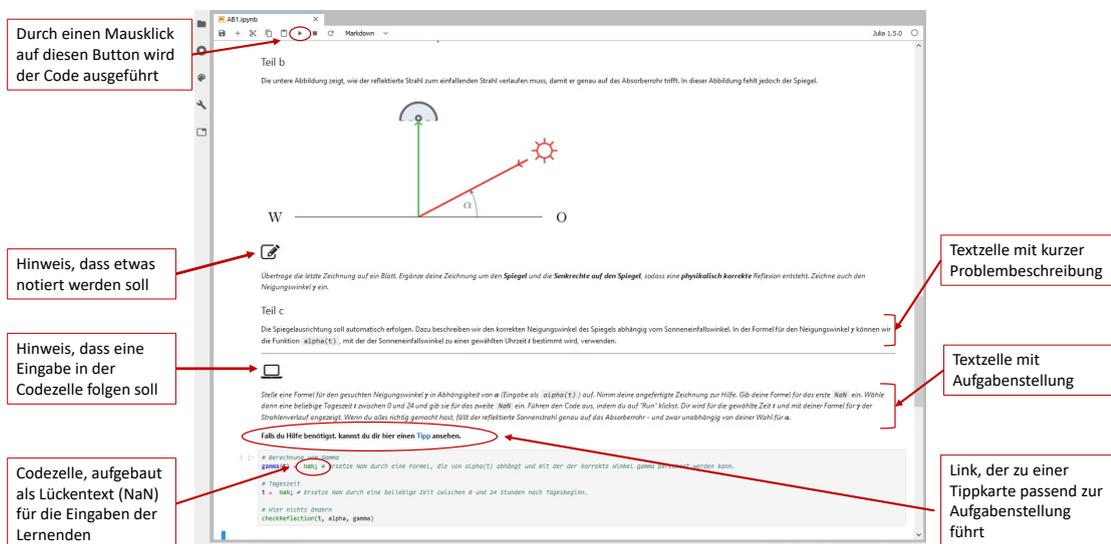


Abb. 5.2: Screenshot eines digitalen Arbeitsblatts (Jupyter Notebook) aus dem Lernmaterial zu Solarkraftwerken

durch das Material führen. Die Verwendung von Textfeldern ist nicht auf narrativen Text beschränkt: Es kann weiteres (interaktives) Material, zum Beispiel Videos, Bilder oder Tabellen, eingefügt werden. Um verschiedene Textdesigns zu ermöglichen, beispielsweise fetten Text oder Überschriften, und Bilder oder Videos einzubinden, wird die Einstellung *Markdown*<sup>4</sup> verwendet. Diese bietet einen intuitiven und reichhaltigen Satz einfacher Befehle, um gewünschte Text- und Medieneinstellungen zu erreichen. Die Textfelder bieten zudem die Möglichkeit mathematische Gleichungen und Formeln darzustellen (in  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Notation) oder auf andere Dokumente, Arbeitsblätter oder Tipps zu verweisen. Diese können direkt an der Stelle verlinkt werden, an der sie benötigt werden.

### Codezellen

Codezellen ermöglichen es, Code zu schreiben und auszuführen. Der Code kann in einer von mehreren Programmiersprachen, unter anderem *Python*, *Julia* oder *R*, geschrieben werden. Immer, wenn eine Codezelle ausgeführt wird, werden die implementierten Berechnungen durchgeführt. Variablen, die in einer ausgeführten Codezelle definiert werden, bleiben dabei verfügbar, sodass sie auch in anderen Codezellen des gleichen Jupyter Notebooks verwendet werden können.

## 5.3 Bereitstellung des digitalen Lernmaterials

Im Rahmen von beiden Unterrichtsformaten (Modellierungstage und -wochen) kommen Jupyter Notebooks zum Einsatz. Diese werden jedoch auf unterschiedliche Art und Weise bereitgestellt: im Falle der Modellierungstage / Unterrichtsreihen über *JupyterHub* und bei den Modellierungswochen über die Plattform *CoCalc*.

<sup>4</sup>Weitere Informationen unter: <https://markdown.de>, letzter Aufruf: 19.11.2021.

### JupyterHub

Die Jupyter Notebooks, die für die angeleiteten Modellierungsaktivitäten entwickelt wurden, werden auf einem Server des KIT, auf dem die Software JupyterHub<sup>5</sup> läuft, bereitgestellt. Das Material kann in jedem Internetbrowser verwendet werden, ohne dass eine zusätzliche Software installiert werden muss. Darüber hinaus laufen alle Berechnungen auf Servern des KIT, sodass das Material auch auf Geräten mit beschränkten Ressourcen (bspw. einem Tablet) geöffnet und ausgeführt werden kann. Die einzige Voraussetzung für den Einsatz des Lernmaterials ist eine stabile Internetverbindung und ein Webbrowser. Der Zugriff auf das Material erfolgt über <https://workshops.cammp.online>. Zu Beginn der angeleiteten Modellierungsprojekte erstellen alle Lernende ein Konto, um sich auf JupyterHub anzumelden, greifen auf das digitale Lernmaterial zu und bearbeiten dieses. Alle Lernenden haben einen eigenen Account mit einem eigenen, bearbeitbaren Satz an Lernmaterial. Die Lernenden können nur auf das eigene Material zugreifen und somit keine Änderungen an den Notebooks anderer Schülerinnen und Schüler vornehmen. Im Folgenden wird die Bereitstellung des stärker angeleiteten, digitalen Lernmaterials auf JupyterHub als **Workshop-Plattform** bezeichnet.

### CoCalc

Die Jupyter Notebooks für die längerfristigen, offenen Modellierungsprojekte (Modellierungswochen) werden nicht auf der Workshop-Plattform bereitgestellt. Der Grund dafür ist, dass die Lernenden während dieser Projekte kollaborativ an den Problemen und insbesondere an der Entwicklung von Programmcode arbeiten sollen. Eine wichtige Voraussetzung dafür ist, dass die Lernenden gleichzeitig am selben Notebook arbeiten können. Ein Werkzeug, welches kollaboratives Programmieren ermöglicht, ist CoCalc.<sup>6</sup> CoCalc ist Open Source, wird aktiv weiterentwickelt und erlaubt die Erstellung von Jupyter Notebooks mit verschiedenen Programmiersprachen (Python, Julia, R oder Octave). Darüber hinaus verfügt CoCalc über zahlreiche zusätzliche Funktionalitäten, die auf der Workshop-Plattform nicht vorhanden sind. Dazu gehören ein  $\text{\LaTeX}$ -Editor oder ein Chat-System. Diese zusätzlichen Funktionen erhöhen jedoch die Komplexität und machen die Bedienung weniger intuitiv. Die Erfahrung hat gezeigt, dass CoCalc besser für Projekte geeignet ist, die sich über einen längeren Zeitraum erstrecken. Dagegen ist die Workshop-Plattform aufgrund ihrer Übersichtlichkeit besser geeignet für kürzere Projekte (wie die beschriebenen Modellierungstage).

Neben CoCalc existieren verschiedene weitere Plattformen, die kollaboratives Programmieren erlauben. Ein Beispiel ist Google Colab,<sup>7</sup> was jedoch nicht Open Source ist.

---

<sup>5</sup>Weitere Informationen unter: <https://jupyter.org/hub>, letzter Aufruf: 04.11.2021.

<sup>6</sup>Weitere Informationen unter: <https://cocalc.com>, letzter Aufruf: 04.11.2021.

<sup>7</sup>Weitere Informationen unter: <https://colab.research.google.com>, letzter Aufruf: 04.11.2021.

# 6 Ziele, Aufbau und Elemente des digitalen Lehr- und Lernmaterials

In diesem Kapitel werden zentrale Entscheidungen und Ziele bei der Entwicklung des Lernmaterials zu beiden Problemstellungen (Empfehlungssysteme und Solarkraftwerk) beschrieben. Der Fokus liegt auf dem Lernmaterial für Projekte mit kürzerem Zeitrahmen (mathematische Modellierungstage oder Unterrichtsreihen) und damit auf dem stärker angeleitet konzipierten Material.

## 6.1 Übergeordnete Zielsetzungen und didaktisch-methodische Entscheidungen

Unabhängig von möglichen mathematisch-inhaltlichen Lernzielen (Erarbeitung oder Anwendung von Inhalten) wurde das digitale Lernmaterial für stärker angeleitete Modellierungsaktivitäten (Modellierungstage / Unterrichtsreihen) im Hinblick auf folgende Zielsetzungen entwickelt und eingesetzt:

- Indem die Lernenden angeleitet durch kleinere Aufgaben mathematische Formeln entwickeln und problemorientiert Aufgaben bearbeiten beziehungsweise lösen, erfassen sie die **Bedeutung von Mathematik** für das Lösen realer Probleme mit dem Fokus auf Optimierungsproblemen.
- Das Lernmaterial soll es den Lernenden ermöglichen, auch in **heterogenen Lerngruppen selbstbestimmt an den Aufgaben zu arbeiten**. Dazu werden zahlreiche gestufte Hilfen zu einzelnen Aufgaben, vertiefendes Informationsmaterial und Zusatzaufgaben bereitgestellt. Zur Förderung des eigenständigen Arbeitens im eigenen Lerntempo werden die Lösungen der Lernenden nach der Eingabe in digitale Arbeitsblätter automatisch überprüft. Die Lernenden erhalten eine individuelle Rückmeldung. Bei fehlerhaften Eingaben werden die Lösungen nicht direkt offengelegt. Stattdessen werden die Lernenden dazu angeregt, selbst nach Fehlern zu suchen. Selbstwirksamkeitserwartungen der Lernenden sollen so gestärkt werden (vgl. [Schwarzer und Jerusalem, 2002](#)).<sup>1</sup>

---

<sup>1</sup>Das Konzept der Selbstwirksamkeitserwartungen wurde maßgeblich durch die sozial-kognitive Theorie Banduras (1977) geprägt. „Mit Selbstwirksamkeitserwartung bezeichnet man [...] das Vertrauen in die eigene Kompetenz, auch schwierige Handlungen in Gang setzen und zu Ende führen zu können“ (Schwarzer und Jerusalem, 2002, S. 39). Als zentralen Einflussfaktor auf die Steigerung der Selbstwirksamkeitserwartung hebt Bandura (1977) das erfolgreiche Überwinden von Aufgaben beziehungsweise Barrieren hervor.

- Das Lernmaterial ermöglicht ein **hohes Maß an Schüleraktivität** und je nach Rahmenbedingungen mehr oder weniger offenes, unangeleitetes Arbeiten. Die Offenheit wird insbesondere im Hinblick auf individuelle Lernwege und das Verfolgen kreativer Lösungsstrategien (beim Lösen der Optimierungsprobleme) realisiert.
- Wenngleich der Schwerpunkt der vorliegenden Arbeit und auch des konzipierten Lernmaterials auf der mathematischen Optimierung liegt, so greifen die entwickelten Modelle dennoch Inhalte aus ganz unterschiedlichen Bereichen der Mathematik (Geometrie, Analysis, Stochastik, lineare Algebra) auf. Durchaus ist eines der Ziele des Lernmaterials, dass die Bedeutung von (in der Schule seltener erkennbaren) **innermathematischen Vernetzungen** für das Lösen realer Probleme erfahrbar gemacht wird.
- Die Lernenden bearbeiten die Materialien in kleinen Gruppen. Sie tauschen sich über Ideen und Fragen bei der Entwicklung der Modelle aus. Es soll deutlich werden, dass Teamarbeit für das Lösen realer, komplexer Problemstellungen hilfreich oder gar notwendig ist. **Team- und Kommunikationsfähigkeiten** werden nicht nur durch den Austausch in Kleingruppen gefördert, sondern auch durch zahlreiche Diskussionen im Plenum. Bei diesen werden Teilergebnisse gesichert, Fragen aufgeworfen und Ideen der Lernenden diskutiert.
- Das Lernmaterial ermöglicht den Lernenden das **vollständige Durchlaufen des Modellierungsprozesses**. Der Fokus liegt dabei auf der **problemorientierten** Bearbeitung des Materials.

## 6.2 Elemente des Lehr- und Lernmaterials für angeleitetes Modellieren

Die folgenden Elemente sind im stärker angeleiteten Lernmaterial zu den beiden Problemstellungen (Solarkraftwerk und Empfehlungssysteme) zentral.<sup>2</sup>

### 6.2.1 Präsentationsfolien

Die Präsentationsfolien dienen dazu,

- a) einen Einstieg in die Problemstellung zu geben,
- b) die Zwischensicherungen und Diskussionen im Plenum anzuleiten,
- c) die absolvierten Modellierungsschritte in einer abschließenden Diskussion mit den Lernenden zu resümieren, finale Ergebnisse zu validieren und zu interpretieren, Grenzen der Modelle aufzuzeigen und Modellverbesserungsideen zu sammeln.

---

<sup>2</sup>Die folgenden Elemente des digitalen Lernmaterials wurden insbesondere in Zusammenarbeit mit Kirsten Wohak, Maïke Gerhard und Maren Hattebuhr erprobt. Eine detaillierte Beschreibung findet sich in [Gerhard et al. \(im Druck\)](#). Dieser Abschnitt orientiert sich daran.

In den Präsentationen a) bis c) ist der an Blum (1985) angelehnte vierschrittige Modellierungskreislauf (vgl. Abschn. 1.1) ein wiederkehrendes Element. Dieser soll den Lernenden in gemeinsamen Diskussionen Orientierung hinsichtlich bereits absolvierter sowie anstehender Modellierungsschritte bieten und den Aufbau von Metawissen über den Modellierungsprozess unterstützen (vgl. Hinrichs, 2008, S. 33).

### 6.2.2 Arbeitsblätter

Die Arbeitsblätter wurden als Jupyter Notebooks basierend auf der Programmiersprache Julia<sup>3</sup> realisiert (vgl. Abschn. 5.2). Ein wichtiger Aspekt, warum Julia gewählt wurde, ist die einfache Syntax, die dem sogenannten Pseudocode nahe kommt. Das heißt, das Lesen und Schreiben von Julia ist vergleichsweise einfach und ähnlich wie das Niederschreiben der Lösungen / Formeln / Algorithmen mit Stift und Papier (vgl. Gerhard et al., im Druck).

#### Gestaltung der Textzellen

Die Arbeitsblätter beginnen stets mit einem Textfeld, in dem das Ziel des jeweiligen Arbeitsblattes benannt wird und gegebenenfalls weiterführende Informationen bereitgestellt werden. Die weiteren Textfelder beinhalten Aufgabenstellungen, kleinere Zwischenrésumés, Erklärvideos und Abbildungen. In die Textfelder sind Icons (bspw.  oder ) integriert. Diese machen kenntlich, wie die jeweilige Aufgabenstellung zu bearbeiten ist (vgl. Abb. 5.2).

#### Gestaltung der Codezellen

Im entwickelten Lernmaterial für die angeleiteten Modellierungsprojekte müssen die Codezellen von den Lernenden entsprechend einer Aufgabenstellung, die in einem vorangegangenen Textfeld beschrieben wird, bearbeitet beziehungsweise ergänzt werden. An diesem Punkt verändern die Lernenden das digitale Arbeitsblatt aktiv. Im Lernmaterial für die angeleiteten Projekte wird ein *Lückentext-Ansatz* verfolgt. Alle von den Lernenden zu ergänzenden Lücken im Code werden durch einen bestimmten Platzhalter gekennzeichnet, nämlich `NaN`<sup>4</sup> (vgl. Abb. 5.2). Diese Platzhalter müssen von den Lernenden zum Beispiel durch Formeln, Zahlen oder Gleichungen ersetzt werden. Dies reduziert die Menge an Code, die die Lernenden schreiben müssen, und stellt sicher, dass keine Programmierkenntnisse erforderlich sind. Der Fokus wird auf die eigentliche Aufgabe gelegt. Die von den Lernenden zu vervollständigenden Code-Eingaben gehen in den angeleiteten Projekten nicht über die Bedienung eines grafischen Taschenrechners hinaus.

---

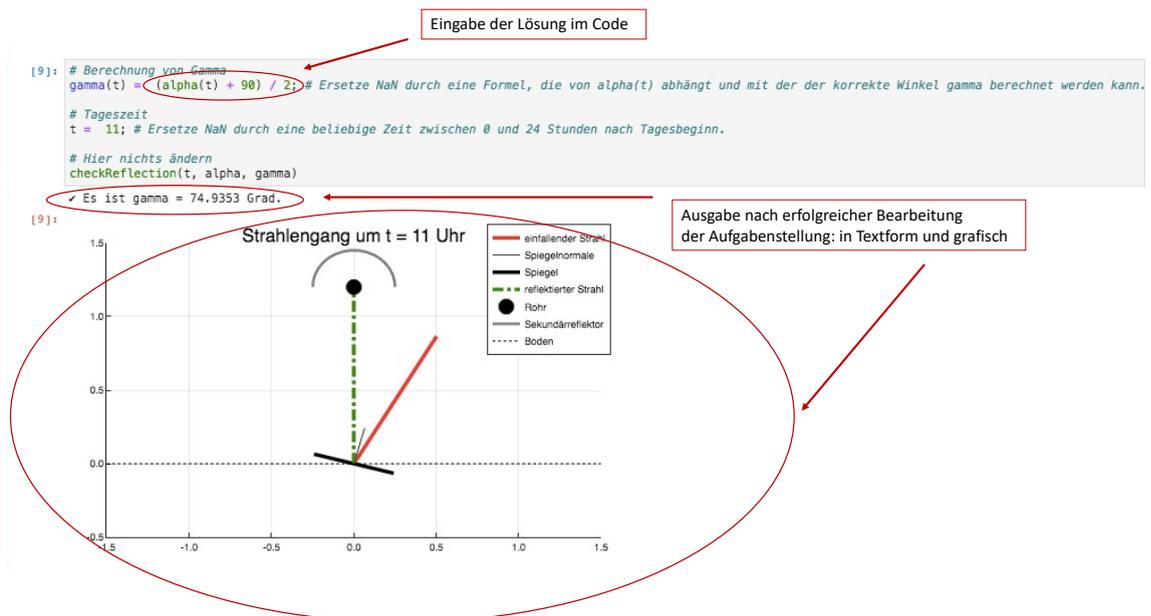
<sup>3</sup>Eine Ausnahme stellt das Arbeitsblatt 1 des Workshops zu den Empfehlungssystemen dar. Dieses wurde in Python entwickelt, da die Visualisierungsmöglichkeiten des Datensatzes in Python vielfältiger waren. Weitere Informationen zu Julia unter: [www.julia.org](http://www.julia.org), letzter Aufruf: 20.07.2021.

<sup>4</sup>`NaN` steht für *Not a Number* und ist ein gängiger Indikator für undefinierte Variablen in Programmiersprachen.

## Die Ausgabe

Bei der Ausführung von Code erhalten die Lernenden eine Ausgabe. Jupyter Notebooks zeigen die Ausgabe / die Ergebnisse direkt unter der Codezelle an (vgl. Abb. 6.1). Julia-Code erlaubt es, verschiedene Formen von Ausgaben zu erstellen, zum Beispiel Bilder, Tabellen, Plots oder Ergebnisse von mathematischen Berechnungen. Dies ermöglicht es den Lernenden, ihre Ergebnisse schnell zu überprüfen oder verschiedene Eingaben auszuprobieren, um den Zusammenhang zwischen Eingabe und Ergebnis zu untersuchen.

In manchen Fällen ist es schwierig, die Lernenden selbst ihre Ergebnisse und Eingaben überprüfen zu lassen. Um dem entgegenzuwirken, wurden Funktionen geschrieben, die die Lernenden so früh wie möglich darüber informieren, ob die getätigten Eingaben korrekt sind oder nicht. Diese sogenannten Überprüfe-Funktionen finden Einsatz, indem eine einzelne Codezeile (z. B. `checkReflection` in Abb. 6.1) unter den Zeilen ausgeführt wird, in denen die Lernenden Code-Lücken ausfüllen. Der Code für die Überprüfung ist für die Lernenden nicht sichtbar. Dadurch bleibt das Arbeitsblatt übersichtlich. Der Fokus liegt auf der eigentlichen Modellierungsaufgabe, während gleichzeitig wertvolles Feedback gegeben wird. Der Funktionsaufruf wird durch einen Kommentar im Code (`# Hier nichts ändern`) gekennzeichnet und soll von den Lernenden nicht geändert werden.



**Abb. 6.1:** Screenshot der Rückmeldung auf einem digitalen Arbeitsblatt aus dem Lernmaterial zu Solarkraftwerken. Eine Lösung der Aufgabe wurde bereits eingetragen.

Im Falle eines häufigen Fehlers<sup>5</sup> in der Eingabe fängt die Überprüfe-Funktion den Fehler im Hintergrund ab und liefert Hinweise, die die Lernenden bei der Verbesserung ihrer Eingabe unterstützen sollen. Dies vereinfacht eigenständige Korrekturen

<sup>5</sup>Mit häufig ist hier gemeint, dass der Fehler in bereits durchgeführten Modellierungsveranstaltungen wiederholt beobachtet wurde.

und unterstützt eine *personalisierte* Form des Feedbacks. Im Falle der unangeleiteten Bearbeitung offener Probleme (wie bspw. im Rahmen einer Modellierungswoche) ist ein derartiges Feedbacksystem nicht umsetzbar.

### 6.2.3 Erklärvideos

In die digitalen Arbeitsblätter sind an verschiedenen Stellen Erklärvideos eingebunden. In diesen werden weiterführende Informationen geliefert, bisherige Modellierungsschritte resümiert oder komplexere Erläuterungen aus den Plenumsphasen visuell und zum wiederholten Anschauen bereitgestellt.

### 6.2.4 Gestufte Hilfen

Die Lernenden können bei zahlreichen Aufgaben auf gestufte Hilfen, im Folgenden auch Tippkarten genannt, zugreifen. Diese Tippkarten wurden als separate Jupyter Notebooks realisiert und können über einen Link an der entsprechenden Aufgabenstellung geöffnet werden (vgl. Abb. 5.2). Die Tippkarten können Hinweise, Skizzen, Gleichungen oder erläuternde Beispiele enthalten. Je nach Aufgabe und deren Schwierigkeitsgrad stehen mehrere Tippkarten zur Verfügung. Im Hinblick auf eine On-Demand-Lernunterstützung, bei der sowohl Über- wie auch Unterforderung vermieden wird, sollen die Lernenden selbst entscheiden, ob und wann sie auf einen Tipp zugreifen. Die Tipps decken dabei ein breites Spektrum von einem wenig hilfreichen bis zu einem sehr präzisen Hinweis ab (schrittweise ansteigende Hilfen). Zudem gibt es zu einzelnen Aufgabenstellungen Tipps, die gleichwertig nebeneinanderstehen, jedoch verschiedene Aspekte der Aufgabe beleuchten. Beispielsweise werden im Lernmaterial zur Solarenergie bei einer Aufgabenstellung Kenntnisse a) rund um Richtungsvektoren und b) zu Sinus und Kosinus im rechtwinkligen Dreieck benötigt. Zu beiden Themen liegen separate Tippkarten bereit. Die Lernenden entscheiden, ob und welche Tippkarten sie öffnen.

### 6.2.5 Vertiefendes Zusatz- und Informationsmaterial

Zu beiden Problemstellungen wurden zusätzliche (optionale) Materialien erstellt, die auf den jeweiligen Arbeitsblättern über Links aufgerufen werden und sich in separaten Jupyter Notebooks öffnen. Die Zusatz- und Informationsmaterialien wurden als Methode der weiteren Differenzierung entwickelt und bieten interessierten Lernenden zusätzliche Aufgaben oder vertiefende Informationen (sowohl inner- als auch außermathematisch). Ergänzend zu den umfangreicheren Zusatzblättern sind kleinere Zusatzaufgaben in die Arbeitsblätter integriert, die von schnelleren Lernenden während oder nach der Bearbeitung der Standardaufgaben des Arbeitsblattes bearbeitet werden können. Die Zusatzaufgaben und -blätter können neben den Tippkarten als weiteres differenzierendes Element betrachtet werden, welches den Einsatz des Materials in heterogenen Lerngruppen ermöglichen soll. Schwierigkeitsgrad und

Umfang des Materials sollen somit flexibel an unterschiedliche Leistungsstände und Vorkenntnisse angepasst werden können.

### 6.2.6 Antwort- und Dokumentationsblätter

Um den Modellierungsprozess bei der Bearbeitung der Problemstellungen zu dokumentieren (dies bietet sich insbesondere bei Projekten an, die über mehrere Schulstunden hinweg durchgeführt werden) und Ergebnisse in Diskussionsphasen im Plenum direkt verfügbar zu haben, können die Lernenden vorbereitete Antwort- und Dokumentationsblätter nutzen. Diese liegen als bearbeitbare pdf-Dateien vor.

## 6.3 Elemente des Lernmaterials für unangeleitetes Modellieren

Für die eigenständige Bearbeitung der offenen (Teil-)Projekte wird wenig Lernmaterial bereitgestellt, da die Lernenden selbstständig und kreativ arbeiten und möglichst unangeleitet eigene Modellierungsansätze verfolgen sollen.

Das für die Durchführung von **Modellierungswochen** bereitgestellte Lernmaterial umfasst

- Präsentationsfolien für die Vorstellung des realen Problems durch die Betreuungsperson,
- eine kurze, ein- bis zweiseitige Problembeschreibung (als pdf) und
- ein Jupyter Notebook mit Hinweisen zur Verwendung der digitalen Werkzeuge und gegebenenfalls erste Einleseroutinen für die zur Verfügung stehenden Daten.

Im Rahmen dieser Arbeit wurde überdies Material für **Modellierungstage mit ansteigendem Offenheitsgrad** entwickelt. Konkret arbeiten die Lernenden zunächst stärker angeleitet mit dem in Abschnitt 6.2 beschriebenen Lernmaterial, um einen niederschweligen Einstieg in die Problemstellung und die Verwendung der digitalen Werkzeuge zu erhalten. An diversen Stellen des Modellierungsprozesses besteht dann die Möglichkeit von den stärker angeleiteten Bausteinen des Materials zum offenen Modellieren überzugehen. Dazu wurden Arbeitsblätter (Jupyter Notebooks) konzipiert, in denen die bereits im angeleiteten Teil des Projektes entwickelten Formeln und Funktionen geladen werden. Diese stehen dann für den direkten Einsatz durch die Lernenden bereit.

Ein Beispiel ist die eigenständige Entwicklung und Implementierung eines Verfahrens zur Optimierung der Positionen der Spiegel eines solarthermischen Kraftwerks. Die Zielfunktion basiert auf einem Modell für den Energieumsatz, welches die Lernenden zuvor angeleitet entwickelt haben. Die Funktion zur Beschreibung der Energie wird auf dem Arbeitsblatt zum unangeleiteten Modellieren bereitgestellt, sodass die Lernenden bei der Entwicklung eines Optimierungsverfahrens und beim Schreiben

von eigenem Programmcode direkt darauf zugreifen können (vgl. Abschn. 12.2.4). Im Gegensatz zum stärker angeleiteten Material erhalten die Lernenden also *keine* kleinschrittigen Aufgabenstellungen, die den Modellierungsprozess vorstrukturieren. Die Schülerlösungen können dadurch logischerweise nicht automatisch überprüft werden.

## 7 Beispielhafter Ablauf eines Online-Modellierungstages

Im Folgenden wird der exemplarische Ablauf von ein- bis zweitägigen Online-Modellierungstagen am Beispiel des Workshops *Optimierung eines Solarkraftwerks* beschrieben. Der Schwerpunkt liegt hier auf den technischen und organisatorischen Aspekten. Inhalte und Lernmaterial des Workshops werden in Teil III detailliert dargestellt. Das digitale Lernmaterial, das während des Modellierungstages eingesetzt wird, kann, wie im [Vorwort](#) erläutert, abgerufen werden.

Ein Modellierungstag dauert typischerweise etwa fünf Arbeitsstunden. Je nach Lerngruppe können die verschiedenen Elemente der Modellierungstage zeitlich, aber auch in ihrer Reihenfolge und Anzahl variieren. In manchen Lerngruppen kann es zum Beispiel sinnvoll sein, mehr Diskussionsphasen einzubauen und damit die einzelnen Arbeitsphasen häufiger zu unterbrechen. Die im Folgenden angegebenen Zeiten sind Durchschnittswerte, die auf Erfahrungswerten mit Lernenden beruhen. Sie können daher je nach Lerngruppe (stark) variieren und dienen lediglich als Richtwerte oder ungefähre Einschätzung.

### **Technischer Check-in (15 Min.)**

Wenn die Lernenden Mattermost und Jitsi zum ersten Mal verwenden, wird mit einem kurzen technischen Check-in begonnen. Die Lernenden treten einer Jitsi-Videokonferenz im Plenumskanal bei und werden gebeten, ihre Stummschaltung kurz aufzuheben (und optional die Kamera einzuschalten), um zu überprüfen, ob Mikrofon und Audio richtig funktionieren.

### **Modellierungsvortrag (optional, 15–20 Min.)**

Im Modellierungsvortrag wird der Prozess der mathematischen Modellierung an einem ausgewählten Beispiel, welches nicht in Zusammenhang mit dem Thema des Workshops steht, beschrieben. Ein vielfach verwendetes Problem ist die Modellierung und Optimierung des Designs eines Flugzeugs. Die zentralen Schritte des Modellierungsprozesses werden erläutert und in den an [Blum \(1985\)](#) angelehnten vierschrittigen Modellierungskreislauf eingebettet (vgl. [Abb. 1.1](#)). Es wird deutlich gemacht, dass mathematisches Modellieren (offensichtlich oder versteckt) Grundlage für zahlreiche Anwendungen und Technologien aus dem Alltag der Lernenden ist. Zudem werden exemplarische Berufs- und Studienmöglichkeiten aufgezeigt, in denen mathematische Modellierung relevant ist. Der Modellierungsvortrag ist lediglich optionaler Bestandteil des Workshops. Wird dieser ausgelassen, so wird bei dem

nachfolgenden Einstieg in die Problemstellung intensiver auf den Modellierungsprozess und dessen idealtypische Schritte eingegangen.

### **Einstieg in die Problemstellung (15–20 Min.)**

Der eigentliche Modellierungstag beginnt mit einer Jitsi-Videokonferenz im Plenumskanal, an der alle Lernenden teilnehmen. Die Lehrkraft hält einen kurzen Einführungsvortrag und stellt dabei die Problemstellung des Modellierungstages vor: *Wie sollte ein solarthermisches Kraftwerk gebaut und betrieben werden, damit der Energieumsatz maximiert wird?* In der Einführung beschreibt die Lehrkraft kurz den Aufbau eines sogenannten Fresnelkraftwerks (vgl. Kap. 9). Flache Spiegel fokussieren die Sonnenstrahlen auf ein Absorberrohr. Das Rohr enthält ein Wärmeträgermedium, zum Beispiel Wasser, das erhitzt und, im Falle von Wasser, verdampft wird. Mittels einer Dampfturbine wird elektrische Energie erzeugt.

Das Problem wird mithilfe von Präsentationsfolien erklärt, die per Bildschirmfreigabe gezeigt werden. Zwischendurch werden kurze Videos gezeigt. Die Lernenden werden in die Diskussion einbezogen, indem sie ihr Vorwissen über Solarenergiesysteme einbringen. Wie im Präsenzunterricht heben die Lernenden ihre (virtuellen) Hände, um sich zu Wort zu melden und Fragen zu stellen.

### **Übergangsphase (15 Min.)**

Nach der Einführung in das Problem gibt die Lehrkraft einen kurzen Überblick über die Verwendung von Mattermost während der Gruppenarbeit. Außerdem zeigt die Lehrkraft durch Freigabe ihres Bildschirms, wie auf das digitale Lernmaterial zugegriffen werden kann. Sie erklärt den Aufbau der Arbeitsblätter (Jupyter Notebooks) und, wie diese zu verwenden sind. Die Lernenden melden sich anschließend auf der Workshop-Plattform an.

### **Erste Arbeitsphase (30–45 Min.)**

Die Lernenden beginnen mit der Bearbeitung des digitalen Lernmaterials. Für die Kommunikation in Kleingruppen treten zwei bis drei Lernende einem Kanal in Mattermost bei. In diesem Kanal starten die Lernenden selbstständig eine eigene Videokonferenz (über Jitsi). Außerdem können sie den Chat in ihrem Kanal nutzen, um Links, Formeln und anderes Material auszutauschen. Während der Arbeit an den digitalen Arbeitsblättern diskutieren die Lernenden Ideen oder Fragen – so, als wären sie im selben Raum. Um Überlegungen zu skizzieren, haben die Lernenden die Möglichkeit, ein Whiteboard-Tool zu verwenden.

Auf dem ersten Arbeitsblatt beschäftigen sie sich mit der richtigen Ausrichtung der Spiegel eines Solarkraftwerks. Die Neigung der Spiegel soll so variiert werden, dass die Sonnenstrahlen zu jeder Tageszeit auf das Rohr reflektiert werden (vgl. Abb. 10.1). Haben die Lernenden Schwierigkeiten mit bestimmten Aufgabenteilen auf den Arbeitsblättern, öffnen sie die verlinkten Tippkarten selbstständig. Falls darüber hinaus Fragen oder Schwierigkeiten bestehen, wenden sie sich über eine kurze Nachricht im Chat an die Lehrkraft. Die Lehrkraft ist während der Arbeitsphasen in den Gruppen über den Chat im Plenumskanal erreichbar. Um die Arbeit und den

---

Fortschritt der Lernenden zu verfolgen, kann die Lehrkraft die Lernenden auffordern, kurz in den Chat des Plenumskanals zu schreiben, wenn sie mit einem neuen Arbeitsblatt beginnen. Die Lehrkraft kann zudem jederzeit den Videokonferenzen der einzelnen Gruppen beitreten.

### **Erste Zwischensicherung (10–15 Min.)**

Die Lehrkraft nutzt den Plenumskanal, um die erste Plenumsdiskussion anzukündigen. Der Fokus liegt auf der Vorstellung der Ergebnisse durch die Lernenden. Teile der Lösungen können auf ein virtuelles Whiteboard gezeichnet oder von den Lernenden per Bildschirmfreigabe präsentiert werden. In der Zwischensicherung werden zudem offene Fragen und Modellverbesserungsideen der Lernenden diskutiert.

### **Weitere Arbeits- und Sicherungsphasen (je 30–45 Min.)**

Die weiteren Arbeitsphasen sind analog zur ersten Arbeitsphase aufgebaut. Die Lernenden bearbeiten das Lernmaterial weiterhin selbstständig in kleinen Teams. An geeigneten Stellen versammelt die Lehrkraft die Lernenden wieder zu Plenumsdiskussionen. Um sicherzustellen, dass den schnelleren Gruppen die interessanten Probleme nicht ausgehen, steht umfangreiches zusätzliches Material zur Verfügung, zum Beispiel kleinere Zusatzaufgaben, Infoblätter sowie offene Teilprobleme.

### **Abschlussdiskussion (20–30 Min.)**

In der Abschlussdiskussion wird der gesamte Modellierungsprozess reflektiert. Die erzielten Ergebnisse werden im Hinblick auf das reale Problem interpretiert und validiert. Die Lernenden beteiligen sich aktiv an der Diskussion, indem sie ihre Ergebnisse mündlich, per Bildschirmfreigabe oder auf einem Whiteboard präsentieren. Es werden Grenzen des Modells und Ideen zu dessen Verbesserung diskutiert. Am Ende des Modellierungstages erhalten die Lernenden einen schriftlichen, digitalen Fragebogen. In diesem werden unter anderem die Erfahrungen mit und Meinungen zu Inhalt, Aufbau und Organisation des Online-Modellierungstages abgefragt, mit dem Ziel, die Workshops kontinuierlich zu verbessern.

### **Vorstellung und Auswahl offener Teilprojekte (10–15 Min.)**

Steht für die Bearbeitung der Problemstellung mehr Zeit zur Verfügung, beispielsweise im Rahmen von anderthalb bis zweitägigen Modellierungstagen oder längerfristigen Unterrichtsreihen, so kann die Bearbeitung von offenen Problemstellungen angeschlossen werden. Diese werden kurz von der Lehrkraft beschrieben. Handelt es sich um eine programmierlastige offene Aufgabe (bspw. die Implementierung eines eigenen Optimierungsverfahrens), so erhalten die Lernenden ein Jupyter Notebook, welches Tipps zur Syntax der verwendeten Programmiersprache bereithält. Zudem werden in diesem Notebook bereits von den Lernenden entwickelte Funktionen bereitgestellt, die direkt eingesetzt oder weiterentwickelt werden können (vgl. Abschn. 6.3).

### **Bearbeitung offener Teilprojekte (120–240 Min.)**

Bei der Bearbeitung der offenen Teilprojekte arbeiten die Lernenden wie bisher in ihrer Kleingruppe, nun jedoch ohne kleinschrittige Aufgabenstellungen und wiederholte Sicherungsphasen. Erfahrungsgemäß wird das Whiteboard in dieser Phase intensiv von den Lernenden genutzt, beispielsweise um geometrische Ideen zu skizzieren oder um Teilschritte eines Algorithmus in Form von Pseudocode zu notieren.

### **Präsentation der Ergebnisse der offenen Teilprojekte (5 Min. pro Gruppe)**

Bei der Präsentation im Plenum ist es den Lernenden freigestellt, ob sie ihre Ergebnisse via Bildschirmübertragung des (weiter-)entwickelten Jupyter Notebooks, des Whiteboards oder eigener Präsentationsfolien vorstellen.

### **Ablauf im Rahmen einer Unterrichtsreihe**

Anstatt das Material zusammenhängend im Rahmen von Modellierungstagen durchzuführen, kann und wurde dieses, erstreckt über mehrere Wochen, im Rahmen von Unterrichtsreihen eingesetzt. Die oben genannten Phasen finden sich auch bei dieser Organisationsform wieder. In diesem Fall bietet es sich an, Zwischendiskussionen und (zusätzliche) Zusammenfassungen am Ende einer jeden Unterrichtsstunde einzubauen. Insbesondere, wenn zwischen einzelnen Unterrichtsstunden längere Zeiträume liegen, ist es sinnvoll, bisherige Modellierungsschritte zu Beginn jeder Unterrichtsstunde im Plenum zu wiederholen, um den roten Faden bei der Bearbeitung der Problemstellungen nicht zu verlieren.

## **Teil III**

# **Unterrichtsmaterial zur Optimierung eines solarthermischen Kraftwerks**



## 8 Einleitung

In Wissenschaft und Wirtschaft wird intensiv daran geforscht erneuerbare Energieträger wirtschaftlich und effizient zu gestalten. Zu den erneuerbaren Energien gehören unter anderem Wasserkraft, Bioenergie, Sonnenenergie, Geothermie und Windenergie. Welche erneuerbare Energiequelle für eine effiziente Energieversorgung geeignet ist, hängt stark von regionalen Aspekten (wie bspw. Wetter und Geographie) ab. Dieser Teil der Arbeit dreht sich um das Thema der Solarenergienutzung. Genauer geht es um die Modellierung und Optimierung von solarthermischen Kraftwerken mit planaren Spiegeln, sogenannte Fresnelkraftwerke (vgl. Abb. 8.1). Diese eignen sich besonders in sonnenreichen Gegenden, wie beispielsweise Spanien oder Kalifornien (vgl. [Hattebuhr et al., 2015](#), S. 356).



**Abb. 8.1:** Fresnelkraftwerk Puerto Erado 2 in Südspanien, Quelle: EBL

Im Folgenden wird zunächst der Energieumsatz eines vereinfachten Fresnelkraftwerks modelliert. Anschließend wird zum Kern dieses Teils der Arbeit und auch des entwickelten Lernmaterials übergegangen: der Optimierung verschiedener Kraftwerksparameter. Dazu werden unterschiedliche Formulierungen der Optimierungsprobleme und Möglichkeiten zu deren Lösung beschrieben. Es wird umfassend herausgestellt, dass die Problemstellung sowohl hinsichtlich der Optimierungsformulierungen als auch hinsichtlich diverser Algorithmen zum Lösen der Probleme enorm facettenreich ist. Dies stellt einen guten Ausgangspunkt dar, um Schülerinnen und Schüler Probleme im Bereich des Optimierens auf kreative und offene Art und Weise angehen zu lassen. Inwieweit diese Möglichkeiten ausgeschöpft und im Rahmen von digitalem Lernmaterial umgesetzt wurde, wird anschließend in Kapitel 12 beschrieben. Konkret wurde Lernmaterial für zwei Workshops entwickelt. Die Workshops richten sich aufgrund ihrer jeweiligen mathematischen Schwerpunkte an unterschiedliche Zielgruppen: Lernende der Mittel- oder Oberstufe.

Bei der folgenden Modellierung und Optimierung eines Solarkraftwerks werden allgemeine Aussagen und (auch von Lernenden denkbare) Herangehensweisen an die Optimierung herausgearbeitet. Dabei wird eindrücklich ersichtlich, dass das Lösen und Formulieren von Optimierungsproblemen mathematisch reichhaltig ist und (den Lernenden) die Möglichkeit bietet, reale Problemstellungen auf unterschiedliche Weise anzugehen.

## 9 Basiswissen zu solarthermischen Kraftwerken

In solarthermischen Kraftwerken wird die Strahlung der Sonne als Energiequelle genutzt. Mithilfe von Spiegeln wird das Sonnenlicht auf einen Absorber gebündelt. Der Absorber besteht bei den betrachteten Kraftwerken aus einem Rohr, welches ein Wärmeträgerfluid (bspw. Wasser oder ein Thermoöl) enthält. Das Fluid wird durch die konzentrierte Sonnenstrahlung stark erhitzt. So wird die absorbierte Energie der Sonne zunächst in Wärmeenergie umgewandelt. Wird Wasser als Wärmeträgermedium verwendet, kann der entstehende Wasserdampf direkt genutzt werden, um die thermische Energie mittels einer Dampfturbine in elektrische Energie umzuwandeln (vgl. Abb. 9.1). Solarthermische Kraftwerke haben gegenüber Photovoltaik den Vorteil, dass es technisch einfacher ist Energie zu speichern, indem beispielsweise Speichertanks mit heißem geschmolzenen Salz integriert werden. Auf diese Weise kann Strom aus Solarenergie auch in der Nacht oder zu sonnenarmen Zeiten bedarfsgerecht zur Verfügung gestellt werden (vgl. [BINE Informationsdienst](#), [FIZ Karlsruhe](#), 2013, S. 4).

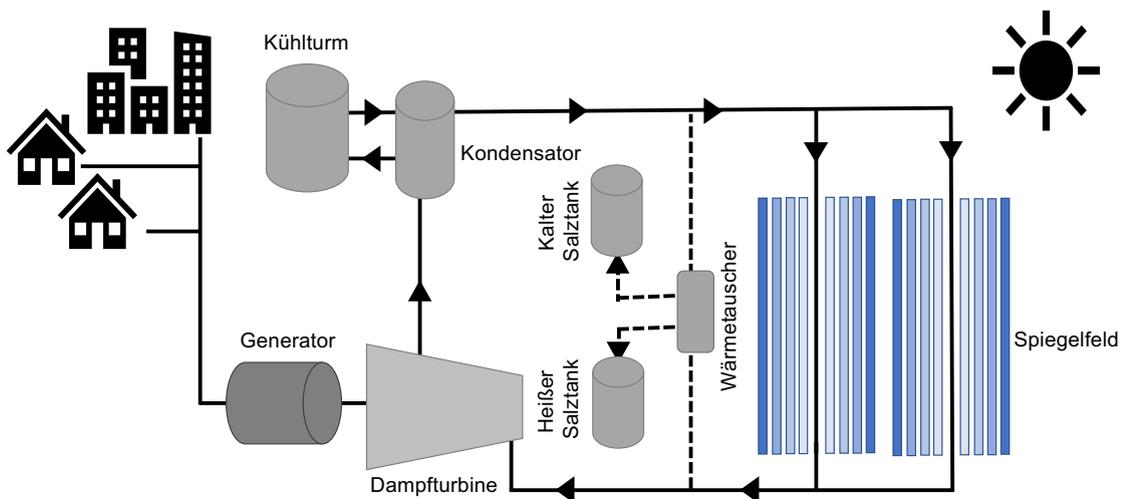


Abb. 9.1: Schematischer Aufbau eines Fresnelkraftwerks

Im Bereich der solarthermischen Kraftwerke gibt es verschiedene Konzepte. Die zwei bekanntesten sind **Solarturmkraftwerke** mit zentralen, relativ kleinen Absorbern und **Solarfarmkraftwerke**, bei denen die Absorber mehrere hundert Meter lang sind. Im entwickelten Lernmaterial liegt der Fokus auf den Solarfarmkraftwerken, deren Funktionsweise nachfolgend beschrieben wird. Eine kurze Beschreibung der Solarturmkraftwerke, inklusive bildlicher Darstellung, ist im Ausblick zu finden (vgl.

Abschn. 13.4).

Weltweit sind bereits zahlreiche solarthermische Kraftwerke in Betrieb. Dennoch wird weiterhin intensiv daran geforscht, die einzelnen Komponenten und Konzepte zu optimieren. Geplante Anlagen werden bereits vor der Konstruktion auf ihre Wirtschaftlichkeit überprüft. Ziel ist es, Kosten zu minimieren und zugleich eine optimale Energieausbeute zu erzielen, indem die vorhandene Sonneneinstrahlung so gut wie möglich genutzt wird (vgl. [BINE Informationsdienst, FIZ Karlsruhe, 2013, S. 3](#)). Die Optimierung der Kraftwerke wird in der Praxis und auch im Rahmen des entwickelten Lernmaterials basierend auf numerischen Simulationen durchgeführt.

### **Solarfarmkraftwerke**

In Solarfarmkraftwerken sind die Spiegel auf einem riesigen Feld parallel zueinander aufgestellt. Die Spiegel fokussieren das Sonnenlicht auf Absorberrohre, die über den Spiegeln angebracht sind (vgl. [Abb. 8.1](#)). Im Laufe des Tages werden die Spiegel der Sonne mithilfe von Motoren oder Seilantrieben nachgeführt (vgl. [BINE Informationsdienst, FIZ Karlsruhe, 2013, S. 4](#)). Man unterscheidet zwischen Kraftwerken mit Parabolrinnenkollektoren und planaren Fresnelkollektoren.

Bei Parabolrinnenkraftwerken wird für jedes Absorberrohr eine parabolisch geformte Spiegelrinne als Reflektorfläche verwendet. Diese wird der Sonne nachgeführt, sodass sich das Absorberrohr immer in deren Brennpunkt befindet. Dies bietet einen hohen Effizienzgrad, da immerzu die komplette Spiegelfläche zur Sonne geöffnet ist. Dem entgegen stehen jedoch hohe Kosten für Herstellung und Reinigung der Parabolspiegel (vgl. [BINE Informationsdienst, FIZ Karlsruhe, 2013; Krahforst, 2016, S. 105](#)).

Anstelle einer Parabolrinne wird bei Fresnelkraftwerken eine Reihe von planaren Spiegeln verwendet. Mehrere Spiegelreihen fokussieren dabei das einfallende Licht auf ein Absorberrohr (vgl. [Abb. 9.2](#)). Um reflektierte Sonnenstrahlen aufzufangen, die das Absorberrohr nur knapp verfehlen, befindet sich über jedem Rohr ein gekrümmter Spiegel, der sogenannte Sekundärreflektor,<sup>1</sup> welcher diese Strahlen mithilfe einer zweiten Spiegelung doch noch auf das Rohr fokussiert. Wesentliche Vorteile von Fresnelkraftwerken sind die geringeren Kosten für Herstellung, Reinigung und Wartung der planaren Spiegel. Nachteil ist, dass sich die Spiegel gegenseitig verschatten können und nicht die komplette Spiegelfläche ausgenutzt wird.

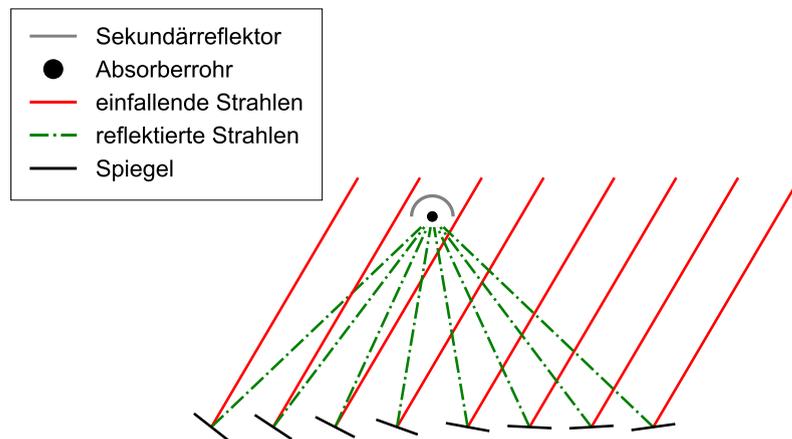
Im folgenden Abschnitt werden die Daten beschrieben, die bei der Modellierung des Energieumsatzes von Solarkraftwerken und der darauf basierenden Optimierung verschiedener Kraftwerksparameter zum Einsatz kommen.

## **9.1 Die Sonnenstrahlung – Datengrundlage**

Wesentlich für die Berechnung der umgesetzten Energie eines Kraftwerks ist die Stärke der Sonnenstrahlung an dem Ort, an dem das Kraftwerk steht oder gebaut

---

<sup>1</sup>Der Sekundärreflektor ist meist parabolisch geformt. Das Absorberrohr befindet sich in dessen Brennpunkt (vgl. [Günther, 2004, S. 22](#)). In nachfolgenden Skizzen wird der Sekundärreflektor der Einfachheit halber halbkreisförmig dargestellt.



**Abb. 9.2:** Schematische Reflexion von Sonnenstrahlen in einem Fresnelkraftwerk dargestellt im Querschnitt

werden soll. Im entwickelten Lernmaterial wird Daggett, ein sonniger Ort in Kalifornien, als Standort für das Fresnelkraftwerk betrachtet. Gewählt wurde Daggett, da für diesen Standort die notwendigen, im Folgenden beschriebenen Daten vorliegen und dort tatsächlich Solarkraftwerke in Betrieb sind. Grundsätzlich ließe sich auch mit Daten eines anderen sonnenreichen Standorts, wie Sevilla (Südspanien) oder Upington (Südafrika), arbeiten.<sup>2</sup>

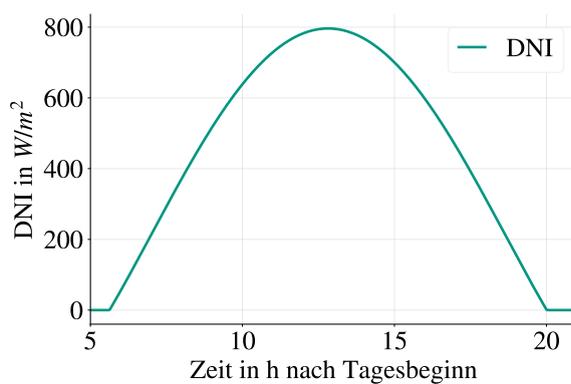
Die Strahlung der Sonne kann in **direkte** und **diffuse Strahlung** unterteilt werden. Die diffuse Strahlung kommt unter anderem durch Reflexion und Ablenkung von Sonnenstrahlen durch die Atmosphäre und Wolken zustande. Die direkte Strahlung hingegen kommt, wie der Name schon sagt, direkt von der Sonne. Da nur die direkte Strahlung gebündelt auf den Absorber konzentriert werden kann, wird lediglich diese Strahlung bei der Modellierung des Kraftwerks berücksichtigt (vgl. Richter, 2017, S. 9). Die Bestrahlungsstärke der direkten Strahlung (engl. *direct normal irradiance*, DNI) kann für jeden Ort auf der Erde mithilfe einer Messung bestimmt werden. Die Bestrahlungsstärke besitzt die Einheit  $\text{W}/\text{m}^2$ . Sie beschreibt die Leistung der Strahlung, die auf eine senkrecht zur Strahlung stehende Fläche von einem Quadratmeter trifft. Zur Messung der Bestrahlungsstärke der direkten Sonnenstrahlung kommen sogenannte Pyrheliometer zum Einsatz (vgl. Duffie und Beckmann, 2013, S. 44). Die Bestrahlungsstärke der direkten Sonnenstrahlung kann außerdem mithilfe einer zweiten Methode ermittelt werden: der Verwendung von meteorologischen Modellen. Das sogenannte **Clear Sky MRM-Modell** ist ein Beispiel für ein solches meteorologisches Modell. Derartige Modelle gehen vereinfachend davon aus, dass die Sonneneinstrahlung ein symmetrisches Verhalten aufweist. Beispielsweise wird die Sonneneinstrahlung in manchen Modellen als tagessymmetrisch (um den Zeitpunkt, zu dem die Sonne mittags am höchsten steht) angenommen. Zudem wird angenommen, dass der Himmel stets wolkenlos ist und keine wetterbedingten Schwankungen der Sonnenstrahlung auftreten (vgl. Kambezidis und Psiloglou, 2008, S. 357).

<sup>2</sup>Nahel Sevilla steht das Solarkraftwerk *PS 20* und bei Upington das Solarkraftwerk *Khi Solar One*, siehe <https://solarpaces.nrel.gov/projects>, letzter Aufruf: 06.12.2021.

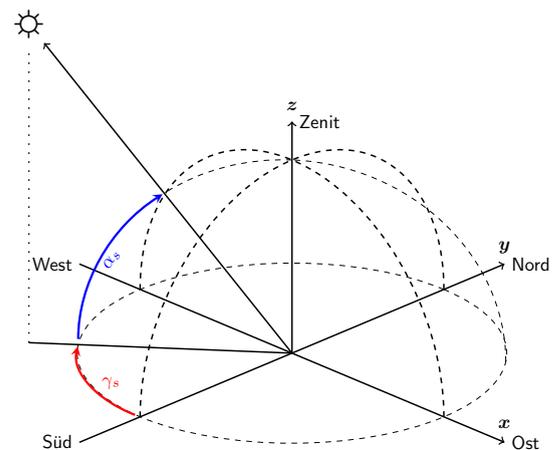
**Tab. 9.1:** Übersicht über vorliegende Daten zur Bestrahlungsstärke der direkten Sonnenstrahlung in Daggett basierend auf einem MRM-Modell

Tag	Zeit in h nach Tagesbeginn	Bestrahlungsstärke in $W/m^2$
1	7.91 <sup>3</sup>	0
1	8.01	3.68
1	8.11	20.14
⋮	⋮	⋮
1	15.91	263.75
1	16.01	251.21
⋮	⋮	⋮
365	17.61	13.57

Für das entwickelte Lernmaterial liegt ein Datensatz aus einem Clear Sky MRM-Modell vor. Dieser beinhaltet für verschiedene Tageszeiten aller 365 Tage eines Jahres die Bestrahlungsstärke der direkten Sonnenstrahlung in Daggett. Ein Auszug aus diesem Datensatz ist in Tabelle 9.1 dargestellt. Im Lernmaterial wird lediglich ein einziger Tag im Juni betrachtet. Die Daten von Sonnenauf- bis Sonnenuntergang dieses Tages wurden mit der Methode der kleinsten Fehlerquadrate durch ein (abgeschnittenes) Polynom approximiert. Das Ergebnis ist in Abbildung 9.3 dargestellt. Die Zeit wird in Stunden nach Tagesbeginn (d. h. 15:45 Uhr entspricht 15.75 h) angegeben.



**Abb. 9.3:** Approximierte Bestrahlungsstärke der direkten Sonnenstrahlung am 21. Juni in Daggett

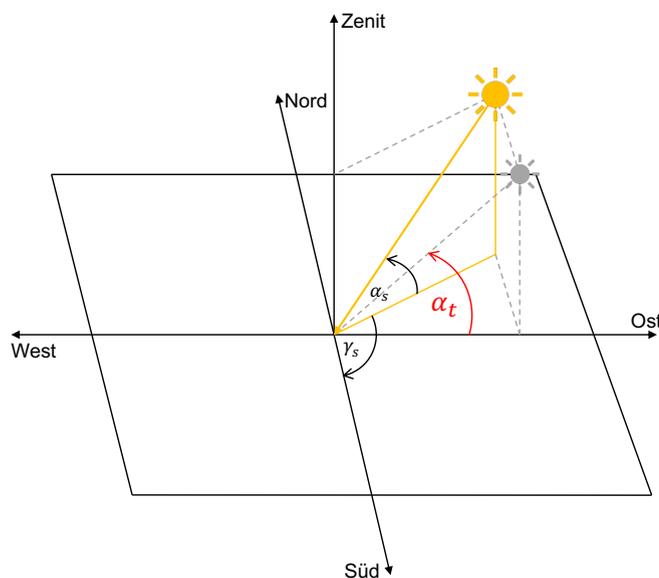


**Abb. 9.4:** Azimut  $\gamma_s$  und Altitude  $\alpha_s$

Um die Sonnenstrahlung in den Spiegelkraftwerken wie gewünscht auf den Absorber zu reflektieren, müssen die Spiegel entsprechend ausgerichtet werden. Dazu werden Daten benötigt, die die Richtung der einfallenden Sonnenstrahlen beschreiben. Mit den geografischen Daten (Längen- und Breitengrad) des Ortes Daggett kann die

Sonnenposition relativ zur Position des Kraftwerks berechnet werden. Die Position der Sonne wird dabei durch zwei Winkel, Azimut und Altitude (auch Sonnenhöhe oder Elevation genannt), beschrieben (vgl. Abb. 9.4). Für jede beliebige Tageszeit ergibt sich die Position der Sonne und letztlich die Richtung der einfallenden Sonnenstrahlen relativ zur Tangentialebene durch den Ort Daggett an der Erdkugel (vgl. Duffie und Beckmann, 2013, S. 13–14).

Da das Solarkraftwerk im Lernmaterial lediglich im Querschnitt betrachtet wird, wird die Richtung der einfallenden Sonnenstrahlen vereinfacht durch einen einzigen sogenannten **transversalen Sonneneinfallswinkel**  $\alpha_t$  beschrieben (vgl. Günther, 2004, S. 26). Dieser ist in Abbildung 9.5 dargestellt. Um diesen Winkel zu erhalten, wird die Position der Sonne in die Ebene projiziert, die durch die Ost-West-Achse und die Zenit-Achse verläuft. Der Winkel, der von der Ost-Achse und den projizierten Sonnenstrahlen eingeschlossen wird, stellt den im Lernmaterial verwendeten transversalen Einfallswinkel dar. Wie die Daten zu Stärke und Richtung der Sonnenstrahlen in die Modellierung einfließen, wird in Kapitel 10 erläutert.



**Abb. 9.5:** Beschreibung der einfallenden Sonnenstrahlen über den transversalen Sonneneinfallswinkel  $\alpha_t$

Im entwickelten Lernmaterial werden die Daten zum Stand der Sonne und zur Stärke der Sonnenstrahlung nicht mit den Lernenden diskutiert. Stattdessen dienen diese Daten als Basis für die Modellierung und Optimierung des Kraftwerks. Weiterführendes Lernmaterial, welches die Daten stärker beleuchtet und sowohl physikalische Konzepte aus dem Bereich Astronomie als auch mathematische Konzepte aus den Bereichen Geometrie und Optimierung verbindet, wurde im Rahmen einer von mir co-betreuten Bachelorarbeit von Hoeffler (2020) entwickelt.

# 10 Modellierung eines Fresnelkraftwerks

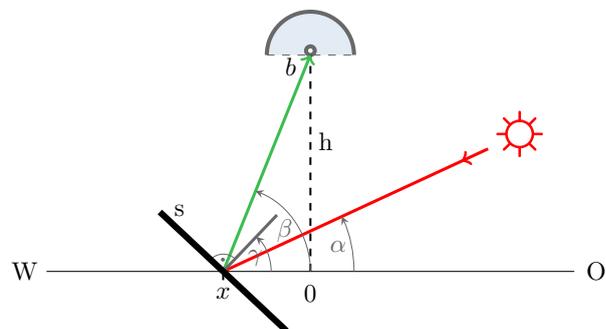
Die in diesem Kapitel beschriebene Modellierung der umgesetzten Leistung eines Fresnelkraftwerks wurde großteils in Arbeiten von [Frank et al. \(2017\)](#), [Hattebuhr et al. \(2015\)](#), [Krahforst \(2016\)](#) und [Roeckerath \(2012\)](#) erläutert. Der Vollständigkeit halber und im Hinblick auf das Verständnis der nachfolgenden Optimierung werden die wesentlichen Modellierungsschritte dennoch kurz erläutert.

Zunächst wird eine (stark) vereinfachte Situation (vgl. [Frank et al., 2017](#); [Hattebuhr et al., 2015](#); [Krahforst, 2016](#); [Roeckerath, 2012](#)) betrachtet. Dazu werden zu Beginn des Modellierungsprozesses die folgenden Vereinfachung vorgenommen:

1. Es wird zunächst nur ein einzelner Spiegel betrachtet.
2. Der Horizont verläuft tangential zum Erdboden durch den Spiegelmittelpunkt.
3. Wir betrachten den Querschnitt eines Kraftwerks und reduzieren das Problem auf den zweidimensionalen Fall.
4. Alle Sonnenstrahlen fallen perfekt parallel ein.
5. Der Spiegel kann exakt um jeden Winkel geneigt werden.
6. Die Spiegel sind perfekt. Sie haben keine Unebenheiten und reflektieren exakt und ohne Verluste.

Im Laufe der Modellierung werden einzelne Vereinfachungen schrittweise verworfen und das Modell der Realität angenähert.

Abbildung 10.1 stellt die vereinfachte Situation schematisch dar. Dabei steht die Variable  $x$  für die Position des Spiegelmittelpunktes relativ zum Nullpunkt. Der Nullpunkt befindet sich senkrecht unter dem Rohr. Dementsprechend ergibt eine Verschiebung des Spiegels nach links (gen Westen) negative und eine Verschiebung nach rechts (gen Osten) positive Werte für  $x$ . Die Höhe des Absorberrohrs über dem Horizont ist durch  $h$  beschrieben, die Sekundärreflektorbreite durch  $b$  und die Spiegelbreite durch  $s$ .



**Abb. 10.1:** Ein um  $x$  horizontal verschobener Spiegel der Länge  $s$

Auch die für die Modellierung relevanten Winkel sind Abbildung 10.1 zu entnehmen. Dabei steht  $\gamma$  für den Neigungswinkel des Spiegels,  $\alpha$  für den Sonneneinfallswinkel<sup>1</sup> und  $\beta$  für den Ausfallswinkel der Sonnenstrahlen.

Für den 21. Juni liegt der Sonneneinfallswinkel  $\alpha$  abhängig von der aktuellen Tageszeit  $t \in [0, 24)$  für Daggett (Kalifornien) vor. Dieser wird als Datengrundlage für die Modellierung des Kraftwerks eingesetzt. Der Einfallswinkel sowie alle weiteren Winkel werden im **Gradmaß** angegeben.

Die bei der Modellierung verwendeten Variablen sind zur Übersicht in Tabelle 10.1 zusammengefasst.

**Tab. 10.1:** Variablen und Funktionen bei der Modellierung des Fresnelkraftwerks

Symbol	Beschreibung
$\alpha$	Einfallswinkel der Sonnenstrahlen
$\gamma$	Neigungswinkel des Spiegels
$\beta$	Ausfallswinkel der reflektierten Sonnenstrahlen
$h$	Höhe des Absorberrohrs über dem Horizont
$s$	Breite der Spiegel
$b$	Breite des Sekundärreflektors
$t$	Tageszeit in h
$a$	Anteil der Leistung, die das Rohr erreicht
$e$	Länge der bestrahlten Strecke senkrecht zu den einfallenden Sonnenstrahlen
$H_i$	$i$ -ter Spiegel
$x$	Position eines Spiegels relativ zum Absorberrohr
$x_i$	Position des Spiegels $H_i$ relativ zum Absorberrohr
$N$	Anzahl der Spiegel
$\mathbf{x}$	Spiegelkoordinatenvektor
$f$	Fehlerstrecke bei gestörter Einstellung des Neigungswinkels
$p$	bestrahlte Strecke auf Rohrhöhe
$\epsilon$	Fehler bei der Einstellung des Neigungswinkels
$I_{\text{sun}}$	Bestrahlungsstärke der direkten Sonnenstrahlung
$P_{\text{mirror}}$	Funktion, die die Leistung am Spiegel beschreibt
$P_{\text{tube}}$	Funktion, die die Leistung am Rohr beschreibt
$P_{\text{tubeAtmLoss}}$	Funktion, die die Leistung am Rohr unter atmosphärischen Abschwächungseffekten beschreibt
$P_{\text{tubeError}}$	Funktion, die die Leistung am Rohr bei fehlerhafter Spiegelausrichtung beschreibt
$E_{\text{tube}}$	Funktion, die den Energieumsatz am Rohr beschreibt
$E_{\text{tubeEffects}}$	Funktion, die den Energieumsatz unter Verschattungseffekten beschreibt
$\mathbf{v}_{\text{sun}}$	Richtungsvektor der einfallenden Sonnenstrahlen

---

<sup>1</sup>Der hier eingeführte Sonneneinfallswinkel  $\alpha$  entspricht dem in Abschnitt 9.1 beschriebenen transversalen Einfallswinkel  $\alpha_t$ .

## 10.1 Ausrichtung der Spiegel

Der gewünschte Ausfallswinkel  $\beta$  ist von der Position  $x$  des Spiegels und der Höhe des Absorberrohrs  $h$  über dem Boden abhängig (vgl. Abb. 10.1). Mithilfe verschiedener rechtwinkliger Dreiecke ergibt sich für den Ausfallswinkel  $\beta$  die Fallunterscheidung<sup>2</sup>

$$\beta = \begin{cases} \arctan\left(\frac{h}{|x|}\right) & \text{für } x < 0 \\ 90^\circ & \text{für } x = 0 \\ 180^\circ - \arctan\left(\frac{h}{x}\right) & \text{für } x > 0. \end{cases} \quad (10.1)$$

Da sich der Sonnenstand im Laufe des Tages ändert, muss der Spiegel im Tagesverlauf gedreht werden. Damit der reflektierte Sonnenstrahl genau auf das Absorberrohr trifft, soll der Neigungswinkel des Spiegels an den Sonneneinfallswinkel angepasst werden. Unter Berücksichtigung des Reflexionsgesetzes ergibt sich für den erforderlichen Neigungswinkel  $\gamma$

$$\gamma = \frac{1}{2} \cdot (\alpha + \beta). \quad (10.2)$$

## 10.2 Modellierung von Leistung und umgesetzter Energie

Ausgehend von der vereinfachten Situation (lediglich ein Spiegel) wird ein erstes Modell für die Leistung des Kraftwerks entwickelt. Anschließend wird die Optimierung der Rohrhöhe basierend auf dem erweiterten Modell durchgeführt. Die Optimierung bietet Anlass zur Validierung und Bewertung des Modells. Es wird ersichtlich, dass verschiedene reale Phänomene in dem bisherigen Modell außer Acht gelassen wurden und der Einbau von Modellverbesserungen notwendig ist.

### 10.2.1 Leistung am Absorberrohr

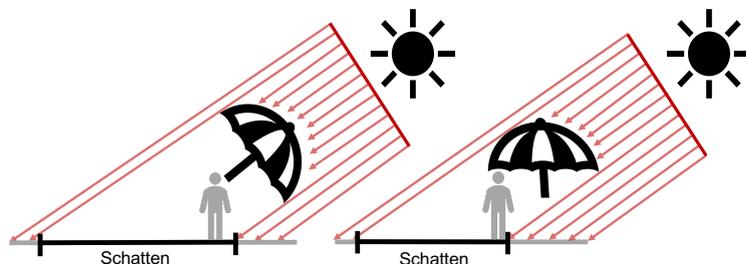
Zu den wichtigsten Kennzahlen eines Kraftwerkes gehört die Leistung  $P$  (engl. *power*). Sie bezeichnet die in einer bestimmten Zeitspanne  $\Delta t$  umgesetzte Energie  $\Delta E$  bezogen auf diese Zeitspanne ( $P = \Delta E / \Delta t$ ). Über die Leistung kann ermittelt werden, wie viel Energie im Kraftwerk umgesetzt wird.

Als Ausgangspunkt für die Berechnung der Leistung am Spiegel wird die Bestrahlungsstärke der direkten Sonnenstrahlung  $I_{\text{sun}}$  verwendet (vgl. Abschn. 9.1). Diese ist über den Tag gesehen nicht konstant, sondern hängt von der Tageszeit ab. Die Stärke der Sonnenstrahlung wird für gewöhnlich in der Einheit Watt pro Quadratmeter (=  $\text{W}/\text{m}^2$ ) angegeben. Da die Problemstellung auf den Querschnitt eines Kraftwerks

<sup>2</sup>Die Abhängigkeiten zwischen verschiedenen Größen, bspw. die Abhängigkeit des Ausfallswinkels von der Rohrhöhe und der Spiegelposition  $\beta(h, x)$ , werden in diesem Kapitel nicht immer explizit angegeben. Dies dient der Übersichtlichkeit von längeren Formeln.

und damit auf den zweidimensionalen Fall reduziert wurde, wird mit der Einheit Watt pro Meter ( $= \text{W/m}$ ) gearbeitet.<sup>3</sup>

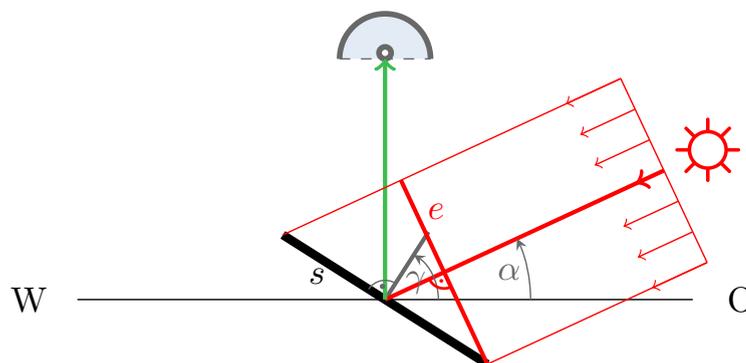
Die Bestrahlungsstärke  $I_{\text{sun}}$  beschreibt die Leistung, die übertragen wird, wenn die Sonnenstrahlen *senkrecht* auf eine Fläche von einem Quadratmeter treffen. Treffen die Strahlen nicht senkrecht auf, so ist die Leistung geringer. Es gilt: je spitzer oder stumpfer der Einfallswinkel, desto geringer die Leistung. Dieser Effekt wird als **Kosinuseffekt** bezeichnet. Ein Gedankenexperiment verdeutlicht diesen Effekt:



**Abb. 10.2:** Visualisierung des Kosinuseffekts am Beispiel des Schattenwurfs. Links fallen die Sonnenstrahlen senkrecht auf den Schirm, rechts treffen die Sonnenstrahlen schräg auf

Wird ein Sonnenschirm so gegen die Sonne gehalten, dass dieser senkrecht zu den Sonnenstrahlen steht (vgl. Abb. 10.2 links), so können mehr Strahlen abgefangen werden als in dem Fall, in dem die Sonnenstrahlen schräg auf den Schirm auftreffen (vgl. Abb. 10.2 rechts). Dies wird auch durch den Schatten ersichtlich, der im linken Fall (Schirm steht senkrecht zu den Strahlen) länger ist als im rechten Fall.

Analog verhält es sich bei der Sonnenstrahlung, die auf den Spiegel fällt. In Abbildung 10.3 ist die Länge der effektiven Strecke  $e$ , die orthogonal zu den Sonnenstrahlen steht, zu erkennen. Sie nimmt die *gleiche Leistung* durch die Sonnenstrahlung auf wie der Spiegel. Sie steht jedoch senkrecht auf den Sonnenstrahlen – was eine direkte Berechnung der zugeführten Leistung unter Verwendung von  $I_{\text{sun}}$  ermöglicht.



**Abb. 10.3:** Darstellung der Hilfsstrecke mit Länge  $e$ , die senkrecht auf den Sonnenstrahlen steht

<sup>3</sup>Diese Vereinfachung lässt sich auch so interpretieren, dass die Tiefe von Spiegel und Rohr auf 1 m festgelegt wird.

Die Länge der Strecke  $e$  lässt sich mithilfe geometrischer Überlegungen berechnen (vgl. [Hattebuhr et al., 2015](#), S. 358). Eine mögliche Lösung lautet

$$e = s \cdot \cos(\gamma - \alpha). \quad (10.3)$$

Die Leistung am Spiegel  $P_{\text{mirror}}$  ergibt sich dann durch

$$P_{\text{mirror}} = I_{\text{sun}} \cdot e. \quad (10.4)$$

Ziel ist nun, die Leistung am Rohr, die im Folgenden mit  $P_{\text{tube}}$  bezeichnet wird, zu berechnen. Dazu wird zunächst angenommen, dass diese mit der Leistung am Sekundärreflektor übereinstimmt. Zur Erinnerung: der Sekundärreflektor dient dazu, die Strahlen, die von unten knapp am Rohr vorbei auf ihn fallen, doch noch auf das Rohr zu reflektieren.

Im Allgemeinen entspricht die Leistung am Sekundärreflektor nicht der bereits modellierten Leistung am Spiegel. Dies begründet sich zum einen dadurch, dass das reflektierte Strahlenpaket breiter als der Sekundärreflektor sein kann (vgl. [Abb. 10.4](#)). Zum anderen treffen die reflektierten Strahlen bei einem horizontal verschobenen Spiegel ( $x \neq 0$ ) schräg auf das Rohr, womit erneut der Kosinuseffekt berücksichtigt werden muss. Dazu kann eine weitere Strecke  $e'$ , die senkrecht auf den reflektierten Strahlen steht, konstruiert werden (vgl. [Abb. 10.4](#)). An dieser wird die gleiche Leistung umgesetzt wie am Sekundärreflektor und damit am Absorberrohr.

Die Strecke  $e'$  ergibt sich durch

$$e' = b \cdot \cos(|90^\circ - \beta|). \quad (10.5)$$

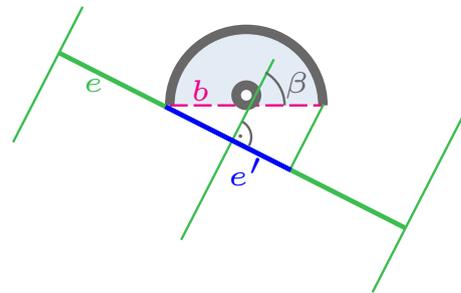
Der relative Anteil der Leistung am Spiegel  $a$ , der tatsächlich am Absorberrohr ankommt, lautet

$$a = \min\left(1, \frac{e'}{e}\right). \quad (10.6)$$

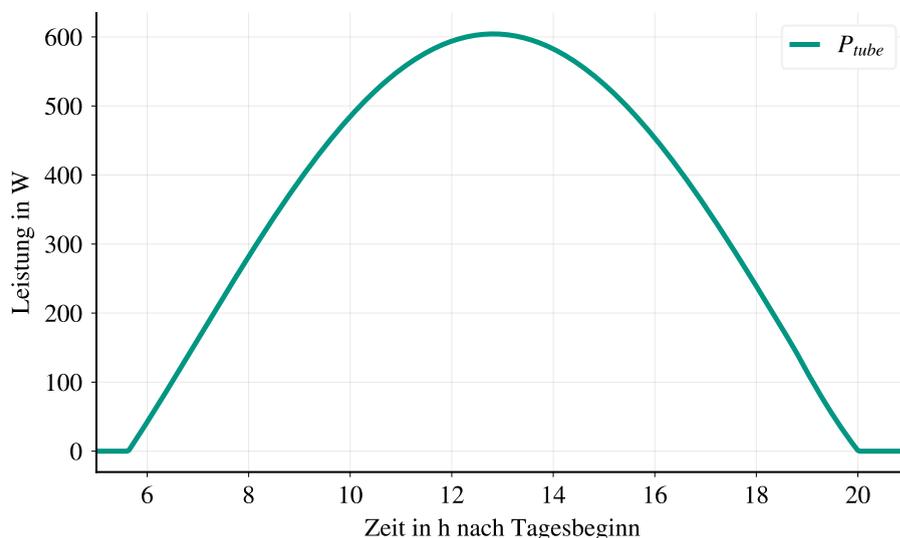
Für die am Absorberrohr generierte Leistung  $P_{\text{tube}}$  erhalten wir damit

$$P_{\text{tube}}(t, b, h, s, x) = P_{\text{mirror}}(t, b, h, s, x) \cdot a. \quad (10.7)$$

Die Funktion  $P_{\text{tube}}$  beschreibt die momentane Leistung am Rohr abhängig von Tageszeit  $t$ , Breite des Sekundärreflektors  $b$ , Spiegelposition  $x$  und Spiegelbreite  $s$  (vgl. [Abb. 10.5](#)). Dabei wurde zunächst angenommen, dass der Schatten, der vom Sekundärreflektor auf den Spiegel fällt, vernachlässigt werden kann.



**Abb. 10.4:** Kosinuseffekt am Sekundärreflektor



**Abb. 10.5:** Leistung am Rohr im Tagesverlauf generiert durch einen Spiegel unter dem Rohr ( $b = 0.8$  m und  $s = 1.1$  m)

## 10.2.2 Energieumsatz eines Tages

Mithilfe der (momentanen) Leistung lässt sich berechnen, wie viel Energie ein Kraftwerk über einen gewissen Zeitraum (bspw. einen Tag oder ein Jahr) umsetzt. Die innerhalb eines Tages umgesetzte Energie entspricht dem Integral von 0 bis 24 über der Funktion  $P_{\text{tube}}$  von  $t$  (momentane Leistung am Rohr). Das Integral wird über eine Riemann-Summe approximiert und nicht analytisch berechnet. Gründe dafür sind:

- Bei Simulationen in Forschung und Industrie existiert oft keine geschlossene Darstellung der Leistung (als Funktion der Zeit) oder diese ist zu komplex, um sie zu integrieren. Dies wird auch bei unserem Modell der Fall sein, wenn im späteren Verlauf Effekte wie Schattenwurf berücksichtigt werden.
- Aufgrund von Unsicherheiten in den Daten (bspw. zur Stärke der Sonneneinstrahlung) täuscht das analytische Vorgehen eine Genauigkeit vor, die die Daten nicht einhalten können (vgl. [Krahforst, 2016](#), S. 114).

Die innerhalb eines Tages umgesetzte Energie wird im Folgenden durch die Funktion  $E_{\text{tube}}$ , die von den Parametern  $b$ ,  $h$ ,  $s$  und  $x$  abhängt, beschrieben. Diese Energie kann näherungsweise bestimmt werden durch

$$E_{\text{tube}}(b, h, s, x) = \sum_{j=1}^n P_{\text{tube}}(t_j, b, h, s, x) \cdot \Delta t.$$

Dabei wird angenommen, dass der Tag in  $n \in \mathbb{N}$  Zeitintervalle der Länge  $\Delta t$  eingeteilt ist und dass die Leistung auf jedem der Zeitintervalle konstant ist. Der Zeitschritt  $\Delta t$  sollte nicht zu klein gewählt werden. Zum einen sind die Daten zu Stärke und Stand der Sonne nicht genau genug, um einen sehr kleinen Zeitschritt zu begründen. Zum anderen nimmt die Laufzeit der Simulation mit kleiner werdendem Zeitschritt zu (vgl. [Krahforst, 2016](#), S. 115). Wird der Zeitschritt hingegen zu groß

gewählt, so führt dies zu ungenauen Ergebnissen.<sup>4</sup> Als Ergebnis für die über einen Tag umgesetzte Energie ergibt sich 5426.1 Wh mit  $b = 0.8$  m,  $h = 6$  m,  $s = 1.1$  m,  $x = -2$  m und  $\Delta t = 0.1$  h.

### 10.2.3 Energie am Rohr erzeugt durch mehrere Spiegel

Wie in Abbildung 9.2 zu erkennen ist, reflektieren mehrere parallel aufgestellte Spiegel das Sonnenlicht auf das Rohr. Eine naheliegende Erweiterung des Modells stellt die Berechnung der Energie umgesetzt durch alle Spiegel dar. Dabei nehmen wir zunächst an, dass keine Effekte (wie Schattenwurf) zwischen den einzelnen Spiegeln auftreten. Unter dieser Annahme ergibt sich die umgesetzte Energie durch Aufsummieren der Energie, die durch die einzelnen Spiegel umgesetzt wird. Das Modell für die Energie über einen Zeitraum kann damit erweitert werden zu

$$E_{\text{tube}}(b, h, s, \mathbf{x}) = \sum_{j=1}^n \sum_{i=1}^N P_{\text{tube}}(t_j, b, h, s, x_i) \cdot \Delta t. \quad (10.8)$$

Hier bezeichnet  $x_i$  für  $i = 1, \dots, N$  den Mittelpunkt eines beliebigen Spiegels  $H_i$ . Die Mittelpunkte der  $N$  Spiegel werden im Positionsvektor  $\mathbf{x} = (x_1, x_2, \dots, x_N)$  zusammengefasst.

## 10.3 Modellierung von Effekten zwischen Spiegeln

Eines der übergeordneten Ziele bei der Simulation von Kraftwerken ist nicht nur die Modellierung der Effizienz oder der Energieausbeute eines Kraftwerks, sondern insbesondere die Maximierung dieser durch eine geeignete Wahl verschiedener Kraftwerkparameter. Besonders bei der Optimierung der Spiegelpositionen sollten Effekte wie a) die Verschattung einfallender Strahlen durch nebenstehende Spiegel, b) die Verschattung durch den Sekundärreflektor sowie c) die Blockierung reflektierter Strahlen durch nebenstehende Spiegel vor der Optimierung berücksichtigt werden. Die Vereinfachung, dass die Spiegel keinerlei Einfluss aufeinander haben, wird verworfen. In diesem Abschnitt wird lediglich die *Verschattung einfallender Sonnenstrahlen durch nebenstehende Spiegel* (vgl. Abb. 10.6) modelliert. Die Blockierung reflektierter Sonnenstrahlen durch nebenstehende Spiegel und die Verschattung einfallender Sonnenstrahlen durch den Sekundärreflektor werden im Anhang A.1 skizziert.

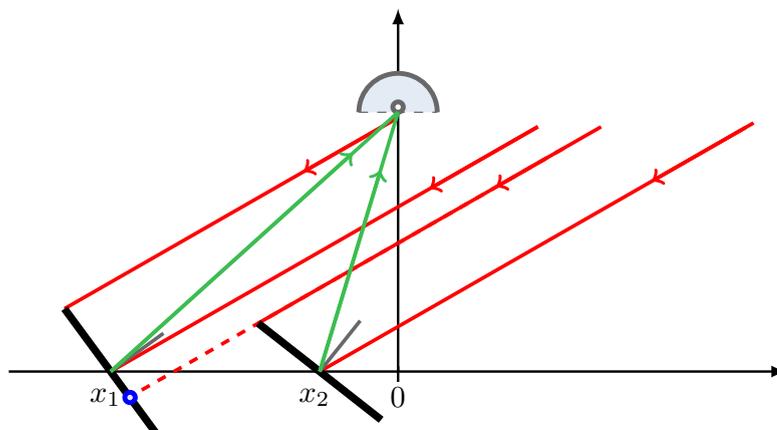
Bei der Modellierung der Verschattungseffekte können verschiedene Ansätze gewählt werden.

1. Einzelne aufgrund der Geometrie des Kraftwerks explizit ausgewählte Strahlen können verfolgt und die Verschattung geometrisch exakt modelliert werden.

<sup>4</sup>In der Praxis wird der Energieumsatz üblicherweise über einen längeren Zeitraum, beispielsweise ein Jahr, simuliert. Unter anderem Kistler (1986, S. 67) weist die Wahl eines einstündigen Zeitschrittes als guten Kompromiss zwischen Genauigkeit und Rechenaufwand aus. Da wir nur einen Tag betrachten, werden hier kleinere Zeitschritte, bspw.  $\Delta t = 0.1$  h, eingesetzt.

2. Der in der Forschung gängige Ansatz eines Raytracers kann genutzt werden. Bei diesem werden die Effekte näherungsweise modelliert.

Da der geometrische Ansatz von Lernenden leichter eigenständig modelliert werden kann, wird dieser in der Folge dargestellt.



**Abb. 10.6:** Skizze des Schattenwurfs für zwei Spiegel auf den Positionen  $x_1$  und  $x_2$

### Geometrisch exakte Modellierung

Bei der geometrisch exakten Modellierung können unterschiedliche Ansätze zur Beschreibung der Spiegel und der Sonnenstrahlen verfolgt und implementiert werden. Denkbare Ansätze, die unterschiedliche Vor- und Nachteile – insbesondere im Hinblick auf die spätere Durchführung mit Lernenden – haben, sind:

- Die Beschreibung von Spiegel und Sonnenstrahlen durch Graphen linearer Funktionen. Diese Modellierung greift jedoch nicht für senkrecht einfallende oder reflektierte Strahlen, hat jedoch den Vorteil, dass die Modellierung mit Lernenden der Mittelstufe durchgeführt werden kann.
- Die Modellierung von Spiegel und Sonnenstrahlen durch (mathematische) Strahlen beziehungsweise Halbgeraden in Parameterform. Dieser Ansatz kann auch für die Beschreibung senkrechter Strahlen genutzt werden.

Wir verfolgen die Beschreibung von Spiegeln und Strahlen über den in der linearen Algebra beziehungsweise analytischen Geometrie zu verortenden Ansatz. Die Spiegel und Sonnenstrahlen werden als Halbgeraden beziehungsweise (mathematische) Strahlen und Strecken modelliert.

### Beschreibung der Spiegel als Strecken

Da die Spiegel auf beiden Seiten begrenzt sind, werden sie als Strecken beschrieben. Als Aufpunkt kann der Mittelpunkt  $(x_i|0)$  eines Spiegels  $H_i$  genutzt werden. Unter Verwendung der Koordinaten der rechten Spiegelecke  $(x_r(H_i)|y_r(H_i))$  kann die

Strecke, die den Spiegel beschreibt, über die Menge

$$\left\{ \begin{pmatrix} x_i \\ 0 \end{pmatrix} + \lambda \begin{pmatrix} x_r(H_i) - x_i \\ y_r(H_i) \end{pmatrix} \mid \lambda \in [-1, 1] \right\}$$

dargestellt werden. Die Koordinaten der Spiegelecke ergeben sich bei gegebener Spiegelausrichtung und fester Spiegelbreite über trigonometrische Überlegungen, die hier nicht weiter ausgeführt werden.

Um bestimmen zu können an welchen Stellen Verschattungen auftreten, müssen auch die Sonnenstrahlen mathematisch beschrieben werden.

### Beschreibung der einfallenden Sonnenstrahlen als Halbgeraden

Der Richtungsvektor<sup>5</sup>  $\mathbf{v}_{sun}$  eines einfallenden Sonnenstrahls lässt sich wiederum über die Verwendung von Sinus und Kosinus im rechtwinkligen Dreieck bestimmen. Damit ergibt sich der Richtungsvektor der einfallenden Strahlen durch

$$\mathbf{v}_{sun} = \begin{pmatrix} \cos(\alpha) \\ \sin(\alpha) \end{pmatrix},$$

wobei dies dem Richtungsvektor entspricht, der der Sonne entgegen zeigt. Durchaus wären auch andere Ansätze zur Bestimmung des Richtungsvektors denkbar, wie die Verwendung von Rotationsmatrizen.

Ein einfallender Strahl, der auf einen beliebigen Punkt  $(a_1, a_2)^T$  trifft, kann letztlich geschrieben werden als Halbgerade

$$\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} + \lambda \begin{pmatrix} \cos(\alpha) \\ \sin(\alpha) \end{pmatrix} \quad \text{mit } \lambda \geq 0.$$

In ähnlicher Weise lassen sich auch reflektierte Sonnenstrahlen als Halbgeraden modellieren (vgl. Anh. A.1).

Aufgrund der Annahme, dass eine feste herstellerbedingte Breite für alle Spiegel  $s$  vorgegeben ist, kann eine mögliche Verschattung zu einem beliebigen Zeitpunkt  $t$  nur durch direkt benachbarte Spiegel auftreten. Die tatsächlich beleuchteten Bereiche eines Spiegels werden über zahlreiche Schnittpunktsberechnungen zwischen den Strecken, die die Spiegel beschreiben, und ausgewählten einfallenden Strahlen bestimmt.

### Schattenwurf durch nebenstehende Spiegel

Exemplarisch wird der Schatten auf dem Spiegel  $H_{i_0}$  modelliert (vgl. Abb. 10.6). Je nach aktuellem Einfallswinkel  $\alpha$  können drei Fälle unterschieden werden:

<sup>5</sup>Im Material für die Lernenden werden Variablen, die Vektoren darstellen, nicht fett gedruckt. Stattdessen wird die Schreibweise mit Vektorpfeilen (z. B.  $\vec{v}$ ) verwendet, da dies der im Mathematikunterricht üblicherweise genutzten Notation entspricht.

**1. Fall:**  $\alpha < 90^\circ$  (Vormittag)

Es wird überprüft, ob der rechtsstehende Spiegel  $H_{i_0+1}$  einen Schatten auf den betrachteten Spiegel  $H_{i_0}$  wirft. Es wird davon ausgegangen, dass rechts neben dem Spiegel  $H_{i_0}$  mindestens ein weiterer Spiegel steht. Der einfallende Sonnenstrahl, der durch die linke Ecke  $(x_l(H_{i_0+1})|y_l(H_{i_0+1}))$  des Spiegels  $H_{i_0+1}$  verläuft, wird durch

$$g : \mathbf{y} = \begin{pmatrix} x_l(H_{i_0+1}) \\ y_l(H_{i_0+1}) \end{pmatrix} + \lambda \mathbf{v}_{sun} \quad \text{mit } \lambda \geq 0 \quad (10.9)$$

modelliert und die Strecke des Spiegels  $H_{i_0}$  durch

$$f : \mathbf{y} = \begin{pmatrix} x_{i_0} \\ 0 \end{pmatrix} + \mu \begin{pmatrix} x_r(H_{i_0}) - x_{i_0} \\ y_r(H_{i_0}) \end{pmatrix} \quad \text{mit } \mu \in [-1, 1]. \quad (10.10)$$

Anschließend werden die Parameter  $\lambda_0$  und  $\mu_0$  im Schnittpunkt von Halbgerade und Strecke bestimmt. Falls  $\mu_0 \in [-1, 1]$  und  $\lambda_0 \geq 0$  gilt, liegt eine Verschattung vor. Das Intervall  $[-1, \mu_0]$  enthält die Werte des Parameters  $\mu$ , die die nicht verschattete Spiegelstrecke beschreiben. Für den einfallenden Sonnenstrahl, der durch die *rechte Ecke* des Spiegels  $H_{i_0+1}$  verläuft, wird analog vorgegangen.

**2. Fall:**  $\alpha > 90^\circ$  (Nachmittag)

Analog zu Fall 1 wird eine Verschattung durch den Spiegel  $H_{i_0-1}$  überprüft.

**3. Fall:**  $\alpha = 90^\circ$  (Mittag)

Analog zu Fall 1 und 2 wird eine Verschattung durch die Spiegel  $H_{i_0+1}$  und  $H_{i_0-1}$  überprüft.

Insgesamt ergibt sich für den Spiegel  $H_{i_0}$  eine Menge an Werten für den Parameter  $\mu$ , die die Spiegelstrecke beschreibt, die tatsächlich beleuchtet wird. Nach

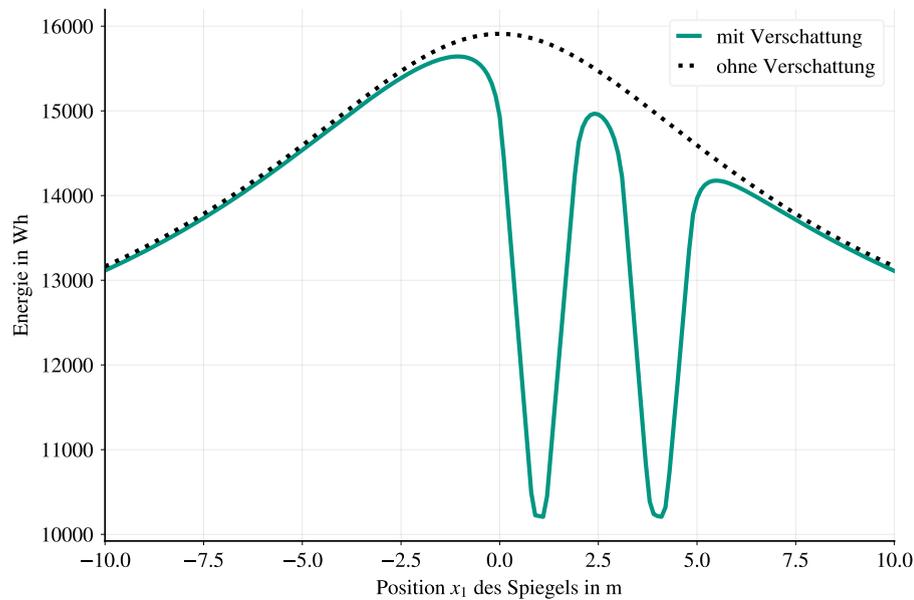
- a) der Modellierung der Blockierung reflektierter Sonnenstrahlen,
- b) der Verschattung durch den Sekundärreflektor und
- c) basierend auf weiteren geometrischen Überlegungen zur Berechnung des Anteils der Sonnenstrahlen, der auf den Sekundärreflektor trifft (vgl. Anh. A.1),

ergibt sich schließlich die Funktion  $E_{\text{tubeEffects}}$  mit  $E_{\text{tubeEffects}}(b, h, s, \mathbf{x})$ , die den Energieumsatz eines Tages unter Berücksichtigung sämtlicher Verschattungseffekte beschreibt.

Der Einfluss der Verschattung einfallender Strahlen wird in Abbildung 10.7 deutlich. Zwei Spiegel stehen bereits fest auf den Positionen  $x_2 = 1$  m und  $x_3 = 4$  m. Es wird ersichtlich, dass die umgesetzte Energie abnimmt, wenn der variable Spiegel auf Position  $x_1$  näher an die beiden feststehenden Spiegel herangerückt wird.

**‡ Ein Blick in Richtung Optimierung**

Im Hinblick auf die spätere Optimierung können wir festhalten, dass die Funktion mit Verschattungseffekten mehrere lokale Maximalpunkte aufweist. Diese Beobachtung wird bei der Optimierung der Spiegelpositionen in Abschnitt 11.2 erneut aufgegriffen.



**Abb. 10.7:** Energie in Abhängigkeit der Position  $x_1$  eines Spiegels. Zwei weitere Spiegel sind fest auf den Positionen  $x_2 = 1$  m und  $x_3 = 4$  m positioniert ( $b = 0.8$  m,  $h = 6$  m  $s = 1.1$  m,  $\Delta t = 0.05$  h).

Zudem weist die Funktion, die die Leistung am Rohr unter Verschattungseffekten beschreibt, Knickstellen auf. Durch die numerische Integration beim Übergang von Leistung zu Energie werden diese Knickstellen zwar etwas geglättet, dennoch wird mit einer Zielfunktion gearbeitet, die nicht überall differenzierbar ist.

### Approximative Modellierung (Raytracer)

Wir haben geometrisch exakt einzelne, ausgewählte Strahlen mathematisch dargestellt und bestimmt, wo diese auf den Spiegel beziehungsweise den Sekundärreflektor auftreffen. In der Forschung wird vielfach ein approximatives Modell gewählt, ein sogenannter Raytracer. Der Name (engl.: *ray* = Strahl und *trace* = Verfolgung / Spur) erlaubt bereits Rückschlüsse auf die Funktionsweise des Modells: Es werden Strahlen verfolgt, jedoch nicht nur einzelne Strahlen, die aufgrund der Geometrie des Kraftwerks ausgewählt werden, sondern sehr viele Strahlen. Der Einfluss der Verschattung auf den Energieumsatz wird dann basierend auf den verfolgten Strahlen approximativ modelliert. In der Praxis existieren verschiedene Simulationstools, die bei der Planung und Optimierung von Solarkraftwerken eingesetzt werden können. Beispiele stellen die Tools SolTrace (vgl. [Wendlin, 2003](#)) oder SolFast (vgl. [Roccia et al., 2012](#)) dar. Viele dieser Simulationstools basieren auf sogenannten *Monte-Carlo-Raytracern*. Bei diesen werden Millionen *zufällige* Strahlen erzeugt, verfolgt und die erwartete Leistung des Kraftwerks simuliert (vgl. [Richter et al., 2016](#), S. 3). Ein Vorteil von Monte-Carlo-Raytracern ist die hohe Genauigkeit der simulierten Leistung, wenn viele Strahlen verfolgt werden. Damit einher geht jedoch der Nachteil, dass diese Methoden sehr rechenintensiv sind. Dies ist insbesondere bei Optimierungsverfahren problematisch, bei denen wiederholte Auswertungen der Simulation erforderlich sind.

# 11 Optimierung verschiedener Kraftwerkparameter

Bei der Optimierung diverser Kraftwerkparameter können verschiedene Ziele verfolgt werden. In der Forschung kommen als Zielkriterien beispielsweise die *jährliche Gesamtenergie*, die *optische Effizienz* oder die sogenannte *Levelized Cost of Electricity* zum Einsatz (vgl. Richter et al., 2016, S. 241; Lutchman et al., 2014, S. 1429).

Bei den in diesem Kapitel diskutierten Optimierungsproblemen wird eine der folgenden Funktionen als Zielfunktion verwendet:

- $P_{\text{tube}}$  mit  $P_{\text{tube}}(t, b, h, s, x_i)$ , die die Leistung beschreibt, die zu einem **festen Zeitpunkt**  $t$  durch **einen** Spiegel auf Position  $x_i$  erzeugt wird (vgl. Abschn. 11.1 zur Optimierung der Rohrhöhe).
- $E_{\text{tube}}$  mit  $E_{\text{tube}}(b, h, s, \mathbf{x})$  beziehungsweise  $E_{\text{tubeEffects}}$  mit  $E_{\text{tubeEffects}}(b, h, s, \mathbf{x})$ , die die Energie über einen Tag erzeugt durch **mehrere** Spiegel auf den Positionen  $\mathbf{x} = (x_1, x_2, \dots, x_N)$  mit oder ohne Verschattungseffekte beschreibt (vgl. Abschn. 11.2 zur Optimierung der Spiegelpositionen).

Es wurden unterschiedliche Zielfunktionen für die Optimierung gewählt, da dies auch im entwickelten Lernmaterial der Fall ist. Im Lernmaterial für die Mittelstufe wird die Rohrhöhe basierend auf dem Modell für die Leistung optimiert. Die umgesetzte Energie wird nicht als Zielkriterium verwendet, da die Lernenden den Modellierungsschritt „Leistung zu Energie“ nicht durchführen.

In den beiden folgenden Abschnitten wird zunächst die Optimierung der Rohrhöhe (Kern des Workshops für die Mittelstufe) und anschließend die Optimierung der Spiegelpositionen (Kern des Workshops für die Oberstufe) beschrieben.

## 11.1 Optimierung der Absorberrohrhöhe

Ziel ist es, die Absorberrohrhöhe zu bestimmen, bei der die **maximale Leistung** umgesetzt wird. Als Zielfunktion der Optimierung wird die Funktion  $P_{\text{tube}}$  mit  $P_{\text{tube}}(t, b, h, s, x)$  betrachtet (vgl. Abb. 11.1). Formal wird das folgende nichtlineare Optimierungsproblem

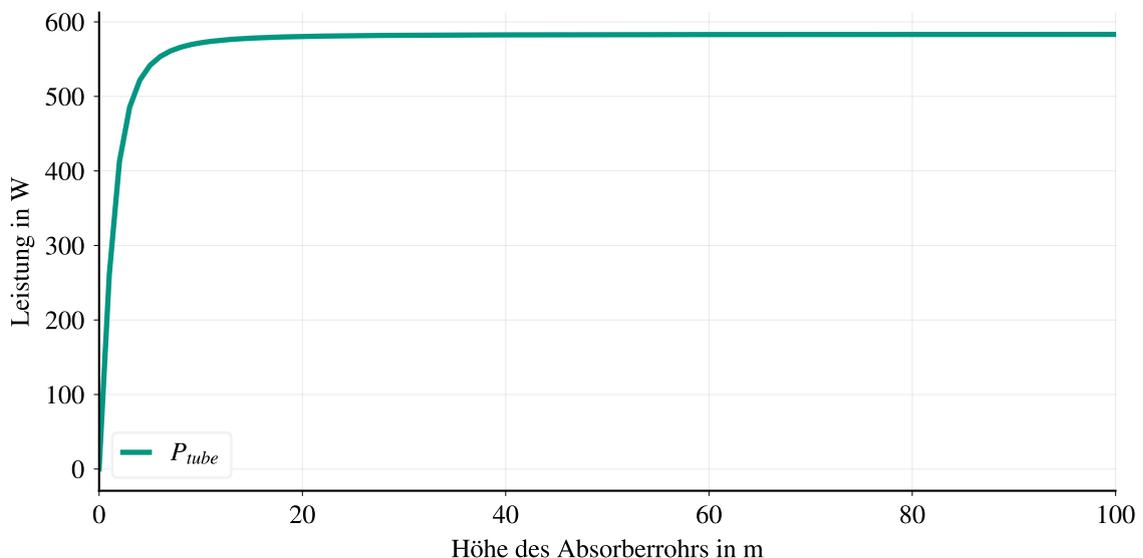
$$\begin{array}{ll} \min_h & - P_{\text{tube}}(t, b, h, s, x) \\ \text{u. d. Nb.} & h \geq 0 \end{array} \quad (11.1)$$

gelöst. Die Nebenbedingung  $h \geq 0$  wird dabei auch als *Nichtnegativitätsbedingung* bezeichnet (vgl. Stein, 2021b, S. 2). Diese wird jedoch im Folgenden vernachlässigt, da sie wie aus Abbildung 11.1 ersichtlich wird, keinen Einfluss auf die optimale Rohrhöhe hat. Die Parameter  $t, b, s$  und  $x$  sind fest gewählt.

Die Optimierung der Rohrhöhe wird zunächst basierend auf dem Modell für die Leistung, beschrieben durch die Funktion  $P_{\text{tube}}$ , durchgeführt, welche in Abschnitt 10.2.1 entwickelt wurde. Bei diesem Modell wurden verschiedene reale Phänomene, wie beispielsweise der Einfluss der Atmosphäre auf die reflektierte Sonnenstrahlung, vernachlässigt. Wie sich in diesem Abschnitt zeigen wird, existiert, basierend auf diesem einfachen Modell, kein Minimalpunkt des Optimierungsproblems (11.1) und damit keine optimale Rohrhöhe. Das Modell wird deswegen um zwei Modellverbesserungen erweitert. Dies liefert letztlich ein Optimierungsproblem, bei dem ein Minimalpunkt und damit eine optimale Rohrhöhe existiert.

### 11.1.1 Optimierung mit einfachem Modell

Der Graph der Funktion  $P_{\text{tube}}$  ist in Abbildung 11.1 dargestellt. Die Funktion  $P_{\text{tube}}$  ist konkav und nähert sich für  $h \rightarrow \infty$  asymptotisch der Leistung an, die bei einem Ausfallswinkel von  $\beta = 90^\circ$  erreicht wird. Es existiert kein Minimalpunkt des Optimierungsproblems. Die Leistung wird maximal, wenn das Rohr „unendlich“ hoch gebaut wird.



**Abb. 11.1:** Leistung in Abhängigkeit von der Höhe des Absorberrohrs ( $t = 11$  h,  $b = 0.8$  m,  $s = 1.1$  m,  $x = -2$  m)

Um das Lösen des nichtlinearen Optimierungsproblems (11.1) nicht nur durch Ablezen am Graphen, sondern zudem, wie in der Praxis üblich, mithilfe eines numerischen Verfahrens vorzunehmen, wurde das Optimierungspaket *Optim.jl*<sup>1</sup> der Programmier-

<sup>1</sup>Weitere Informationen zum Paket Optim.jl sind zu finden unter <https://julianlsolvers.github.io/Optim.jl/stable>, letzter Aufruf: 03.08.2021.

sprache Julia eingesetzt. Dieses bietet die Möglichkeit verschiedene Algorithmen (0., 1. oder 2. Ordnung) zur Lösung von unrestringierten oder boxrestringierten nichtlinearen Optimierungsproblemen einzusetzen. Da die Zielfunktion ( $-P_{\text{tube}}$ ) des Minimierungsproblems differenzierbar ist<sup>2</sup>, wurde ein Verfahren 1. Ordnung, konkret das Gradientenverfahren eingesetzt (vgl. Abschn. 2.2.1). Das Verfahren wird abgebrochen, sobald die Norm des Gradienten in der aktuellen Iterierten eine Toleranzgrenze unterschreitet oder, wenn eine maximale Anzahl an Iterationen überschritten wird. Das Verfahren terminiert im vorliegenden Fall, da die gegebene Toleranzgrenze von  $10^{-8}$  unterschritten wurde. Für einen Startwert von  $h_0 = 3$  m liefert das Verfahren eine Höhe von  $h = 3.371 \cdot 10^6$  m mit Leistung 583.3 W.

Bezugnehmend zur Diskussion in Abschnitt 2.1.1 rund um die Existenz und Eindeutigkeit von Minimalpunkten, wird hier unterstrichen, dass (strikte) Konvexität der Zielfunktion  $-P_{\text{tube}}$  nicht ausreicht, um die Existenz eines Minimalpunktes zu garantieren, wenn die zulässige Menge unbeschränkt ist.

Motiviert durch die Erkenntnis, dass das Problem nicht lösbar ist und das Rohr salopp gesprochen ins Unendliche gebaut werden müsste, werden Modellverbesserungen diskutiert und umgesetzt. Dazu können verschiedene Informationen und natürliche Phänomene berücksichtigt werden, die bei der bisherigen Modellierung außer Acht gelassen wurden. Zum Beispiel:

- Materialkosten
- Atmosphärische Effekte und damit verbundene Verluste
- Unsicherheiten und Fehler bei der Ausrichtung der Spiegel
- Wartungs- und Reinigungsaufwand.

### 11.1.2 Modellerweiterung: atmosphärische Abschwächung

Bisher sind wir davon ausgegangen, dass die Energie eines Sonnenstrahls immer gleich bleibt – egal wie lang die Strecke ist, die der reflektierte Strahl vom Spiegel bis zum Rohr zurücklegt. Dies entspricht nicht der Realität.

Tatsächlich **sinkt** die Energie eines Lichtstrahls aufgrund der sogenannten **atmosphärischen Abschwächung**, wenn die zurückgelegte Strecke wächst. Zur Modellierung dieses physikalischen Effekts kann ein einfaches Modell von [Leary und Hankins \(1979\)](#) eingesetzt werden. Dieses beschreibt abhängig von der zurückgelegten Strecke  $d$  wie stark das reflektierte Sonnenlicht abgeschwächt wird. Das Modell ist für Entfernungen von bis zu 1000 m ausgelegt. [Schmitz et al. \(2006\)](#) erweiterten dieses für Entfernungen von mehr als 1000 m.

Basierend auf [Leary und Hankins \(1979\)](#) und [Schmitz et al. \(2006\)](#) wird der Anteil

---

<sup>2</sup>Die Parameter  $t, b, s$  und  $x$  wurden so gewählt, dass in Formel (10.6) für alle  $h > 0$  stets  $\frac{e'}{e} > 1$  gilt. Für andere Parameterwahlen weist die Zielfunktion aufgrund der Minimumfunktion eine Knickstelle auf und ist nicht überall differenzierbar. Dies wird hier nicht weiter diskutiert. Stattdessen sei auf die Modellierung von Einstellungsfehlern verwiesen, bei der die Optimierung mit einer nichtglatten Zielfunktion diskutiert wird (vgl. Abschn. 11.1.3).

der Leistung, die tatsächlich am Ziel (in unserem Fall dem Rohr) ankommt, durch die Funktion  $\text{atmLoss}$  mit

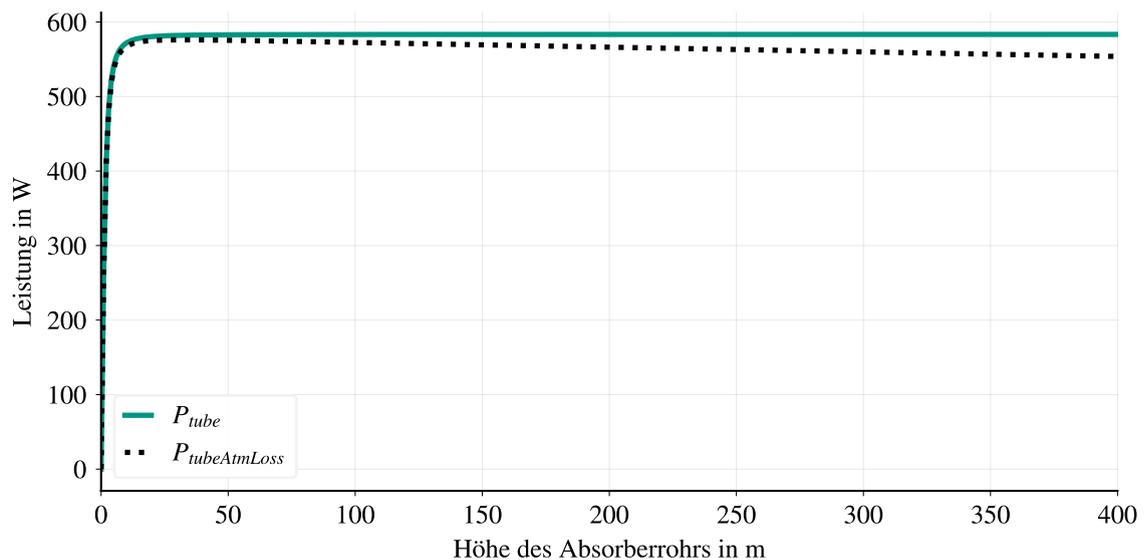
$$\text{atmLoss}(d) = \begin{cases} 0.99321 - 1.176 \cdot 10^{-4}d + 1.97 \cdot 10^{-8}d^2 & \text{für } d \leq 1000 \text{ m} \\ e^{-1.106 \cdot 10^{-4}d} & \text{für } d > 1000 \text{ m} \end{cases} \quad (11.2)$$

modelliert. Die zurückgelegte Strecke  $d$  wird über den Abstand zwischen Spiegelmittelpunkt  $x$  und der Rohrmitte gemäß

$$d = \sqrt{h^2 + x^2} \quad (11.3)$$

beschrieben. Dabei wird angenommen, dass alle vom Spiegel reflektierten Strahlen die gleiche Strecke zurücklegen wie der Strahl, der vom Spiegelmittelpunkt reflektiert wird. Die neue Funktion  $P_{\text{tubeAtmLoss}}$ , die die Leistung am Rohr beschreibt, ist gegeben durch

$$P_{\text{tubeAtmLoss}}(t, b, h, s, x) = P_{\text{tube}}(t, b, h, s, x) \cdot \text{atmLoss}(h, x). \quad (11.4)$$



**Abb. 11.2:** Leistung in Abhängigkeit von der Höhe des Absorberrohrs mit (gestrichelte Linie) und ohne (durchgezogene Linie) atmosphärischer Abschwächung ( $t = 11 \text{ h}$ ,  $b = 0.8 \text{ m}$ ,  $s = 1.1 \text{ m}$ ,  $x = -2 \text{ m}$ )

Der zugehörige Graph ist in Abbildung 11.2 dargestellt. Dieser weist einen konkaven und ab einer Höhe von ungefähr 33 m monoton fallenden Verlauf auf.

### ⚡ Zurück zur Optimierung

Es wird bereits anschaulich ersichtlich, dass die Funktion  $P_{\text{tubeAtmLoss}}$  ihr Maximum und die Funktion  $-P_{\text{tubeAtmLoss}}$  somit ihr Minimum annimmt. Die Existenz eines globalen Minimalpunktes lässt sich nicht nur anschaulich, sondern auch theoretisch über untere Niveaumengen und den *Verschärften Satz von Weierstraß* begründen

(vgl. Abschn. 2.1.1). Die Funktion  $-P_{\text{tubeAtmLoss}}$  ist stetig und beispielsweise für  $c = -540$  erhalten wir als untere Niveaumenge das abgeschlossene Intervall

$$\text{Niv}(-P_{\text{tubeAtmLoss}}, c) = \{h \geq 0 \mid -P_{\text{tubeAtmLoss}}(t, b, h, s, x) \leq c\} = [5.17, 643.08].$$

Die Niveaumenge ist kompakt, sodass gemäß Satz 2 ein globaler Minimalpunkt existiert. Aufgrund der strikten Konvexität der Zielfunktion ist dieser zudem eindeutig.

Die Anwendung des Gradientenverfahrens zur Minimierung von  $-P_{\text{tubeAtmLoss}}$  liefert als Lösung für die Rohrhöhe  $h = 32.4$  m mit einer Leistung von 576.1 W. Das Ergebnis ist nun zwar theoretisch konstruierbar, aber immer noch sehr hoch. Im Folgenden wird deswegen eine zweite Modellerweiterung beschrieben, die in Abschnitt 11.1.4 mit der Berücksichtigung der atmosphärischen Effekte kombiniert wird.

### 11.1.3 Modellerweiterung: Berücksichtigung von Unsicherheiten

Die Spiegel in einem Fresnelkraftwerk werden mithilfe von Motoren im Laufe des Tages gemäß des Sonnenstandes ausgerichtet. Die notwendige Ausrichtung eines Spiegels wurde mithilfe des Neigungswinkels  $\gamma$  bestimmt. Da es durchaus vorkommt, dass einzelne Motoren ungenau arbeiten, entspricht die tatsächliche Ausrichtung der Spiegel nicht immer exakt der gewünschten (vgl. Kincaid et al., 2019, S. 386). Ziel ist es, die Rohrhöhe so zu wählen, dass die Leistung am Rohr unter Berücksichtigung des Einflusses von Einstellungsfehlern maximal ist. Auch bei kleineren Einstellungsfehlern des Spiegels sollen also gute, oder besser gesagt optimale, Werte für die Leistung erreicht werden. Damit wird ein **Optimierungsproblem unter Unsicherheiten** gelöst. Wie in Abschnitt 2.1.2 beschrieben, gibt es Forschungsgebiete, wie die stochastische oder die robuste Optimierung, die sich Optimierungsproblemen unter Unsicherheiten aus unterschiedlichen Blickwinkeln widmen.

#### Modellierung des Einflusses von Einstellungsfehlern auf die Leistung

Wird der Spiegel nicht korrekt eingestellt, so führt dies dazu, dass der mittlere reflektierte Strahl nicht direkt auf das Rohr fällt. Stattdessen verfehlt er das Rohr (links oder rechts) um eine Fehlerstrecke  $f$  (vgl. Abb. 11.3). Der fehlerbehaftete Neigungswinkel wird mit  $\gamma^*$  bezeichnet. Dieser ergibt sich aus dem gewünschten Neigungswinkel durch Addition des Fehlers  $\epsilon$  (vgl. Krahforst, 2016, S. 115 f.)

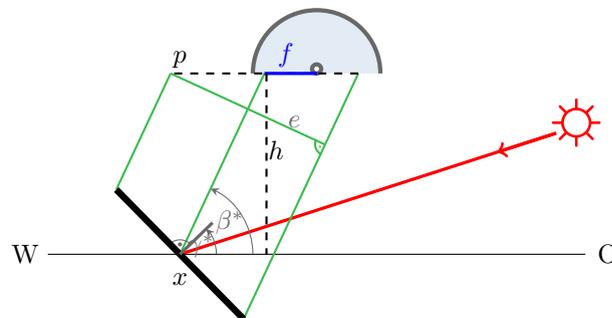
$$\gamma^* = \gamma + \epsilon. \tag{11.5}$$

Wird der Fehler klein gewählt, so trifft der mittlere reflektierte Strahl zwar nicht mehr das Rohr, jedoch weiterhin den Sekundärreflektor. Wird der Fehler hingegen groß gewählt, so trifft der Strahl nicht einmal mehr auf den Sekundärreflektor. Wird der Neigungswinkel  $\gamma$  nicht korrekt eingestellt, so führt dies auch zu einem fehlerhaften Ausfallswinkel  $\beta^*$ . Dieser ergibt sich über

$$\begin{aligned} \beta^* &= 2\gamma^* - \alpha = 2\gamma + 2\epsilon - \alpha \\ &= \beta + 2\epsilon. \end{aligned}$$

Wenn der tatsächliche Ausfallswinkel vom gewünschten Wert abweicht, hat dies Einfluss auf die angestrahlte Strecke auf Rohrhöhe (vgl. Abb. 11.3). Diese Strecke überdeckt den Sekundärreflektor gegebenenfalls nur noch teilweise oder gar nicht mehr. Mit der Funktion  $P_{\text{tube}}$  kann die Leistung am Rohr nicht mehr exakt beschrieben werden.

Im Folgenden wird die Länge der Fehlerstrecke  $f$  exemplarisch für einen Spiegel auf Position  $x < 0$  bestimmt und der Einfluss der Rohrhöhe  $h$  auf die Fehlerstrecke untersucht.



**Abb. 11.3:** Skizze der Reflexion bei fehlerhafter Spiegeleinstellung

Unter Zuhilfenahme von Abbildung 11.3 und Verwendung des Tangens ergibt sich

$$\tan(\beta^*) = \frac{h}{|x| - f} \quad (11.6)$$

und damit

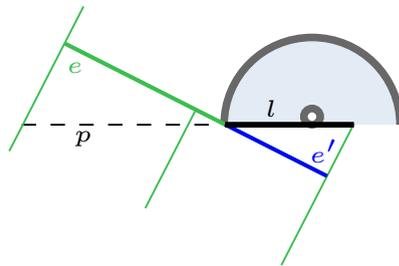
$$f = |x| - \frac{h}{\tan(\beta^*)}. \quad (11.7)$$

Damit die Formel für die Fehlerstrecke verwendet werden kann, muss der fehlerbehaftete Ausfallswinkel  $\beta^*$  im Intervall  $(0^\circ, 90^\circ)$  liegen. Somit darf auch der Fehler nicht beliebig groß werden. Wir nehmen für die folgende Modellierung an, dass der Fehler genügend klein ist.

Zur korrekten Bestimmung der Leistung am Rohr muss die Abweichung um die Fehlerstrecke  $f$  in das Modell aufgenommen und der Anteil der vom Spiegel abgegebenen Leistung, der das Rohr tatsächlich erreicht, neu bestimmt werden. Die Länge der bestrahlten Strecke auf Rohrhöhe wird mit  $p$  bezeichnet (vgl. Abb. 11.3). Mithilfe von Stufen- und Wechselwinkelbeziehungen sowie Anwenden des Sinus oder Kosinus im rechtwinkligen Dreieck ergibt sich für die Strecke  $p$

$$p = \frac{e}{\sin(\beta^*)} = \frac{e}{\cos(90^\circ - \beta^*)}.$$

Wie wir von bisherigen Modellierungsschritten wissen, entspricht die Leistung, die auf der Strecke  $p$  anliegt, der Leistung auf der Hilfsstrecke  $e$  (vgl. Abb. 11.3). Die Leistung, die am Sekundärreflektor ankommt, entspricht jedoch nur dem Anteil, den



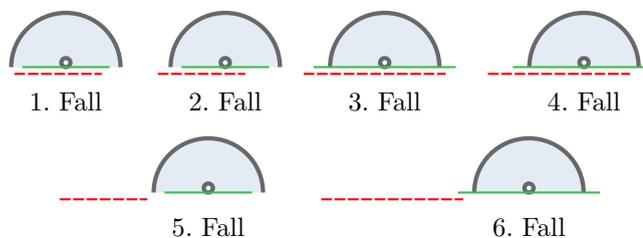
**Abb. 11.4:** Länge des bestrahlten Bereichs des Sekundärreflektors  $l$  bei fehlerhafter Spiegeleinstellung

die Strecke  $l$  an  $p$  ausmacht (vgl. Abb. 11.4). Dieser relative Anteil wird mit  $a_{\text{error}}$  bezeichnet. Mit diesem ergibt sich für die Leistung am Rohr

$$P_{\text{tubeError}} = I_{\text{sun}} \cdot e \cdot a_{\text{error}}.$$

Abhängig von der Größe des Fehlers können unterschiedliche Situationen auftreten, die wiederum in sechs Fälle unterteilt werden können. Diese sechs Fälle sind in Abbildung 11.5 schematisch dargestellt. Die untere, gestrichelt eingezeichnete Strecke stellt jeweils den Bereich dar, der bei fehlerhafter Einstellung des Neigungswinkels tatsächlich bestrahlt wird (also  $p$  basierend auf  $\gamma^*$ ). Die obere Strecke stellt den bestrahlten Bereich dar, wenn der mittlere reflektierte Strahl wie gewünscht auf die Mitte des Rohrs trifft (also  $p$  basierend auf  $\gamma$ ).

- Der Fehler ist so klein, dass bei fehlerhafter Einstellung genau so viele Strahlen auf den Sekundärreflektor treffen wie bei korrekter Einstellung (1. und 3. Fall).
- Ein Teil der Strahlen, die ohne fehlerhafte Einstellung den Reflektor getroffen hätten, fällt jetzt links oder rechts neben diesen (2. und 4. Fall).
- Der Fehler ist so groß, dass alle Strahlen den Sekundärreflektor verfehlen (5. und 6. Fall).



**Abb. 11.5:** Sechs Fälle, die unter Fehlereinfluss am Sekundärreflektor auftreten können

Der gesuchte Anteil, der am Sekundärreflektor ankommt, ist gegeben durch

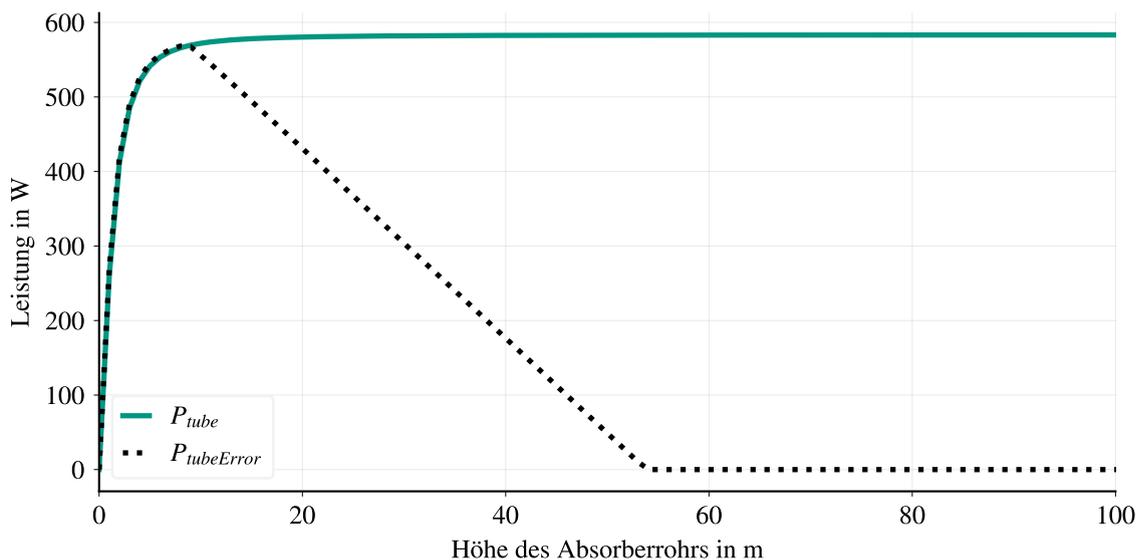
$$a_{\text{error}} = \frac{l}{p}.$$

Die Länge der Strecke  $l$  lässt sich wiederum über verschiedene geometrische Überlegungen bestimmen. Für die ersten drei Fälle aus Abbildung 11.5 erhalten wir

beispielsweise

$$l = \begin{cases} p & \text{im 1. Fall} \\ p - \left(f - \frac{b-p}{2}\right) & \text{im 2. Fall} \\ b & \text{im 3. Fall.} \end{cases}$$

Wird in der Funktion  $P_{\text{tube}}$  der Faktor  $a$  durch  $a_{\text{error}}$  ersetzt, ergibt sich die Funktion  $P_{\text{tubeError}}$ , welche die Leistung am Rohr unter Berücksichtigung etwaiger Einstellungsfehler beschreibt. Der Graph der Funktion  $P_{\text{tubeError}}$  ist in Abhängigkeit der Rohrhöhe in Abbildung 11.6 dargestellt. Der Funktionsgraph gleicht bis zu einer Höhe von circa 9 m dem Graphen der Funktion  $P_{\text{tube}}$ . Der Einfluss der Fehleinstellung ist so gering, dass der Sekundärreflektor bis zu einer Höhe von circa 9 m weiterhin vollständig bestrahlt wird. Bei einer Höhe von knapp 9 m wird die maximale Leistung erreicht. Für noch größere Werte der Rohrhöhe führt die stetig anwachsende Länge der Fehlerstrecke  $f$  dazu, dass immer mehr Strahlen den Sekundärreflektor verfehlen – der Graph hat in diesem Bereich einen streng monoton fallenden Verlauf. Ab einer Rohrhöhe von circa 53 m erreicht keine Leistung mehr das Rohr. Die reflektierten Strahlen verfehlen den Sekundärreflektor vollständig.



**Abb. 11.6:** Leistung am Rohr in Abhängigkeit von der Rohrhöhe mit (gestrichelte Linie) und ohne (durchgezogene Linie) Berücksichtigung von Einstellungsfehlern ( $\epsilon = 0.5^\circ$ ,  $t = 11$  h,  $b = 0.8$  m,  $s = 1.1$  m,  $x = -2$  m)

### 🔪 Zurück zur Optimierung

In Abbildung 11.6 wird bereits anschaulich ersichtlich, dass die Funktion  $P_{\text{tubeError}}$  ihr Maximum und die Zielfunktion des Minimierungsproblems  $-P_{\text{tubeError}}$  damit ihr Minimum annimmt. Über den *verschärften Satz von Weierstraß* lässt sich erneut theoretisch begründen, dass ein globaler Minimalpunkt existiert: Die Funktion  $-P_{\text{tubeError}}$  ist stetig und beispielsweise für das Niveau  $c = 100$  ist die untere Niveaumenge von  $-P_{\text{tubeError}}$  ein abgeschlossenes Intervall und damit kompakt. Dies

garantiert die Existenz eines globalen Minimalpunktes. Da  $-P_{\text{tubeError}}$  zudem strikt konvex auf dem Intervall  $[0, 53]$  ist, ist dieser eindeutig.

In Abbildung 11.6 ist erkennbar, dass der Graph der Funktion  $P_{\text{tubeError}}$  zwei Knickstellen aufweist. An diesen Stellen ist die Funktion nicht differenzierbar. Das Gradientenverfahren ist somit nicht direkt anwendbar, da der Gradient an besagten Stellen nicht existiert. Auch für nichtglatte Probleme existieren verschiedene Verfahren, wie die in Abschnitt 2.2 bereits benannten gradientenfreien Verfahren oder das sogenannte Subgradientenverfahren, welches sich für nichtglatte, konvexe Probleme eignet. Das Subgradientenverfahren wurde (insbesondere mangels eines geeigneten Softwarepaketes in Julia) nicht angewandt. Eine detaillierte Beschreibung des Verfahrens und dessen Vor- und Nachteile sind beispielsweise bei Alt (2004, S. 69–70) zu finden. Zur Optimierung wurde stattdessen das Optimierungspaket *BlackBoxOptim*<sup>3</sup> eingesetzt, welches eine globale Suche basierend auf einem metaheuristischen Verfahren durchführt. Genauer wird ein Evolutionärer Algorithmus (EA), die sogenannte Differentialevolution (engl. *Differential Evolution*, kurz: DE) genutzt. Hierbei handelt es sich um ein gradientenfreies Verfahren. Für weiterführende Erläuterungen zur DE seien interessierte Leserinnen und Leser auf Weicker (2015, S. 170–172) verwiesen. Einen kurzen Einblick in die Grundidee von EAs liefert der folgende Exkurs.

### Exkurs: Evolutionäre Algorithmen

Evolutionäre Algorithmen (EA) sind von der Natur, genauer von der biologischen Evolution, inspiriert. Weicker (2015, S. 20) charakterisiert die EAs wie folgt:

„Evolutionäre Algorithmen kombinieren den Computer als universelle Rechenmaschine mit dem allgemeinen Problemlösungspotential der natürlichen Evolution. So wird im Computer ein Evolutionsprozess künstlich simuliert, um für ein nahezu beliebig wählbares Optimierungsproblem möglichst gute Näherungswerte an eine exakte Lösung zu erzeugen. Dabei wird ein beliebiges abstraktes Objekt, das eine mögliche Lösung für ein Problem darstellt, wie ein Organismus behandelt. Dieses wird durch Anwendung von so genannten evolutionären Operatoren variiert, reproduziert und bewertet. Diese Operatoren nutzen in der Regel Zufallszahlen für ihre Veränderungen an den Individuen.“

EAs gehören zu den stochastischen, metaheuristischen Verfahren. Es kann somit nicht stets garantiert werden,<sup>4</sup> dass eine optimale Lösung des gegebenen Optimierungsproblems gefunden wird (vgl. Weicker, 2015, S. 20). EAs eignen sich unter anderem für hoch-dimensionale Probleme, bei denen wenige bis keine Informationen über die Zielfunktion vorliegen oder bei denen die (approximative) Bestimmung von Gradienten nicht sinnvoll oder nicht möglich ist.

EAs können in verschiedene Teilgebiete unterteilt werden. Zu den vier wichtigsten gehören genetische Algorithmen, Evolutionsstrategien, evolutionäre Programmierung

---

<sup>3</sup>Weitere Informationen zum Paket *BlackBoxOptim* sind zu finden unter: <https://github.com/robertfeldt/BlackBoxOptim.jl>, letzter Aufruf: 31.08.2021.

<sup>4</sup>Durchaus existieren theoretische Aussagen, die für bestimmte Klassen von EAs und unter gewissen Bedingungen an das Optimierungsproblem die Konvergenz gegen optimale Lösungen garantieren (vgl. He und Yu, 2001).

und genetische Programmierung (vgl. Weicker, 2015, S. 127). Im Folgenden wird kurz die Grundidee eines sehr einfachen evolutionären Verfahrens, eines Selektions-Mutations-Verfahrens, skizziert (vgl. Alt, 2011, S. 24):

1. Wähle einen Startpunkt  $\mathbf{x}^{(0)} \in \mathbb{R}^n$  und setze  $k := 0$ . Prüfe, ob ein gegebenes Abbruchkriterium erfüllt ist. Falls ja, stoppe das Verfahren, andernfalls gehe zu Schritt 2.
2. **Mutation:** Ändere (bzw. mutiere) den aktuellen Punkt  $\mathbf{x}^{(k)}$  zufällig und erhalte so einen neuen Punkt  $\tilde{\mathbf{x}}^{(k)}$ .
3. **Selektion:** Falls  $f(\tilde{\mathbf{x}}^{(k)}) < f(\mathbf{x}^{(k)})$ , setze  $\mathbf{x}^{(k+1)} := \tilde{\mathbf{x}}^{(k)}$ , andernfalls setze  $\mathbf{x}^{(k+1)} := \mathbf{x}^{(k)}$ .
4. Falls das Abbruchkriterium nicht erfüllt ist, gehe zu Schritt 2, andernfalls beende das Verfahren.

Als Abbruchkriterium kann unter anderem das Erreichen einer maximalen Anzahl an Iterationen, das Überschreiten eines Zeitlimits oder die Konvergenz des Verfahrens in den letzten Iterationen verwendet werden.

In Schritt 2 des Verfahrens sind diverse Mutationsoperationen denkbar. Eine simple Möglichkeit ist die Festlegung der Mutation gemäß

$$\tilde{\mathbf{x}}_i^{(k)} = \mathbf{x}_i^{(k)} + \sigma_k \left( \mathbf{r}_i^{(k)} - 0.5 \right) \quad \text{für } i = 1, \dots, n,$$

mit Zufallsvektor  $\mathbf{r}^{(k)} \in \mathbb{R}^n$ , dessen Komponenten in  $[0, 1]$  liegen und einer Schrittweite  $\sigma_k > 0$  (vgl. Alt, 2011, S. 24). In komplexeren Verfahren wird oft nicht nur mit einem Punkt  $\mathbf{x}^{(k)}$ , sondern mit einer Menge an Punkten, der sogenannten Population, gearbeitet. Bei diesen Algorithmen wird dann neben den Operationen *Selektion* und *Mutation* zudem eine *Rekombination* von zwei oder mehr Punkten aus der Population der vorherigen Iteration durchgeführt.

In dem beschriebenen, einfachen Verfahren werden keine Nebenbedingungen berücksichtigt. Es wäre denkbar diese im Mutationsschritt zu berücksichtigen. Dazu könnten Mutationen, die die Nebenbedingungen nicht erfüllen, verworfen werden und (zufällig) neue Mutationen erzeugt werden. Alternativ ließe sich ein restringiertes Problem in ein unrestringiertes Problem überführen, indem die Nebenbedingungen beispielsweise über einen Strafterm in die Zielfunktion integriert werden.

Die Anwendung des EAs liefert bei einem Einstellungsfehler von  $\epsilon = 0.5^\circ$  und für einen Spiegel auf Position  $x = -2$  m als Lösung für die Rohrhöhe  $h = 8.7$  m mit einer Leistung von 570.8 W. Der erhaltene optimale Wert scheint realistisch und konstruierbar. Wird der Spiegel auf Position  $x = -5$  m gesetzt, so ergibt sich eine optimale Höhe von  $h = 9.7$  m. Die Schwankung der optimalen Höhe in Abhängigkeit von der gewählten Spiegelposition legt nahe, dass eine Erweiterung des Modells für die Leistung am Rohr – erzeugt durch **alle** aufgestellten Spiegel – sinnvoll wäre.

### Weitere Fehlerquellen

Bisher wurde lediglich die Einstellung des Spiegels als Fehlerquelle diskutiert. Beispiele für weitere Fehlerquellen sind verschmutzte Spiegeloberflächen, die zu einer

abweichenden Reflexion führen, oder die Ablenkung der Sonnenstrahlen durch die Atmosphäre und ein daraus resultierender abweichender Sonneneinfallswinkel  $\alpha$ .

### 11.1.4 Kombination beider Modellerweiterungen

In diesem Abschnitt werden die Modellerweiterungen *atmosphärische Abschwächung* und *Einstellungsfehler* kombiniert und die Optimierung der Rohrhöhe erneut durchgeführt. Die Zielfunktion, bei der beide Modellerweiterungen berücksichtigt werden, wird  $P_{\text{tubeErrorAtmLoss}}$  genannt. Sie ergibt sich durch

$$P_{\text{tubeErrorAtmLoss}}(t, b, h, s, x, \epsilon) = P_{\text{tubeError}}(t, b, h, s, x, \epsilon) \cdot \text{atmLoss}(h, x).$$

Das Optimierungsproblem wurde erneut mit dem EA des Pakets *BlackBoxOptim* gelöst. Liegt kein Fehler oder ein kleiner Fehler vor ( $\epsilon = 0^\circ$  oder  $\epsilon = 0.1^\circ$ ), so entspricht der gefundene Optimalpunkt dem Ergebnis der Optimierung mit der Zielfunktion  $P_{\text{tubeAtmLoss}}$ . Die optimale Höhe lautet ungefähr  $h = 32$  m. Wird der Fehler größer, so entspricht das Ergebnis für die optimale Rohrhöhe dem Ergebnis der Optimierung mit der Zielfunktion  $P_{\text{tubeError}}$ , das heißt einer Höhe von circa  $h = 9$  m. Der Einfluss des fehlerhaften Neigungswinkels ist dann der dominierende Faktor.

Für einen Einstellungsfehler von  $\epsilon = 0.5^\circ$ , der für fünf Spiegel auf den Positionen  $\mathbf{x} = (-4, -2, 0, 2, 4)$  fest gewählt wird, entspricht die optimale Rohrhöhe  $h = 8.1$  m ( $t = 11$  h,  $b = 0.8$  m,  $s = 1.1$  m, ohne Berücksichtigung von Verschattungseffekten).

### 11.1.5 Monte-Carlo-Methode

Ein und denselben Fehler für alle Spiegel festzulegen und basierend auf der resultierenden Funktion für die Leistung am Rohr die optimale Höhe zu bestimmen, spiegelt die Realität nicht wider, da der Fehler auf einen beliebigen Spiegel  $H_i$  nicht bekannt ist. Womöglich arbeiten einige Spiegelmotoren äußerst korrekt, andere hingegen mit deutlich größerem Fehler. Um eine Lösung für die Rohrhöhe zu finden, die robust gegenüber verschiedenen Fehlerszenarien ist, wird die Optimierung basierend auf einer Monte-Carlo-Methode durchgeführt. Monte-Carlo-Methoden stellen eine leicht umsetzbare Methode zur Durchführung von Simulationen und numerischen Integrationen dar. Ein Nachteil ist jedoch, dass diese Methoden sehr langsam konvergieren können (vgl. Caffisch, 1998, S. 1).

Monte-Carlo-Methoden beruhen auf dem Gesetz der großen Zahlen. Dieses existiert in einer schwachen und einer starken Variante:

**Satz 11** (Gesetz der großen Zahlen).

Sei  $X^{(1)}, X^{(2)} \dots$  eine Folge von  $n$  identisch verteilten, unabhängigen Zufallsvariablen mit endlichem Erwartungswert  $\mathbb{E}[X]$ . Dann gilt:

Das arithmetische Mittel

$$S_n := \frac{1}{n} \sum_{i=1}^n X^{(i)}$$

konvergiert gegen  $\mathbb{E}[X]$  für  $n \rightarrow \infty$ .

Für die Konvergenz gilt im Falle des **schwachen Gesetzes der großen Zahlen**

$$\lim_{n \rightarrow \infty} \mathbb{P} \left[ \left| S_n - \mathbb{E}[X] \right| > \epsilon \right] = 0 \quad \forall \epsilon > 0,$$

das heißt es liegt Konvergenz in Wahrscheinlichkeit vor.

Im Falle des **starken Gesetzes der großen Zahlen** liegt sogar fast sichere Konvergenz vor. Das heißt es gilt

$$\mathbb{P} \left[ \lim_{n \rightarrow \infty} S_n = \mathbb{E}[X] \right] = 1$$

(vgl. [Sullivan, 2015, S. 178](#)).

Im vorliegenden Fall ist die Monte-Carlo-Methode die Anwendung des Gesetzes der großen Zahlen auf eine Zufallsvariable  $X = u(Z)$ , wobei  $u$  der auszuwertenden Funktion und  $Z$  einer Zufallsvariablen beziehungsweise einem Zufallsvektor entspricht, die gemäß einer Wahrscheinlichkeitsverteilung  $P$  verteilt ist. Angenommen es können unabhängige identisch verteilte Zufallsvariablen  $Z^{(1)}, Z^{(2)} \dots$  mit Wahrscheinlichkeitsverteilung  $P$  und Wahrscheinlichkeitsdichtefunktion  $f$  erzeugt werden. Dann ist die  $n$ -te Monte-Carlo-Näherung gegeben durch

$$\mathbb{E}_{Z \sim P}[u(Z)] = \int_{-\infty}^{\infty} u(z) f(z) dz \approx \frac{1}{n} \sum_{i=1}^n u(Z^{(i)})$$

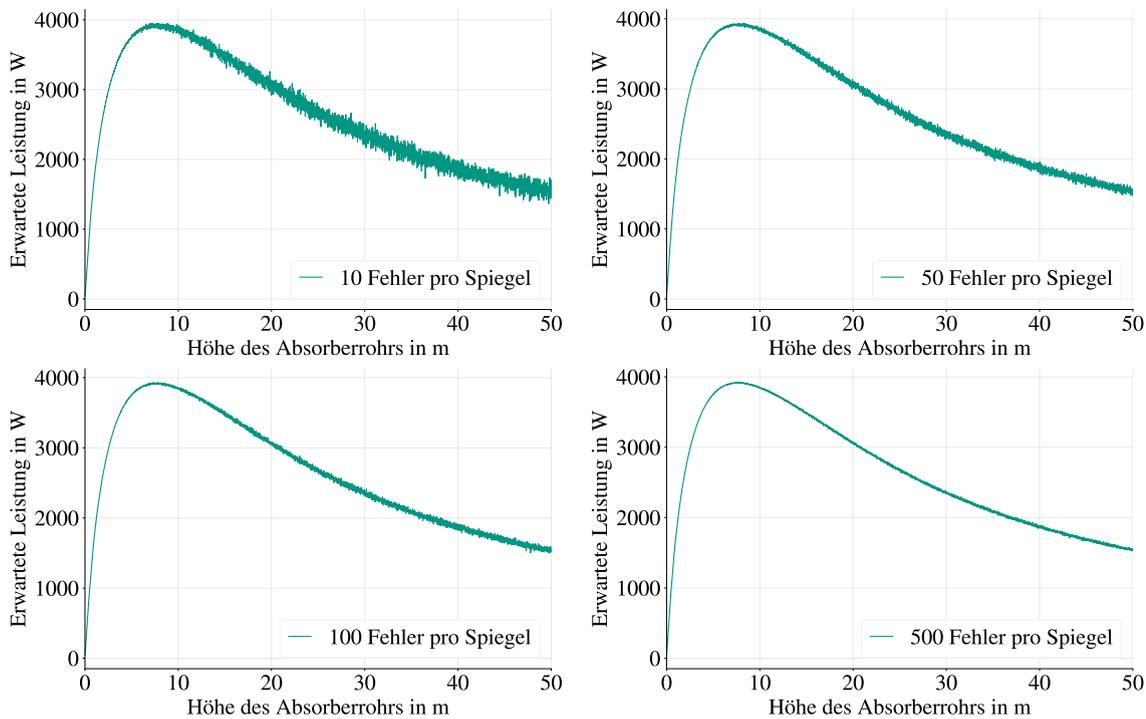
(vgl. [Sullivan, 2015, S. 179](#)).

In unserem Fall kann als auszuwertende Funktion  $u$  die Funktion zur Beschreibung der Leistung am Rohr  $P_{\text{tubeError}}$  betrachtet werden. Der Zufallsvektor  $Z$  hat die Dimension  $N$ , die der Anzahl der betrachteten Spiegel entspricht. Eine Realisierung der Zufallsvariable beschreibt dann ein mögliches Fehlerszenario. Genauer lässt sich die Monte-Carlo-Methode in der betrachteten Anwendung wie folgt beschreiben: Es werden für jeden Spiegel gemäß einer Wahrscheinlichkeitsverteilung  $P$  mit Wahrscheinlichkeitsdichtefunktion  $f$  zufällig verschiedene Fehler und damit viele verschiedene Szenarien generiert. Für jedes Szenario wird die Leistung am Absorberrohr berechnet. Anschließend wird das **arithmetisch Mittel der Leistung** bestimmt. Die in der Optimierung betrachtete Zielfunktion entspricht dann dem durch Monte-Carlo-Integration angenähertem Erwartungswert der Leistung am Rohr

$$\mathbb{E}[P_{\text{tubeError}}(Z)] = \int_{-\infty}^{\infty} P_{\text{tubeError}}(z) f(z) dz \approx \frac{1}{n_{\text{error}}} \cdot \sum_{k=1}^{n_{\text{error}}} P_{\text{tubeError}}(t, b, h, s, \mathbf{x}, Z^{(k)}),$$

wobei  $Z^{(1)}, Z^{(2)}, \dots$  unabhängig und identisch verteilte Zufallsvektoren darstellen. Anschließend kann die optimale Höhe des numerisch approximierten Erwartungswertes für die Leistung am Rohr bestimmt werden.

Die Monte-Carlo-Methode wurde zur Berechnung des Erwartungswerts der Leistung am Rohr angewendet und für verschiedene Werte der Höhe  $h$  wiederholt. Der Verlauf des Erwartungswerts in Abhängigkeit der Höhe  $h$  ist in [Abbildung 11.7](#) für verschiedene Anzahlen an pro Spiegel generierten Fehlern dargestellt. Zur Erzeugung der Fehler wurde eine Normalverteilung  $\mathcal{N}(0, 0.7)$  gewählt. Es wird ersichtlich, dass die Oszillationen in den Graphen kleiner sind, wenn die Anzahl der pro Spiegel generierten Fehler größer gewählt werden.



**Abb. 11.7:** Monte-Carlo-Simulation der erwarteten Leistung am Rohr für unterschiedliche Werte für  $n_{\text{error}}$  mit  $t = 11$  h,  $b = 0.8$  m,  $s = 1.1$  m,  $\mathbf{x} = (-5.5, -4, -2.5, -1, 1, 2.5, 4, 5.5)$

### 📌 Zurück zur Optimierung

Basierend auf der beschriebenen Simulation für die unter Fehlereinfluss zu erwartende Leistung wird erneut die Optimierung der Rohrhöhe vorgenommen. Da die Zielfunktion (je nach Anzahl betrachteter Fehler) stark oszilliert, ist bereits klar, dass zahlreiche globale und lokale Maximalpunkte existieren. Der Einsatz des Gradientenverfahrens würde je nach gewähltem Startpunkt lediglich einen lokalen Optimalpunkt liefern. Zur Optimierung wurde deswegen ein gradientenfreies Verfahren gewählt, welches lediglich Informationen über die Funktionswerte an ausgewählten Stellen nutzt. Konkret wurde erneut ein EA des Pakets *BlackBoxOptim* verwendet. Mit dem EA ergibt sich als optimale Rohrhöhe  $h = 7.0$  m mit einer Leistung von 2574.0 W ( $\mathbf{x} = (-4, -2, 0, 2, 4)$ ,  $t = 11$  h,  $b = 0.8$  m,  $s = 1.1$  m, ohne Berücksichtigung von Verschattungseffekten). Zur Erzeugung der Fehler wurde eine Normalverteilung mit  $\mathcal{N}(0, 0.7)$  gewählt.

## 11.2 Optimierung des Layouts eines Spiegelfeldes

Bei der Planung und Konstruktion von solarthermischen Kraftwerken spielt insbesondere das Layout des Spiegelfeldes, das heißt die Position der Spiegel relativ zum Absorber eine wesentliche Rolle. Grund dafür ist, dass die Spiegelpositionen einen entscheidenden Einfluss auf die Effizienz des Kraftwerks, wie auch auf die Kosten für Konstruktion und Wartung haben (vgl. Richter et al., 2016, S. 241). Die Effizienz des

Kraftwerks hängt insofern von den Positionen der Spiegel ab, als dass dadurch zum einen Verschattungs- und Blockierungseffekte determiniert werden und zum anderen der Energiebeitrag eines einzelnen Spiegels insbesondere aufgrund des Kosinuseffektes beeinflusst wird. Aufgrund dessen werden die Spiegelpositionen basierend auf der Simulation des Kraftwerks vor Konstruktion optimiert.

Das Layoutoptimierungsproblem kann allgemein geschrieben werden als<sup>5</sup>

$$\min_{\mathbf{x}} \quad - E_{\text{tube}}(\mathbf{x}) \quad (11.8a)$$

$$\text{u. d. Nb.} \quad l_b \leq x_i \leq u_b \quad 1 \leq i \leq N, \quad (11.8b)$$

wobei die Nebenbedingung (11.8b) fordert, dass die Spiegel im Bereich eines gegebenen Kraftwerksareals von  $l_b$  bis  $u_b$  stehen.

Das Optimierungsproblem kann sowohl hinsichtlich der Wahl der Zielfunktion als auch hinsichtlich der Modellierung weiterer Nebenbedingungen variiert werden. Lassen wir für die  $N$  Variablen  $x_1, \dots, x_N$  Werte in  $\mathbb{R}$  zu, so ergibt sich ein kontinuierliches Optimierungsproblem. Schränken wir die Variablen hingegen auf vordefinierte Punkte im Bereich des Kraftwerks ein, so liefert dies ein diskretes Problem. Zudem existieren verschiedene Verfahren, die zur Lösung des Problems in den unterschiedlichen Formulierungen eingesetzt werden können.

Im Folgenden wird kurz erläutert, wie sich diverse Formulierungen des Layoutoptimierungsproblems in der Forschung rund um solarthermische Kraftwerke entwickelt haben. Anschließend werden die in der Forschung eingesetzten Formulierungen und Verfahren (in elementarer Form) auf die Optimierung des Layouts unseres vereinfachten Fresnelkraftwerks übertragen. Dazu wird das Optimierungsproblem von ganz unterschiedlichen Standpunkten aus (sowohl diskret als auch kontinuierlich) angegangen. Dabei werden Vor- und Nachteile der einzelnen Formulierungen diskutiert. Auch werden exemplarische Ergebnisse gezeigt, bei denen weniger die Performanz im Vordergrund stand, als vielmehr das Herausarbeiten zentraler Charakteristika der verschiedenen Formulierungen und Algorithmen, die auch im entwickelten Lernmaterial aufgegriffen wurden.

### 11.2.1 Ein Blick in die Forschung

Bei der Optimierung des Layouts eines Solarkraftwerks werden in der Forschung verschiedene Ansätze verfolgt. Gängige Herangehensweisen, die in der Literatur primär für Solarturmkraftwerke diskutiert werden (vgl. Abschn. 13.4), die sich jedoch auch auf die Optimierung der Spiegelpositionen in einem Fresnelkraftwerk erweitern und anwenden lassen, werden im Folgenden beschrieben.

<sup>5</sup>Die Optimierung wurde in allen nachfolgend vorgestellten Formulierungen des Problems für die beiden Modelle  $E_{\text{tube}}(\mathbf{x})$  und  $E_{\text{tubeEffects}}(\mathbf{x})$  durchgeführt. Bei der Beschreibung der unterschiedlichen Formulierungen des Optimierungsproblems wird die Unterscheidung zwischen den Zielfunktionen aus Gründen der Übersichtlichkeit nicht vorgenommen. In der Diskussion der Ergebnisse wird dann hinsichtlich der beiden Modelle (mit und ohne Verschattungseffekte) differenziert.

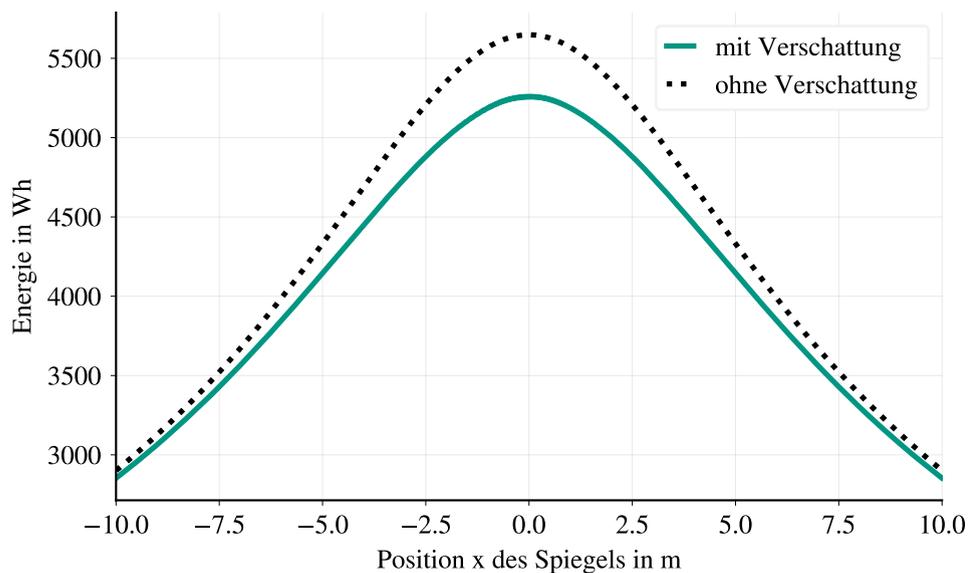
Bei den **kontinuierlichen Formulierungen** des Optimierungsproblems kann zwischen **musterbasierten Methoden** und **freien Variablen Methoden** unterschieden werden:

- Bei den **freien Variablen Methoden** wird direkt über die Koordinaten der Spiegel optimiert. In den meisten Simulationen in der Forschung, bei denen das gesamte Spiegelfeld (im dreidimensionalen Fall) betrachtet wird, wird über die  $(x, y)$ -Koordinaten optimiert. Da wir lediglich den Querschnitt eines Kraftwerks betrachten, ist in unserer Modellierung nur die  $x$ -Koordinate relevant. Aufgrund der Komplexität der eingesetzten Simulation ist eine der größten Hürden bei den freien Variablen Methoden die hohe Rechenzeit (vgl. [Lutchman et al., 2014](#), S. 1429). Für die Lösung des Optimierungsproblems mit freien Koordinatenvariablen gibt es verschiedene Ansätze. Diese reichen von gradientenbasierten Methoden (vgl. [Lutchman et al., 2014](#), S. 1433), über Ansätze der nichtlinearen Programmierung bis hin zu Metaheuristiken wie Partikelschwarm- oder evolutionäre Algorithmen (vgl. [Richter et al., 2016](#), S. 242). Historisch wurden die Layoutoptimierungsprobleme in der Forschung nicht über freie Variablen Formulierungen angegangen, da das Problem in dieser Formulierung hochdimensional und komplex ist. Stattdessen wurden die Positionsvariablen auf gewisse Muster oder vorgegebene Positionen eingeschränkt. Dies führt dazu, dass gefundene Lösungen nicht notwendigerweise optimal sind, dafür jedoch Vorteile, wie die Anwendbarkeit erprobter Lösungsverfahren und meist damit einhergehend eine reduzierte Rechenzeit, erzielt werden können.
- Bei den **musterbasierten Methoden** wird ein geometrisches Muster vorgegeben, entsprechend dem die Spiegel positioniert werden. Diese Muster hängen von gewissen Parametern ab. Anstatt direkt über die Koordinaten eines Spiegels zu optimieren wird über die Parameter des Musters optimiert. Bei der Layoutoptimierung eines Solarturmkraftwerks kommen als Muster beispielsweise Spiralen, wie die Fermat'sche Spirale, zum Einsatz, die lediglich von zwei Parametern abhängt (vgl. [Noone et al., 2012](#), S. 11). Dies hat den Vorteil, dass die Anzahl der Variablen von einer Vielzahl an Koordinaten auf wenige Parameter reduziert werden kann. Ein Nachteil ist, dass die gefundene Lösung für das Spiegelfeldlayout gegebenenfalls schlechter wird (vgl. [Richter et al., 2016](#), S. 241).

Bei den **diskreten Formulierungen** wird das Spiegelfeld zuvor in vordefinierte, mögliche Positionen unterteilt (diskretisiert). Die Spiegelkoordinaten können dann nur noch eine Kombination dieser vordefinierten Positionen annehmen. Zum Lösen dieses Problems werden unter anderem sogenannte **Field Growth Methoden** eingesetzt. Bei diesen werden die Spiegel iterativ auf die vordefinierten Positionen gesetzt. In den einzelnen Iterationen wird für jede noch mögliche Position (Positionen, die den Mindestabstand nicht einhalten werden gelöscht) die Gesamtenergie aus bereits besetzten und der noch zu betrachtenden Position berechnet. Wird ein Greedy-Verfahren gewählt, so wird der zu setzende Spiegel in jeder Iteration auf die zu diesem Zeitschritt beste Position gesetzt (vgl. [Sánchez und Romero 2006](#), S. 863; [Richter et al. 2016](#), S. 241). Bei den diskreten Formulierungen des Layoutoptimie-

rungsproblems muss zwischen der Güte der gefundenen Lösung und der Rechenzeit des Algorithmus abgewogen werden. Abhängig von der Feinheit der Diskretisierung ergibt sich nicht unbedingt das optimale Layout. Wird die Diskretisierung jedoch sehr fein gewählt, so nimmt die Rechenzeit zu.

Die unterschiedlichen Formulierungen des Optimierungsproblems werden im Folgenden auf unser Fresnelkraftwerk übertragen und exemplarische Ergebnisse der Optimierung basierend auf der in Kapitel 10 beschriebenen Simulation diskutiert. Dabei wird umfassend ersichtlich, dass das Lösen und Formulieren von Optimierungsproblemen mathematisch äußerst reichhaltig ist.



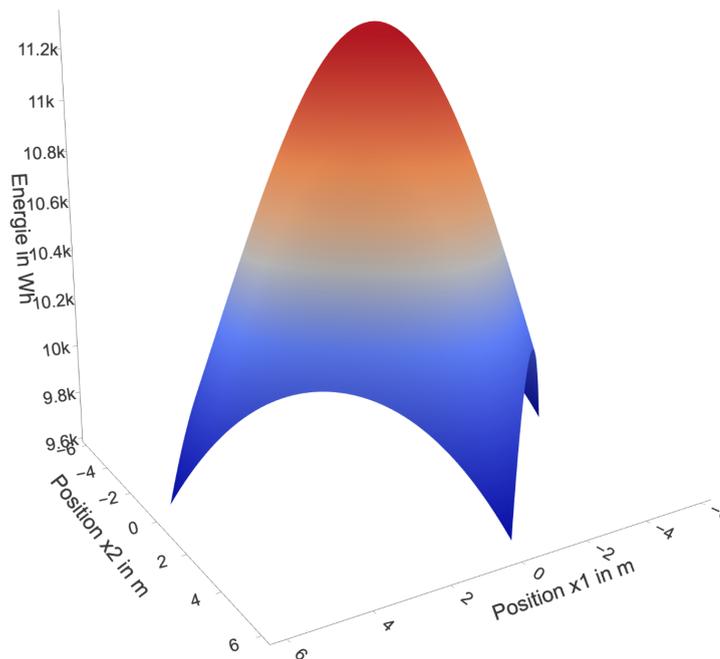
**Abb. 11.8:** Energieumsatz gemäß dem Modell ohne Verschattungseffekte ( $E_{\text{tube}}$ ) und gemäß dem Modell mit Verschattungseffekten ( $E_{\text{tubeEffects}}$ ) in Abhängigkeit von der Position  $x$  eines Spiegels ( $b = 0.8 \text{ m}$ ,  $h = 6 \text{ m}$ ,  $s = 1.1 \text{ m}$ ,  $\Delta t = 0.05 \text{ h}$ )

### 11.2.2 Kontinuierliche Formulierungen

Bei allen Formulierungen wird der Energieumsatz eines Tages maximiert. Dabei wird die Optimierung zum einen basierend auf dem Modell ohne Verschattungseffekte, beschrieben durch  $E_{\text{tube}}$ , und zum anderen basierend auf dem Modell mit Verschattungseffekten, beschrieben durch  $E_{\text{tubeEffects}}$ , durchgeführt. Beide Funktionen hängen von dem Spiegelkoordinatenvektor  $\mathbf{x}$  ab, alle weiteren Parameter sind fest ( $\Delta t = 0.05 \text{ h}$ ,  $b = 0.8 \text{ m}$ ,  $s = 1.1 \text{ m}$ ,  $h = 6 \text{ m}$ ).

Betrachten wir Abbildungen 11.8 und 11.9 so wird im Falle von einem beziehungsweise zwei Spiegeln bereits anschaulich ersichtlich, dass die Zielfunktion ohne Verschattungseffekte einen eindeutigen globalen Optimalpunkt besitzt. Bezugnehmend zu den theoretischen Grundlagen rund um Existenz und Eindeutig globaler Optimalpunkte (vgl. Abschn. 2.1.1), ließe sich dies für das auf  $-5 \text{ m}$  bis  $5 \text{ m}$  eingeschränkte

Kraftwerksareal auch theoretisch diskutieren: Die Funktion  $E_{\text{tube}}$  ist eingeschränkt auf das Kraftwerksareal von  $-5\text{ m}$  bis  $5\text{ m}$  konkav. Die Zielfunktion des Minimierungsproblems ( $-E_{\text{tube}}$ ) ist folglich auf dem eingeschränkten Areal konvex, womit jeder lokale Minimalpunkt (sofern ein solcher existiert) zugleich globaler Minimalpunkt ist. Selbiges gilt für die Funktion  $E_{\text{tubeEffects}}$ , wenn lediglich ein Spiegel auf dem Kraftwerksareal positioniert wird (vgl. Abb. 11.8). Für alle anderen Fälle ( $N \geq 2$ ) weist die Funktion bedingt durch Verschattungseffekte zahlreiche lokale Maximalpunkte auf (vgl. Abb. 11.10). In diesem Fall kann nicht garantiert werden, dass ein lokales Optimierungsverfahren einen globalen Minimalpunkt des Optimierungsproblems liefert.



**Abb. 11.9:** Energieumsatz in Abhängigkeit von den Positionen  $x_1$  und  $x_2$  zweier Spiegel ( $b = 0.8\text{ m}$ ,  $h = 6\text{ m}$ ,  $s = 1.1\text{ m}$ )

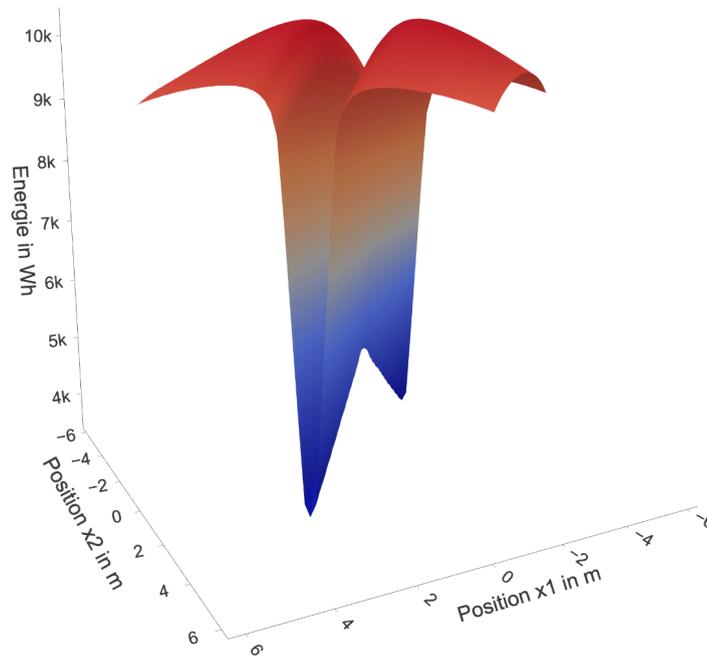
Die ersten drei folgenden, kontinuierlichen Formulierungen des Optimierungsproblems gehören zum **freien Variablen Ansatz**. Die Formulierungen werden kleinschrittig beschrieben, um zu verdeutlichen, welchen Einfluss die einzelnen Nebenbedingungen auf die Minimalpunkte des Optimierungsproblems und damit auf das Layout des Spiegelfeldes haben. Die vierte kontinuierliche Formulierung folgt einem **musterbasierten Ansatz**.

### Freie Variablen Optimierung

Nachfolgend wird vernachlässigt, dass das Absorberrohr befestigt sein muss. Je nach Konstruktion des Kraftwerks wäre sonst eine Nebenbedingung nötig, die einen Mindestabstand zur Rohrbefestigung sicherstellt.

#### Formulierung 1: Spiegel im Areal des Kraftwerks

Die einzige Nebenbedingung bei dieser Formulierung ist, dass die Spiegel im Areal



**Abb. 11.10:** Energieumsatz gemäß dem Modell mit Verschattungseffekten in Abhängigkeit von den Positionen  $x_1$  und  $x_2$  zweier Spiegel ( $b = 0.8$  m,  $h = 6$  m,  $s = 1.1$  m)

des Kraftwerks, welches durch das Intervall  $[l_b, u_b]$  beschrieben wird, positioniert werden müssen. Das Optimierungsproblem lautet

$$\begin{array}{ll} \min_{\mathbf{x}} & -E_{\text{tube}}(\mathbf{x}) \\ \text{u. d. Nb.} & l_b \leq x_i \leq u_b \quad \text{für } 1 \leq i \leq N. \end{array} \quad (11.9)$$

### Formulierung 2: Mindestabstand zwischen Spiegeln

Die Formulierung 1 wird um eine Nebenbedingung erweitert, die einen Mindestabstand  $d_{\min}$  zwischen den Spiegeln fordert. Um eine Kollision zwischen den Spiegeln zu verhindern, sollte der Abstand  $d_{\min}$  mindestens so groß wie die Breite der Spiegel  $s$  gewählt werden. Das Optimierungsproblem lautet

$$\begin{array}{ll} \min_{\mathbf{x}} & -E_{\text{tube}}(\mathbf{x}) \\ \text{u. d. Nb.} & |x_i - x_j| \geq d_{\min} \quad \text{für } 1 \leq i \neq j \leq N \\ & l_b \leq x_i \leq u_b \quad \text{für } 1 \leq i \leq N. \end{array} \quad (11.10)$$

### Formulierung 3: Aufsteigende Sortierung der Spiegel

Lösen des Optimierungsproblems in Formulierung 2 würde zu mehreren (bezüglich des Layouts identischen) Minimalpunkten mit gleichem Zielfunktionswert führen. Werden beispielsweise die erhaltenen Positionen der Spiegel  $i$  und  $j$  vertauscht, so liefert dies zwar aus mathematischer Sicht eine neue Lösung mit gleichem Zielfunktionswert, aufgrund der Tatsache, dass alle Spiegel identisch sind, impliziert dies

jedoch das gleiche Layout. Um diese mehrfachen (gleichwertigen) Minimalpunkte zu vermeiden, wird eine Nebenbedingung eingeführt, die eine aufsteigende Sortierung der Spiegel sichert. Das Optimierungsproblem lautet

$$\begin{array}{llll}
 \min_{\mathbf{x}} & -E_{\text{tube}}(\mathbf{x}) & & \\
 \text{u. d. Nb.} & |x_i - x_j| \geq d_{\min} & \text{für } 1 \leq i, j \leq N & (11.11) \\
 & x_{i+1} - x_i \geq 0 & \text{für } 1 \leq i \leq N - 1 & \\
 & l_b \leq x_i \leq u_b & \text{für } 1 \leq i \leq N. & 
 \end{array}$$

Dieses Problem kann zusammengefasst werden zu

$$\begin{array}{llll}
 \min_{\mathbf{x}} & -E_{\text{tube}}(\mathbf{x}) & & \\
 \text{u. d. Nb.} & x_i - x_{i+1} \leq -d_{\min} & \text{für } 1 \leq i \leq N - 1 & (11.12) \\
 & l_b \leq x_1 & & \\
 & x_N \leq u_b. & & 
 \end{array}$$

Das Problem lässt sich auch in der Notation schreiben, die vielfach in der Fachliteratur zu finden ist, nämlich

$$\begin{array}{llll}
 \min_{\mathbf{x}} & -E_{\text{tube}}(\mathbf{x}) & & \\
 \text{u. d. Nb.} & A \cdot \mathbf{x} \leq \mathbf{b} & & (11.13) \\
 & -x_1 \leq -l_b & & \\
 & x_N \leq u_b, & & 
 \end{array}$$

wobei

$$A = \begin{pmatrix} 1 & -1 & 0 & 0 & 0 & \cdots \\ 0 & 1 & -1 & 0 & 0 & \cdots \\ 0 & 0 & 1 & -1 & 0 & \cdots \\ 0 & 0 & 0 & 1 & -1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix} \text{ und } \mathbf{b} = \begin{pmatrix} -d_{\min} \\ -d_{\min} \\ -d_{\min} \\ -d_{\min} \\ -d_{\min} \\ \vdots \end{pmatrix}.$$

Es liegt ein kontinuierliches Optimierungsproblem mit nichtlinearer Zielfunktion und linearen Nebenbedingungen vor.

### Numerische Verfahren und Ergebnisse

Die Lösung von nichtlinearen, restringierten Optimierungsproblemen ist analytisch keine einfache Aufgabe. Es existieren jedoch verschiedene numerische Verfahren zum Lösen derartiger Probleme. Zu den gradientenbasierten Verfahren zählen:

- Sequentielle Quadratische Programmierung (SQP) oder allgemeiner sequentielle Approximationsverfahren (vgl. [Lutchman et al., 2014](#), S. 1434),
- Innere-Punkte-Methoden und Barriereverfahren,

- Straftermverfahren und
- Augmented Lagrange Verfahren beziehungsweise Multiplikatorverfahren (vgl. Nocedal und Wright, 2006, S. 422–424; Stein, 2021b, S. 189–221).

Darüber hinaus gibt es zahlreiche gradientenfreie Verfahren, die wie der Name bereits sagt, ohne die Berechnung des Gradienten auskommen. Diese Verfahren werden vielfach eingesetzt, wenn die Zielfunktion nicht glatt ist oder die Auswertung der Zielfunktion äußerst rechenintensiv ist – wie es bei realen Simulationen des Energieumsatzes eines Kraftwerks der Fall ist. Beispiele für gradientenfreie Verfahren sind evolutionäre Algorithmen, die unter anderem von Richter (2017) zur Optimierung des Layouts von Solarturmkraftwerken eingesetzt wurden, oder sogenannte Random Search-Verfahren (vgl. Lutchman et al., 2014, S. 1434).

Zur Optimierung der Spiegelpositionen basierend auf den Modellen ohne oder mit Verschattungseffekten ( $E_{\text{tube}}$  oder  $E_{\text{tubeEffects}}$ ) wurde erneut ein EA eingesetzt. Da der eingesetzte EA (eine Differential evolution) nicht für die Optimierung von nichtlinearen Optimierungsproblemen **mit Nebenbedingungen** ausgelegt ist, wurde das Problem (11.13) umformuliert. Dazu wurden die Nebenbedingungen als Strafterm in die Zielfunktion integriert und das unrestringierte Problem

$$\min_{\mathbf{x}} -E_{\text{tube}}(\mathbf{x}) + \mu \sum_{i=1}^{N-1} \max\{0, g_i(\mathbf{x})\} \quad (11.14)$$

gelöst, wobei  $g_i(\mathbf{x}) = x_i - x_{i+1} + d_{\min}$  und  $\mu > 0$ . Damit wurde der bereits in Abschnitt 2.2.2 beschriebene Ansatz der Integration einer  $\ell_1$ -Straftermfunktion genutzt. Das unrestringierte Optimierungsproblem (11.14) kann dann mit dem EA gelöst werden.

Wird das Optimierungsproblem (11.13) für einen Spiegel ( $N = 1$ ) gelöst, so ergibt sich  $x_1 = 0$  m als optimale Spiegelposition. Der Spiegel sollte folglich direkt unter dem Rohr platziert werden. Dies ist sowohl mit als auch ohne Berücksichtigung der Verschattung durch den Sekundärreflektor der Fall (vgl. Abb. 11.8). Das Ergebnis lässt sich dadurch begründen, dass der Einfluss des Kosinuseffektes für einen Spiegel direkt unter dem Rohr minimal ist.

Wird die Optimierung gemäß Formulierung 1 für zwei Spiegel ( $N = 2$ ) durchgeführt, ergibt sich *ohne Berücksichtigung von Verschattungseffekten* als optimale Positionen  $x_1 = 0$  m =  $x_2$  (vgl. Abb. 11.9). Die Spiegel würden kollidieren, was die Berücksichtigung der Nebenbedingung für den Mindestabstand notwendig macht.

Die Berücksichtigung des Mindestabstands (Formulierung 2) liefert im Falle von zwei Spiegeln sowohl  $\mathbf{x} = \left(-\frac{d_{\min}}{2}, \frac{d_{\min}}{2}\right)$  als auch  $\mathbf{x} = \left(\frac{d_{\min}}{2}, -\frac{d_{\min}}{2}\right)$  als Lösungen. Das Optimierungsproblem ist auch bei Einschränkung des Definitionsbereichs der Zielfunktion auf das Kraftwerksareal von  $-5$  m bis  $5$  m nicht konvex, da die Nebenbedingung für den Mindestabstand keine konvexe Menge beschreibt. Das Problem hat keinen eindeutigen Optimalpunkt.

Wird nun zusätzlich eine aufsteigende Sortierung der Spiegelpositionen gefordert (Formulierung 3), so werden die genannten gleichwertigen, mehrfachen Lösungen vermieden (und die linearen Nebenbedingungen beschreiben nun wiederum eine konvexe Menge). Im Falle von zwei Spiegeln ergibt sich als optimale Lösung für die Spiegelpositionen die symmetrische Platzierung links und rechts vom Rohr gemäß

$\mathbf{x} = \left(-\frac{d_{\min}}{2}, \frac{d_{\min}}{2}\right)$ . Wird die Optimierung basierend auf dem Modell *mit Verschattungseffekten* durchgeführt, so ergibt sich als optimale Positionierung zweier Spiegel  $\mathbf{x} = (-1.04, 1.09)$  bei einem Mindestabstand von  $d_{\min} = 1.1$  m. Am Graph der Zielfunktion in Abbildung 11.10 ist erkennbar, dass die umgesetzte Energie geringer wird, wenn die Spiegel nah aneinander rücken, da der Einfluss der Verschattungseffekte dann groß ist.

Die Ergebnisse der Optimierung basierend auf Formulierung 3 sind Tabelle 11.1 zu entnehmen. Übergeordnet ist zu erkennen, dass das Ergebnis für die Spiegelpositionen von der Feinheit der Unterteilung des Tages in äquidistante Zeitschritte  $\Delta t$  bei der numerischen Integration abhängt.

**Tab. 11.1:** Ergebnisse der freien Variablen Optimierung mit Mindestabstand  $d_{\min} = 1.1$  m und  $b = 0.8$  m,  $h = 6$  m,  $s = 1.1$  m basierend auf Formulierung 3

N	$\Delta t$ in h	Modell	opt. Positionen $\mathbf{x}^*$	Energieumsatz in Wh
2	0.05	o. Versch.	$(-0.55, 0.55)$	11248.8
4	0.05	o. Versch.	$(-1.65, -0.55, 0.55, 1.65)$	22134.6
4	0.5	o. Versch.	$(-1.65, -0.55, 0.55, 1.65)$	22127.7
4	2.0	o. Versch.	$(-1.63, -0.53, 0.57, 1.67)$	22264.1
9	0.05	o. Versch.	$(-4.39, -3.29, -2.19, -1.09, 0.01, 1.11, 2.21, 3.31, 4.41)$	46596.1
2	0.05	m. Versch.	$(-1.04, 1.09)$	10234.7
4	0.05	m. Versch.	$(-2.6, -0.84, 0.80, 2.58)$	19536.1
9	0.05	m. Versch.	$(-4.73, -3.4, -2.06, -0.81, 0.45, 1.69, 3.03, 4.28, 5.68)$	39457.8

### Musterbasierte Optimierung

Im Falle unseres vereinfachten Kraftwerks stellt die Optimierung des Abstands zwischen den Spiegeln eine Möglichkeit dar, um das Problem über eine musterbasierte Methode anzugehen. Das Optimierungsproblem kann dann gemäß

$$\begin{aligned} \min_{\Delta x, x_1} & & -E_{\text{tube}}(\mathbf{x}) \\ \text{u. d. Nb.} & & \Delta x \geq d_{\min} \\ & & l_b \leq x_1 \\ & & x_1 + (N - 1) \cdot \Delta x \leq u_b \end{aligned} \tag{11.15}$$

formuliert werden. Hier beschreibt  $\Delta x$  den äquidistanten Abstand zwischen zwei benachbarten Spiegeln

$$\Delta x = |x_i - x_{i+1}| \text{ für } i = 1, \dots, N - 1.$$

Weiterhin wird die Startposition des Musters, und damit die Position  $x_1$  des Spiegels ganz links auf dem Kraftwerksareal als Variable gewählt. Insgesamt ergibt sich der Positionsvektor durch

$$\mathbf{x} = \begin{pmatrix} x_1 + 0 \cdot \Delta x \\ x_1 + 1 \cdot \Delta x \\ x_1 + 2 \cdot \Delta x \\ \vdots \\ x_1 + (N - 1) \cdot \Delta x \end{pmatrix} = \sum_{j=0}^{N-1} (x_1 + j \cdot \Delta x) \cdot e_{j+1},$$

wobei  $e_j \in \mathbb{R}^N$  dem  $j$ -ten Einheitsvektor entspricht.

Gegenüber dem freien Variablen Ansatz kann die Anzahl der Entscheidungsvariablen des Optimierungsproblems stark reduziert werden (von  $N$  auf 2).

Die Festlegung auf ein bestimmtes Muster stellt eine richtungsweisende Modellentscheidung dar, die sich im Hinblick auf die Umsetzung im Lernmaterial für eine Diskussion mit den Lernenden anbietet. Die Wahl des Musters lässt Raum für subjektive Entscheidungen, die Einfluss auf Eigenschaften des mathematischen Modells sowie die resultierende Lösung haben. Wird ein anderes Muster zugrunde gelegt, so kann die Optimierung zu einem gänzlich anderen Ergebnis für die optimalen Spiegelpositionen führen. Mögliche Modifikationen der musterbasierten Formulierung sind:

- Die Festlegung des Wertes für die Position des ersten Spiegel  $x_1$ , womit die Optimierung nur über die Variable  $\Delta x$  durchgeführt werden würde.
- Die Anzahl der zu setzenden Spiegel  $N$  kann als weitere Variable mit der oberen Schranke  $N \leq \frac{u_b - l_b}{d_{\min}}$  eingeführt werden. Damit würde über drei Variablen  $x_1$ ,  $\Delta x$  und  $N$  optimiert werden.
- Der Abstand zwischen zwei benachbarten Spiegeln wird nicht äquidistant, sondern gemäß eines beliebigen anderen Musters gewählt.

**Tab. 11.2:** Ergebnisse der Layoutoptimierung mit Mindestabstand  $d_{\min} = 1.1$  m und  $b = 0.8$  m,  $h = 6$  m,  $s = 1.1$  m,  $\Delta t = 0.05$  h basierend auf dem musterbasierten Ansatz

N	Modell	opt. Positionen $\mathbf{x}^*$	Energieumsatz in Wh
2	o. Versch.	(-0.55, 0.55)	11248.8
4	o. Versch.	(-1.65, -0.55, 0.55, 1.65)	22134.6
9	o. Versch.	(-4.4, -3.3, -2.2, -1.1, 0.0, 1.1, 2.2, 3.3, 4.4)	46329.8
2	m. Versch.	(-1.04, 1.01)	10099.0
4	m. Versch.	(-2.6, -0.9, 0.79, 2.49)	19303.9
9	m. Versch.	(-5.4, -4.06, -2.72, -1.38, -0.04, 1.3, 2.64, 3.98, 5.32)	39190.8

### Numerische Verfahren und Ergebnisse

Wie bei der freien Variablen Formulierung wurde die Optimierung mit einem EA (des Pakets *BlackBoxOptim*) durchgeführt. Die Ergebnisse der Optimierung können Tabelle 11.2 entnommen werden.

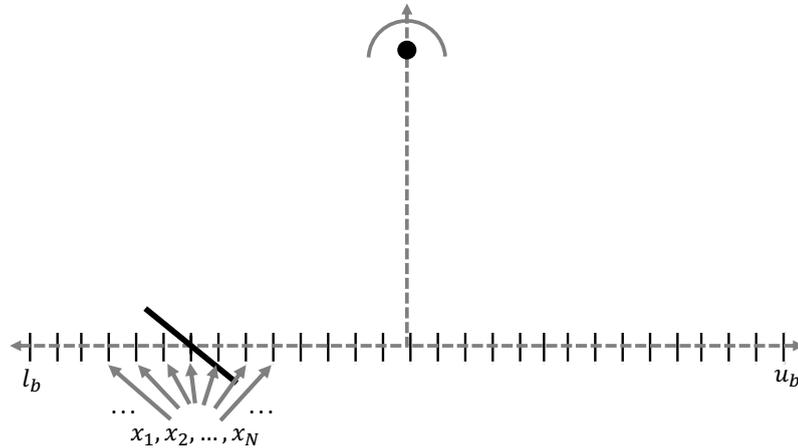
Auf Grundlage des Modells ohne Verschattungseffekte, werden die optimalen Positionen durch den Kosinuseffekt dominiert. Der maximale Energieumsatz ergibt sich, wenn die Spiegel so nah wie möglich aneinander und symmetrisch links und rechts um das Rohr positioniert werden. Der optimale Spiegelabstand entspricht dem Mindestabstand  $d_{\min}$ . Die Ergebnisse stimmen in diesem Fall mit den Ergebnissen der freien Variablen Optimierung überein.

Bei dem Modell mit Verschattungseffekten unterscheiden sich die gefundenen Positionen bei der musterbasierten und der freien Variablen Optimierung. Der Schattenschwurf führt dazu, dass es nicht mehr optimal ist, die Spiegel maximal nah aneinander und damit stets im gleichen Abstand zu positionieren. Der musterbasierte Ansatz führt deswegen zu schlechteren Ergebnissen als der freie Variablen Ansatz.

### 11.2.3 Diskrete Formulierungen

Die Einschränkung der Spiegelpositionen auf eine Menge an vordefinierten Positionen führt gegebenenfalls zu Einbußen hinsichtlich der Güte der gefundenen Lösung für die Positionen im Hinblick auf das ursprüngliche Optimierungsproblem. Es hat aber den Vorteil, dass zahlreiche Algorithmen existieren, die zur Lösung des Problems eingesetzt werden können. Konkret wird das Problem im Folgenden als binäres nichtlineares Optimierungsproblem formuliert. Diese Art Probleme werden auch nichtlineare 0-1-Programme genannt.

Bei der diskreten Formulierung des Layoutoptimierungsproblems werden zunächst zahlreiche Positionen  $\{y_1, \dots, y_M\}$ , auf denen die Spiegel potentiell positioniert wer-



**Abb. 11.11:** Diskrete Betrachtung des Layoutoptimierungsproblems

den können, festgelegt. Das Areal des Kraftwerks wird diskretisiert (vgl. Abb. 11.11). Das Problem kann dann auf unterschiedliche Weise formuliert werden:

**Formulierung 1:**

Das Problem kann aus Sicht der einzelnen Spiegel betrachtet werden. Es wird gefragt, ob ein Spiegel  $H_i$  auf der Position  $y_p$  stehen sollte oder nicht. Für jeden Spiegel  $H_i$  wird die binäre Variable  $x_{ip}$  eingeführt, die angibt, ob Spiegel  $H_i$  an Position  $y_p$  steht ( $x_{ip} = 1$ ) oder nicht ( $x_{ip} = 0$ ). Wir erhalten das binäre Optimierungsproblem

$$\min_{x_{i,p}} \quad -E_{\text{tube}}(\mathbf{x}) \quad (11.16a)$$

$$\text{u. d. Nb.} \quad \sum_{p=1}^M x_{ip} = 1 \quad \text{für } 1 \leq i \leq N \quad (11.16b)$$

$$\left| \sum_{p=1}^M x_{ip} y_p - \sum_{p=1}^M x_{jp} y_p \right| \geq d_{\min} \quad \text{für } 1 \leq i \neq j \leq N \quad (11.16c)$$

$$x_{ip} \in \{0, 1\} \quad \text{für } 1 \leq i \leq N; 1 \leq p \leq M, \quad (11.16d)$$

wobei der Positionsvektor geschrieben werden kann als

$$\mathbf{x} = \begin{pmatrix} \sum_{p=1}^M x_{1p} y_p \\ \vdots \\ \sum_{p=1}^M x_{Np} y_p \end{pmatrix}.$$

Die Nebenbedingung (11.16c) kann weggelassen werden, wenn der Mindestabstand bereits in der Wahl der möglichen Positionen  $y_p$  berücksichtigt wird. Dies würde das Problem leichter machen, die Feinheit der Diskretisierung jedoch einschränken.

Lösen des Optimierungsproblems (11.16) würde zu mehrfachen, im Hinblick auf das Spiegelfeldlayout jedoch identischen, Lösungen für die Spiegelpositionen führen. Um dies zu umgehen, wird erneut gefordert, dass die Spiegel aufsteigend sortiert werden.

Hierzu kann die Nebenbedingung (11.16c) zu

$$\sum_{p=1}^M x_{ip} y_p + d_{\min} \leq \sum_{p=1}^M x_{i+1p} \cdot y_p \quad \text{für } 1 \leq i \leq N - 1 \quad (11.17)$$

abgeändert werden.

### Formulierung 2:

Um gleichwertige (mehrfache) Lösungen für die Spiegelpositionen zu verhindern, kann das Optimierungsproblem auch aus Sicht der möglichen Positionen  $y_p$  formuliert werden. Dazu wird für jede vordefinierte Position  $i$  die binäre Variable  $z_i$  eingeführt. Diese gibt an, ob an Position  $y_i$  ein Spiegel steht ( $z_i = 1$ ) oder nicht ( $z_i = 0$ ). Das Optimierungsproblem lautet

$$\min_{z_1, \dots, z_N} \quad -E_{\text{tube}}(\mathbf{x}) \quad (11.18a)$$

$$\text{u. d. Nb.} \quad \sum_{i=1}^M z_i \leq N \quad (11.18b)$$

$$-|y_j z_j - y_i z_i| \leq -d_{\min} \cdot z_i z_j \quad \text{für } 1 \leq i < j \leq M \quad (11.18c)$$

$$z_i \in \{0, 1\} \quad \text{für } 1 \leq i \leq M, \quad (11.18d)$$

wobei der Positionsvektor geschrieben werden kann als

$$\mathbf{x} = \{y_i \mid \text{für alle } 1 \leq i \leq M \text{ für die } z_i = 1\}.$$

Bei dieser Formulierung wird nicht zwischen den einzelnen Spiegeln unterschieden, womit gleichwertige, im Hinblick auf das Spiegelfeldlayout identische Positionierungen nicht möglich sind.

### Numerische Verfahren und Ergebnisse

Nichtlineare 0-1-Programme können als Spezialfall von nichtlinearen ganzzahligen Optimierungsproblemen oder allgemeiner von nichtlinearen gemischt-ganzzahligen Optimierungsproblemen (engl. *Mixed-Integer-Nonlinear-Programs*, MINLP) betrachtet werden. Entsprechend können Algorithmen aus diesem Bereich angewandt werden (vgl. Hansen, 1979, S. 54). Bei der Auswahl von Algorithmen für MINLPs wird meist zwischen konvexen und nichtkonvexen Problemen<sup>6</sup> unterschieden. Algorithmen, die zum Lösen konvexer MINLPs eingesetzt werden, sind beispielsweise Outer Approximation-Methoden oder Branch-and-Bound-Methoden (vgl. Schewe und Schmidt, 2019, S. 125; Hemker, 2008, S. 28). Auch für nichtkonvexe MINLPs existieren verschiedene exakte Algorithmen, auf die in dieser Arbeit jedoch nicht detailliert eingegangen wird. Eine Übersicht liefern beispielsweise Burer und Letchford (2012). Dort werden zudem (meta-)heuristische Ansätze, wie unter anderem Partikelschwaralgorithmen, aufgeführt.

---

<sup>6</sup>Ein ganzzahliges oder gemischt-ganzzahliges Problem wird als konvex bezeichnet, wenn das Problem, welches sich bei Vernachlässigung der Ganzzahligkeitsbedingungen ergibt, konvex ist.

Zur Lösung des diskreten Optimierungsproblems fanden in dieser Arbeit zwei simple Verfahren Einsatz: eine Brute-Force-Methode und ein Greedy-Algorithmus.

Die **Brute-Force-Methode** wurde zum Lösen des Problems in Formulierung 1 eingesetzt. Bei dieser Methode wird zunächst für sämtliche Kombinationen an *zulässigen* Spiegelpositionierungen der Zielfunktionswert berechnet. Es werden letztlich die Positionierungen ausgegeben, die den höchsten Zielfunktionswert besitzen. Dieses Vorgehen ist rechenintensiv und führt, je nach Diskretisierung des Spiegelfeldes, nicht zu guten Ergebnissen. Ein Vorteil ist jedoch, dass die Rechnungen parallelisiert werden können.

Weniger rechenintensiv ist die Anwendung eines **Greedy-Algorithmus**. Die Spiegel werden bei diesem Algorithmus nach und nach auf den vordefinierten Positionen  $y_1, \dots, y_M$  des Spiegelfeldes positioniert und zwar so, dass der zu setzende Spiegel in der aktuellen Iteration zur maximalen Energie führt. Bei dem Greedy-Algorithmus wäre es nicht sinnvoll eine aufsteigende Positionierung der Spiegelpositionen zu fordern, da die Spiegel lediglich auf einer Seite des Kraftwerksareals positioniert werden würden. Demzufolge wird als Nebenbedingung lediglich der Mindestabstand berücksichtigt. Der Greedy-Algorithmus kann zum einen als Lösungsstrategie aufgefasst werden. Alternativ kann dieser auch als eigene Formulierung verstanden werden: das Optimierungsproblem wird nicht als mehrdimensionales Problem formuliert, sondern es werden viele eindimensionale Probleme betrachtet, die nacheinander gelöst werden. Der Greedy-Algorithmus ließe sich nicht nur im diskreten Fall, sondern auch im Fall der kontinuierlichen Formulierung des Problems einsetzen. In jeder Iteration wäre dann ein eindimensionales, kontinuierliches Optimierungsproblem zu lösen.

Die Ergebnisse der Optimierung basierend auf dem Modell mit und ohne Verschattungseffekte sind Tabellen 11.3 und 11.4 zu entnehmen.<sup>7</sup>

**Tab. 11.3:** Ergebnisse der Optimierung mit einer Brute-Force-Methode mit Mindestabstand  $d_{\min} = 1.1 \text{ m}$ ,  $b = 0.8 \text{ m}$ ,  $h = 6 \text{ m}$ ,  $s = 1.1 \text{ m}$  und  $\Delta t = 0.05 \text{ h}$ . Die Variable  $n_{\text{Pieces}}$  gibt an, in wie viele Intervalle das Kraftwerksareal unterteilt wurde (Feinheit der Diskretisierung).

N	Modell	$n_{\text{Pieces}}$	opt. Positionen $\mathbf{x}^*$	Energieumsatz in Wh	Rechenzeit in s
2	o. Versch.	100	(-0.6, 0.6)	11239.8	12.6
2	o. Versch.	250	(-0.56, 0.56)	11247.1	63.2
2	m. Versch.	100	(-1.20, 1.0)	10231.4	61.4
2	m. Versch.	250	(-1.04, 1.12)	10233.5	332.3

Das Kraftwerksareal von  $l_b = -10.0 \text{ m}$  bis  $u_b = 10.0 \text{ m}$  wurde äquidistant in  $n_{\text{Pieces}}$  Intervalle unterteilt. Die Intervallgrenzen geben die möglichen Positionen an. Sowohl beim Brute-Force-Ansatz als auch beim Greedy-Algorithmus hängen die Rechenzeit

<sup>7</sup>Die Berechnungen wurden mit einem 1.8 GHz Dual-Core Intel Core i5 Prozessor und 8 GB 1600 MHz DDR3 Arbeitsspeicher durchgeführt.

und die Güte der gefundenen Lösung für die Spiegelpositionen von der gewählten Diskretisierung des Kraftwerksareals ab. Wird eine feinere Diskretisierung des Spiegelfeldes gewählt, so führt dies zu besseren Ergebnissen, gleichzeitig nimmt jedoch die Rechenzeit enorm zu. Die Optimierung mit der Brute-Force-Methode wurde lediglich für zwei Spiegelpositionen durchgeführt, da die Optimierung bereits in diesem Fall sehr rechenintensiv ist. Der Greedy-Algorithmus hat gegenüber der Brute-Force-Methode den Nachteil, dass das Ergebnis für die Spiegelpositionen insbesondere bei einer geraden Anzahl zu positionierender Spiegel schlechter ist. Ein wesentlicher Vorteil des Greedy-Algorithmus ist, dass die Rechenzeit gegenüber der Brute-Force-Methode deutlich reduziert ist.

**Tab. 11.4:** Ergebnisse der Optimierung mit einem Greedy-Algorithmus mit Mindestabstand  $d_{\min} = 1.1 \text{ m}$ ,  $b = 0.8 \text{ m}$ ,  $h = 6 \text{ m}$ ,  $s = 1.1 \text{ m}$  und  $\Delta t = 0.05 \text{ h}$

N	Modell	$n_{\text{Pieces}}$	opt. Positionen $\mathbf{x}^*$	Energieumsatz in Wh	Rechenzeit in s
2	o. Versch.	100	(0.0, -1.2)	11185.3	0.6
2	o. Versch.	250	(0.0, -1.12)	11199.3	0.8
4	o. Versch.	250	(0.0, -1.12, 1.12, -2.24)	22036.5	2.2
9	o. Versch.	250	(0.0, -1.12, 1.12, -2.24, 2.24, -3.36, 3.36, -4.48, 4.48)	46199.9	8.3
2	m. Versch.	100	(0.0, 1.8)	10120.9	2.6
2	m. Versch.	250	(0.08, -1.84)	10139.3	4.9
4	m. Versch.	250	(0.08, -1.84, 1.92, -3.52)	19392.1	11.8
9	m. Versch.	250	(0.08, -1.84, 1.92, -3.52, 3.6, -5.12, 5.2, -6.64, 6.72)	38598.9	46.2

### 11.2.4 Evaluation der Ergebnisse

*Welche Formulierung ist denn nun am Besten?* Auf diese Frage gibt es letztlich keine richtige oder falsche Antwort. Stattdessen wird deutlich, was mathematisches Modellieren auszeichnet: Modellieren ist kein deterministischer Prozess, der stets in einer eindeutigen besten Lösung mündet. Vielfach entscheiden ökonomische, technische oder ökologische Überlegungen, die je nach individueller Gewichtung der Problemlösenden auf unterschiedliche (mathematische) Modelle mit spezifischen Vor- und Nachteilen hinauslaufen, was als praktikabel und damit gut zu bewerten ist. Soll das Optimierungsproblem mit wenig Rechenaufwand und einem schnell implementierten Algorithmus gelöst werden und ist es dabei unwichtig, ob die gefundene Lösung für die Spiegelpositionen dafür *etwas* schlechter ist? Dann wird womöglich

ein Greedy-Ansatz zum Lösen des Problems in der diskreten Formulierung oder der musterbasierte Ansatz (und damit ein weniger hochdimensionales Problem) gewählt. Gegebenenfalls ist jedoch zentral, dass „auf den Millimeter genau“ die bestmögliche Positionierung der Spiegel gefunden wird. Dann wählen wir womöglich den freien Variablen Ansatz – vorausgesetzt wir haben die entsprechende Rechenpower und sind bereit längere Rechenzeiten in Kauf zu nehmen.

## 11.3 Zwischenfazit und Konsequenzen für die Modellierung mit Lernenden

Aus den vorherigen Kapiteln wurde ersichtlich, dass die Modellierung und Optimierung des vereinfachten Fresnelkraftwerks auf diversen schulmathematischen Inhalten basiert und zugleich die Diskussion wichtiger Fragestellungen der mathematischen Optimierung aufgreift. So erlaubt die Optimierung der Absorberrohrhöhe Charakteristika der Zielfunktion anhand des Funktionsgraphen zu untersuchen und damit auf anschaulicher Ebene Begriffe wie Existenz und Eindeutigkeit von Optimierungsproblemen zu diskutieren. Durch den Einbau diverser Modellerweiterungen wurde das nichtkonvexe Problem, ohne optimalen und insbesondere ohne realistischen Optimalpunkt, in ein Problem überführt, für das ein eindeutiger Optimalpunkt existiert, der zudem konstruierbar ist.

Bei der Optimierung der Spiegelpositionen wurde deutlich, dass diverse Möglichkeiten existieren, wie ein und dieselbe Optimierungsfrage als mathematisches Optimierungsproblem formuliert und gelöst werden kann. Dies stellt einen hervorragenden Ausgangspunkt für die Entwicklung von Lernmaterial dar, welches kreatives und möglichst offenes Modellieren seitens der Lernenden erlaubt.

Die Gestaltung des Lernmaterials und der Grad der Anleitung bei der Durchführung der Modellierungsprojekte sollte dabei unbedingt an bisherige Erfahrungen der Lernenden mit offenen Modellierungsaufgaben angepasst werden. Insbesondere bei Schülerinnen und Schülern, die bisher kaum Erfahrungen mit derartigen Aufgaben gesammelt haben, besteht die Gefahr einer Überforderung, wenn sogleich verlangt wird, vollkommen unangeleitet an die Modellierung heranzugehen. Deswegen wurde Lernmaterial variierenden Offenheitsgrades entwickelt. Auf der einen Seite ist Material entstanden, welches den Lernenden erlaubt eigene Ansätze unangeleitet und selbstständig zu verfolgen und computergestützt umzusetzen (vgl. Abschn. 12.2.4). Demgegenüber liegt stärker angeleitetes Material bereit, welches eine Auswahl der oben genannten Optimierungsansätze aufgreift und den Lernenden ermöglicht diese angeleitet durch kleinere Aufgabenstellungen anzuwenden und vergleichend zu diskutieren (vgl. Abschn. 12.2.3). Je nach zeitlicher Rahmenbedingung ist klar, dass nur ein Teil der unterschiedlichen Ansätze mit den Lernenden erarbeitet werden kann. Auf Möglichkeiten das Lernmaterial zu den verschiedenen Optimierungsansätzen je nach Rahmenbedingungen flexibel einzusetzen, wird in Abschnitt 12.3 eingegangen.

# 12 Umsetzung im Lernmaterial

In diesem Kapitel wird Lernmaterial für zwei Workshops, einer für die Mittelstufe und einer für die Oberstufe, vorgestellt. Anhand des Materials soll deutlich werden, wie ein sinnvoller Umgang mit Optimierungsproblemen bei der Bearbeitung realer, komplexer Problemstellungen im Rahmen von mathematischen Modellierungsaktivitäten gelingen kann.

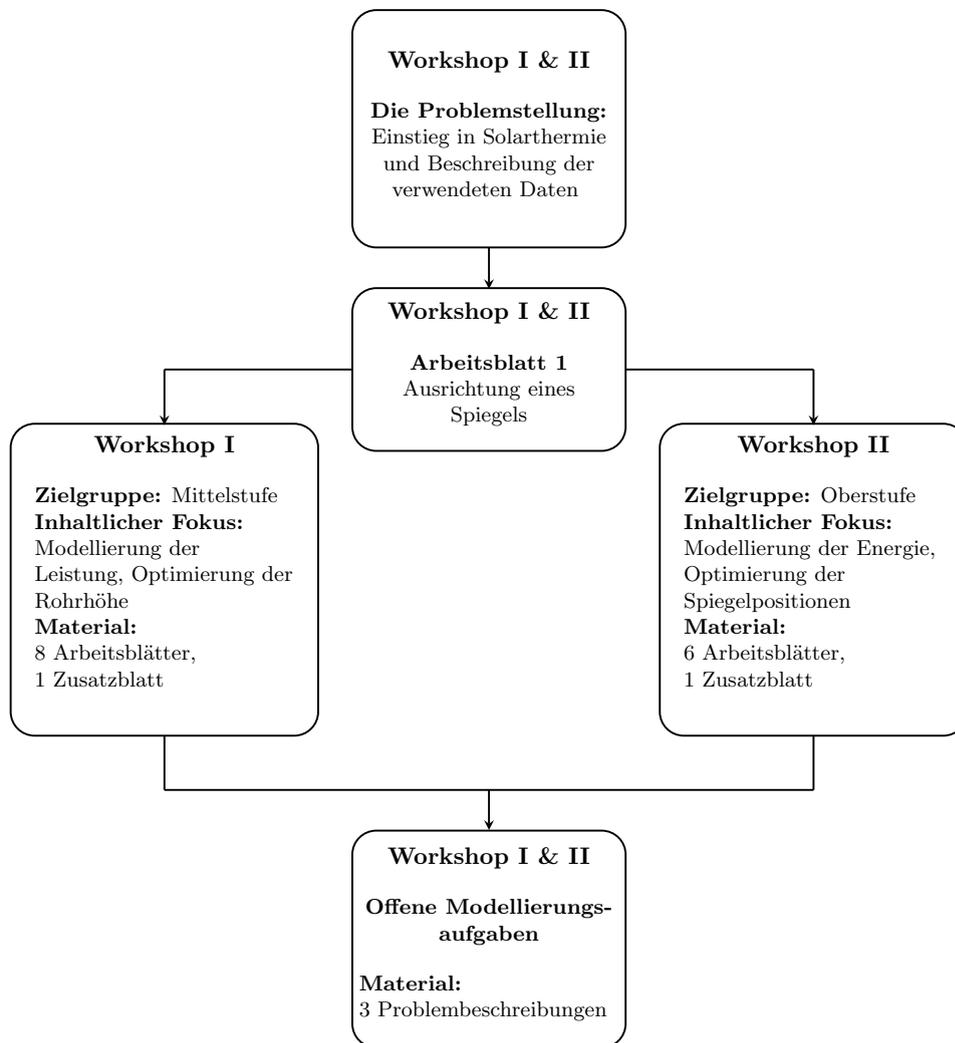
Die Zuordnung der Inhalte zu den beiden Workshops ist Abbildung 12.1 zu entnehmen. Die Inhalte sind nach Lernmaterial gegliedert, welches die Optimierung der Rohrhöhe (Workshop I, Mittelstufe) aufgreift und solches, welches die Optimierung der Spiegelpositionen (Workshop II, Oberstufe) thematisiert. Es wird ersichtlich, dass der Einstieg in die Thematik und die Modellierung der Spiegelausrichtung bei beiden Workshops identisch ist. Das Lernmaterial beider Workshops wurde für die Durchführung im Rahmen von mathematischen Modellierungstagen entwickelt. Das Material, welches zu Beginn der beiden Workshops eingesetzt wird, ist stärker angeleitet konzipiert. Der Modellierungsprozess der Lernenden wird durch kleinere Aufgaben und Diskussionsfragen strukturiert. Zusätzlich wurde Lernmaterial entwickelt, welches das unangeleitete Bearbeiten verschiedener offener Teilprobleme ermöglicht (vgl. Abschn. 12.2.4). Dieses kann zu verschiedenen Zeitpunkten an das stärker angeleitete Material angebunden und der Modellierungsprozess damit geöffnet werden. Eine detaillierte, tabellarische Übersicht über den Aufbau des Materials und dessen schulmathematische Anknüpfungspunkte ist für den Mittelstufenworkshop in Anhang B.1 und für den Oberstufenworkshop in Anhang C.1 zu finden.

## 12.1 Voraussetzungen und Ziele

Die Voraussetzungen und Ziele des Unterrichtsmaterials werden hier separat für die beiden Zielgruppen (Mittel- oder Oberstufe) beschrieben. Dabei wird lediglich auf die inhaltlichen Hauptziele eingegangen. Die jeweiligen Teilziele der einzelnen Bausteine des Materials werden in den folgenden Abschnitten bei der Darstellung des Lernmaterials deutlich. Ziele, die unabhängig von mathematisch-inhaltlichen Lernzielen mit den Modellierungsaktivitäten verfolgt werden, wurden in Abschnitt 6.1 beschrieben.

### **Workshop I (Mittelstufe)**

Der Schwerpunkt des Workshops I liegt auf der Modellierung der Leistung des Kraftwerks und der Optimierung der Absorberrohrhöhe. Dabei finden zahlreiche mathematische Begriffe und Konzepte Anwendung, die aus dem Geometrieunterricht der



**Abb. 12.1:** Übersicht über die Inhalte des Workshops für die Mittelstufe (Workshop I) und für die Oberstufe (Workshop II)

Sekundarstufe I bekannt sind. Der Workshop richtet sich vornehmlich an Lernende ab Klasse 9.<sup>1</sup>

Für die Bearbeitung des Lernmaterials wird vorausgesetzt, dass die Lernenden im Bereich Geometrie im Umgang mit Winkelpaaren, der Winkelsumme im Dreieck, Stufen- sowie Wechselwinkeln und dem Satz des Pythagoras vertraut sind. Zudem ist ein Verständnis von relativen Anteilen und insbesondere von funktionalen Zusammenhängen notwendig. Günstig ist es, wenn die Lernenden trigonometrische Beziehungen im rechtwinkligen Dreieck (Sinus, Kosinus und Tangens) bereits kennen. Dies ist jedoch keine notwendige Voraussetzung, da diese bei Bedarf durch entsprechendes Lernmaterial anschaulich und problemorientiert eingeführt werden. Überdies ist die

<sup>1</sup>Die genaue curriculare Einordnung hängt vom Lehrplan des betrachteten Bundeslandes, den schulinternen Curricula sowie der jeweiligen Lerngruppe ab.

Anwendung des Reflexionsgesetzes (Physik) notwendig. Sofern die Lernenden dieses nicht präsent haben, liegt entsprechendes Informationsmaterial bereit.

Mit dem Workshop werden die folgenden Ziele verfolgt (das Zusatzmaterial ist nicht inbegriffen):

Die Lernenden ...

- entwickeln basierend auf bekannten oder, im Falle trigonometrischer Beziehungen, gegebenenfalls problemorientiert eingeführten geometrischen Überlegungen ein Modell für Spiegelausrichtung, Leistung auf dem Spiegel und Leistung am Absorberrohr.
- führen die Optimierung der Rohrhöhe basierend auf dem entwickelten, einfachen Modell für die Leistung (vgl. Abschn. 11.1.1) durch Ablesen der Lösung am Funktionsgraphen und unter Verwendung eines Optimierungstools als Blackbox durch. Sie interpretieren die Ergebnisse im Hinblick auf das reale Problem.
- diskutieren anhand des Zielfunktionsgraphen, dass bei dem einfachen Modell keine optimale Rohrhöhe existiert. Sie sammeln Ideen zur Verbesserung des Modells.
- setzen ausgewählte Modellerweiterungen um. Dabei kommen wiederum vermehrt geometrische Überlegungen zum Einsatz.
- führen nach jeder Modellerweiterung die Optimierung der Rohrhöhe durch Ablesen der Lösung am Funktionsgraphen und basierend auf dem Optimierungstool durch. Sie diskutieren, inwieweit die gefundene optimale Rohrhöhe im Hinblick auf die Konstruktion des Kraftwerks sinnvoll ist. Zudem erfassen sie die Änderung des Verlaufs des Zielfunktionsgraphen und diskutieren basierend auf dieser Änderung, inwieweit eine konstruierbare optimale Rohrhöhe existiert.

Es wird bereits ersichtlich, dass die Optimierung im Unterrichtsmaterial für die Mittelstufe primär durch das Ablesen von Extremstellen am Funktionsgraphen durchgeführt wird. Dieses Vorgehen ließe sich gemäß [Schupp \(1997, S. 8\)](#) der Strategie „Optimieren durch Finitisieren“ zuordnen.

### **Workshop II (Oberstufe)**

Der Schwerpunkt des Workshops für die Oberstufe liegt weniger auf der physikalisch-technischen Modellierung der Leistung des Kraftwerks basierend auf geometrischen Überlegungen. Stattdessen steht die **Modellierung von Optimierungsproblemen** und die Entwicklung sowie Anwendung von Verfahren zur Lösung solcher Probleme im Fokus. Der Workshop richtet sich an Lernende der Oberstufe ab Klasse 10/11. Voraussetzungen für die Bearbeitung des Workshops ist ein sicherer Umgang mit Funktionen und Ungleichungen sowie Kenntnisse trigonometrischer Beziehungen im rechtwinkligen Dreieck. Sollen die Lernenden zudem die Verschattungseffekte modellieren, so ist es günstig, wenn die Darstellung von Geraden in Parameterform

und die Bestimmung von Schnittpunkten zwischen eben diesen unter Verwendung von technischen Hilfsmitteln (Taschenrechner / Computer) beherrscht wird.

Mit dem Workshop werden die folgenden Ziele verfolgt (Ziele des Zusatzmaterials werden dabei nicht aufgegriffen):

Die Lernenden ...

- entwickeln ein Modell für die Ausrichtung der Spiegel und wenden dabei verschiedene Winkelbeziehungen an. Anschließend wird das Modell für die Leistung am Rohr entweder gegeben oder die Lernenden erarbeiten dieses analog zum Mittelstufenworkshop.
- entwickeln ein Modell für die über einen Tag umgesetzte Energie. Genauer modellieren sei den Energieumsatz über eine Produktsomme.
- führen die Optimierung der Position **eines** Spiegels durch, indem sie die Extremstelle am Funktionsgraphen ablesen. Sie interpretieren das Ergebnis im Hinblick auf die reale Situation. Anschließend erweitern sie das Modell für zwei Spiegel und lesen die optimalen Positionen am nun dreidimensionalen Funktionsgraphen ab. Sie diskutieren, inwieweit die gefundenen optimalen Positionen realistisch konstruierbar sind, und sammeln Ideen zur Verbesserung des Modells.
- entwickeln eine Nebenbedingung (in Form einer Ungleichung) für den Mindestabstand zwischen **zwei** Spiegeln. Sie berücksichtigen die Nebenbedingung bei der Optimierung der Spiegelpositionen. Sie bestimmen die optimalen Positionen durch systematisches Einsetzen in die Zielfunktion.
- sammeln Ideen, wie die Optimierung der Spiegelpositionen für mehr als zwei Spiegel automatisiert werden kann. Sie notieren Schritte eines möglichen Optimierungsverfahrens und diskutieren ihre Ansätze mit ihren Mitlernenden.
- diskutieren bei der Anwendung verschiedener Optimierungsformulierungen und Algorithmen, inwieweit Existenz und Eindeutigkeit einer optimalen Positionierung gegeben sind. Sie wägen die Güte der gefundenen Lösung für die Spiegelpositionen gegen den Zeitaufwand des eingesetzten Verfahrens ab.
- modellieren den Schattenwurf durch nebenstehende Spiegel für einen Spezialfall (feste Tageszeit und feste Spiegelpositionen). Sie nutzen dazu die Beschreibung von Spiegeln und Sonnenstrahlen als (Halb-)Geraden in Parameterform und berechnen Schnittpunkte zwischen Geraden.
- führen die Optimierung der Spiegelpositionen für das erweiterte Modell mit Verschattungseffekten durch und diskutieren die erhaltenen Ergebnisse basierend auf den bereits kennengelernten Optimierungsansätzen.

Der Fokus des Materials für die Oberstufe liegt auf der Diskussion zentraler Fragestellungen beim Lösen von Optimierungsproblemen in der Praxis (Eindeutigkeit, Existenz, Güte und Rechenzeit). Zentral ist dabei die Brainstorming-Phase zu den eigenen Ideen für ein Optimierungsverfahren. Algorithmisches Denken kann und soll in diesem Workshop in besonderem Maße geschult werden.

## 12.2 Darstellung des Lernmaterials

In diesem Abschnitt wird das Lernmaterial dargestellt. Dabei werden die für die Bearbeitung des Materials notwendigen inhaltlichen Kompetenzen aufgezeigt. Bei der Beschreibung des Materials wird nicht im Detail auf einzelne Aufgabenstellungen eingegangen. Stattdessen sei auf das Lernmaterial in Anhang B und C beziehungsweise erneut auf die Möglichkeit verwiesen, auf das Material über die Workshop-Plattform zuzugreifen (vgl. [Vorwort](#)).

Zuerst wird das Lernmaterial vorgestellt, welches zu Beginn der beiden Workshops für die Mittelstufe (Workshop I) und für die Oberstufe (Workshops II) identisch ist. Anschließend wird hinsichtlich der beiden Workshops differenziert. Abschließend wird auf Lernmaterial für verschiedene offene Modellierungsprobleme eingegangen. Es sei bereits hier erwähnt, dass das Material die Möglichkeit einer flexiblen (workshopübergreifenden) Kombination bietet. Dies wird in Abschnitt 12.3 ausführlicher erläutert.

Für die Entwicklung des Lernmaterials wurden die in Teil II beschriebenen digitalen Werkzeuge (Jupyter Notebook bereitgestellt auf JupyterHub, vgl. Kap. 5) und methodischen Bausteine (vgl. Kap. 6) ein- beziehungsweise umgesetzt.

### 12.2.1 Workshopübergreifendes Lernmaterial

#### Folien für die Einstiegspräsentation

Im Einstiegsvortrag (vgl. Anh. B.5.1) erhalten die Lernenden einen kurzen Überblick über die Solarthermie und einen Einblick in die Funktionsweise von Fresnelkraftwerken. Dabei werden die Lernenden mit ihrem Vorwissen rund um die Solarenergie in die Diskussion eingebunden. Abschließend werden erste Vereinfachungen vorgenommen. Das reale und vereinfachte Problem werden in den vierschriftigen Modellierungskreislauf angelehnt an [Blum \(1985\)](#) eingebettet. Dieser wird als strukturgebendes Element auch in nachfolgenden Diskussionsphasen aufgegriffen.

#### Arbeitsblatt 1: Ausrichtung der Spiegel

Das erste Arbeitsblatt wurde angelehnt an das Material aus dem Workshop von [Krahforst \(2016\)](#) hinsichtlich des verwendeten digitalen Werkzeugs (Jupyter Notebook / Julia statt Matlab) und einzelner Formulierungen von Aufgabenstellungen angepasst. Ziel des Arbeitsblattes ist die Einstellung des Normalenwinkels, sodass die gewünschte Reflexion auf das Absorberrohr erzielt wird. Die Lernenden wenden dazu das Reflexionsgesetz an. Dabei wird vereinfachend nur ein Spiegel direkt unter dem Rohr betrachtet. Diverse *Tippkarten* sind auf dem Arbeitsblatt verlinkt.

### 12.2.2 Lernmaterial zum Workshop I

#### Antwort- und Dokumentationsblätter

Die Lernenden werden an verschiedenen Stellen auf den Arbeitsblättern aufgefor-

dert ihre Ergebnisse zu dokumentieren oder Ideen für Modellverbesserungen zu verschriftlichen. Dazu liegen **Antwortblätter** bereit, auf denen Ergebnisse und Antworten strukturiert dokumentieren werden können. Dies soll insbesondere in den Diskussionsphasen dazu führen, dass die Lernenden übersichtlich und zügig auf ihre Ergebnisse zugreifen können ohne verschiedene bearbeitete Arbeitsblätter öffnen zu müssen. Für den Einsatz des Materials in einem zeitlich ausgedehnten Rahmen, beispielsweise im Zuge einer Unterrichtsreihe, liegen **Dokumentationsblätter** bereit, auf denen die Lernenden wichtige Formeln bei der Entwicklung des Modells für Leistung und Energie dokumentieren sollen.

### **Arbeitsblatt 2 und 3: Die Leistung am Spiegel und am Rohr**

Nachdem die gewünschte Spiegelausrichtung erzielt wurde, entwickeln die Lernenden ein Modell für die Leistung, die den Spiegel erreicht (AB 2) und anschließend für die Leistung, die am Sekundärreflektor beziehungsweise dem Absorberrohr ankommt (AB 3). Erneut wird vereinfachend nur ein Spiegel betrachtet, der direkt unter dem Rohr positioniert ist. Auch hier liegt der Fokus auf geometrischen beziehungsweise trigonometrischen Überlegungen und Zusammenhängen in rechtwinkligen Dreiecken. Diese beiden Arbeitsblätter existierten in ähnlicher Form bereits von [Krahforst \(2016\)](#) und werden deswegen nicht weiter beschrieben.

### **Arbeitsblatt 4 (Langversion): Horizontal verschobene Spiegel**

Auf diesem Arbeitsblatt erweitern die Lernenden ihr Modell auf horizontal nach recht oder links verschobene Spiegel. Auch hier liegt der mathematische Fokus auf der Geometrie. Dieses Arbeitsblatt wurde angelehnt an das Material von [Krahforst \(2016\)](#) leicht modifiziert.

### **Arbeitsblatt 4 (Kurzversion): Infoblatt zu horizontal verschobenen Spiegeln**

Ziel des Workshops sollte es sein, nicht nur die physikalisch-technische Modellierung des Kraftwerks und dessen Leistung zu erarbeiten, wie es im bisherigen Material von [Krahforst \(2016\)](#) der Fall war, sondern zusätzlich die Optimierung der Absorberrohrhöhe durchzuführen. Da das ursprüngliche Arbeitsblatt 4 in der Langversion viel Zeit beansprucht, die angewendete Mathematik jedoch der von Arbeitsblättern 1 bis 3 entspricht, wurde eine Kurzversion des Arbeitsblattes 4 erstellt. Die Lernenden erhalten dort eine Beschreibung, welche weiteren Parameter einbezogen werden müssen (die Rohrhöhe  $h$  und die Spiegelposition  $x$ ), um die Ausrichtung und die Leistung auch im Falle horizontal verschobener Spiegel modellieren zu können.

An dieser Stelle des Workshops kann innerhalb der Lerngruppe differenziert werden, indem schnellere Lernende das Arbeitsblatt 4 in der Langversion bearbeiten. Zudem besteht die Möglichkeit, das im Folgenden beschriebene Zusatzblatt zu bearbeiten, auf dem das Modell für die Leistung auf mehrere Spiegel erweitert werden kann.

### **Zusatzblatt: Leistung erzeugt durch mehrere Spiegel**

Auf diesem Arbeitsblatt erhalten die Lernenden die Aufgabe, ihr Modell auf mehrere

Spiegel zu erweitern. Dazu summieren sie die durch mehrere Spiegel generierte Leistung auf. Alternativ kann die Berechnung über eine for-Schleife realisiert werden. Zur Verwendung von for-Schleifen liegt eine *Tippkarte* bereit, auf die die Lernenden bei Bedarf zugreifen können.

### **Zusatzblatt: Über einen Tag umgesetzte Energie**

Dieses Zusatzblatt ist analog zu Arbeitsblatt 2 des Workshops für die Oberstufe aufgebaut und wird hier nicht beschrieben.

### **Folien für die Zwischendiskussion**

Die bisherigen Modellierungsschritte werden resümiert und Teilergebnisse der Lernenden gesichert. Dazu liegen Präsentationsfolien bereit, auf denen zentrale Skizzen der Arbeitsblätter 1 – 4 abgebildet sind (vgl. Anh. B.5.2). Anhand dieser können die Lernenden geometrische Überlegungen erläutern. Es wird diskutiert, welche Parameter des Kraftwerks vor Konstruktion optimiert werden könnten. Abschließend wird das Leitziel des zweiten Teils des Workshops benannt: Die Optimierung der Absorberrohrhöhe, basierend auf dem entwickelten Modell für die Leistung am Rohr.

### **Antwortblatt zur Optimierung**

Für die Optimierung der Absorberrohrhöhe erhalten die Lernenden ein Antwortblatt, auf dem sie die Ergebnisse für die optimale Rohrhöhe basierend auf den unterschiedlichen Modellen dokumentieren. Ziel ist, einen übersichtlichen Vergleich zu ermöglichen, wie sich die Modellerweiterungen auf die optimale Rohrhöhe auswirken.

### **Arbeitsblatt 5: Optimierung der Rohrhöhe**

Das Modell für die Leistung am Rohr ist eine Funktion, die von mehreren Parametern abhängt. Die Parameter Spiegelbreite  $s$ , Sekundärreflektorbreite  $b$  und Tageszeit  $t$  werden auf konkrete Werte festgelegt. Zur besseren Vergleichbarkeit innerhalb der Lerngruppe wählen alle Schülerinnen und Schüler die gleichen Werte. Ziel des Arbeitsblattes ist die Bestimmung der optimalen Höhe des Absorberrohrs. Die Lernenden können zu Beginn des Arbeitsblattes auf ein Infoblatt zugreifen, auf dem die drei wesentlichen Bausteine von Optimierungsproblemen (Zielfunktion, Variablen, Nebenbedingungen) beschrieben und auf das vorliegende Problem übertragen werden.

Die Lernenden bestimmen die optimale Rohrhöhe durch Ablesen am Funktionsgraphen (vgl. Abb. 11.1). Fahren sie mit der Maus über den Graphen, so werden die Funktionswerte angezeigt. Der Graph der Zielfunktion steigt ab einer Höhe von circa 10 – 20 m sehr langsam. Es liegt nahe, als *guten* Wert eine Höhe im Bereich von 10 – 20 m zu wählen, was von den Lernenden auch vielfach genannt wird. Dies ist vor dem Hintergrund von Wirtschaftlichkeitsfaktoren wie Materialkosten durchaus sinnvoll, wenngleich dies nicht einer optimalen Lösung des betrachteten Problems im (rein) mathematischen Sinne entspricht. Um dies deutlicher zu machen, geben die Lernenden ihre Lösung für die Rohrhöhe im digitalen Arbeitsblatt ein und lassen sich den zugehörigen Funktionswert ausgeben. Anschließend sollen sie den Wert

für die gefundene Höhe leicht erhöhen und sich den Funktionswert erneut ausgeben lassen. Es wird ersichtlich, dass für wachsendes  $h$  immer größere Zielfunktionswerte erreicht werden. Motiviert durch die Tatsache, dass das Ablesen am Funktionsgraphen nicht genau, umständlich und in der Praxis vielfach nicht möglich ist, kommt dann ein Optimierungstool<sup>2</sup> zum Einsatz. Die Funktionsweise des Tools wird nicht mit den Lernenden diskutiert. Stattdessen wird das Tool als Blackbox eingesetzt. Das „Optimieren durch Ablesen am Funktionsgraphen“ und der Vergleich mit den Ergebnissen des Optimierungstools zieht sich auch im weiteren Verlauf des Workshops als roter Faden durch das Lernmaterial.

Die Lernenden diskutieren, inwieweit die gefundenen Ergebnisse realistisch sind. Zentrale Erkenntnis des Arbeitsblattes ist, dass keine optimale Rohrhöhe existiert, da der Graph der Zielfunktion einen monoton steigenden, asymptotischen Verlauf aufweist. Die Lernenden sammeln Modellverbesserungsideen, die in der anschließenden Plenumsdiskussion diskutiert werden.

Dieses Arbeitsblatt macht ersichtlich, dass die Optimierung eine Validierung des Modells ermöglicht hat, die zu der Erkenntnis führt, dass die Realität, will heißen die tatsächlich umgesetzte Leistung, noch unzureichend durch das mathematische Modell beschrieben wird.

### Folien für die Zwischendiskussion

In der Zwischendiskussion werden die Erkenntnisse der Lernenden im Hinblick auf die optimale Rohrhöhe diskutiert. Dabei werden sowohl konstruktionstechnisch sinnvolle oder gar konstruierbare Ergebnisse, als auch die mathematische Lösung aufgegriffen. Die Eigenschaft der Zielfunktion, streng monoton zu wachsen, wird als Grund für die Nichtexistenz einer optimalen Rohrhöhe diskutiert. Zugunsten der Problemorientierung werden Begriffe wie *Koerzivität* oder *Konveritat / Konkavitat* nicht explizit eingeführt.

Die bisherigen Modellierungsschritte werden in den vierschrittigen Modellierungskreislauf eingeordnet. Anschließend werden Modellverbesserungsideen der Lernenden diskutiert. Motiviert durch die Erkenntnis, dass das Problem keine optimale Lösung besitzt und das Rohr unendlich hoch gebaut werden müsste, bearbeiten die Lernenden im Anschluss eine der beiden folgenden Modellerweiterungen in beliebiger Reihenfolge: a) atmospharische Effekte oder b) Berucksichtigung einer fehlerhafte Spiegelausrichtungen.

### Arbeitsblatt 6: Modellerweiterung – atmospharische Effekte

Ziel des Arbeitsblattes ist es, das Modell fur die Leistung am Rohr um atmospharische Abschwachungseffekte zu erweitern. Basierend auf dem erweiterten Modell wird die Optimierung der Rohrhohe erneut durchgefuhrt.

Die Lernenden erhalten den Arbeitsauftrag, eine Formel fur die Lange der Strecke

<sup>2</sup>Obleich am Verlauf des Graphen ersichtlich wird, dass das Problem keinen Minimalpunkt besitzt, kommt das Optimierungstool hier zum Einsatz. Grund ist, dass diese Erkenntnis erfahrungsgemaß nicht trivial fur die Lernenden ist (vgl. Abschn. 13.2). Stattdessen wurde wiederholt beobachtet, dass die Lernenden als „optimale“ durch Ablesen am Graphen erhaltene Rohrhohe zumeist einen Wert in einem realistisch konstruierbaren Bereich benannten.

aufzustellen, die ein Lichtstrahl vom Spiegelmittelpunkt bis zum Rohr zurücklegt. Die Formel kann durch Anwenden des Satz des Pythagoras aufgestellt werden. Haben die Lernenden Schwierigkeiten, können sie auf eine verlinkte *Tippkarte* zugreifen. Auf dieser ist eine Skizze der betrachteten Situation mit eingezeichnetem rechtwinkligen Dreieck zu sehen. Zudem erhalten sie den Hinweis, den Satz des Pythagoras anzuwenden. Anschließend erhalten die Lernenden einen kurzen Informationstext, in dem ein in der Solarenergieforschung vielfach verwendetes Modell zur Beschreibung der Abschwächung des Lichts in Abhängigkeit der zurückgelegten Strecke eingeführt wird.

Ausgehend von dem erweiterten Modell für die Leistung am Rohr wird die Optimierung der Rohrhöhe zunächst grafisch und anschließend mithilfe des Optimierungstools durchgeführt. Am Verlauf des Graphen der Zielfunktion wird ersichtlich, dass ein globaler Minimalpunkt des Optimierungsproblems existiert. Die Lernenden validieren, inwieweit die gefundene optimale Rohrhöhe praktikabel ist.

### **Arbeitsblatt 7: Modellerweiterung – fehlerhafte Spiegelausrichtung**

Ziel des Arbeitsblattes ist es, das Modell für die Leistung am Rohr so zu erweitern, dass kleinere Einstellungsfehler bei der Ausrichtung der Spiegel berücksichtigt werden. Basierend auf dem erweiterten Modell wird die Optimierung der Rohrhöhe erneut durchgeführt. Die Modellierung wird lediglich für einen horizontal nach links verschobenen Spiegel ( $x < 0$ ) vorgenommen, da sonst weitere Fallunterscheidungen und eine repetitive Anwendung geometrischer Überlegungen notwendig wäre.

Um eine fehlerhafte Spiegelausrichtung in das Modell zu integrieren, leiten die Lernenden die Länge der Fehlerstrecke  $f$  mithilfe trigonometrischer Zusammenhänge her (s. Gleichung (11.7) in Abschn. 11.1.3). Haben sie Schwierigkeiten, so können sie auf eine *Tippkarte* zugreifen. Auf dieser ist eine Skizze der betrachteten Situation mit eingezeichnetem rechtwinkligen Dreieck zu sehen. Zudem erhalten sie den Tipp den Tangens zu nutzen. Anschließend bestimmen die Lernenden die Länge der Strecke, die auf Höhe des Rohres bestrahlt wird und letztlich den Anteil, der tatsächlich auf den Sekundärreflektor trifft. Auch hier sind geometrische Überlegungen in rechtwinkligen Dreiecken zentral. Diverse *Tippkarten* mit unterstützenden Skizzen liegen bereit.

Haben die Lernenden die Leistung unter Fehlereinfluss für einen nach links verschobenen Spiegel modelliert, so können sie die Optimierung der Rohrhöhe mit dem erweiterten Modell für einen fest gewählten Fehler durchführen. Auch hier wird die Optimierung erneut grafisch und mithilfe des Optimierungstools vorgenommen. Am Graph der Zielfunktion wird erkennbar, dass eine global optimale Lösung für die Rohrhöhe existiert. Die Lernenden sollen das Ergebnis für die Rohrhöhe im Hinblick auf die Konstruktion des Kraftwerks validieren.

In einer Zusatzaufgabe können sich schnellere Lernende Gedanken zu weiteren Fehlerquellen bei der Modellierung des Kraftwerks machen.

### **Arbeitsblatt 8: Kombination beider Modellerweiterungen**

Nach der Bearbeitung einer oder beider Modellerweiterungen (AB 6 oder 7), ist das Ziel dieses Arbeitsblattes, den Einfluss der Kombination beider Modellerweiterungen

auf die optimale Rohrhöhe zu untersuchen.

Zunächst variieren die Lernenden die Größe des Einstellungsfehlers und vergleichen ihre Ergebnisse mit den Ergebnissen von Arbeitsblatt 6 oder 7. Zudem erklären sie den Verlauf des Graphen sowie die Ergebnisse der Optimierung. Es wird ersichtlich, dass die optimale Rohrhöhe für einen Einstellungsfehler nahe Null durch die atmosphärischen Abschwächungseffekte bestimmt wird. Ist der Fehler größer, so dominiert der Einfluss des Einstellungsfehlers und die optimale Rohrhöhe entspricht den Ergebnissen von Arbeitsblatt 7.

Anschließend haben die Lernenden die Möglichkeit die Position des Spiegels und die Tageszeit zu variieren und den Einfluss auf die optimale Rohrhöhe zu untersuchen. Es wird ersichtlich, dass sowohl die Tageszeit als auch die Position des Spiegels einen Einfluss auf die optimale Höhe haben. Die Lernenden erhalten den Auftrag Ideen für Modellverbesserungen zu sammeln. Für die folgenden beiden Modellverbesserungen liegen Zusatzblätter bereit:

- Die Optimierung der Rohrhöhe basierend auf dem auf mehrere Spiegel und auf den Energieumsatz erweiterten Modell.
- Die Berücksichtigung von verschiedenen Fehlern bei der Spiegeleinstellung (Monte-Carlo-Methode) und die anschließende Optimierung der Rohrhöhe.

Beide Zusatzblätter werden im Folgenden kurz beschrieben.

#### **Zusatzblatt: Erweiterung auf mehrere Spiegel und die umgesetzte Energie**

Auf diesem Zusatzblatt können die Lernenden die Optimierung der Rohrhöhe basierend auf dem Modell durchführen, welches auf den Energieumsatz eines Tages und auf mehrere Spiegel erweitert wurde. Die Modellerweiterungen müssen die Lernenden nicht selbst vornehmen. Die Lernenden vergleichen und dokumentieren die Ergebnisse der Optimierung mit den bisherigen Ergebnissen von Arbeitsblättern 6 – 8. Es wird ersichtlich, dass die Modellerweiterungen wiederum zu anderen Ergebnissen für die optimale Rohrhöhe führen.

#### **Zusatzblatt: Optimierung der Rohrhöhe unter variablem Einstellungsfehler**

Ziel dieses Zusatzblattes ist es, dass die Lernenden die Grundidee der Monte-Carlo-Methode kennenlernen und basierend auf dieser Methode die Optimierung der Rohrhöhe erneut ausführen. Dazu wird die Grundidee der Monte-Carlo-Methode zunächst beschrieben. Anschließend wenden die Lernenden die bereits implementierte Monte-Carlo-Methode zur Modellierung der erwarteten Leistung am Rohr unter fehlerhafter Spiegelausrichtung an. Die Zielfunktion der erwarteten Leistung in Abhängigkeit der Rohrhöhe wird grafisch dargestellt (vgl. Abb. 11.7). Die Lernenden variieren die Anzahl der pro Spiegel generierten Fehler und beobachten, welchen Einfluss die Änderung auf den Graph der Zielfunktion hat. Zudem begründen sie, welchen Wert sie für die Anzahl der Fehler pro Spiegel wählen würden. Anschließend führen die Lernenden die Optimierung der Rohrhöhe mit einem Optimierungstool aus und vergleichen ihre Ergebnisse mit den Ergebnissen von Arbeitsblatt 7.

### Folien für die Abschlussdiskussion

Für die Abschlussdiskussion liegen Folien bereit, auf denen wesentliche Leitfragen für die Diskussion und Sicherung der Ergebnisse aufgegriffen werden. Die Lernenden sollen sich in dieser Diskussion insbesondere zu folgenden Punkten einbringen:

- Welche Erkenntnisse und Beobachtungen sie bei der Optimierung der Rohrhöhe gesammelt haben.
- Welche Rohrhöhe sie letztlich für die Konstruktion des Kraftwerks wählen würden und warum.
- Welche Ideen sie für Modellverbesserungen haben.

Abschließend wird ein Ausblick auf weiterführende Fragen aus der Forschung zu solarthermischen Kraftwerken gegeben. Beispiele sind die Optimierung von Reinigungsrouten oder die optimale Verkabelung der Spiegel.

## 12.2.3 Lernmaterial zum Workshop II

Im Workshop für die Oberstufe entwickeln die Lernenden zunächst das Modell für die Ausrichtung eines Spiegels, der direkt unter dem Rohr positioniert ist. Die Modellierung der Leistung am Rohr und die Erweiterung auf horizontal verschobene Spiegel brauchen die Lernenden nicht vorzunehmen. Stattdessen erhalten sie die Funktion  $P_{\text{tube}}$  als Ausgangspunkt für die anstehende Optimierung. Diese beschreibt die Leistung am Rohr in Abhängigkeit von den Parametern Tageszeit  $t$ , Spiegelposition  $x$ , Reflektorbreite  $b$ , Spiegelbreite  $s$  und Absorberrohrhöhe  $h$ . Die Herleitung der Funktion  $P_{\text{tube}}$  (AB 2 – 4 von Workshop I, vgl. Abschn. 12.2.2) ist nicht Bestandteil dieses Workshops. Sie kann jedoch als differenzierende Zusatzaufgabe herausgegeben werden.

Der Schwerpunkt des Workshops für die Oberstufe liegt auf der Formulierung von Optimierungsproblemen und der Entwicklung von Verfahren zum Lösen eben dieser Probleme. In diesem Workshop wird am Beispiel der Optimierung der Spiegelpositionen eindrücklich deutlich, dass die Modellierung von Optimierungsproblemen mathematisch enorm facettenreich ist und die Möglichkeit bietet, beim Lösen realer Probleme unterschiedliche Strategien zu verfolgen und gegeneinander abzuwägen.

### Arbeitsblatt 2: Energie über einen Tag

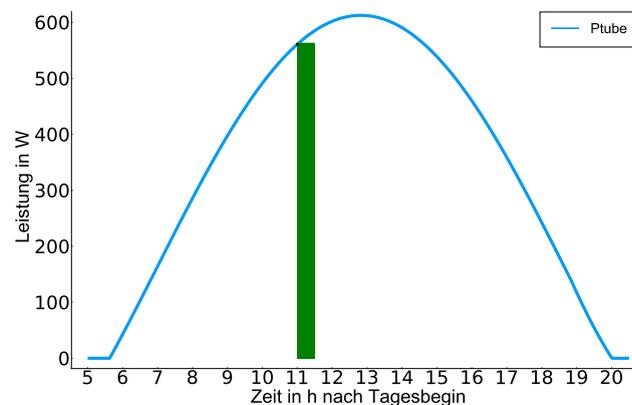
Ziel dieses Arbeitsblattes ist es, mithilfe der (momentanen) Leistung zu berechnen, wie viel Energie ein Kraftwerk über einen Tag umsetzt. Die Lernenden erhalten den Hinweis, zur Berechnung der Energie eine for-Schleife zu nutzen. Falls die Verwendung von for-Schleifen neu für die Lernenden ist, können sie an dieser Stelle auf eine Tippkarte zugreifen. Auf dieser wird deren Verwendung an einem Beispiel verdeutlicht. Das Grundgerüst für die for-Schleife ist im Code auf dem Arbeitsblatt bereits vorhanden und muss von den Lernenden lediglich im Sinne eines Lückentexts vervollständigt werden.

Für die Bearbeitung der Aufgabe ist Vorwissen zur Integralrechnung keine notwendige Voraussetzung. Das Prinzip von Unter-, Ober- oder Zwischensummen kann

von den Lernenden (gegebenenfalls mit Unterstützung) erarbeitet werden. Ist die Integralrechnung bereits bekannt, so kann die näherungsweise Berechnung von Integralen über Produktsummen aufgegriffen werden. Bei Schwierigkeiten können die Lernenden auf eine zweistufige *Tippkarte* zugreifen. In der *ersten Stufe* erhalten sie die Formel  $\Delta E = P \cdot \Delta t$ . Zudem wird am Beispiel einer Strahlungsquelle mit konstanter Leistung der Zusammenhang zwischen Leistung und Energie verdeutlicht. In der *zweiten Stufe* wird an einem Beispielzeitraum von  $t_1 = 11.0$  h bis  $t_2 = 11.5$  h erläutert, dass die umgesetzte Energie in dieser halben Stunde näherungsweise durch Einsetzen in die Formel  $\Delta E = P \cdot \Delta t$  und somit gemäß

$$\Delta E_{(11.0 \text{ bis } 11.5)} = P_{\text{tube}}(t^*, b, h, s, x) \cdot (11.5 - 11.0)$$

berechnet werden kann. Hierbei beschreibt  $t^*$  einen beliebigen Zeitpunkt zwischen  $t_1 = 11.0$  h und  $t_2 = 11.5$  h. Für  $t^* = 11.0$  h wird die approximative Berechnung der umgesetzten Energie in Abbildung 12.2 veranschaulicht. Die Lernenden sollen, ausgehend von diesem Beispiel, den Energieumsatz eines Tages berechnen. Wenn gleich im bisherigen Lernmaterial nicht umgesetzt, so wäre es perspektivisch denkbar, anstelle eines halbstündigen Intervalls bereits zwei Viertelstundenintervalle zu betrachten. Zum einen wäre die Berechnung damit genauer, zum anderen wird so nicht nur der Produktaspekt sondern auch der Summenaspekt der näherungsweise Berechnung des Integrals über Produktsummen angedeutet.



**Abb. 12.2:** Näherungsweise Berechnung der umgesetzten Energie im Zeitraum von  $t_1 = 11.0$  h bis  $t_2 = 11.5$  h

### Antwortblätter zu Arbeitsblättern 3 und 4

Zu den Arbeitsblättern 3 und 4 liegen Antwortblätter bereit. Auf diesen dokumentieren die Lernenden ihre Ergebnisse sowie Modellverbesserungsideen und halten Erkenntnisse zur Optimierung fest.

### Arbeitsblatt 3: Optimierung der Spiegelpositionen

Ziel dieses Arbeitsblattes ist, die Optimierung der Position eines beziehungsweise zweier Spiegel vorzunehmen.

Zunächst bestimmen die Lernenden die optimale Position **eines** Spiegels durch Ablesen am Funktionsgraphen und interpretieren das Ergebnis im Hinblick auf die reale

Situation. Die optimale Position des Spiegels ist direkt unter dem Rohr. Dies lässt sich dadurch begründen, dass der Kosinuseffekt in diesem Fall am geringsten ist.

Die Lernenden erweitern ihr Modell zur Beschreibung des Energieumsatzes auf zwei Spiegel und notieren getroffene Modellvereinfachungen. Die einfachste und in diesem Fall gefragte Erweiterung ist die Addition der durch je einen Spiegel umgesetzten Energien. Verschattungseffekte werden vernachlässigt.

Die Lernenden erhalten den interaktiven Graphen der Zielfunktion für das auf zwei Spiegel erweiterte Modell (vgl. Abb. 11.9), lesen die optimalen Positionen am Graphen ab und validieren das Ergebnis. Im mathematischen Sinne ist die Positionierung beider Spiegel direkt unter dem Rohr optimal. Da dies im Hinblick auf die Realität nicht sinnvoll ist, formulieren die Lernenden anschließend eine Nebenbedingung, die einen Mindestabstand zwischen den Spiegeln fordert. Diese durch eine Ungleichung beschriebene Nebenbedingungen berücksichtigen sie bei der Optimierung der beiden Spiegelpositionen. Die optimalen Positionen werden durch systematische Variation der Spiegelpositionen und wiederholtes Berechnen der Zielfunktionswerte bestimmt. Motiviert durch die Tatsache, dass die Suche nach den optimalen Spiegelpositionen bei mehr als zwei Spiegeln durch systematisches Variieren nicht möglich ist, sollen die Lernenden erste Ideen zur Automatisierung der Optimierung formulieren. Explizite Schritte, die dann dem Computer überlassen werden,<sup>3</sup> sollen notiert werden. Die Ansätze der Lernenden werden in einer anschließenden Plenumsdiskussion gesammelt.

### Folien für die Zwischendiskussion

Die Ergebnisse der Optimierung der Position eines sowie zweier Spiegel werden im Plenum diskutiert. Dazu liegen Präsentationsfolien bereit (vgl. Anh. C.14.1). Es wird festgehalten, dass die Lernenden ein Optimierungsproblem mit Nebenbedingungen aufgestellt und zunächst händisch gelöst haben. Zentral in dieser Plenumsphase ist die Diskussion der Optimierungsstrategien und Ansätze, die die Lernenden gesammelt haben.

### Arbeitsblatt 4: Übersicht über Arbeitsblätter zu Optimierungsansätzen

Dieses Arbeitsblatt dient lediglich als Übersicht über mögliche Ansätze zur Optimierung, die von den Lernenden bearbeitet werden können. Vier Optimierungsarbeitsblätter sind verlinkt. Auf diesen können die Lernenden

- die Brute-Force-Methode und
- den Greedy-Algorithmus zum Lösen des diskreten Problems einsetzen sowie
- die Optimierung basierend auf einer musterbasierten Formulierung und
- basierend auf der freien Variablen Formulierung durchführen.

---

<sup>3</sup>In dem hier beschriebenen stärker angeleiteten Lernmaterial sind diverse Optimierungsverfahren bereits implementiert und können von den Lernenden auf im Folgenden beschriebenen Arbeitsblättern direkt angewandt werden. Im Lernmaterial zur offenen Modellierung (vgl. Abschn. 12.2.4) schreiben die Lernenden hingegen selbst Programmcode.

In allen Fällen sind die Optimierungsverfahren bereits implementiert und können direkt eingesetzt oder um Nebenbedingungen erweitert werden. Ziel der vier Arbeitsblätter ist die Diskussion der Ergebnisse im Hinblick auf Güte der gefundenen Lösung für die Spiegelpositionen, Rechenzeit sowie Einschätzungen zur Eindeutigkeit der Lösung und Existenz einer solchen. Dazu liegen **Antwortblätter** bereit, auf denen die Lernenden ihre Ergebnisse strukturiert dokumentieren sollen.

### **Arbeitsblatt zur Brute-Force-Methode**

Die Lernenden wenden die Brute-Force-Methode zur Optimierung der Positionen zweier Spiegel an. Sie variieren die Feinheit der Diskretisierung, dokumentieren ihre Ergebnisse und diskutieren Vor- und Nachteile des Verfahrens sowie Ideen zur Verbesserung. Wird eine feine Diskretisierung des Kraftwerksareals gewählt, so steigt die Rechenzeit. Gleichzeitig wird die gefundene Lösung besser. Zu bedenken ist, dass das Optimierungsproblem in der bisherigen Formulierung keine eindeutige optimale Positionierung der Spiegel liefert. Stattdessen erhalten die Lernenden zwei Kombinationen der Spiegelpositionen mit gleichem Wert für die umgesetzte Energie, die jedoch zum identischen Layout des Spiegelfeldes führen.

Um mehrere Lösungen für die Spiegelpositionen mit identischem Spiegelfeldlayout zu vermeiden, erhalten die Lernenden den Arbeitsauftrag, eine Nebenbedingung für die Positionen  $x_1$  und  $x_2$  zu formulieren, die nicht nur den Mindestabstand fordert, sondern zusätzlich die genannten gleichwertigen Positionierungen verhindert. Bei Schwierigkeiten können die Lernenden auf eine *Tippkarte* zugreifen. Auf dieser erhalten sie den Hinweis einen der Operatoren  $>$  oder  $<$  zu verwenden, um eine Ungleichung aufzustellen, die eine aufsteigende oder absteigende Sortierung der Spiegelpositionen  $x_1$  und  $x_2$  liefert. Nach Eingabe der Nebenbedingung im Code führen die Lernenden die Optimierung erneut aus und bewerten, welchen Einfluss die Hinzunahme der Nebenbedingung auf Rechenzeit und Eindeutigkeit der gefundenen Spiegelpositionen hat.

Da die Anzahl der zu testenden Kombinationen mit wachsender Spiegelanzahl enorm ansteigt und damit auch die Rechenzeit stark zunimmt, ist der Algorithmus auf diesem Arbeitsblatt nur für zwei Spiegel implementiert.

### **Arbeitsblatt zum Greedy-Algorithmus**

Zunächst wird der Greedy-Algorithmus beschrieben. Anschließend können die Lernenden diesen zur Optimierung von beliebig vielen Spiegelpositionen anwenden. Dabei wird der Mindestabstand berücksichtigt. Die Lernenden erhalten den Arbeitsauftrag, die Diskretisierung des Kraftwerksareals und die Anzahl der zu positionierenden Spiegel zu variieren und den Einfluss auf die Ergebnisse zu untersuchen. Auf dem Antwortblatt sollen die Lernenden Vor- und Nachteile des Algorithmus notieren, den Einfluss der Diskretisierung auf Rechenzeit und die Güte der gefundenen Lösung für die Spiegelpositionen beschreiben und Ideen für die Verbesserung des Verfahrens notieren.

Vorteil dieses Verfahrens ist, dass es einfach umzusetzen ist und leicht auf viele Spiegel erweitert werden kann. Das Verfahren liefert jedoch eine Positionierung, die insbesondere bei einer geraden Anzahl an Spiegeln nicht optimal ist. Bei zwei Spie-

geln platziert der Greedy-Algorithmus den ersten Spiegel direkt unter dem Rohr. Die Lernenden haben auf Arbeitsblatt 3 jedoch festgestellt, dass eine Positionierung symmetrisch um das Rohr zu einem höheren Energieumsatz führt.

### **Arbeitsblatt zum musterbasierten Ansatz**

Auf diesem Arbeitsblatt werden nicht unmittelbar die Positionen  $x_1, x_2, x_3, \dots$  optimiert, stattdessen wird lediglich der Abstand zwischen den Spiegeln als Variable betrachtet und optimiert. Dabei wird gefordert, dass der Abstand zwischen zwei benachbarten Spiegeln  $\Delta x$  immer gleich groß ist.

Zunächst stellen die Lernenden eine Formel auf, mit der die Positionen  $x_2, x_3$  bis  $x_N$  in Abhängigkeit von einer fest gewählten Position für den linksstehenden Spiegel und vom Abstand  $\Delta x$  berechnet werden können. Ziel ist es, die Zielfunktion von Arbeitsblatt 3 so umzuschreiben, dass diese von lediglich zwei Variablen (der Position des Spiegels ganz links und dem Spiegelabstand) abhängt.

Anschließend nutzen die Lernenden ein Optimierungstool, um den Spiegelabstand zu bestimmen, der zum maximalen Energieumsatz führt. Die Position des Spiegels ganz links auf dem Spiegelfeld, wird zunächst nicht als Variable berücksichtigt, sondern fest gewählt. Die Lernenden variieren die Position dieses fest gewählten Spiegels und führen die Optimierung erneut aus. Sie notieren Beobachtungen sowie Verbesserungsideen. Es wird ersichtlich, dass das Ergebnis für den optimalen Spiegelabstand stark von der Position des linksstehenden Spiegels abhängt. Deshalb berücksichtigen die Lernenden diese ebenfalls als Variable. Sie erhalten damit ein zweidimensionales Optimierungsproblem. Die Lernenden vergleichen die Ergebnisse des ein- und zweidimensionalen Problems und bewerten diese im Hinblick auf die reale Situation. Sie notieren Vor- oder Nachteile der musterbasierten Optimierung hinsichtlich Rechenzeit, Güte der gefundenen Lösung, Existenz und Eindeutigkeit einer optimalen Lösung und sammeln mögliche Ideen zur Verbesserung.

### **Arbeitsblatt zur freien Variablen Optimierung**

Die Lernenden führen die Optimierung zunächst für **zwei** Spiegel durch. Dabei wird die Nebenbedingung für den Mindestabstand berücksichtigt. Bei dem verwendeten Optimierungstool können Startwerte für die einzelnen Variablen festgelegt werden. Die Lernenden variieren den Startwert für die Position des ersten Spiegels  $x_1$  und diskutieren dessen Einfluss auf die gefundene Lösung für die Spiegelpositionen. Sie dokumentieren ihre Beobachtungen und diskutieren Vor- und Nachteile des Verfahrens sowie Ideen zur Verbesserung. Für die Position des zweiten Spiegels  $x_2$  brauchen die Lernenden keinen Startwert festzulegen. Dieser wird im Hintergrund zufällig gesetzt.

Es ist zu beobachten, dass das Optimierungsproblem in der bisherigen Formulierung keine eindeutige optimale Positionierung der Spiegel liefert. Stattdessen existieren zwei Kombinationen für  $x_1$  und  $x_2$ , mit gleichem Zielfunktionswert. Diese Kombinationen sind bis auf Vertauschung der Spiegel identisch. Da die Spiegel nicht unterscheidbar sind, implizieren sie das gleiche Spiegelfeldlayout. Um die genannten mehrfachen (im Hinblick auf das Spiegelfeldlayout identischen) Positionierungen zu verhindern, formulieren die Lernenden (analog zum Arbeitsblatt zur Brute-Force-

Methode) eine Nebenbedingung für die Positionen  $x_1$  und  $x_2$ , die nicht nur den Mindestabstand berücksichtigt, sondern zusätzlich die genannten gleichwertigen Lösungen verhindert. Bei Schwierigkeiten liegt eine *Tippkarte* bereit. Anschließend führen die Lernenden die Optimierung mit der neuen Nebenbedingung erneut aus. Sie bewerten, welchen Einfluss die Hinzunahme der Nebenbedingung auf Rechenzeit sowie Eindeutigkeit der gefundenen Positionierung hat. Dazu variieren sie erneut den Startwert. Die Optimierung liefert unabhängig vom Startwert nur noch eine Lösung für die Spiegelpositionen.

### **Infoblatt: Globales oder lokales Optimum?**

Die Lernenden wenden das Optimierungstool, welches auf dem Arbeitsblatt zur freien Variablen Optimierung eingesetzt wurde, auf verschiedene eindimensionale Funktionen mit zahlreichen lokalen wie auch globalen Extremwerten an. Durch Variation des Startwertes wird deutlich, dass nicht immer ein globales Optimum gefunden wird.

### **Zwischendiskussion**

Für die Zwischendiskussion liegen Präsentationsfolien bereit. Die Ergebnisse der Optimierung mit den verschiedenen Algorithmen / Formulierungen des Problems werden diskutiert. Die Lernenden stellen die Funktionsweise der Verfahren beziehungsweise die Grundideen der Optimierungsansätze vor. Anschließend werden Vor- und Nachteile diskutiert. Es wird auf Existenz und Eindeutigkeit einer optimalen Lösung für die Spiegelpositionen sowie Güte und Rechenzeit eingegangen. Die Lernenden benennen Ideen zur Verbesserung der jeweiligen Ansätze.

An dieser Stelle kann der Workshop entweder stärker angeleitet fortgeführt (s. AB 5 zur Modellierung von Verschattungseffekten) oder geöffnet werden (vgl. Abschn. 12.2.4).

### **Arbeitsblatt 5: Verschattungseffekte**

Ziel dieses Arbeitsblattes ist a) die Erweiterung des Modells für die umgesetzte Energie um Verschattungs- und Blockierungseffekte (vgl. Abschn. 10.3) und b) die anschließende Untersuchung des Einflusses dieser Effekte auf die optimalen Spiegelpositionen.

Auf dem Arbeitsblatt wird lediglich die *Verschattung von einfallenden Strahlen durch nebenstehende Spiegel* von den Lernenden modelliert. Die Modellierung wird dabei nicht mit rein geometrischen Überlegungen angegangen. Stattdessen kommen Ansätze der Vektoralgebra zum Einsatz. Dies spiegelt zum einen das tatsächliche Vorgehen in der Forschung wieder, zum anderen erlaubt dieser Ansatz eine unproblematische Erweiterung des Modells auf den dreidimensionalen Fall. Falls die Lernenden die Parameterform von Geraden nicht aus der Schule kennen oder eine Erinnerung benötigen, so liegt ein *Infoblatt* bereit, auf dem diese anhand von Beispielen und mit diversen Visualisierungen erläutert wird.

Zur Modellierung der Verschattungseffekte beschreiben die Lernenden einfallende Sonnenstrahlen und den Spiegel durch (Halb-)Geraden beziehungsweise Strecken

in Parameterform und berechnen, angeleitet durch kleinere Aufgabenstellungen, Schnittpunkte zwischen eben diesen. Dabei kommt Wissen zum Sinus und Kosinus in rechtwinkligen Dreiecken, zum Satz des Pythagoras sowie zu relativen Anteilen zum Tragen. Auch hier liegen an verschiedenen Stellen *gestufte Tippkarten* mit unterstützenden Visualisierungen bereit.

Nachdem die Lernenden an einem Beispiel modelliert haben, wie einfallende Strahlen durch benachbarte Spiegel beeinflusst werden, wird das Modell im Hintergrund, das heißt für die Schülerinnen und Schüler nicht sichtbar, um die Blockierung reflektierter Strahlen und die Verschattung durch den Sekundärreflektor ergänzt. Die Lernenden nutzen dieses ergänzte Modell für die erneute Optimierung der Spiegelpositionen. Die umgesetzte Energie wird dazu basierend auf der weiterentwickelten Funktion (mit Verschattungs- und Blockierungseffekten) und der ursprünglichen Funktion (ohne Verschattungseffekte) in Abhängigkeit von

- der Position  $x$  eines Spiegels (vgl. Abb. 11.8) und
- der Positionen  $x_1$  und  $x_2$  von zwei Spiegeln (vgl. Abb. 11.10)

grafisch dargestellt. Die Lernenden lesen am Graphen ab, inwieweit sich die optimalen Spiegelpositionen gegenüber dem Modell ohne Verschattungseffekte geändert haben, und erklären den Verlauf der beiden Graphen im Hinblick auf die reale Situation.

### **Arbeitsblatt 6: Optimierung der Spiegelpositionen mit erweitertem Modell**

Auf diesem Arbeitsblatt können die verschiedenen Optimierungsansätze von Arbeitsblatt 4 auf das erweiterte Modell (mit Verschattungseffekten) angewandt werden. Ziel des Arbeitsblattes ist es, dass die Lernenden erkunden, wie sich die Erweiterung ihres Modells zum einen auf die optimalen Positionen, aber auch auf die Rechenzeit der Algorithmen auswirkt. Da bei dem erweiterten Modell zahlreiche Schnittpunktberechnungen durchgeführt werden, dauert die Optimierung bei allen Algorithmen länger als zuvor.

### **Abschlussdiskussion**

Für die Abschlussdiskussion liegen Präsentationsfolien bereit. Es wird zunächst aufgegriffen, wieso zur Beschreibung von Sonnenstrahlen und Spiegeln nicht Graphen linearer Funktionen gewählt wurden. Zudem wird diskutiert, inwieweit sich die Ergebnisse für die optimalen Spiegelpositionen bei dem neuen Modell geändert haben. Die Lernenden begründen, für welchen Optimierungsansatz sie sich entscheiden würden. Abschließend werden Modellverbesserungsideen der Lernenden aufgegriffen und ein Ausblick auf weiterführende Fragen im Bereich der Solarenergieforschung gegeben.

## **12.2.4 Lernmaterial zum unangeleiteten Modellieren**

Um offenes, unangeleitetes Modellieren auch in kürzeren Unterrichtsformaten (bspw. ein- bis zweitägigen Modellierungstagen) zu ermöglichen, wurden Arbeitsblätter (Ju-

pyter Notebooks) vorbereitet, auf denen bereits von den Lernenden entwickelte Funktionen zur direkten Verwendung zur Verfügung stehen. Konkret wurden Arbeitsblätter zu drei Problemstellungen entwickelt. Die Jupyter Notebooks liegen wie die bisherigen Arbeitsblätter aus den stärker angeleiteten Teilen der Workshops auf der Workshop-Plattform (JupyterHub, vgl. Abschn. 5.3) bereit. Bei der Bearbeitung der offenen Problemstellungen werden die Lernenden ermutigt, im Team zu arbeiten und Ideen sowie Strategien gemeinsam auf einem Whiteboard festzuhalten.

### **Übersichtsblatt zu offenen Problemstellungen**

Die drei offenen Problemstellungen werden beschrieben und die zu den Problemen bereitliegenden Jupyter Notebooks verlinkt. Zudem werden verschiedene Beispiele gezeigt, wie in der Programmiersprache Julia Funktionen und Datenpunkte visualisiert werden können. Falls einzelne Lernende vorziehen, mit vorstrukturierten Aufgaben weiterzuarbeiten, so finden sie einen Link zu Arbeitsblatt 5 des Mittelstufenworkshops, auf dem sie die Optimierung der Rohrhöhe durchführen können.

### **Arbeitsblatt zu Problem 1: Optimierungsverfahren implementieren**

Die Lernenden implementieren ein Optimierungsverfahren zur Positionsoptimierung. Dabei können sie sowohl ganz eigene Ansätze als auch einen der Ansätze verfolgen, die sie auf bisherigen Arbeitsblättern zur Optimierung kennengelernt haben. Bereits von den Lernenden entwickelte Funktionen, wie beispielsweise für die Einstellung des Neigungswinkels  $\gamma$  oder für den Energieumsatz über einen Tag  $E_{\text{tube}}$ , werden in einer Codezelle geladen und können von den Lernenden direkt aufgerufen werden. Zudem ist das Grundgerüst einer Funktion gegeben, die die Anzahl der zu positionierenden Spiegel sowie die Grenzen des Kraftwerksareals als Eingabe und den Positionsvektor als Array mit zunächst leeren Einträgen als Ausgabe liefert. Die Schritte ihres Optimierungsverfahrens implementieren die Lernenden selbstständig innerhalb des gegebenen Grundgerüsts. Auf dem Arbeitsblatt sind *Tippkarten* zur Syntax und Verwendung von if-Bedingungen und for-Schleifen verlinkt. Zudem wird beschrieben, wie Einträge in einem Array gespeichert werden können.

### **Arbeitsblatt zu Problem 2: Verschattung modellieren**

Die Lernenden modellieren die Verschattung durch nebeneinanderstehende Spiegel eigenständig. Um den Einstieg in die offene Modellierung zu erleichtern, erhalten die Lernenden die Hinweise,

- mit der Modellierung des Schattenwurfs zu einer beliebigen, aber festen Tageszeit und für zwei Spiegel auf festgelegten Positionen zu beginnen und das Modell erst anschließend zu verallgemeinern.
- die Entwicklung des Modells zunächst mit Stift und Papier (bzw. auf dem Whiteboard) vorzunehmen und erst anschließend zu implementieren.

Haben die Lernenden ausreichend Zeit, um ihr Modell zu implementieren, so können sie eine Codezelle nutzen, in der sämtliche Funktionen zur Modellierung der umgesetzten Energie ohne Verschattungseffekte implementiert sind. Diese können

die Lernenden beliebig erweitern oder abändern, um Verschattungseffekte zu integrieren. *Tippkarten* zu for-Schleifen und if-Abfragen sind verlinkt.

### **Arbeitsblatt zu Problem 3: Beliebige Kraftwerksparameter optimieren**

Die Lernenden wählen selbst, welche Kraftwerksparameter sie optimieren. Sie erhalten die Tipps,

- zunächst nur *einen* Parameter auszuwählen, dessen optimalen Wert sie bestimmen (z. B. die Breite des Sekundärreflektors) und sich anschließend zu überlegen, wie sie die Optimierung auf mehrere Parameter ausweiten können (bspw. um gleichzeitig Spiegelbreite und Reflektorbreite zu optimieren).
- dass es helfen kann, die Zielfunktion der Optimierung grafisch darzustellen.

*Tippkarten* zu for-Schleifen und if-Abfragen sind verlinkt.

### **Abschlussdiskussion**

Zur Diskussion der Ergebnisse der offenen Modellierungsprobleme liegen keine Präsentationsfolien bereit. Die Diskussion wird entlang der Ergebnisse der Lernenden geführt. Die Lernenden stellen dazu ihre auf den Whiteboards entwickelten Modelle, ihren Code oder ihre auf Präsentationsfolien zusammengefassten Ergebnisse vor.

### **Abgrenzung zu bestehendem Material**

Bereits 2011 wurde das Problem der Modellierung und Optimierung eines Fresnelkraftwerks erstmals im Rahmen einer Modellierungswoche ausgerichtet vom Schülerlabor CAMMP der RWTH Aachen behandelt. Damals beschäftigte sich eine kleine Gruppe von Schülerinnen und Schülern der Oberstufe unangeleitet mit der offenen Problemstellung. Der Lösungsrahmen war in keiner Weise vorgegeben (vgl. [Roeckerath, 2012](#)). Im Jahr 2012 wurde die Problemstellung durch die Entwicklung von computergestützten Arbeitsmaterialien für den Einsatz in größeren Lerngruppen aufbereitet. Dadurch wurde auch die Durchführung in einem zeitlich begrenzten Rahmen (ein Projekttag) ermöglicht (vgl. [Krahforst, 2016](#); [Roeckerath, 2012](#)). Die entwickelten Arbeitsmaterialien beschränkten sich auf die Modellierung der Spiegelausrichtung und der am Rohr ankommenden Leistung. Die Optimierung und die Modellierung der umgesetzten Energie wurde nicht umgesetzt.

In der Literatur finden sich durchaus verschiedene Lernmaterialien zu den in dieser Arbeit aufgegriffenen Algorithmen wieder. Hier sei beispielsweise auf die Diskussion der Brute-Force-Methode beziehungsweise die Anwendung eines Greedy-Algorithmus (der Dijkstra-Algorithmus) zum Finden einer kürzesten Wegstrecke hingewiesen (vgl. [Mühlehner, 2021](#), S. 54; [Lehrerinnenfortbildung Baden-Württemberg, 2016](#); [Lutz-Westphal, 2006](#), S. 62). In der vorliegenden Arbeit ist der Blick auf die Optimierung jedoch ganzheitlicher. Es wurde nicht nur eine Klasse von Optimierungsproblemen (bspw. kombinatorische Probleme) betrachtet, sondern die gesamte

Bandbreite ausgenutzt und dabei unter anderem kontinuierliche und diskrete Ansätze und deren Vor- und Nachteile gegenübergestellt.

Auch die Monte-Carlo-Methode wurde bereits in der didaktischen Literatur aufgegriffen und dabei im Hinblick auf die stochastische Näherung des Kreisflächeninhalts diskutiert (vgl. [Kuntze, 2018](#), S. 170; [Greefrath et al., 2016](#), S. 234 f.).

## 12.3 Flexibler Einsatz des Lernmaterials

Das Lernmaterial, wenngleich es zugeordnet zu den beiden Workshops I und II vorgestellt wurde, kann flexibel kombiniert werden. Verschiedene Möglichkeiten, für derartige Kombinationen, werden hier kurz aufgeführt. Auch wird auf die Möglichkeit einer frühzeitigen Öffnung des Modellierungsprozesses eingegangen.

- **Zeitlich ausgedehntes Projekt für die Oberstufe:** Lernende der Oberstufe bearbeiten zunächst die Arbeitsblätter 1 – 4 des Mittelstufenworkshops und entwickeln das Modell für die Leistung am Rohr eigenständig. Erst anschließend wird das Modell auf die umgesetzte Energie erweitert und die Optimierung der Spiegelpositionen durchgeführt.
- **Zeitlich ausgedehntes Projekt für die Mittelstufe:** Lediglich Arbeitsblatt 5 des Oberstufenworkshops greift mit der Parameterform von Geraden Inhalte auf, die in vielen Bundesländern erst in den Lehrplänen der Sekundarstufe II vorkommen. Die notwendigen Vorkenntnisse für die Bearbeitung der Arbeitsblätter 2 – 4 des Oberstufenworkshops sind zum Ende der Mittelstufe gegeben. Damit könnten Lernende der Mittelstufe die genannten Arbeitsblätter zur Optimierung der Spiegelpositionen durchführen.
- **Frühzeitiger Übergang zur offenen Modellierung:** Im Workshop für die Oberstufe kann bereits nach Arbeitsblatt 3 zur offenen Modellierung übergegangen werden.
- **Zeitlich verkürzte Projekte:** Der Mittelstufenworkshop kann bereits nach der Bearbeitung des ersten Arbeitsblattes zur Optimierung der Rohrhöhe beendet werden (AB 5). Der Oberstufenworkshop kann bereits nach der Optimierung auf Arbeitsblatt 3 oder der Bearbeitung von einem (bis vier) Algorithmen-Arbeitsblättern beendet werden. Der Modellierungskreislauf wird in beiden Fällen dennoch vollständig durchlaufen.

## 12.4 Lehrplananbindung

In diesem Abschnitt wird die Anbindung des entwickelten Lernmaterials an die Lehrpläne des Fachs Mathematik der exemplarisch gewählten Bundesländer<sup>4</sup> Nord-

---

<sup>4</sup>Die genannten Bundesländer wurden gewählt, da die Mehrheit der durchgeführten Modellierungsveranstaltungen mit Lernenden dieser Ländern stattgefunden hat.

rhein-Westfalen (NRW), Baden-Württemberg (BW) und Rheinland-Pfalz (RLP) sowie übergeordnet an die Bildungsstandards der Kultusministerkonferenz (KMK) beschrieben. Es wird ersichtlich, dass das Lernmaterial zahlreiche in den Lehrplänen verankerte Inhalte aufgreift. Das entwickelte Material (oder Teile dessen) eignen sich sowohl zur Vertiefung und Anwendung bereits gelernter Inhalte, können aber auch im Rahmen eines problemorientierten Einstiegs in neue Themen eingesetzt werden. Die innermathematische Reichhaltigkeit des Lernmaterials ermöglicht es den Lernenden die Bedeutung der Vernetzung mathematischer Inhalte für das Lösen realer Probleme aktiv zu erfahren. Wissen aus unterschiedlichen Themenbereichen des Mathematikunterrichts findet eine authentische Anwendung.

Beide Workshops sind zudem außermathematisch reichhaltig und bieten damit die Möglichkeit, fächerübergreifende oder fächerverbindende Projekte durchzuführen. Deswegen werden zudem Lehrplanbezüge zu weiteren Schulfächern, insbesondere Informatik und Physik, aufgezeigt.

### Mathematiklehrpläne

Bei der Modellierung der Leistung des Kraftwerks finden zahlreiche mathematische Begriffe und Konzepte Anwendung, die in den betrachteten Lehrplänen (BW, RLP, NRW) im Inhaltsfeld **Geometrie** für die Sekundarstufe I verankert sind. Dazu zählen insbesondere Beziehungen im rechtwinkligen Dreieck (Satz des Pythagoras, Stufen-, Wechsel-, Scheitelwinkel, Winkelsumme im Dreieck). Zentral ist zudem die Anwendung von trigonometrischen Beziehungen (Sinus, Kosinus und Tangens). Diese sind in BW Teil des Lehrplans für die Klassen 9/10. Die Lernenden sollen unter anderem „Streckenlängen und Winkelweiten unter Nutzung der Längenverhältnisse Sinus, Kosinus, Tangens bestimmen“ ([Bildungsplan BW Mathe Sek. I und II, 2016](#), S. 32). Die genannten Inhalte sind auch in RLP für die Klassen 9/10 (vgl. [Lehrplan RLP Mathe Sek. I, 2007](#), S. 91) und in NRW bis zum Ende der Sekundarstufe I verankert (vgl. [Lehrplan NRW Mathe Sek. I, 2019](#), S. 34).

Die Modellierung des Schattenwurfs über **Geraden in Parameterform** und die Bestimmung von Schnittpunkten zwischen Geraden sind in den untersuchten Lehrplänen für die Sekundarstufe II verankert (vgl. [Bildungsplan BW Mathe Sek. I und II, 2016](#); [Lehrplan NRW Mathe Sek. II, 2014](#); [Lehrplan RLP Mathe Sek. II, 2015](#)). Zugleich lassen sich bei der Modellierung des Schattenwurfs Verknüpfungen zur Analysis herstellen, indem der verfolgte Ansatz der Beschreibung durch Graphen linearer Funktionen gegenübergestellt wird. Gemeinsam mit den Lernenden kann diskutiert werden, dass sich die Beschreibung über lineare Funktionen per Definition nicht für senkrechte Sonnenstrahlen eignet.

Der Umgang mit Funktionen und der Zusammenhang von Funktionsgleichungen und Funktionsgraphen ist sowohl bei der Optimierung der Absorberrohrhöhe (Workshop I) als auch bei der Optimierung der Spiegelpositionen (Workshop II) von zentraler Bedeutung. Insbesondere im Workshop I kann der Wechsel des Parameters, der als Variable betrachtet wird,<sup>5</sup> dazu beitragen das Verständnis **funktionaler Zusammenhänge** zu festigen.

---

<sup>5</sup>Zuerst wird die Leistung am Rohr in Abhängigkeit der Zeit, später in Abhängigkeit der Rohrhöhe betrachtet.

Die Modellierung der Energie ausgehend von der momentanen Leistung des Kraftwerks bietet die Möglichkeit, den Unterricht zur **Integralrechnung** zu bereichern. (vgl. [Bildungsplan BW Mathe Sek. I und II, 2016](#); [Lehrplan NRW Mathe Sek. II, 2014](#); [Lehrplan RLP Mathe Sek. II, 2015](#)). Der Fokus wird dabei auf den Produktsumentenaspekt des Integralbegriffs gelegt: „Interpretiert man das Integral als Grenzwert von Ober-, Unter- bzw. Zwischensummen oder berechnet man konkret Produktsummen aus Größen, so wird der Produktsumentenaspekt des Integralbegriffs in den Vordergrund gestellt“ ([Greefrath et al., 2016](#), S. 239).

Im Lernmaterial zur Optimierung der Rohrhöhe liegt der Fokus auf dem Ablesen global optimaler Lösungen am Funktionsgraphen und der Diskussion, inwieweit Verlauf und Eigenschaften des Funktionsgraphen Aussagen über die Existenz einer Lösung erlauben. Die Untersuchung von Funktionsgraphen auf Extremstellen findet in der Sekundarstufe I bereits am Beispiel von quadratischen Funktionen statt. So fordert der Lehrplan RLPs für die Klassen 9/10, dass die Lernenden „kennzeichnende Eigenschaften von Graphen quadratischer Funktionen (Parabeln) kennen und in Sachsituationen nutzen“ ([Lehrplan RLP Mathe Sek. I, 2007](#), S. 100). Zudem wird dort als Hinweis genannt, dass das grafische Lösen von Optimierungsaufgaben durch Ermitteln des Scheitelpunkts von Parabeln im Rahmen dieses Themas Anwendung finden kann. Auch im Lehrplan NRWs für die Sekundarstufe I wird das Lösen von Extremwertproblemen bei der Behandlung quadratischer Funktionen benannt (vgl. [Lehrplan NRW Mathe Sek. I, 2019](#), S. 33). Das Lernmaterial zur Optimierung der Rohrhöhe greift die Diskussion von Eigenschaften des (Ziel-)Funktionsgraphen eines Optimierungsproblems auf und ermöglicht das Lösen dieses (unabhängig von einer konkreten Funktionenklasse) auf anschauliche Weise. Das Ablesen von Extremstellen am Graphen spielt in Teilen auch bei der Optimierung der Spiegelpositionen eine Rolle – jedoch liegt der Fokus eher auf dem Vergleich verschiedener Formulierungen des Problems und auf dessen Lösung mithilfe numerischer Verfahren.

Im Lehrplan RLPs für die Sekundarstufe II werden Anregungen und Themenvorschläge für sowohl fächerübergreifende wie auch fächerverbindende Unterrichtseinheiten benannt. Ein Themenvorschlag ist die **Monte-Carlo-Methode**: „Bei der Planung und Durchführung von Simulationen mithilfe von Zufallszahlen (Monte-Carlo-Methode) erfahren die Schülerinnen und Schüler die Bedeutung dieser Arbeitsmethode, die in verschiedenen Studiengängen und Berufsfeldern eine zunehmend größere Rolle spielt“ ([Lehrplan RLP Mathe Sek. II, 2015](#), S. 53). Der Stochastikunterricht kann durch die „Erarbeitung von stochastischen Simulationen zum Lösen von Anwendungsproblemen“ bereichert werden ([Lehrplan RLP Mathe Sek. II, 2015](#), S. 71). Die Simulation der fehlerhaften Ausrichtung der Spiegel bei der Optimierung der Rohrhöhe ließe sich als relevantes Anwendungsproblem einsetzen, welches, wie weiter unten diskutiert wird, zudem die Chance bietet Inhalte des Informatik- und Physikunterrichts aufzugreifen.

Algorithmen sind unter der Leitidee *Algorithmus und Zahl* in den Bildungsstandards der KMK verankert. Diese Leitidee umfasst „die Kenntnis, das Verstehen und das Anwenden mathematischer Verfahren, die prinzipiell automatisierbar und damit einer Rechnernutzung zugänglich sind“ ([KMK Mathe Sek. II, 2012](#), S. 18). Insbesondere das Lernmaterial zur Optimierung der Spiegelpositionen eignet sich in

besonderem Maße, um **algorithmisches Denken** im Mathematikunterricht zu fördern. Dies wird nicht nur in den Bildungsstandards, sondern auch in den Lehrplänen gefordert.

### Physiklehrpläne

Bereits in der Präambel der KMK Bildungsstandards zum Fach Physik wird die Relevanz des Themas herausgestellt:

„Als eine der ältesten Wissenschaften ist die Physik seit jeher in ein Wechselspiel mit Technik und Gesellschaft eingebunden. Sowohl historische als auch aktuelle Entwicklungen verdeutlichen die Notwendigkeit der Betrachtung gesellschaftlich relevanter Problemstellungen wie der Energieversorgung oder des Klimawandels aus physikalischer und technischer Sicht“ (KMK Physik Sek. II, 2020, S. 11).

Unter anderem in NRW wird die nachhaltige Energieversorgung im Lehrplan der Sekundarstufe I expliziert. Dort ist die Energieversorgung als eigenes Inhaltsfeld ausgewiesen:

„Die gesicherte Versorgung der Welt mit Energie ist in den letzten Jahren zu einem zentralen Thema geworden. Die weltweit gestiegene Nachfrage vor allem nach elektrischer Energie zwingt dazu, gerade mit Blick auf den Klimawandel, nach regenerativen Energiequellen zu suchen“ (Lehrplan NRW Physik Sek. I, 2019, S. 19).

Damit ist das Thema *Erneuerbare Energie und Solarenergie* bereits übergeordnet hervorragend für ein fächerübergreifendes Lernen mit dem Fach Physik geeignet. Bezugnehmend zu expliziten Inhalten des Lernmaterials eignet sich beispielsweise die Einstellung des Neigungswinkels des Spiegels, um das Reflexionsgesetz zu wiederholen und damit Bezüge zum Themenfeld **Optik** herzustellen (vgl. Lehrplan RLP Physik Sek. I, o. D.; Lehrplan NRW Physik Sek. I, 2019; Bildungsplan BW Physik Sek. I und II, 2016).

Weitere Anknüpfungspunkte bieten die Themenfelder **Energie** (insbesondere Energieversorgung) und Wärmelehre. So fordert der Bildungsplan BWs, dass Lernende nach Klasse 7/8 „Beispiele für die Speicherung von Energie in verschiedenen Energieformen in Alltag und Technik nennen und beschreiben (unter anderem Lageenergie, Bewegungsenergie, thermische Energie)“, „Möglichkeiten der Energieversorgung mithilfe von Energieübertragungsketten beschreiben (zum Beispiel Wasserkraftwerk, Kohlekraftwerk)“ und „den Zusammenhang von Energie und Leistung beschreiben ( $P = \frac{\Delta E}{\Delta t}$ )“ können (Bildungsplan BW Physik Sek. I und II, 2016, S. 14).

Im Themenfeld **Wärmelehre** wird dort benannt, dass die Lernenden nach Klasse 9 „verschiedene Arten der Energieversorgung unter physikalischen, ökologischen, ökonomischen und gesellschaftlichen Aspekten vergleichen und bewerten (zum Beispiel fossile Brennstoffe, Kernenergie, Windenergie, Sonnenenergie)“ können (Bildungsplan BW Physik Sek. I und II, 2016, S. 22).

### Informatiklehrpläne

Das stärker angeleitete Lernmaterial, in dem die Lernenden primär ausgewiesene

Lücken im Code ausfüllen, bietet die Möglichkeit einen ersten Einblick in den Aufbau von Kontrollstrukturen (for-Schleifen und if-Abfragen) zu erhalten. So wird die numerische Integration bei der Erweiterung des Modells von der Leistung auf die über einen Tag umgesetzte Energie über eine for-Schleife realisiert, deren Grundgerüst bereits gegeben ist. Kontrollstrukturen sind in den Oberstufenlehrplänen Informatik in allen drei Bundesländern verankert (vgl. [Bildungsplan BW Info Sek. II, 2016](#); [Lehrplan NRW Info Sek. II, 2014](#); [Lehrplan RLP Info Sek. II, o. D.](#)) und finden sich zum Teil bereits in Lehrplänen der Unterstufe wieder (vgl. [Lehrplan NRW Info Sek. I, 2021](#), S. 18).

Die Entwicklung, Implementierung, Anwendung und Bewertung von **Algorithmen** spielten bei der Optimierung der Spiegelpositionen eine zentrale Rolle. Dieser Teil des Lernmaterials greift damit diverse Kompetenzen auf, die für den Informatikunterricht der Oberstufe ausgewiesen werden. Als Beispiel sei auf den Lehrplan für die Einführungsphase in Nordrhein-Westfalen verwiesen. Dort wird im Inhaltsfeld *Algorithmen* gefordert, dass die Schülerinnen und Schüler einfache Algorithmen und Programme analysieren, erläutern und modifizieren können. Zudem sollen sie eigene Algorithmen entwerfen und implementieren (vgl. [Lehrplan NRW Info Sek. II, 2014](#), S. 23). Eine ähnliche Beschreibung findet sich auch im Lehrplan BWs wieder (vgl. [Bildungsplan BW Info Sek. II, 2016](#), S. 17). Insbesondere das offene Lernmaterial bietet die Möglichkeit, Algorithmen nicht nur theoretisch zu entwickeln, sondern auch zu implementieren. Die Lernenden können hier unter anderem eigene Verfahren zur Optimierung diverser Kraftwerksparameter oder ein Modell für den Schattenwurf implementieren.

Besonders schön knüpft das Lernmaterial an die Forderung des Lehrplans RLPs zum Inhaltsbereich *Algorithmisches Problemlösen* an. Dort wird die Problemorientierung bei der Anwendung von Kontrollstrukturen und der Entwicklung von Algorithmen hervorgehoben (vgl. [Lehrplan RLP Info Sek. II, o. D.](#), S. 29). Dies erfüllt das entwickelte Lernmaterial in besonderem Maße.

### Anbindung an weitere Schulfächer

Der Lehrplan für das Basisfach **Naturwissenschaften und Technik**, welches bis dato im Rahmen eines Schulversuchs in BW umgesetzt wird, greift die Notwendigkeit einer nachhaltigen Energieversorgung auf und fordert, dass die Lernenden „die Herausforderungen, die sich beim Zusammenwirken von Bereitstellung, Speicherung und Transport elektrischer Energie in Versorgungsnetzen ergeben“ ([Bildungsplan BW NwT Sek. II, 2019](#), S. 17) erkennen.

Auch an den Lehrplan des Fachs **Informatik, Mathematik, Physik** (IMP), welches in BW in Klasse 7 bis 10 als Profulfach unterrichtet wird, lassen sich diverse Anknüpfungspunkte aufzeigen (vgl. [Bildungsplan BW IMP Sek. I, 2018](#), S. 11). Als ein Beispiel sei die Diskussion und Modellierung des Sonnenstands sowie der Stärke der Sonnenstrahlung genannt, die dem inhaltlichen Kompetenzbereich *Erde und Weltall* zugeordnet werden kann (vgl. [Hoeffler, 2020](#)). Zudem wird im Lehrplan IMPs im Teilbereich Informatik die *Variation von Problemstellungen* betont:

„In besonderer Weise bietet der informatische Zugang zur Problemlösung auch die Möglichkeit, über die Grenzen der ursprünglichen Aufgabenstellung hinaus zu denken. Durch veränderte Anforderungen, Rahmenbedingungen oder Variation der Problemgröße werden informatische Lösungskonzepte verdeutlicht, nachvollziehbar gemacht oder hinterfragt. Fragestellungen der Art „Wie hätte man es anders machen können?“, „Ist die gefundene Lösung optimal?“ oder „Wo enden die Einsatzbereiche der gefundenen Lösung?“ gehören zum Standardrepertoire des Informatikunterrichts“ ([Bildungsplan BW IMP Sek. I, 2018](#), S. 12).

Diese Fragen sind mitnichten nur im Informatikunterricht zentral, sondern spielen (oder sollten es) auch im Mathematikunterricht beim Lösen realer Problemstellungen eine Rolle. Das entwickelte Lernmaterial bietet damit durchaus die Chance, orientiert an realen Fragestellungen Projekte durchzuführen, die die Bereiche Informatik und Mathematik verbinden.

# 13 Erfahrungswerte und didaktische Reflexion

Um eine erste Bewertung des Lernmaterials vornehmen zu können, wurde dieses in diversen Modellierungsveranstaltungen mit Lernenden erprobt und basierend auf den Erfahrungen iterativ verbessert und erweitert. Ziel der Erprobungen war es, *erste Einschätzungen* über die Eignung des Materials im Bezug auf Interesse und Motivation der Lernenden, Lernzuwachs, Angemessenheit des Schwierigkeitsgrades und Einsatzmöglichkeiten in heterogenen Lerngruppen treffen zu können. Für Aussagen zum Kompetenzzuwachs und zum langfristigen Lernerfolg wären weiterführende Untersuchungen notwendig, die nicht Bestandteil dieser Arbeit sind.

## 13.1 Durchgeführte Modellierungsveranstaltungen

In Tabelle 13.1 sind die zum entwickelten Lehr- und Lernmaterial durchgeführten Modellierungsveranstaltungen aufgeführt. Insgesamt haben 147 Lernende, davon 96 männlich (m), 49 weiblich (w) und 2 divers (d), teilgenommen. Bis auf die beiden Unterrichtsreihen in den Informatikkursen haben die Veranstaltungen mit Lernenden stattgefunden, die sich explizit für diese angemeldet haben. Es ist davon auszugehen, dass diese Lernenden Interesse an Mathematik beziehungsweise mathematischer Modellierung hatten.

## 13.2 Evaluation

Die Auswertung der Erprobungen erfolgt basierend auf den Beobachtungen, die die Durchführenden während der Modellierungsveranstaltungen gemacht und dokumentiert haben. Zudem füllten die Lernenden nach den Modellierungsveranstaltungen eine schriftliche Befragung aus. Diese dient dazu, die persönlichen Erfahrungen der Lernenden, ihre Meinung zum Lernmaterial, zu Ablauf und Thema des Workshops sowie zum Schwierigkeitsgrad abzufragen.

### Zentrale Beobachtungen

Zunächst werden die Beobachtungen, die während der Durchführung der Modellierungsveranstaltungen von den Dozierenden gemacht wurden, zusammenfassend dargelegt.

**Tab. 13.1:** Durchgeführte Modellierungsveranstaltungen zum Thema Solarenergie. Die mit \* markierten Veranstaltungen haben in Präsenz stattgefunden, die übrigen digital. Die mit \*\* markierte Veranstaltung wurde von der Lehrkraft der Lerngruppe durchgeführt. Thema I  $\hat{=}$  Optimierung der Rohrhöhe, Thema II  $\hat{=}$  Optimierung der Spiegelpositionen, Thema III  $\hat{=}$  Bearbeitung offener Modellierungsprobleme

Veranstaltung	Anzahl Lernende	Jgst.	Zeitlicher Rahmen
Modellierungstage MINT-EC* (DE), I + III	18 (9m/9w)	11–12	10/2019 11h (1.5 Tage)
Schülerpraktikum KIT* (BW), II + III	3 (3m/0w)	10–11	02/2020 10h (2 Tage)
Modellierungstag MINT-EC (DE)*, II	16 (12m/6w)	10–13	02/2020 5h (1 Tag)
Offene Modellierungstage RWTH (NRW), II	8 (4m/4w)	10–11	04/2020 8.5h (1.5 Tage)
U-Reihe Informatikkurs Gym.** (BW), II + III	12 (9m/3w)	11	04–07/2020 11h (7 x 1.5h)
Modellierungstage MINT-EC (DE), I + II + III	29 (19m/8w/2d)	10–13	08/2020 12h (2 Tage)
Schülerpraktikum KIT* (BW), II + III	2 (1m/1w)	10	10/2020 15h (3 Tage)
Offener Modellierungstag KIT (BW), I	1 (1m/0w)	12	10/2020 5h (1 Tag)
U-Reihe Modellierungs-AG (NRW), I	7 (4m/3w)	9	01–02/2021 9h (4.5 x 2h)
U-Reihe Modellierungs-AG (BW), I	9 (7m/2w)	9–10	01–02/2021 9h (4.5 x 2h)
U-Reihe Matheprofis KIT (BW), I	5 (4m/1w)	9–10	02/2021 7.5h (5 x 1.5h)
U-Reihe Informatikkurs Gym. (BW), II + III	12 (10m/2w)	12	04–06/2021 6h (4 x 1.5h)
Modellierungstage MINT-EC 2021 (DE), II + III	25 (15m/10w)	10–13	08/2021 11h (2 Tage)

Zur besseren Übersicht werden die Beobachtungen nach den drei Themen, Optimierung der Rohrhöhe, Optimierung der Spiegelpositionen und offenen Modellierung,

sortiert. Bei der Darstellung der Beobachtungen wird insbesondere auf Schwierigkeiten der Lernenden bei der Bearbeitung der Lernmaterialien eingegangen. Es wird erläutert, welche Änderungen am Material vorgenommen wurden, um den Schwierigkeiten zu begegnen.

### Optimierung der Rohrhöhe

Bei der Bearbeitung des Lernmaterials zur Optimierung der Rohrhöhe war zu beobachten, dass die Lernenden keine Probleme mit dem Vorgehen *Optimieren durch Ablesen am Funktionsgraphen* hatten. Die Lernenden hatten kaum Fragen und arbeiteten überwiegend konzentriert. Bei der Optimierung mit dem stark vereinfachten Modell (ohne atmosphärische Abschwächung und ohne fehlerhafte Spiegeleinstellung) nannten die Lernenden vielfach einen Wert von circa 3 – 10 m als zu wählende optimale Höhe, auch wenn dies nicht der optimalen Lösung des Problems im mathematischen Sinne entsprach. Als Begründung wurde genannt, dass eine weitere Erhöhung der Rohrhöhe kaum zu einer Verbesserung der Leistung führt. Im Hinblick auf Materialkosten würde sich dies nicht lohnen. In der Diskussion wurde letztlich erkannt, dass der steigende Verlauf des Zielfunktionsgraphen dazu führt, dass formal keine optimale Rohrhöhe existiert. Es wurde von den Lernenden wiederholt angemerkt, dass kein sinnvolles Ergebnis für die optimale Rohrhöhe gefunden werden kann.

Neben der Berücksichtigung von Materialkosten nannten die Lernenden verschiedene weitere Modellverbesserungsideen, beispielsweise die Berücksichtigung des Aufwands für die Konstruktion oder eines möglichen Energieverlusts der Sonnenstrahlen. Zudem schlugen einzelne Lernende vor, bei der Optimierung ein Abbruchkriterium festzulegen. Konkret wurde genannt, die Optimierung abubrechen sobald die Steigung kleiner ist als ein gewisser Schwellwert.

Bei der Erarbeitung der Modellerweiterung um atmosphärische Abschwächungseffekte (AB 6) wurden keine Schwierigkeiten beobachtet. Die Lernenden äußerten in den Diskussionen, dass basierend auf dem neuen Modell eine realistisch konstruierbare optimale Rohrhöhe existiert.

Bei der Modellierung der fehlerhaften Spiegeleinstellung (AB 7) hatten die Lernenden vermehrt Schwierigkeiten bei der Bearbeitung der geometrischen Aufgaben. Insbesondere die Modellierung der Fallunterscheidungen (vgl. Abb. 11.5) dauerte lange und führte bei den Lernenden gehäuft zu Fragen. Demzufolge wurde die Aufgabe gekürzt, sodass die Lernenden nur noch drei von sechs Fällen selbst modellieren mussten. Zudem wurde eine Tippkarte mit einer Skizze zu relevanten Größen zur Verfügung gestellt. Das Ablesen der optimalen Rohrhöhe am Funktionsgraphen und die Diskussion, inwieweit der veränderte Verlauf des Graphen zur Existenz einer optimalen Rohrhöhe führte, stellte keine Schwierigkeit dar.

Das Zusatzblatt zur Monte-Carlo-Methode wurde lediglich von zwei Kleingruppen (vier Lernenden) bearbeitet. In der anschließenden Diskussion mit den Lernenden erläuterten sie ihre Erkenntnisse und begründeten den Einsatz der Methode wie folgt: „Man sollte mehrere Fehler berücksichtigen, da man ja gar nicht genau weiß, wie groß der Fehler ist“ (Zitat einer teilnehmenden Person).

In den ersten Erprobungen lag lediglich die Langversion von Arbeitsblatt 4 vor. Dies führte dazu, dass die Lernenden viel Zeit für die geometrische und physikalische Modellierung auf den Arbeitsblättern 1 – 4 benötigten. Damit blieb weniger Zeit für die anschließende Optimierung. Aus diesem Grund wurde die Kurzversion von Arbeitsblatt 4 entwickelt.

#### **Optimierung der Spiegelpositionen**

Bei der Modellierung der Energie ausgehend von der momentanen Leistung (AB 2) war bei den ersten Erprobungen zu beobachten, dass einzelne Lernende Schwierigkeiten hatten, den Zusammenhang der beiden physikalischen Größen korrekt zu erfassen und anzuwenden. Deswegen wurde eine zweistufige Tippkarte entwickelt, die sowohl den Zusammenhang zwischen den Einheiten als auch die Grundidee der numerischen Integration an einem kleinen Beispiel verdeutlicht. In den nachfolgenden Durchführungen war beim (virtuellen) Rundgang durch die Gruppen vermehrt zu beobachten, dass die Lernenden von diesen Tippkarten Gebrauch machten. Zudem äußerten die Lernenden seltener Verständnisfragen zu dieser Aufgabe.

Bei der Optimierung der Position **eines** Spiegels waren keine Schwierigkeiten beim Ablesen der optimalen Position am Graphen zu beobachten. In einzelnen Erprobungen waren die Lernenden in der Lage das abgelesene Ergebnis (senkrecht unter dem Rohr) durch den Kosinuseffekt zu begründen.

Auch das Ablesen der optimalen Positionen **zweier** Spiegel am dreidimensionalen Funktionsgraphen verlief ohne zu beobachtende Schwierigkeiten. Die Lernenden argumentierten in den Diskussionen, dass die optimale Positionierung (beide Spiegel senkrecht unter dem Rohr) im Hinblick auf die Realität nicht sinnvoll ist:

„Es wäre optimal beide Spiegel direkt unter das Rohr zu stellen. Das ist aber schlecht, da dann nur ein Spiegel wirklich effektiv Sonnenstrahlen reflektiert. Man sollte einen Abstand voraussetzen“ (Zitat einer teilnehmenden Person).

Als weitere Modellverbesserungsideen nannten die Lernenden vielfach den Einbau von Verschattungseffekten.

In der ersten Version des Lernmaterials lag keine Tippkarte zum Aufstellen der Ungleichung, die als Nebenbedingung des Optimierungsproblems einen Mindestabstand sichern sollte, bereit. Da in den ersten Durchführungen wiederholt beobachtet wurde, dass die Lernenden damit Schwierigkeiten hatten, wurde eine Tippkarte erstellt, in der denkbare Situationen der Missachtung des Mindestabstands für zwei Spiegel visualisiert wurde.

Zentral im Workshop zur Optimierung der Spiegelpositionen war die Brainstorming-Phase, in der die Lernenden eigene Ideen für die Entwicklung eines Optimierungsverfahrens sammeln sollten. In den ersten Erprobungen war zu beobachten, dass Lernende den Sinn der Aufgabenstellungen wiederholt anzweifeln, da „es ja klar ist, dass die Spiegel einfach so nah wie möglich aneinander gestellt werden sollten“. Deswegen wurde eine Zwischenüberlegung / ein Gedankenexperiment zu dieser Aufgabenstellung ergänzt: *Wir werden die Funktion  $E_{\text{tube}}$  später erweitern, so dass Verschattungseffekte zwischen Spiegeln eingebaut werden. Kann dein Verfahren*

*auch dann noch sinnvoll angewandt werden?* Diese Ergänzung führte dazu, dass der Vorschlag „möglichst nah aneinander stellen“ seltener als Optimierungsstrategie benannt wurde.

Bei den Erprobungen nannten die Lernenden diverse Ansätze für die Entwicklung eines Optimierungsverfahrens. Die Ideen der Lernenden lassen sich den folgenden Ansätzen zuordnen:

- In allen Durchführungen wurde die Idee benannt, das Spiegelfeld zu diskretisieren und anschließend alle möglichen Kombinationen an Positionen auszuprobieren (**Brute-Force-Methode**):

„Wir legen viele kleine mögliche Positionen fest und setzen dann die Spiegel auf die besten Kombinationen. Wir können die möglichen Positionen dann später auch nochmal feiner wählen“ (Zitat einer teilnehmenden Person).

Vereinzelt wurde bereits vor der Durchführung der Algorithmen-Arbeitsblätter erkannt, dass dieses Vorgehen nicht effizient ist: „Das dauert aber sehr lange und ist ungenau und umständlich“ (Zitat einer teilnehmenden Person).

- **Ableitung berechnen und gleich Null setzen.** Auf diesen Vorschlag erwiderten Mitlernende in den Diskussionsphasen, dass die Funktion zu komplex sei, um sie abzuleiten.
- Auch der **Greedy-Algorithmus** wurde in einzelnen Erprobungen benannt:

„Man berechnet immer die optimale Position des nächsten Spiegels und setzt die Spiegel nach und nach. Oder man lässt den Computer einfach alle Möglichkeiten ausprobieren. Entweder in einem bestimmten Muster also beispielsweise immer eine Einheit weiter zu setzen oder auch komplett zufällige Anordnungen durchzuprobieren“ (Zitat einer teilnehmenden Person).

- Einzelne Lernende schlugen zudem einen **musterbasierten Ansatz** vor:

„Der Abstand von beiden Spiegeln zu  $x = 0$  muss gleich sein. Oder halt der Abstand von beiden Spiegeln zum Rohr. Nur dieser gleiche Abstand soll optimiert werden. Und für die nächsten beiden Spiegel sollte das gleiche wieder gelten“ (Zitat einer teilnehmenden Person).

- **Möglichst nah aneinander:** Bei gerader Spiegelanzahl werden die Spiegel symmetrisch links und rechts um das Rohr positioniert. Bei ungerader Spiegelanzahl wird ein Spiegel unter das Rohr gesetzt und die übrigen symmetrisch rechts und links um das Rohr. In beiden Fällen werden die Spiegel möglichst nah aneinander gesetzt.

Beim Rundgang durch die Kleingruppen waren keine Schwierigkeiten bei der Bearbeitung der vier Optimierungsarbeitsblätter zu beobachten. Die Lernenden äußerten selten Fragen zu den Arbeitsblättern. Im Plenum wurden die Erkenntnisse der Lernenden im Hinblick auf verschiedene Optimierungsansätze und Verfahren diskutiert. Zusammengefasst nannten die Lernenden folgende Aspekte:

Die Lernenden stuften den **Brute-Force-Algorithmus** als langsam ein, sobald eine feine Diskretisierung gewählt wurde. Zudem erkannten sie, dass genauere Ergebnisse für die Spiegelpositionen gefunden werden, wenn die Diskretisierung feiner gewählt wird. Wiederholt wurde vorgeschlagen, die Optimierung zunächst mit einer groben Diskretisierung durchzuführen und diese dann erneut, mit feinerer Diskretisierung, in dem Bereich zu wiederholen, in dem mit dem groben Ansatz Positionierungen mit hohem Zielfunktionswert gefunden wurden. Im Hinblick darauf, ob die optimale Lösung mit diesem Ansatz gefunden werden kann, merkte eine teilnehmende Person exemplarisch an: „Je nachdem in wie viele Stücke man das unterteilt, findet man manchmal die optimale Lösung, aber vielleicht auch nicht.“

Die Lernenden beurteilen den **Greedy-Algorithmus** als nicht optimal, da der erste Spiegel stets unter das Rohr gesetzt wird. Wiederholt wurde darauf verwiesen, dass sich bereits im Falle zweier Spiegel nicht die optimale Lösung für die Spiegelpositionen ergibt:

„Der Algorithmus ist sehr schnell, aber arbeitet sehr kurzsichtig. Er schaut also nicht nach global guten Lösungen. Das heißt die Lösung ist nicht so gut wie beispielsweise bei Algorithmus 4 [freie Variablen Optimierung; Anmerk. der Verf.]. Die Lösung hängt außerdem von der Anzahl der Stücke ab, in die das Kraftwerksareal unterteilt wird“ (Zitat einer teilnehmenden Person).

Der **musterbasierte Ansatz** wurde von Lernenden wiederholt als schlecht eingestuft. Dieser „sucht nur sehr eingeschränkt nach einer optimalen Lösung. Es hängt vom Muster und vom Startwert ab, wie gut die Lösung ist“ (Zitat einer teilnehmenden Person). Das Muster wurde als zu eingeschränkt beurteilt.

Der Ansatz über die **freie Variablen Optimierung** wurde von den Lernenden mehrfach als besonders gut eingestuft, „da er schnell ist und gute Lösungen liefert. Problem ist aber, dass die beste Lösung nicht unbedingt gefunden wird. Man sollte das also von unterschiedlichen Startwerten wiederholen“ (Zitat eines Lernenden zur Optimierung basierend auf dem Modell ohne Verschattungseffekte). Bei der Optimierung mit Verschattungseffekten dauerten die Berechnungen deutlich länger. Dies kommentierte eine Lernende wie folgt:

„Die freie Variablen Optimierung ist zwar mit Schatten langsam, aber für die Planung eines großen Kraftwerks mit zum Beispiel 16 Spiegelreihen kann man ruhig mal drei Minuten Laufzeit abwarten.“

In der Abschlussdiskussion wurde nach weiteren Modellverbesserungsideen gefragt. Die Vorschläge der Lernenden waren vielfältig. Sie nannten unter anderem die Berücksichtigung von Wetter, die Optimierung des Standorts des Kraftwerks, die Optimierung der Form der Spiegel (bspw. Parabolspiegel) und die Berücksichtigung von Verschmutzungen der Spiegel.

### **Bearbeitung offener Problemstellungen**

In den ersten Erprobungen wurden die offenen Problemstellungen ohne vorbereitetes digitales Arbeitsmaterial herausgegeben. Dabei war die Schwierigkeit zu beobachten, dass die Lernenden auf bereits entwickelte Funktionen von zuvor bearbeiteten

Arbeitsblättern zugreifen wollten, dies aber nicht konnten. Deswegen wurden Arbeitsblätter zum unangeleiteten, offenen Modellieren entwickelt. Auf diesen werden bisherige Funktionen geladen und können direkt eingesetzt werden.

Zudem war zu beobachten, dass die Lernenden Schwierigkeiten hatten Kontrollstrukturen in der für sie neuen Programmiersprache zu verwenden. Deswegen wurden Tippkarten mit if-Abfragen und for-Schleifen sowie Hinweise mitsamt Beispielen zum Plotten von Funktionen und Daten entwickelt.

Die Ansätze, Modelle und Algorithmen, die von Schülergruppen bei der Bearbeitung der offenen Problemstellungen verfolgt, entwickelt und zum Teil implementiert wurden, werden im Folgenden zusammengefasst.

### Problem 1: Optimierungsverfahren implementieren

- **Optimierung des Spiegelabstands über Brute-Force.** Die Lernenden schrieben die bereits entwickelte Funktion für die Leistung am Rohr  $P_{\text{tube}}$  so um, dass diese vom Abstand zwischen den einzelnen Spiegeln abhing. Anschließend bestimmten sie den optimalen Abstand mithilfe einer Brute-Force-Methode.
- **Positionierung gemäß Fallunterscheidung.** Die Lernenden unterschieden je nach gerader beziehungsweise ungerader Spiegelanzahl, wie die Spiegel positioniert werden. Sie berücksichtigten dabei nicht die Zielfunktion. Die Frage, ob das Modell auch bei Verschattung noch eine optimale Positionierung liefert, verneinten die Lernenden. In dem Fall würden sie die Funktion integrieren wollen.
- **Optimierung der Spiegelpositionen über Brute-Force.** Diverse Schülergruppen implementierten einer Brute-Force-Methode zur Optimierung der Positionen zweier Spiegel. Dabei nutzten sie zwei ineinander verschachtelte for-Schleifen. Einige Schülergruppen hatten das angeleitete Arbeitsblatt zur Brute-Force-Methode zuvor nicht bearbeitet.

### Problem 2: Verschattung modellieren

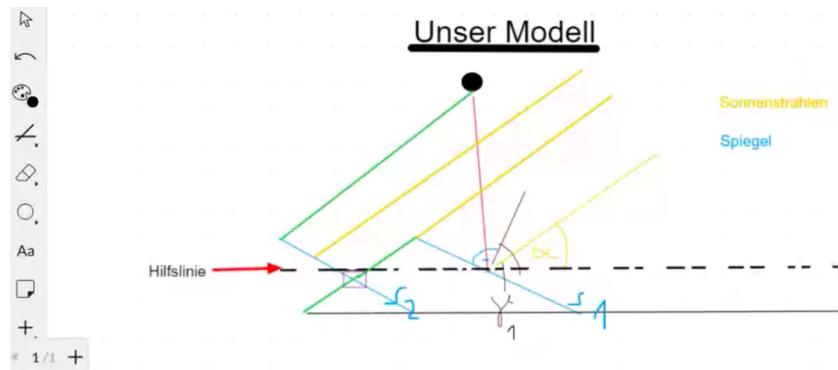
Bei den Schülergruppen, die den Schattenwurf eigenständig modelliert haben, wurde das bereitgestellte Whiteboard besonders intensiv genutzt (vgl. Abb. 13.1). Folgende Ansätze und Modellierungsschritte wurden beobachtet:

- Diverse Lerngruppen beschrieben Spiegel und Sonnenstrahlen durch Graphen linearer Funktionen und berechneten zur Bestimmung verschatteter Bereiche Schnittpunkte zwischen Geraden. Für senkrechte Strahlen nutzten die Lernenden die Beschreibung  $x = a$ , wobei  $a$  die Stelle beschreibt, an der der Strahl die  $x$ -Achse schneidet.
- Eine schnelle Lerngruppe<sup>1</sup> modellierte den Schattenwurf zwischen Spiegeln nicht nur, sondern implementierte diesen in Julia. Die Gruppe führte darü-

<sup>1</sup>Die Gruppe hatte an den zweitägigen Modellierungstagen des MINT-EC 2020 teilgenommen (vgl. Tab. 13.1).

ber hinaus die Optimierung der Sekundärreflektorbreite durch und erstellten eine Präsentation sowie eine GeoGebra Simulation zur Visualisierung der Verschattungseffekte.

- Eine Lerngruppe spielte auf dem Whiteboard verschiedene Szenarien durch, wie die Verschattung durch eine geänderte Konstruktion des Kraftwerks verringert werden könnte. Ihr Vorschlag war das halbkreisförmige Aufstellen der Spiegel um einen runden Absorber.



**Abb. 13.1:** Digitales Whiteboard (bereitgestellt von awwapp), das von einer Lerngruppe während eines Modellierungstages gestaltet wurde

### Problem 3: Beliebige Kraftwerkparameter optimieren

- **Optimierung von Spiegelbreite und Spiegelanzahl.** Die Lernenden definierten die Funktion für die Energie über einen Tag in Abhängigkeit von der Anzahl  $N$  der aufzustellenden Spiegel und von der Spiegelbreite. Zusätzlich recherchierten sie Materialkosten und integrierten diese in ihr Modell. Die Optimierung setzten sie mithilfe einer Brute-Force-Methode um.
- **Optimierung von Spiegelbreite und Spiegelanzahl.** Die Lernenden definierten die Funktion für die Energie über einen Tag in Abhängigkeit von der Anzahl  $N$  der aufzustellenden Spiegel und von der Spiegelbreite. Sie nutzten das Optimierungstool von Julia, welches auf dem Arbeitsblatt zur freien Variablen Optimierung eingesetzt wurde. Bei der Optimierung berücksichtigten sie den Mindestabstand.
- **Optimierung der Breite des Sekundärreflektors.** Die Lernenden betrachteten zwei Spiegel auf festen Positionen und visualisierten den Graph der Zielfunktion in Abhängigkeit der Reflektorbreite. Sie bestimmten die optimale Reflektorbreite durch Ablesen am Funktionsgraphen. Das erste Ergebnis der Lernenden war, dass der Reflektor *sehr* breit gebaut werden sollte. Die Lernenden erkannten, dass die Verschattung durch den Reflektor vernachlässigt wurde. Sie bauten diese mit Stift und Papier in das Modell für die umgesetzte Energie ein.

Stand für die Bearbeitung der offenen Problemstellungen wenig(er) Zeit zur Verfügung, so war das Ergebnis der Lernenden nicht immer ein lauffähiger Code. Eine weitere Herausforderung, die vermehrt zu Fragen von den Lernenden führte, war, dass die im Internet verfügbare Dokumentation zu Julia zum Teil nicht umfänglich vorhanden ist. Zudem war zu beobachten, dass die Lernenden ohne Programmiererfahrung zumeist die Problemstellung 2 (Verschattung) wählten oder weiterhin mit den angeleiteten Arbeitsblättern arbeiten wollten.

In der ersten Erprobung im Rahmen eines mehrtägigen Modellierungsprojektes (Modellierungstage MINT EC 2019) sollten die Lernenden alle Arbeitsblätter des Mittelstufenworkshops bearbeiten, bevor die offenen Modellierungsprobleme herausgegeben wurden. Dies wurde von den Lernenden als zu eintönig bemängelt. Sie hätten sich einen früheren Übergang zu den offenen Problemen gewünscht. Dies wurde in nachfolgenden Erprobungen berücksichtigt.

Zum Teil waren Überprüfe-Funktionen noch fehlerhaft. Korrekte Lösungen der Lernenden wurden bei vereinzelt Aufgabenstellungen nicht akzeptiert. Dies wurde im Laufe der Erprobungen korrigiert. Insbesondere bei den ersten Erprobungen kam es zu Instabilitäten des Servers, auf dem das Lernmaterial bereitgestellt wurde. Die Arbeitsblätter mussten neu gestartet werden oder der Zugriff auf die Arbeitsblätter war für mehrere Minuten nicht möglich.

## Ergebnisse der schriftlichen Befragung

Die Ergebnisse der schriftlichen Befragung werden hier zusammengefasst. Die detaillierten Evaluationsergebnisse aller Erprobungen sind in Anhang G.1 dargestellt. Es ist anzumerken, dass a) nicht alle Lernenden überhaupt die Befragung ausgefüllt haben (daher die Abweichung der Anzahl an Antworten aus der Evaluation zu den in Tabelle 13.1 dargestellten Schülerzahlen) und b) nicht alle Lernenden, die an der Befragung teilgenommen haben, alle Fragen beantwortet haben.

Die Mehrheit der Lernenden gab an, dass sie durch den Workshop *ein besseres* oder *eher ein besseres* Verständnis von mathematischer Modellierung gewonnen haben (122/128). Dies wurde auch wiederholt auf die Frage genannt, was sie durch den Workshop gelernt haben:<sup>2</sup>

- „Neben neuen Informationen zum Thema, habe ich persönlich durch den Workshop das Interesse an der mathematischen Modellierung gewonnen.“
- „Die Prinzipien / Das Konzept von mathematischer Modellierung, ein wenig zu Optimierung“
- „Ich habe gelernt, wie man beim Modellieren eines mathematischen Problems vorgeht und, dass man es immer weiter verbessern kann.“

Folgende Zitate verdeutlichen zudem, dass insbesondere das schrittweise Vorgehen bei der Modellierung und das durchdachte Treffen von Vereinfachungen durch die Workshops deutlich geworden ist:

<sup>2</sup>Rechtschreib- und Interpunktionsfehler der Zitate der Lernenden aus den Evaluationen wurden zur besseren Lesbarkeit korrigiert.

- „Zentral in der Lösung war für mich, klare Vereinfachungen zu wählen, an Stellen wo sie vertretbar sind.“
- „Das kleinschrittige Herangehen an ein komplexes Problem“
- „Wie viele verschiedene Modellierungsansätze es gibt“
- „Modellierung, Reduzierung eines komplexen Sachzusammenhangs (Solarkraftwerk) zu einer (mehr oder weniger) simplen Modellvorstellung“

Die Gestaltung der Arbeitsblätter stuft die Mehrheit der Lernenden als *gut* oder *eher als gut* ein (67/74). Dies wurde auch mehrfach als Antwort auf die Freitextfrage genannt, was ihnen am Workshop besonders gut gefallen hat:

- „Die Vorbereitung der Aufgaben war super.“
- „Die Arbeitsblätter, die sehr gut strukturiert waren und angenehm zu bearbeiten“
- „Es war schön, dass es so viele Grafiken gab, die wir auch verstellen konnten. Es ist gut so ein bisschen visuelle Hilfe zu haben.“

Die automatischen Rückmeldungen stuft die Mehrheit der Lernenden als *hilfreich* oder *eher hilfreich* ein (47/58). Dies wurde auch in den Freitextfeldern betont:

„Die vorgegebenen Codefelder, in denen man seine eigenen Formeln / Algorithmen einfügen muss, finde ich sehr angenehm und intuitiv und die direkte Kontrolle ist sehr hilfreich“ (Zitat aus der Evaluation).

Insgesamt stuften die Lernenden die Vermittlung der Inhalte mehrheitlich als klar ein. Lediglich 4/130 Lernende antworteten auf diese Frage mit *trifft eher nicht zu*. Hingegen gaben 125/130 Lernende *trifft voll zu* oder *trifft eher zu* an.

Im Hinblick auf den Schwierigkeitsgrad gaben 101/127 Lernende an, dass die Aufgaben *nicht* oder *eher nicht* zu einfach waren und 100/129 Lernende gaben an, dass die Aufgaben *nicht* oder *eher nicht* zu schwierig waren. Damit kann der Schwierigkeitsgrad für die Teilnehmenden der Erprobungen als angemessen eingestuft werden. Dies wird auch durch die Antworten auf die Frage, inwieweit die Lernenden die Aufgaben eigenständig bearbeiten konnten, unterstrichen.

Da die Mehrheit der Lernenden sich explizit für die Teilnahme an einem Workshop angemeldet hatte, ist davon auszugehen, dass die Lernenden mathematisch interessiert waren. Dies schränkt die Übertragbarkeit der Ergebnisse auf heterogenere Lerngruppen ein.

Die Tippkarten bewerteten 94/123 Lernende als *hilfreich* oder *eher hilfreich* und 22/123 Lernende als *nicht hilfreich* oder *eher nicht hilfreich*. Die übrigen Befragten kreuzten *nicht beurteilbar* an. Dies ist vermutlich darauf zurückzuführen, dass sie nicht von den Tippkarten Gebrauch machten. Insgesamt kann der Einsatz des Differenzierungselements als sinnvoll eingestuft werden.

Den Umgang mit der Programmiersprache stuften die Lernenden durchaus unterschiedlich ein. So gaben 78/122 Lernende an, dass ihnen der Umgang *eher nicht*

oder *nicht* schwerfiel, wohingegen 44/122 Lernende angaben, dass ihnen der Umgang schwerfiel oder eher schwerfiel. Dieses gemischte Bild unterstreichen auch die Aussagen der Lernenden aus den Freitextfeldern. Einige Lernende antworteten auf die Frage, was sie durch den Workshop gelernt haben, „Programmieren“ oder „Ich habe das Programmieren mit Julia kennengelernt.“ Der Workshop machte einigen Lernenden nicht nur deutlich, dass Mathematik Relevanz für den Alltag hat, sondern zudem die Informatik: „Ich konnte das erste Mal sehen, wie Informatik im Leben umgesetzt und gebraucht wird.“ Demgegenüber kritisierten andere Lernende, die an den offenen Problemen (vgl. Abschn. 12.2.4) gearbeitet haben, dass „freies Programmieren bei schulischen Programmierkenntnissen“ gefordert war. Vermehrt wurde in den Evaluationen die Instabilität des Servers und des Videokonferenztools Jitsi bemängelt. Dies ist auf einzelne Erprobungen zurückzuführen, an denen beides zeitweise nicht stabil lief.

Auf die Frage *Was fandest du an dem Teil zur Optimierung besonders interessant?* nannten die Lernenden, die die Optimierung der Spiegelpositionen durchgeführt hatten, wiederholt die Algorithmen beziehungsweise den Vergleich der verschiedenen Methoden. Folgende Zitate machen zudem ersichtlich, dass der Facettenreichtum der Optimierung ersichtlich geworden ist:

- „Besonders interessant fand ich die verschiedenen Problemstellungen sowie differenzierten Ansichten der Gruppen zur Lösung eines Problems.“
- „Zu sehen, wie durch mathematische Vorgehensweisen eine komplexe Optimierung mit auch mehreren Ideen gelöst werden kann“
- „Die Vielseitigkeit an Möglichkeiten, welche offen stehen und dass bei jeder neuen Idee direkt wieder Probleme und Grenzen aufkommen“
- „Wie viele verschiedene Arten der Optimierung [es] gibt“
- „Ich fand besonders interessant, dass man sehr offen an Optimierungsproblemen arbeiten kann.“

Insbesondere das letzte Zitat unterstreicht, was viele Lernende sowohl auf die Frage *Was fandest du an dem Teil zur Optimierung besonders interessant?* als auch auf die Frage *Hat dir etwas an dem Workshop besonders gut gefallen?* als positiv hervorgehoben haben: das freie / eigenständige Arbeiten.

Auch die Arbeit in Gruppen wurde von zahlreichen Lernenden als positiv oder sinnvoll hervorgehoben:

- „Ich habe gelernt, dass es sinnvoll ist sich zusammenzuschließen und es mit Interessierten besonders Spaß macht an einem Thema zu arbeiten. Außerdem habe ich viel im Bezug auf Studiengänge und Möglichkeiten für die Zukunft gelernt, was mir sehr weitergeholfen hat.“ (*Was hast du für dich persönlich durch die Teilnahme am Workshop gelernt?*)
- „Offene Gruppenarbeit am Ende“ (*Hat dir etwas am Workshop besonders gut gefallen?*)

- „das selbstständige Arbeiten in Kleingruppen“ (*Hat dir etwas am Workshop besonders gut gefallen?*)

Dass die Workshops verdeutlicht haben, dass Mathematik auch in der Realität Anwendung findet, unterstreichen folgende (exemplarische) Zitate der Lernenden auf die Frage, was sie durch den Workshop gelernt haben:

- „Dass es zum Glück auch eine eher mehr spaßige Version von Mathematik gibt, bei der man eben normalerweise gegebene Hilfsmittel, wie Programme benutzt, ganz im Gegensatz zu der meist langweiligen Schulmathe.“
- „Verknüpfung von Mathematik mit Physik und Informatik.“
- „Schulmathe findet tatsächlich Anwendungen. Es macht Spaß und es ist interessant wie weit man Sachen berechnen kann.“
- „Ich habe mit der mathematischen Modellierung eine wichtige Einsatzmöglichkeit der Mathematik kennengelernt, die ich vorher nicht kannte. Nun habe ich konkretere Vorstellungen von der Bedeutung der Mathematik für andere Bereiche.“
- „Die Matheformeln kommen doch nicht nur im Schulbuch vor.“
- „Dass Mathematik sehr spannend und anwendungsbezogen sein kann.“

Kritikpunkt einzelner Lernender war, dass sie sich einen stärkeren Bezug zur aktuellen Forschung im Bereich Solarenergie und einen tieferen Einblick in unterschiedliche Vor- und Nachteile diverser Technologien aus diesem Bereich gewünscht hätten. Konkret schlugen die Lernenden vor, Podiumsdiskussionen mit / Vorträge von Expertinnen und Experten in die Workshops einzubinden.

Die Gesamtbewertung der Workshops fällt mit der durchschnittlichen Schulnote 1.67 durchaus positiv aus. Dies unterstreichen auch die Antworten der Lernenden zu: *Alles in allem hat mir der Workshop gut gefallen*. Lediglich eine befragte Person kreuzte *trifft eher nicht zu* an. Einmal wurde *nicht beurteilbar* angegeben. Die übrigen 128/130 Lernenden gaben entweder *trifft eher zu* oder *trifft voll zu* an.

### 13.3 Didaktische Reflexion

In diesem Abschnitt soll das entwickelte Lernmaterial, die eingesetzten digitalen Werkzeuge und die Problemstellung im Hinblick auf die in Abschnitt 6.1 genannten übergeordneten Ziele und die in Abschnitt 1.3 diskutierten Kriterien für *gute Modellierungsprobleme* reflektiert werden.

Die Energiewende und der fortschreitende Klimawandel werden in den Medien, in der Politik, in Wirtschaft und Forschung vielfältig diskutiert. Im Hinblick auf die Abschaltung der Atomkraftwerke und den anstehenden Kohleausstieg Deutschlands sind effiziente, saubere Energiequellen unabdingbar. Deswegen wird und muss, jetzt

und auch in Zukunft, intensiv an der optimierten Nutzung erneuerbarer Energieformen geforscht werden. Damit ist das Thema dieses Workshops nicht nur heutzutage, sondern vermutlich auch in den nächsten Jahrzehnten als **relevant** zu bewerten. Dass das Bewusstsein für den Klimawandel und die Notwendigkeit für ein Umdenken in der Gesellschaft nicht nur in Forschung und Wirtschaft, sondern auch bei jungen Menschen ein relevantes und viel diskutiertes Thema ist, untermauert beispielsweise die Initiative Fridays for Future.

Die Problemstellung wurde im engen Austausch mit Partnerinnen und Partnern aus der Praxis zunächst für die Durchführung im Rahmen einer Modellierungswoche aufbereitet. Genauer stammte die Problemstellung von der Firma Frenell. Der Austausch wurde in der Folge mit der TSK Flagsol fortgeführt. Nicht nur die genannten Kooperationspartnerinnen und -partner, sondern zahlreiche weitere Forschungseinrichtungen und Firmen<sup>3</sup> forschen intensiv im Bereich Solarthermie. Damit ist die Problemstellung durchaus als **authentisch** zu bewerten.

Die bei den angeleiteten Lernmaterialien zur Lösung des Problems angewandte Mathematik ist ausgehend von Fachbeiträgen und Publikationen didaktisch reduziert worden. Die Grundideen und -ansätze aus Forschung und Technik wurden jedoch beibehalten. Damit ist nicht nur die Problemstellung, sondern zudem die angewandte Mathematik als authentisch einzuordnen.

Bei der Modellierung des solarthermischen Kraftwerks und der Diskussion verschiedenster Ansätze zum Lösen der Optimierungsprobleme ist die **innermathematische Reichhaltigkeit** eindrücklich sichtbar geworden. Die Reichhaltigkeit der Problemstellung und der Facettenreichtum der Optimierung wurde im Lernmaterial vielfältig ausgenutzt. Das Lernmaterial zur Optimierung der Spiegelpositionen erlaubt es den Lernenden, eigene Ansätze zur Optimierung zu formulieren und diese mit in der Forschung gängigen Strategien zu vergleichen.

Das Material ermöglicht, sowohl im Hinblick auf den Modellierungsprozess im Allgemeinen als auch beim Formulieren und Lösen der Optimierungsprobleme im Besonderen, mathematisch kreativ tätig zu sein und eigene Ideen und Ansätze computergestützt umzusetzen. Dabei wird insbesondere ersichtlich, dass Optimieren mehr ist als das Kriterium *Ableitung gleich Null*.

Die Problemstellung ist auch **außermathematisch reichhaltig**. So sind bei der Modellierung des Kraftwerks physikalische Grundlagen zentral. Der Einsatz des Computers beziehungsweise die Entwicklung von eigenem Programmcode hebt zudem die Vernetzung zur Informatik hervor<sup>4</sup>. Die inner- und außermathematische Reichhaltigkeit im Lernmaterial wurde bereits in Abschnitt 12.4 detailliert herausgearbeitet. Damit erfüllt das Lernmaterial, was unter anderem Heiliö und Pohjolainen (2016) als Kriterium für gute Modellierungsprobleme unterstreichen: **Interdisziplinarität**. Auch die von Heiliö und Pohjolainen (2016) hervorgehobene Bedeutung von Teamarbeit für das Lösen realer Probleme mithilfe mathematischer Modellie-

<sup>3</sup>Exemplarisch seien das Forschungszentrum Jülich, das Deutsche Zentrum für Luft- und Raumfahrt oder die Firma EBL genannt.

<sup>4</sup>Informatikerinnen und Informatikern würde es wohl widerstreben, wenn von Informatik gesprochen wird, sobald der Computer eingeschaltet wird.

rung wurde in den Erprobungen des Unterrichtsmaterials konsequent umgesetzt und wurde von den Lernenden wiederholt positiv bewertet.

Die (Weiter-)Entwicklung der Modelle und das Formulieren und Lösen der Optimierungsprobleme geschieht stets im Hinblick auf die Problemstellung, der *Modellierung und Optimierung des Fresnelkraftwerks*. Den Lernenden noch unbekanntes Mathematik wird nicht um der Mathematik willen, sondern zum Lösen der gegebenen realen Problemstellung eingeführt (wie bspw. Sinus, Kosinus und Tangens). Das Lernmaterial ist damit durchaus als **problemorientiert** zu bewerten. Für Lernende, die weiterführendes Interesse an den mathematischen Hintergründen haben, liegt an diversen Stellen Vertiefungsmaterial bereit.

Bei der Durchführung der Workshops liegt der Fokus auf der eigenständigen Bearbeitung des Materials durch die Lernenden. Die Arbeitsphasen, in denen sie in Kleingruppen an der Modellierung der Problemstellung arbeiten, überwiegt. Die **Eigenaktivität** der Lernenden steht im Vordergrund.

Die Erprobungen haben gezeigt, dass der **Schwierigkeitsgrad** – zumindest für mathematisch oder informatisch interessierte Lernende – durchaus angemessen ist. Durch das Differenzierungsmaterial (Tippkarten, Zusatzmaterial, Infomaterial, offene Aufgaben), welches wiederholt zum Einsatz kam, können die Lernenden auf ihrem Leistungsstand abgeholt werden. Die flexible Kombination des Lernmaterials ermöglicht, dass spontan auf das Interesse und den Kenntnisstand der Lernenden reagiert werden kann.

Beide Workshops erlauben das vollständige Durchlaufen des Modellierungskreislaufs. Insbesondere das sukzessive Erweitern der Modelle und wiederholte Interpretieren der (durch die Optimierung) gefundenen Lösungen für die Rohrhöhe beziehungsweise für die Spiegelpositionen ist zentraler Bestandteil des Lernmaterials. Die Evaluationsergebnisse unterstreichen, dass die Lernenden durch die Teilnahme an den Workshops ein besseres Verständnis von mathematischer Modellierung gewonnen haben.

Abschließend soll noch auf die **Eignung der gewählten digitalen Werkzeuge** eingegangen werden. Die Verwendung von Jupyter Notebooks erlaubte die Entwicklung übersichtlichen Lernmaterials, bei dem verschiedene Darstellungsformen der Lerninhalte kombiniert werden können (Visualisierungen / Grafiken, Texte, Videos, Code). Auch im Hinblick auf die Aussagen der Lernenden aus den schriftlichen Befragungen, ist die Gestaltung des Materials im Rahmen von Notebooks durchaus als geeignet zu bewerten. Die Möglichkeit, dass die Lernenden ohne Installation von Software auf das Material zugreifen können, spricht zudem für die Bereitstellung der Notebooks über JupyterHub. Da Instabilitäten dieser Bereitstellungsart nach den ersten Erprobungen seltener auftraten wurden, ist auch dies als sinnvoll zu bewerten. Die Wahl der Programmiersprache Julia hatte für die angeleiteten Projekte den Vorteil, dass die Syntax nah an Pseudocode ist und somit wenige Probleme / Fehlermeldungen durch eine fehlerhafte Syntax zu beobachten waren. Nachteil ist jedoch, dass die Dokumentation im Internet gegenüber anderen Programmiersprachen (bspw. Python) lückenhafter ist. Das eigenständige Recherchieren von Befehlen im Rahmen der Bearbeitung offener Problemstellungen stellte damit eine Herausforderung für die Lernenden und die Durchführenden dar.

## 13.4 Ausblick

Das Thema Solarthermie bietet die Möglichkeit sich in beliebige Richtungen weiter zu vertiefen. In diesem Abschnitt werden verschiedene Möglichkeiten für weiterführende Modellierungsaufgaben genannt, die an die Inhalte der Workshops anknüpfen und den Lernenden einen kreativen Einsatz von Mathematik ermöglichen.

### Modellierungsaufgaben aufbauend auf den Mittelstufenworkshop

- Die Lernenden setzen sich intensiver mit den Daten zum Stand der Sonne sowie der Stärke der Sonnenstrahlung auseinander. Der Zusammenhang zwischen der Sonnenposition, gegeben durch die Winkel Azimut und Altitude, und der Richtung der einfallenden Sonnenstrahlen im Raum ließe sich diskutieren (vgl. [Hoeffler, 2020](#)). Außerdem wäre die Approximation von realen Wetterdaten und deren Einbezug in die Modellierung der Leistung des Kraftwerks möglich.
- Die Algorithmen, die den verwendeten Optimierungstools zugrunde liegen, werden mit den Lernenden bei der Optimierung der Rohrhöhe nicht diskutiert. Denkbar ist, die Lernenden eigenständig ein Verfahren entwickeln zu lassen, mit dem die optimale Rohrhöhe bestimmt werden kann. Beispiele für umsetzbare Verfahren sind:
  - Die Unterteilung des Suchbereichs in diskrete Werte und die anschließende Berechnung aller zugehöriger Funktionswerte. Die Höhe mit größtem Zielfunktionswert wird näherungsweise als Lösung ausgegeben.
  - Ein weiteres Verfahren, zu dem im Rahmen einer von mir betreuten Seminararbeit von L. Bihr bereits Lernmaterial entwickelt und einmalig erprobt wurde, ist das schrittweise Voranschreiten in die Richtung, in die die Funktion stärker ansteigt oder fällt. Das Verfahren ist angelehnt an das Gradientenverfahren, wird jedoch didaktisch stark reduziert, um auch für Lernende ab Klasse 9 durchführbar zu sein. Konkret sollte das Verfahren ohne die Verwendung von Ableitungen auskommen. Um die Richtung des steilsten Abstiegs auszumachen, werden Funktionswerte von benachbarten Werten für die Rohrhöhe verglichen. Ein solches durchaus simples Suchverfahren ermöglicht wesentliche Komponenten von Abstiegsverfahren mit den Lernenden zu diskutieren: die Wahl von Startpunkt, Schrittweite und Abbruchkriterium.

### Modellierungsaufgaben aufbauend auf den Oberstufenworkshop

- In einem fächerübergreifenden Projekt von Mathematik und Informatik können die Lernenden einen Raytracer entwickeln und implementieren.
- Neben den Fresnel- und Parabolrinnenkraftwerken gibt es einen weiteren solarthermischen Kraftwerkstyp, der bereits an verschiedenen sonnigen Orten operiert: sogenannte Solarturmkraftwerke. Bei Solarturmkraftwerken wird das



**Abb. 13.2:** Seitenansicht eines Solarturmkraftwerks in Jülich, Quelle: DLR



**Abb. 13.3:** Luftaufnahme eines Solarturmkraftwerks in Jülich, Quelle: DLR

Sonnenlicht mithilfe von Reflektoren auf einen Absorber, der sich in der Spitze eines Solarturms befindet, fokussiert. Ein Solarturmkraftwerk, welches vom Deutschen Zentrum für Luft- und Raumfahrttechnik in Jülich betrieben wird, ist in den Abbildungen 13.2 und 13.3 dargestellt. Die Lernenden können zunächst wesentliche Komponenten eines solchen Kraftwerks recherchieren, ein Modell für die Leistung entwickeln und anschließend verschiedene Parameter (z. B. die Positionen der Spiegel oder die Turmhöhe) optimieren.

- Neben der Simulation von Solarkraftwerken werden in der Praxis auch Testkraftwerke wie das in den Abbildungen 13.2 und 13.3 dargestellte Solarturmkraftwerk gebaut. Diese dienen dazu, neue Komponenten, Verfahren und Aufbauten zu testen, physikalische Eigenschaften des Kraftwerks zu untersuchen und kommerzielle Kraftwerke noch kosteneffizienter planen und bauen zu können. In einem fächerübergreifenden Projekt (z. B. der Fächer Physik, Technik, Mathematik) ließe sich ein solarthermisches Kraftwerk in kleinem Maßstab von Lernenden entwickeln und bauen. Damit würde verstärkt die ingenieurwissenschaftliche Herangehensweise in der Forschung zu erneuerbaren Energien einbezogen.

## **Teil IV**

# **Unterrichtsmaterial zu Empfehlungssystemen und dem Netflix Preis**



# 14 Einleitung

In diesem Teil der vorliegenden Arbeit wird diskutiert, wie ein zeitgemäßer Mathematikunterricht durch die Modellierung realer, schülernaher Probleme aus dem Bereich KI bereichert werden kann.

Zahlreiche Anwendungen und Technologien aus dem Alltag der Lernenden basieren auf Methoden des maschinellen Lernens (ML), einem Teilbereich der KI: sei es, wenn Netflix, Amazon und Co ihren Nutzerinnen und Nutzern neue Produkte vorschlagen, wenn Mails automatisch als Spam erkannt werden oder sich unser Smartphone per Gesichtserkennung entsperrt. Was sich hinter diversen ML-Methoden verbirgt, sind mathematische Ideen, Konzepte und Aussagen, die in elementarer Form in zahlreichen Lehrplänen deutscher Bundesländer verankert sind. Dies macht das Themengebiet ML zu einem aussichtsreichen Kandidaten für die mathematische Modellierung realer Probleme mit Lernenden – und damit auch zu einer guten Möglichkeit, die Bedeutung von Mathematik für Gesellschaft, Alltag und Technik erfahrbar zu machen. Eine entsprechende Gestaltung von Lerngelegenheiten und Lernmaterialien zu Problemstellungen aus dem Bereich KI kann insbesondere einen Beitrag leisten, die erste und dritte Winter'sche Grunderfahrung, die der Mathematikunterricht anstreben und verknüpfen sollte, anzusprechen (vgl. Abschn. 1.2).

Im Rahmen dieser Arbeit wurde Unterrichtsmaterial zu einer ausgewählten Problemstellung entwickelt, die mit ML-Methoden gelöst werden kann. Genauer geht es um die Entwicklung eines Empfehlungssystems für Filme basierend auf einem realen Datensatz des Streaminganbieters Netflix. Im Folgenden werden kurz wesentliche Grundlagen von Data Science (DS) und ML erläutert und die Bedeutung der Optimierung für Methoden aus eben diesen Bereichen hervorgehoben. Anschließend werden unterschiedliche Prinzipien aus Praxis und Forschung, die bei der Modellierung von Empfehlungssystemen Einsatz finden, beschrieben und damit das notwendige außermathematische Hintergrundwissen zum unterrichtspraktischen Einsatz des Materials geliefert. Das ausgewählte Modell, entlang dessen das Lernmaterial entwickelt wurde, wird Schritt für Schritt erläutert. Die Modellierung läuft auf ein Optimierungsproblem hinaus, für das wiederum verschiedene Formulierungsmöglichkeiten und Lösungsstrategien bestehen. Unterschiedliche Formulierungen werden aufgegriffen und Vor- und Nachteile diskutiert. Abschließend wird das Lernmaterial im Überblick dargestellt und Erfahrungen aus der Durchführung von Unterrichtsreihen und Modellierungstagen beschrieben. Zudem wird der Modellierungsprozess einer Lerngruppe auf einer mathematischen Modellierungswoche beleuchtet.

## 14.1 Data Science, maschinelles Lernen und Optimierung

Dieser Abschnitt hat nicht den Anspruch, ein umfassendes Bild der beiden Forschungsbereiche DS beziehungsweise ML zu zeichnen. Vielmehr sollen zentrale Strategien dieser Bereiche herausgehoben werden, die sich im entwickelten Lernmaterial wiederfinden.

### Data Science – Wie geht das?

DS hat in den letzten Jahrzehnten immens an Bedeutung gewonnen. Dies hängt nicht zuletzt mit der Verfügbarkeit von immer größeren Datenmengen wie auch von schnelleren Computern und wachsenden Speicherkapazitäten zusammen, mit denen die Datenmengen verarbeitet werden können. Eine allseits anerkannte Definition von DS sucht man in der Fachliteratur vergeblich (vgl. [Gould, 2021](#), S. 11). Für die Zwecke dieser Arbeit begnügen wir uns mit dem Verständnis von DS als eine höchst interdisziplinäre Wissenschaft, die das Ziel hat, Daten auszuwerten beziehungsweise Wissen aus Daten zu extrahieren. Mathematik (insbesondere Statistik, aber auch Analysis, lineare Algebra, Optimierung etc.) und Informatik (Verarbeitung und Visualisierungen von Daten, Implementierung von Algorithmen etc.) spielen bei der Arbeit mit Daten eine zentrale Rolle. Wesentlich ist zudem Wissen aus dem Bereich, aus dem die Daten stammen (vgl. [Engel, im Druck](#)).

In der Literatur existieren verschiedene Prozessmodelle, die idealtypische Schritte bei der Analyse und Auswertung von Daten zur Beantwortung konkreter Fragestellungen aus und für Wirtschaft und Technik beschreiben. Beispiele sind

- der Sample, Explore, Modify, Model, Assess (SEMMA) Prozess (vgl. [SAS Institute, 2017](#)),
- das Cross-Industry Standard Process for Data-Mining<sup>1</sup> (CRISP-DM) Modell (vgl. [Chapman et al., 2000](#)),
- der Problem-Plan-Data-Analysis-Conclusion (PPDAC) Zyklus (vgl. [Wild und Pfankuch, 1999](#)) und
- der Knowledge Discovery in Databases (KDD) Kreislauf (vgl. [Fayyad et al., 1996](#)).

Eine ausführliche Diskussion verschiedener DS-Modelle ist bei [Costa und Aparicio \(2020\)](#) zu finden. In dieser Arbeit wird der Fokus auf das CRISP-DM Modell gelegt, da das Lernmaterial nah an diesem Modell entwickelt wurde. Das CRISP-DM Modell setzt sich aus den folgenden sechs Phasen zusammen, die auch in [Abbildung 14.1](#) dargestellt sind (vgl. [Chapman et al., 2000](#), S. 13 f.):

1. **Business understanding:** Verstehen der Ziele und der Aufgabenstellung des Projektes; Entwicklung eines vorläufigen Plans, um Ziele zu erreichen

---

<sup>1</sup>Kurz gesagt beschreibt Data-Mining den Prozess, Wissen aus meist großen Datenmengen zu gewinnen. Dabei kommen vielfach Methoden aus Statistik und ML zum Einsatz.

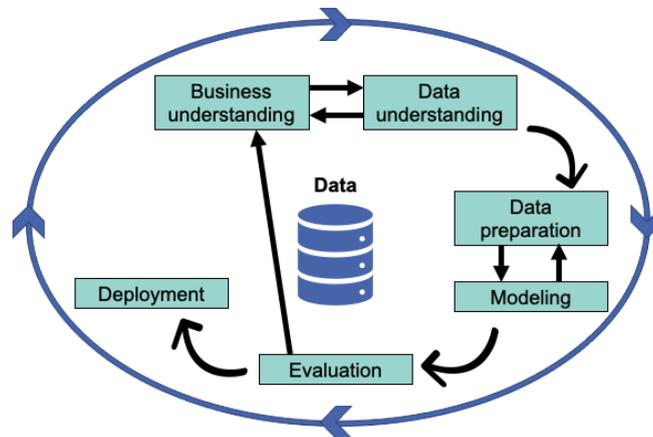


Abb. 14.1: CRISP-DM Modell angelehnt an Chapman et al. (2000, S. 13)

2. **Data understanding:** Erste Datenerhebung und Sichten der verfügbaren Daten; Probleme mit der Datenqualität erkennen; erste Einblicke in die Daten gewinnen
3. **Data preparation:** Vorbereitung der Daten (unter anderem Bereinigen und Transformieren), um finalen Datensatz für die Modellierung zu generieren
4. **Modeling:** Anwendung (verschiedener) geeigneter Modellierungstechniken; optimale Bestimmung von Parametern
5. **Evaluation:** Umfassende Validierung des Modells vor dessen Einsatz in der wirtschaftlichen Anwendung (Wurden gegebenenfalls wichtige Anwendungsaspekte nicht ausreichend berücksichtigt?); Entscheidung ob / wie Ergebnisse verwendet werden
6. **Deployment:** Gewonnenes Wissen so aufbereiten und präsentieren, dass Kundinnen und Kunden dieses nutzen können; Ergebnisse (auch für Laien im Data Mining-Bereich) zugänglich machen

Die sechs Phasen werden nicht stets in dieser Reihenfolge durchlaufen. Vielfach ist es (wie auch im Modellierungskreislauf) notwendig, mehrfach zwischen einzelnen Phasen hin und her zu wechseln. Die Pfeile in Abbildung 14.1 halten lediglich die wichtigsten beziehungsweise häufigsten Schritte fest (vgl. Chapman et al., 2000, S. 13). Hier sei vorweggenommen, dass im entwickelten Lernmaterial für die stärker angeleiteten Projekte (Modellierungstage und Unterrichtsreihen, vgl. Abschn. 17.2.1) insbesondere die Schritte 1, 2, 4 und 5 zentral sind. Die Datenvorbereitung (Schritt 2) findet im Hintergrund, das heißt für die Schülerinnen und Schüler nicht sichtbar, statt. Die Bereitstellung (Schritt 6) spielt im Rahmen der Sicherungsphasen eine Rolle. In diesen präsentieren die Lernenden ihre Ergebnisse mündlich (ggf. unterstützt durch Projektion des bearbeiteten Arbeitsblattes). Aufgrund des zeitlich begrenzten Rahmens wird die Erstellung einer Präsentation, eines Berichts oder gar einer App / Webseite, auf der das Empfehlungssystem bereitgestellt wird, bei den stärker angeleiteten Projekten (Modellierungstage) nicht gefordert. Bei dem offenen, unangeleiteten Veranstaltungsformat (Modellierungswoche, vgl. Abschn. 17.2.2) ist

dies hingegen durchaus möglich. Hier findet die Bereitstellung (Schritt 6) in Form eines Berichts und einer Präsentation ausgeprägter statt.

Wird das CRISP-DM Modell mit dem siebenschrittigen Kreislauf von [Blum und Leiß \(2005\)](#), einem gängigen Metamodell zur Beschreibung des mathematischen Modellierungsprozesses (vgl. [Abb. 1.3](#)), verglichen, so werden schnell gewisse Parallelen erkennbar. Beide Metamodelle beschreiben iterative Prozesse, die nichtlinear verlaufen. Eine mögliche Zuordnung der einzelnen Schritte des CRISP-DM Modells zu den Schritten des siebenschrittigen mathematischen Modellierungskreislaufs ließe sich wie folgt vornehmen: Die Schritte 1 bis 3 des CRISP-DM Modells können dem Schritt des *Verstehens* im mathematischen Modellierungskreislauf zugeordnet werden. Schritt 4 (Modeling) des CRISP-DM Modells umfasst die Schritte *vereinfachen und strukturieren*, *mathematisieren* und *mathematisch arbeiten* des Modellierungskreislaufs. Schritt 5 des CRISP-DM Modells lässt sich dem Schritt *Validieren* und *Interpretieren* und der Schritt 6 dem *Vermitteln* zuordnen.

Durchaus wären vergleichende Diskussionen von weiteren Metamodellen, die zur Beschreibung von Modellierungs- beziehungsweise DS-Prozessen eingesetzt werden, interessant. Hierzu sei auf [Sube \(2019, S. 14 f.\)](#) verwiesen, die Unterschiede und Ähnlichkeiten des PPDAC-Zyklus und des KDD-Kreislaufs mit dem vierschriftigen Modellierungskreislauf von [Blum \(1985\)](#) diskutiert.

Da Daten mittlerweile in jeglichen Bereichen unseres Lebens (Technik, Wissenschaft, Kultur) eine zentrale Rolle spielen, scheint es wesentlich, Lernenden einen verständigen Zugang zu DS-Fragestellung zu erlauben. Lernende sollen zur Teilhabe als mündige Bürgerinnen und Bürger an unserer datenlastigen Gesellschaft gefördert und somit insbesondere im kritischen Umgang mit Daten im Sinne einer umfassenden Datenkompetenz geschult werden. Dies wird sowohl in der Literatur unter anderem von [Engel \(im Druck\)](#), [Gould \(2021\)](#), [Opel et al. \(2019\)](#) als auch von verschiedenen Initiativen wie dem deutschen Stifterverband gefordert. Letzterer hat eine Data-Literacy-Charta herausgebracht, die von zahlreichen Unternehmen, Forschungsinstituten und Privatpersonen (aus Wirtschaft und Forschung) unterstützt wird und mit der „ein gemeinsames Verständnis von Datenkompetenzen und deren Bedeutung für Bildungsprozesse formuliert“ werden soll ([Schüller et al., 2021, S. 1](#)). In dieser Charta heißt es:

„Data Literacy umfasst die Datenkompetenzen, die für alle Menschen in einer durch Digitalisierung geprägten Welt wichtig sind. Sie ist unverzichtbarer Bestandteil der Allgemeinbildung. [...] Data Literacy umfasst die Fähigkeiten, Daten auf kritische Art und Weise zu sammeln, zu managen, zu bewerten und anzuwenden. Wenn Daten Entscheidungsprozesse unterstützen sollen, braucht es kompetente Antworten auf vier grundlegende Fragen:

- **Was will ich mit Daten machen?** Daten und Datenanalysen sind kein Selbstzweck, sondern dienen einer konkreten Anwendung in der realen Welt.

- **Was kann ich mit Daten machen?** Datenquellen und deren Qualität sowie der Stand der technischen und methodischen Entwicklungen eröffnen Möglichkeiten und setzen Grenzen.
- **Was darf ich mit Daten machen?** Alle gesetzlichen Regeln der Datennutzung (z. B. Datenschutz, Urheberrechte und Lizenzfragen) müssen immer mitbedacht werden.
- **Was soll ich mit Daten machen?** Weil Daten eine wertvolle Ressource darstellen, leitet sich daraus ein normativer Anspruch ab, sie zum Wohl von Individuen und Gesellschaft zu nutzen“ (Schüller et al., 2021, S. 1).

Im entwickelten Lernmaterial zum Thema Empfehlungssysteme werden die in der Charta genannten Fragen unterschiedlich detailliert aufgegriffen und mit den Lernenden thematisiert.

Weitere Vorschläge für Unterrichtsmaterial rund um DS und mathematische Modellierung sind beispielsweise bei Sube et al. (2020) und Sube (2019) zu finden. Erste Vorschläge für die Entwicklung von DS-Curricula werden von den Projekten ProDaBi,<sup>2</sup> IDSSP<sup>3</sup> oder IDS<sup>4</sup> vorangetrieben.

### Maschinelles Lernen – Was ist das?

Auch auf die Frage *Was ist ML?* gibt es nicht **die eine Antwort**. Je nachdem von welchem Fachbereich ausgehend recherchiert wird, finden sich diverse Antworten und Ansichten darauf, was ML ist oder was es im Kern auszeichnet: von „ML ist Statistik“ über „ML ist Optimierung“ oder „ML ist das Fitten von Daten“ bis hin zu „ML ist die Entwicklung intelligenter Computerprogramme.“

Eine Beschreibung, die in der Literatur besonders oft zitiert wird, stammt von Mitchell (1997, S. 2):

„A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.“

Am Beispiel von Email-Spamfiltern konkretisiert Mitchell dies wie folgt:

„For example, in learning an email spam filter the task T is to learn a function that maps from any given input email to an output label of spam or not-spam. The performance metric P to be improved might be defined as the accuracy of this spam classifier, and the training experience E might consist of a collection of emails, each labeled as spam or not“ (Mitchell, 2017, S. 2).

---

<sup>2</sup>Projekt Data Science und Big Data in der Schule, weitere Informationen unter: [www.prodabi.de](http://www.prodabi.de), letzter Aufruf: 19.11.2021.

<sup>3</sup>International Data Science in Schools Project, weitere Informationen unter: [www.idssp.org](http://www.idssp.org), letzter Aufruf: 19.11.2021.

<sup>4</sup>Introduction to Data Science Project, weitere Informationen unter: [www.mobilizingcs.org](http://www.mobilizingcs.org), letzter Aufruf: 19.11.2021.

In dieser Arbeit wird nicht versucht *die eine Definition* von ML zu liefern. Stattdessen wird die Auffassung von ML hervorgehoben, die im Laufe der Arbeit relevant ist: ML als eine Disziplin, die sich unterschiedlicher Werkzeuge (insbesondere mathematischer und informatischer Natur) bedient, um ein System beziehungsweise ein Modell zu entwickeln, mit dem eine spezielle<sup>5</sup> Aufgabe erfüllt werden kann. Das gesuchte Modell wird dabei nicht durch komplexe Regelsysteme definiert, die von Expertinnen und Experten festgelegt werden.<sup>6</sup> Stattdessen kommen gegebene Daten und Informationen zum Einsatz, um gewisse Modellparameter bestmöglich festzulegen beziehungsweise zu *lernen*. Diese Grundidee soll hier an einer zentralen Strategie des ML, dem **überwachten Lernen**, verdeutlicht werden. Bei dieser Art des Lernens sind nicht nur die Eingangsdaten (*Input*) gegeben, sondern auch die zugehörigen Ergebnisse (*Output*). Am Beispiel der Klassifizierung von Verkehrschildern könnten die Eingangsdaten Bilder von Stoppschildern (Klasse 1) und von Vorfahrtsschildern (Klasse 2) sein. Der zugehörige Output ist die Klasse eines jeden Bildes. Typischerweise wird der gesamte Datensatz unterteilt in einen **Trainings-** und einen **Testdatensatz**. Basierend auf den Trainingsdaten wird ein Modell gelernt. Das Modell, meist eine Funktion, soll den Zusammenhang zwischen bekannten Eingangsdaten und den Outputs bestmöglich beschreiben. In der Regel wird das Modell nicht im Raum aller möglicher Funktionen gesucht. Stattdessen wird der Suchraum basierend auf Domänenwissen oder aus Komplexitätsgründen auf Klassen von Funktionen eingeschränkt. Diese Funktionen hängen üblicherweise von gewissen Parametern ab, die in der Lernphase **optimal** gewählt werden sollen. Die durch das Lösen eines Optimierungsproblems gefundenen optimalen Parameter legen das Modell fest. Anschließend werden die Testdaten genutzt, um zu validieren, wie gut das Modell für die Anwendung auf „unbekannte“ Daten geeignet ist. Dazu wird das Modell auf die Testdaten angewandt und das erhaltene Ergebnis mit dem tatsächlichen Output verglichen. Ziel ist es, das gelernte Modell dann zur Klassifizierung von unbekanntem Daten, die sich noch nicht im Trainings- oder Testdatensatz befinden (d. h. von Bildern, bei denen noch nicht bekannt ist, welches der beiden Verkehrschilder abgebildet ist), anwenden zu können.

Neben dem überwachten Lernen sind wesentliche weitere Strategien des ML das **unüberwachte** und das **bestärkende Lernen**. In dieser Arbeit wird eine Methode des überwachten Lernens thematisiert.

Dass das Formulieren und Lösen von Optimierungsproblemen zentraler Bestandteil von ML-Methoden ist, wird bei der folgenden Beschreibung eines Empfehlungssystems ersichtlich. Als weiteres Beispiel sei hier die sogenannte Stützvektormethode

---

<sup>5</sup>Ein KI- beziehungsweise ML-System kann bisher lediglich die Aufgaben erfüllen, für die es initial entwickelt wurde. Eine KI, die entwickelt wurde, um Bilder zu klassifizieren, kann folglich nicht zum Übersetzen von Texten oder zum Go Spielen eingesetzt werden. Man spricht deswegen auch von schwacher KI (vgl. Kleesiek et al., 2020, S. 24).

<sup>6</sup>Ein Expertensystem könnte am Beispiel des Spamfilters aus einer komplexen Aneinanderreihungen von Regeln bestehen: „Wenn die Mail mindestens drei kritische Worte enthält *und* wenn Bedingung 2 erfüllt ist *oder* wenn Bedingung 3 *und* ..., *dann* klassifiziere die Mail als Spam.“ Derartige Systeme sind meist hartkodiert und lassen wenig bis keinen Spielraum für eine flexible Anpassung an neue Daten und Informationen (vgl. Hardt und Recht, 2021, S. 6). Expertensysteme (auch regelbasierte Systeme genannt) werden ebenfalls als KI-Systeme bezeichnet, jedoch nicht als *lernende* Systeme.

(SVM) benannt, die unter anderem für Klassifizierungsprobleme eingesetzt wird. Bei dieser wird versucht, die Parameter einer Gerade / Ebene / Hyperebene zu bestimmen, die gegebene Datenpunkte verschiedener Klassen **bestmöglich** trennt. Das Modell basiert auf der Lösung eines konvexen quadratischen Optimierungsproblems unter Nebenbedingungen. Wie sich Klassifizierungsprobleme basierend auf der SVM mit Lernenden angehen lassen, wird bei [Schönbrodt \(2019\)](#), [Schmidt \(2019\)](#) und [Schönbrodt et al. \(2021\)](#) beschrieben. Auch künstliche Neuronale Netze (KNN) fußen letztlich auf einem Optimierungsproblem. Gewisse Parameter sollen so bestimmt werden, dass eine gewählte Fehlerfunktion minimal wird. Möglichkeiten der Umsetzung von Unterrichtssequenzen zu KNNs werden in [Opel et al. \(2019\)](#) und [Janssen \(2020\)](#) diskutiert.

# 15 Basiswissen zu Empfehlungssystemen

Netflix, Amazon und zahlreiche weitere e-Commerce Unternehmen versuchen ihren Nutzerinnen und Nutzern durch individuelle Empfehlungen für neue Produkte, seien es Filme, Bücher oder Kleidung, den Entscheidungsprozess für zukünftige Kauf- oder Konsumententscheidungen zu erleichtern. Durch eine angenehme Nutzererfahrung beziehungsweise ein leichtes Zurechtfinden auf der Plattform sollen die Nutzerinnen und Nutzer an die Unternehmen gebunden, ihr Konsum erhöht und letztlich der Profit des Unternehmens erhöht werden (vgl. [Aggarwal, 2016](#), S. 3). Die Softwaresysteme, die zu diesem Zweck Einsatz finden, werden Empfehlungssysteme genannt (vgl. [Klahold, 2009](#), S. 1; [Jannach et al., 2010](#), S. xiii):

Ein Empfehlungssystem (engl. *Recommender System*) ist ein Softwaresystem, welches das Ziel hat, Nutzerinnen und Nutzer in einem gegebenen Kontext bei der Entscheidungsfindung zu unterstützen. Dazu werden der Nutzerin oder dem Nutzer aus einer Menge an vorhandenen Objekten diejenigen empfohlen, die für diese oder diesen sehr wahrscheinlich am nützlichsten oder interessantesten sind.

Die Anwendungsbereiche von Empfehlungssystemen sind divers. Sie reichen von personalisierten Produktempfehlungen (Amazon, Zalando) über Textempfehlungen (Nachrichtendienste wie Google News (vgl. [Das et al., 2007](#)) oder Twitter), Empfehlungen für Musik (Spotify) oder für Videos und Filme (Netflix, YouTube) bis hin zu Empfehlungen für neue Freundschaften oder Kontakte auf sozialen Netzwerken (facebook, LinkedIn) oder Orte, die besucht werden sollten (Tripadvisor) (vgl. [Ricci et al., 2011](#), S. 3).

Bei der Entwicklung von Empfehlungssystemen werden verschiedene Ziele verfolgt: Die empfohlenen Produkte sollen in erster Linie **relevant** für den betrachteten Nutzenden sein. Sie sollten tatsächlich **neu** sein, die nutzende Person sollte sie also wenn möglich noch nicht konsumiert haben. Die Empfehlungen sollten auch **überraschende und dennoch passende** Produkte beinhalten (vgl. [Aggarwal, 2016](#), S. 4).

## 15.1 Arten von Empfehlungssystemen

Drei wesentliche Arten von Empfehlungssystemen, die sich insbesondere darin unterscheiden, welche Daten und Informationen zur Entwicklung des Empfehlungssystems ausgenutzt werden, sind **Methoden des kollaborativen Filterns**, **inhaltsbasierte Methoden** und **wissensbasierte Methoden**. Alle drei Ansätze werden

im Folgenden beschrieben. Der Fokus liegt auf dem kollaborativen Filtern, da das entwickelte Lehr- und Lernmaterial auf einer solchen Methode beruht. Neben den hier beschriebenen Ansätzen für Empfehlungssysteme gibt es weitere Methoden, die lediglich kurz angerissen werden.

### 15.1.1 Methoden des kollaborativen Filterns

Methoden des kollaborativen Filterns „nutzen die kollaborative Macht von Bewertungen zahlreicher Nutzender aus, um Empfehlungen auszusprechen“ (Aggarwal, 2016, S. 8, eigene Übersetzung). Mit anderen Worten wird Wissen beziehungsweise Daten über die vergangene Interaktion von Nutzenden mit gewissen Artikeln (im Folgenden Items genannt), beispielsweise deren vergangenes Bewertungs- oder Kaufverhalten, ausgenutzt.

Grundannahme von Methoden des kollaborativen Filterns ist, dass das beobachtete Nutzerverhalten, zum Beispiel die Nutzerbewertungen, über zahlreiche Nutzende und Items hinweg stark korrelieren. Es wird davon ausgegangen, dass Nutzende, die in der Vergangenheit ein ähnliches Verhalten oder Interesse gezeigt haben, auch zukünftig Interesse an ähnlichen Produkten haben werden (vgl. Jannach et al., 2010, S. 2; Aggarwal, 2016, S. 9).

Ausgangspunkt für die Entwicklung von Methoden des kollaborativen Filterns sind bereits beobachtete beziehungsweise bekannte Bewertungen (impliziter oder expliziter Natur, vgl. Abschn. 15.1.1.1), die in einer User-Item-Matrix<sup>1</sup>, etwa

$$\begin{array}{c}
 \text{User}_1 \\
 \text{User}_2 \\
 \text{User}_3 \\
 \text{User}_4 \\
 \vdots
 \end{array}
 \begin{pmatrix}
 \text{Item}_1 & \text{Item}_2 & \text{Item}_3 & \text{Item}_4 & \text{Item}_5 & \text{Item}_6 & \dots \\
 * & ? & * & ? & * & * & \dots \\
 ? & * & * & * & ? & * & \dots \\
 * & * & ? & * & ? & ? & \dots \\
 * & * & ? & ? & * & ? & \dots \\
 \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots
 \end{pmatrix}$$

zusammengefasst werden können. Hierbei steht \* für eine Bewertung aus einer gegebenen Bewertungsskala.

Aufgabe ist es nun, für beliebige User-Item-Paare die unbekanntes beziehungsweise fehlenden Bewertungen bestmöglich vorherzusagen. Das Problem kann damit auch als Matrix-Vervollständigungsproblem (engl. *matrix completion problem*) betrachtet werden (vgl. Aggarwal, 2016, S. 3). Das Empfehlungssystem liefert dann die Möglichkeit für einzelne User-Item-Paare eine Aussage zu treffen, inwieweit die Nutzerin oder der Nutzer das betrachtete Item mögen wird (vgl. Jannach et al., 2010, S. 14).

Methoden des kollaborativen Filterns lassen sich in **modellbasierte Methoden** und **speicherbasierte Methoden**, die auch als **Nachbarschaftsmethoden** bezeichnet werden, unterteilen.

<sup>1</sup>Die Bewertungsdaten sind vielfach spärlich vorhanden und die Matrix damit dünnbesetzt (vgl. Abschn. 15.1.1.2). Deswegen werden die Daten typischerweise nicht als Matrix gespeichert. Stattdessen werden die Position und der Wert der bekannten Einträge gespeichert.

- **Modellbasierte Methoden:** Bei den modellbasierten Empfehlungssystemen kommen Methoden des MLs und des Data Minings zum Einsatz. Diesen Methoden liegt vielfach ein parametrisiertes Modell zu Grunde, dessen optimale Parameterwerte basierend auf den gegebenen (Bewertungs-)Daten gelernt werden sollen. Bei diesen Methoden ist das Formulieren und Lösen eines Optimierungsproblems zentral. Wichtige modellbasierte Methoden sind Clusterbildung, Bayes'sche Netze, Entscheidungsbäume, Neuronale Netze und latente Faktormodelle (vgl. [Aggarwal, 2016](#), S. 9; [Klahold, 2009](#), S. 65).
- **Speicherbasierte Methoden:** Bei den Nachbarschaftsmethoden ist die Definition eines geeigneten Maßes, mit dem die Ähnlichkeit zwischen Nutzenden oder zwischen Items gemessen werden kann, zentral. Basierend auf dem gewählten Ähnlichkeitsmaß wird eine Menge an ähnlichen Nutzerinnen und Nutzern beziehungsweise ähnlichen Items bestimmt: die sogenannte Nachbarschaft. In der Regel wird dann basierend auf den  $k$  ähnlichsten Nutzerinnen und Nutzern / Items vorhergesagt, wie ausgewählte Nutzende ein gewisses Item bewerten würden (vgl. [Aggarwal, 2016](#), S. 9). Wird das zukünftige Interesse einer Nutzerin oder eines Nutzers für ein ausgewähltes Item durch eine Nachbarschaft des Items modelliert, spricht man von itembasierten Methoden. Wird hingegen die Nachbarschaft einer Nutzerin oder eines Nutzers bestimmt, spricht man von nutzerbasierten Methoden. Die Nachbarschaftsmethoden werden in Abschnitt 16.3 ausführlicher beschrieben.

Methoden des kollaborativen Filterns benötigen keine Informationen über die einzelnen Produkte (bspw. das Genre eines Filmes) oder die einzelnen Nutzenden (bspw. Alter oder Geschlecht). Dies ist ein wesentlicher Vorteil dieser Methoden, da derartige Informationen nicht immer vollständig gegeben, nicht korrekt oder schwer zu erheben sind.

#### 15.1.1.1 Arten von Bewertungsdaten

Bei der Entwicklung von Empfehlungssystemen basierend auf Methoden des kollaborativen Filterns finden im Wesentlichen zwei verschiedene Arten von Nutzerfeedback Einsatz:

- Eine Möglichkeit ist, die Nutzenden **explizite Bewertungen** auf einer gegebenen Skala abgeben zu lassen. Dabei kommen sowohl Likert-Skalen mit zahlreichen Stufen, wie die fünfstufige-Skala im Falle des von Netflix veröffentlichten Datensatzes, als auch binäre Skalen (like und dislike) zum Einsatz. Ein Nachteil der Verwendung expliziter Bewertungen ist, dass von der nutzenden Person eine zusätzliche Aktion verlangt wird, der sie gegebenenfalls nicht nachkommen möchte (vgl. [Jannach et al., 2010](#), S. 23).
- Eine Alternative (oder Ergänzung) stellt die Verwendung von **impliziten Bewertungen** dar. Hierbei wird indirekt auf das Interesse der Nutzenden geschlossen, indem beispielsweise das Kaufen eines Items, das Anschauen eines Filmes oder eine lange Verweildauer als positive Bewertung betrachtet wird.

Dabei ist zu bedenken, dass diese Annahme nicht stets erfüllt sein muss. Durchaus kommt es vor, dass Nutzende einen Film anschauen oder ein Kleidungsstück bestellen, welches ihnen letztlich doch nicht gefallen hat. Wenngleich das implizite Feedback der Nutzenden schwieriger zu interpretieren ist, so kann es eine wertvolle Information sein, um Nutzenden, die kein oder wenig explizites Feedback abgeben, dennoch Empfehlungen aussprechen zu können. Eine Möglichkeit, wie explizite und implizite Bewertungen kombiniert werden können beschreiben unter anderem [Koren und Bell \(2011, S. 153\)](#).

### 15.1.1.2 Spärlichkeit der Daten und das Kaltstartproblem

Zentrale Herausforderungen von Systemen, die auf dem vergangenen Nutzerverhalten beruhen, sind das Kaltstartproblem und der Umgang mit Spärlichkeit (engl. *sparsity*) der Daten. Beide Herausforderungen werden hier kurz erläutert:

- Vielfach finden kollaborative Filtersysteme in Kontexten Anwendung, in denen eine große Anzahl Items zur Verfügung stehen, von denen die Nutzenden lediglich einen kleinen Bruchteil bewertet haben. Dies führt dazu, dass die Mehrheit der Bewertungen nicht bekannt ist. Man spricht in diesem Fall auch von einem **spärlichen Datensatz**. Die Herausforderung besteht darin, dass möglichst präzise Vorhersagen für zukünftige Bewertungen bestimmt werden sollen, obgleich nur relativ wenige Bewertungen bekannt sind (vgl. [Jannach et al., 2010, S. 23 f.](#)).
- Unter dem **Kaltstartproblem**, welches auch als Spezialfall der Spärlichkeit betrachtet werden kann, versteht man die Schwierigkeit, dass bei Hinzunahme einer neuen Nutzerin oder eines neuen Nutzers zu einem bestehenden System noch keine oder nicht ausreichend viele Bewertungen vorliegen, um ihm eine Empfehlung aussprechen zu können. Dies gilt ebenso für die Hinzunahme von neuen Items oder bei der Entwicklung eines ganz neuen Empfehlungssystems (vgl. [Klahold, 2009, S. 66](#); [Jannach et al., 2010, S. 26](#)).

Eine Möglichkeit, um beiden Problemen zu begegnen, ist die Kombination von kollaborativen Filtermethoden mit anderen Methoden, bei denen zusätzliche Informationen abgefragt und ausgenutzt werden.

### 15.1.2 Weitere Methoden für Empfehlungssysteme

Neben den Methoden des kollaborativen Filterns, die in dieser Arbeit untersucht werden, existieren verschiedene weitere gängige Methoden:

#### **Inhaltsbasierte Methoden**

Bei inhaltsbasierten Methoden spielen die Eigenschaften beziehungsweise Beschreibungen der jeweiligen Items eine zentrale Rolle. Diese Beschreibungen werden mit dem bisherigen Nutzerverhalten kombiniert, um Empfehlungen für neue Produkte auszusprechen. Im Falle von Empfehlungssystemen für Filme könnten wichtige

Informationen das Erscheinungsjahr, das Genre oder die Hauptdarstellenden sein. Diese Art der Systeme sind besonders hilfreich, wenn zwar das vergangene Verhalten einer Nutzerin oder eines Nutzers bekannt ist, jedoch keine Informationen über das Verhalten anderer Nutzerinnen und Nutzer vorliegen (vgl. [Aggarwal, 2016](#), S. 14). Hat die betrachtete nutzende Person beispielsweise in der Vergangenheit insbesondere Actionfilme mit Bruce Willis in der Hauptrolle gut bewertet, so liegt es nahe weitere, noch ungesehene Filme zu empfehlen, in denen Bruce Willis mitspielt.

Ein Vorteil von inhaltsbasierten Methoden gegenüber Methoden des kollaborativen Filterns ist, dass keine riesigen Datenmengen von zahlreichen Nutzenden notwendig sind, um Empfehlungen auszusprechen. Ein weiterer Vorteil ist, dass bei diesen Systemen auch neue Items leicht integriert werden können, wenn die entsprechenden Informationen über die Attribute des Items vorliegen. Ein Nachteil ist, dass diese Systeme meist offensichtliche Empfehlungen generieren – also solche, die zu den Attributen passen, die die nutzende Person in der Vergangenheit konsumiert hat (vgl. [Aggarwal, 2016](#), S. 15). Ein weiterer Nachteil dieser Methoden ist, dass zentrale Attribute der Items nicht notwendigerweise vorliegen oder nicht automatisch generiert werden können. Diese fehlenden Informationen müssen dann manuell und damit zeitaufwendig sowie fehleranfällig hinzugefügt werden (vgl. [Jannach et al., 2010](#), S. 4).

### **Wissensbasierte Methoden**

Wissensbasierte Methoden nutzen, wie die inhaltsbasierten Methoden, Informationen über die Charakteristika der Items. Diese Methoden basieren jedoch nicht auf dem vergangenen beobachteten Nutzerverhalten. Stattdessen werden zusätzliche Informationen über spezifische Eigenschaften oder Anforderungen der Nutzerin oder des Nutzers erhoben und mit den Eigenschaften der Items kombiniert. Bei diesen Systemen hat die nutzende Person die Möglichkeit explizit anzugeben, was sie mag (vgl. [Aggarwal, 2016](#), S. 16). Solche Methoden sind in Bereichen nützlich, in denen Produkte sehr selten gekauft / konsumiert werden. Als Beispiele seien Empfehlungssysteme für den nächsten Autokauf oder die nächste Immobilie genannt (vgl. [Aggarwal, 2016](#), S. 15). Auch lassen sich diese Methoden (in Kombination mit anderen Methoden) einsetzen, um dem Kaltstartproblem zu begegnen. Am Beispiel eines Empfehlungssystems für Filme könnten die Nutzenden bei der erstmaligen Registrierung auf einer Streaming-Plattform beispielsweise angeben, welche Genres sie mögen. Den neuen Nutzenden können dann Filme aus diesem Genre vorgeschlagen werden.

### **Hybride Methoden**

In der Praxis werden vielfach unterschiedliche Methoden für Empfehlungssysteme kombiniert, um die jeweiligen Vorzüge der einzelnen Methoden verbindend auszunutzen. Beispielsweise ließen sich Methoden des Kollaborativen Filterns mit wissensbasierten Methoden kombinieren, um dem Kaltstartproblem bei neuen Nutzerinnen und Nutzern zu begegnen. Neben der Frage danach, welche Methoden sich sinnvoll kombinieren lassen, ist zu entscheiden, wie die Ergebnisse aus verschiedenen Methoden gewichtet und ob die Methoden sequentiell oder parallel eingesetzt werden.

Eine ausführliche Diskussion unterschiedlicher Hybridisierungsansätze findet sich bei [Jannach et al. \(2010, S. 124 ff.\)](#). Eine Übersicht über die beschriebenen Methoden liefert, angelehnt an [Aggarwal \(2016, S. 16\)](#), Tabelle 15.1.

**Tab. 15.1:** Klassifizierung von Empfehlungssystemen angelehnt an [Aggarwal \(2016, S. 16\)](#)

Methoden	Grundidee	Verwendete Daten
Kollaboratives Filtern	Wissen über vergangenes Verhalten der betrachteten und von zahlreichen weiteren nutzenden Personen ausnutzen, um Empfehlungen zu generieren	Bewertungen / Daten zum Verhalten der betrachteten nutzenden Person und der gesamten Nutzerschaft
Inhaltsbasierte Methoden	Wissen über vergangenes Verhalten der betrachteten nutzenden Person und Infos (Attribute) der Items ausnutzen, um Empfehlungen zu generieren	Bewertungen / Daten zum Verhalten der betrachteten nutzenden Person und Attribute der Items
Wissensbasierte Methoden	Empfehlungen basierend auf den expliziten Angaben / Wünschen der nutzenden Person für genau die Items aussprechen, deren Attribute diesen Angaben entsprechen	Spezifikationen der nutzenden Person und Attribute der Items
Hybride Methoden	Methoden von oben genannten Empfehlungssystemen werden entweder sequentiell oder parallel kombiniert	siehe oben

# 16 Entwicklung eines Empfehlungssystems

In diesem Kapitel liegt der Fokus auf der Beschreibung des mathematischen Modells eines Empfehlungssystems für Filme. Der Schwerpunkt des beschriebenen Modells und auch des konzipierten Lernmaterials liegt auf Inhalten der linearen Algebra (Vektoren, Skalarprodukt, Matrizen). Dennoch greift das Modell auch Inhalte aus der Analysis und insbesondere der mathematischen Optimierung auf.

In Anhang E.3 werden verschiedene numerische Experimente auf einem Teil des originalen Datensatzes des Netflix Preises beschrieben. Diese verdeutlichen, wie sich das gewählte Modell, dessen Parameter und verschiedene Modifikationen des zugrundeliegenden Optimierungsproblems auf die Güte des Empfehlungssystems auswirken.

## 16.1 Der Netflix Preis und der zugrundeliegende Datensatz

Im Jahr 2006 schrieb der Streaming-Dienst Netflix, der damals noch als postalischer DVD-Verleihservice agierte, einen Wettbewerb aus. Ziel dieses Wettbewerbs war die Entwicklung eines Modells, welches Vorhersagen konnte an welchen Filmen die Nutzerinnen und Nutzer zukünftig Interesse haben könnten. Genauer musste ein Empfehlungssystem entwickelt werden, welches Nutzerinteressen um 10 % besser vorhersagen konnte als Netflix' damaliges System. Im Zuge des Wettbewerbs veröffentlichte Netflix einen (theoretisch<sup>1</sup>) anonymen Datensatz von 480189 Nutzerinnen und Nutzern, 17700 Filmen und 100480507 zugehörigen Bewertungen von eins (schlechteste Bewertung) bis fünf (beste Bewertung), die die Nutzerinnen und Nutzer für die Filme abgegeben hatten (vgl. [Feuerverger et al., 2012](#), S. 202). Der Datensatz ist mit lediglich 1.18 % bekannten Bewertungen extrem dünn besetzt. Netflix veröffentlichte nicht nur den genannten Datensatz mit bekannten Bewertungen, sondern zudem einen Quiz-Datensatz mit Nutzerinnen und Nutzern sowie Filmen deren Bewertungen Netflix zwar kannte, die jedoch nicht öffentlich gemacht wurden. Das Team, dessen Empfehlungssystem die Bewertungen von ausgewiesenen Nutzerinnen und Nutzern im Quiz-Datensatz um 10 % (gemessen über die Wurzel

---

<sup>1</sup>Im Jahr 2008 zeigten [Narayanan und Shmatikov \(2008\)](#), zwei Forscher der Universität Texas, dass die De-Anonymisierung einzelner Nutzerinnen und Nutzer des Datensatzes durch Kombination mit einem weiteren öffentlich verfügbaren Filmdatensatz teilweise möglich ist. Dies führte sogar dazu, dass eine ursprünglich geplante zweite Auflage des Wettbewerbs abgesagt wurde (vgl. [Lohr, 2010](#)).

der gemittelten Fehlerquadratsumme, kurz RMSE<sup>2</sup>) genauer vorhersagen konnte als Netflix' eigenes System es vermochte, hatte Chancen auf den Hauptpreis von 1 Million US Dollar. Dieser Preis ging im Jahr 2009 an das Team *BellKor's Pragmatic Chaos* (vgl. [Feuerverger et al., 2012](#), S. 203).

Der Netflixdatensatz ist frei im Internet verfügbar<sup>3</sup> und wurde für die Entwicklung von digitalem Lernmaterial heruntergeladen und aufbereitet. Die nachfolgend vorgestellte und didaktisch reduzierte Methode einer Matrix-Faktorisierung wurde von dem siegreichen Team (in Kombination mit weiteren Methoden, die hier nicht beschrieben werden) umgesetzt.

Ausgangspunkt für die folgende Entwicklung des Empfehlungssystems ist die  $n_1 \times n_2$  Bewertungsmatrix  $R$ , bei der jede Zeile für eine Nutzerin / einen Nutzer und jede Spalte für einen Film steht. Hat Person  $i$  den Film  $j$  bereits gesehen und eine Bewertung abgegeben, so hat der Eintrag  $r_{ij}$  der Matrix  $R$  einen ganzzahligen Wert zwischen 1 und 5, welcher der abgegebenen Bewertung entspricht. Ein möglicher Auszug aus dem Datensatz könnte wie in der Bewertungsmatrix  $R$

$$R = \begin{matrix} & \begin{matrix} \text{🎬}_1 & \text{🎬}_2 & \text{🎬}_3 & \text{🎬}_4 & \text{🎬}_5 & \text{🎬}_6 & \dots \end{matrix} \\ \begin{matrix} \text{👤}_1 \\ \text{👤}_2 \\ \text{👤}_3 \\ \text{👤}_4 \\ \text{👤}_5 \\ \vdots \end{matrix} & \begin{pmatrix} 3 & ? & 1 & ? & 1 & 4 & \dots \\ ? & 2 & 4 & 1 & 3 & 1 & \dots \\ 3 & 1 & ? & 3 & ? & ? & \dots \\ 4 & 3 & ? & 4 & 4 & ? & \dots \\ 4 & ? & ? & 4 & ? & 5 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix} \end{matrix} \quad (16.1)$$

dargestellt, aussehen. In diesem Beispiel wurde vereinfacht ein weniger dünn besetzter Teil des Datensatzes betrachtet. Diese Modellvereinfachung wird auch in der Folge beibehalten, da Zusammenhänge in den Daten so leichter veranschaulicht werden können.

## 16.2 Matrix-Faktorisierungen als Modell zur Vorhersage von Bewertungen

Die Grundidee hinter dem entwickelten Empfehlungssystem ist eine Matrix-Faktorisierung: Die  $n_1 \times n_2$  Bewertungsmatrix  $R$  (im Folgenden auch Ratingmatrix genannt) mit fehlenden Einträgen soll durch ein Produkt  $U \cdot M$  aus zwei niedrigdimensionalen Matrizen  $U \in \mathbb{R}^{n_1 \times n_f}$  und  $M \in \mathbb{R}^{n_f \times n_2}$  approximiert werden. Mit anderen Worten werden die Vektoren, die die Nutzerinnen und Nutzer (Zeilen der

<sup>2</sup>RMSE steht hier für Root Mean Squared Error.

<sup>3</sup>Der Datensatz kann bspw. von Kaggle heruntergeladen werden: [www.kaggle.com/netflix-inc/netflix-prize-data](http://www.kaggle.com/netflix-inc/netflix-prize-data), letzter Aufruf: 19.11.2021.

Bewertungsmatrix) beziehungsweise Filme (Spalten der Bewertungsmatrix) repräsentieren, in einen niedrigdimensionalen Eigenschaftsraum<sup>4</sup> überführt (vgl. Hu et al., 2008, S. 32). Der Geschmack der Nutzenden und die Eigenschaften der Filme werden so basierend auf den bisherigen Bewertungen *aller Nutzerinnen und Nutzer* implizit modelliert. Die Methode fällt somit unter die Technik des kollaborativen Filterns (vgl. Koren et al., 2009, S. 31).

Bei der folgenden Beschreibung einer Matrix-Faktorisierungsmethode werden bereits einzelne Kommentare zur möglichen didaktischen Reduktion der Inhalte eingestreut. Die Umsetzung im Lernmaterial wird dann in Kapitel 17 beschrieben. Die bei der Modellierung des Empfehlungssystems verwendeten Variablen sind in Tabelle 16.1 zusammengefasst. Es sei darauf hingewiesen, dass einzelne Variablen, die bereits in den Teilen I und III verwendet wurden, mit neuer Bedeutung belegt werden.

**Tab. 16.1:** Variablen und Operatoren bei der Modellierung eines Empfehlungssystems

Symbol	Beschreibung
$n_1$	Anzahl der Nutzerinnen und Nutzer im Datensatz
$n_2$	Anzahl der Filme im Datensatz
$R$	$n_1 \times n_2$ Bewertungsmatrix
$r_{ij}$	Bewertung von Person $i$ für Film $j$
$\hat{r}_{ij}$	vorhergesagte Bewertung von Person $i$ für Film $j$
$\kappa$	Menge der Indizes $(i, j)$ für die die Bewertung $r_{ij}$ bekannt ist
$n_f$	Anzahl betrachteter Eigenschaften
$U$	Usermatrix mit $U \in \mathbb{R}^{n_1 \times n_f}$
$M$	Moviematrix mit $M \in \mathbb{R}^{n_f \times n_2}$
$\mathbf{u}_{i,\bullet}$	Zeilenvektor $i$ der Matrix $U$ mit $\mathbf{u}_{i,\bullet} \in \mathbb{R}^{1 \times n_f}$ (Uservektor)
$\mathbf{m}_{\bullet,j}$	Spaltenvektor $j$ der Matrix $M$ mit $\mathbf{m}_{\bullet,j} \in \mathbb{R}^{n_f \times 1}$ (Movievektor)
$\lambda$	Regularisierungsparameter
$W$	Gewichtsmatrix des gewichteten Niedrigrang-Matrixapproximationsproblems mit $W \in \mathbb{R}^{n_1 \times n_2}$
$n_i^{\text{user}}$	Anzahl bekannter Bewertungen von Person $i$
$n_j^{\text{movie}}$	Anzahl bekannter Bewertungen von Film $j$
$\mu$	Mittlere Bewertung des Trainingsdatensatzes
$o_i$	Bias von Person $i$
$p_j$	Bias von Film $j$
$\ \cdot\ $	Euklidische Norm / 2-Norm
$\ \cdot\ _F$	Frobenius-Norm

<sup>4</sup>Die Bezeichnung der Variablen  $n_f$  wurde in Anlehnung an die englischsprachige Fachliteratur gewählt. Das  $f$  leitet sich vom englischen Wort für Eigenschaften (*features*) ab. Es deutet die Interpretation der Zeilen von  $U$  bzw. Spalten von  $M$  als Eigenschaftsvektoren an. Dies wird im Folgenden deutlich.

### 16.2.1 Zusammenhänge in den Daten

Verschaffen wir uns zunächst einen Eindruck von den Daten und interessanten Zusammenhängen, die bei der Entwicklung eines Empfehlungssystems ausgenutzt werden können. Dazu wird die beispielhafte Bewertungsmatrix

$$R = \begin{pmatrix} 3 & 2 & 1 & 3 & 1 & 4 \\ 4 & 4 & 5 & 4 & 4 & 5 \\ 1 & 2 & 4 & 1 & 3 & 1 \\ 3 & 2 & 1 & 3 & 1 & 4 \end{pmatrix} \quad (16.2)$$

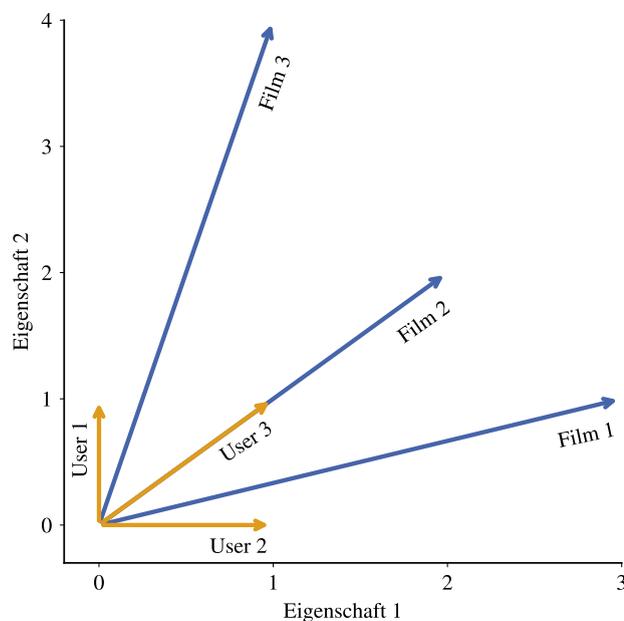
von vier Nutzerinnen und Nutzern sowie sechs Filmen, deren Bewertungen allesamt bekannt sind, betrachtet. Die grundlegende Idee zur Erklärung des Datensatzes ist, dass Filme mit ähnlicher Bewertung gewisse Ähnlichkeiten (bspw. das gleiche Genre, die gleichen berühmten Hauptdarstellenden o. ä.) aufweisen. Diese und weitere Zusammenhänge in der Bewertungsmatrix können ausgenutzt werden, um ein Modell zu entwickeln, das unbekannte Bewertungen von Nutzerinnen und Nutzern vorhersagt.

In einem ersten Schritt wird angenommen, dass sich die Filme leicht bezüglich zweier Eigenschaften einordnen lassen. In dem kleinen Beispiel könnten diese Eigenschaften die Genres **Action** (A) und **Comedy** (C) sein (vgl. [Serrano, 2018](#)). Angenommen im vorliegenden Fall haben die sechs Filme folgende Ausprägungen in Action und Comedy (1 = gering bis 5 = hoch)

$$M = \begin{matrix} & \begin{matrix} \text{🎬}_1 & \text{🎬}_2 & \text{🎬}_3 & \text{🎬}_4 & \text{🎬}_5 & \text{🎬}_6 \end{matrix} \\ \begin{matrix} A \\ C \end{matrix} & \begin{pmatrix} 3 & 2 & 1 & 3 & 1 & 4 \\ 1 & 2 & 4 & 1 & 3 & 1 \end{pmatrix} \end{matrix} \quad (16.3)$$

Diese Informationen werden in der Matrix  $M \in \mathbb{R}^{2 \times 6}$  gespeichert, die im Folgenden als Moviematrix bezeichnet wird. Betrachten wir die Bewertungen der Nutzerinnen und Nutzer in der Bewertungsmatrix  $R$  und die Ausprägungen der Filme bezüglich der beiden betrachteten Eigenschaften in der Moviematrix  $M$ , so sind Rückschlüsse auf den Geschmack der vier Nutzerinnen und Nutzer möglich: Es wird beispielsweise ersichtlich, dass Person 1 und 4 alle Filme höher bewertet haben, die eine höhere Action-Ausprägung haben. Filme mit höherer Comedy-Ausprägung haben diese beiden Personen hingegen schlecht bewertet. Bei Person 3 ist es genau umgekehrt. Person 2 scheint sowohl Filme mit höherer Action-Ausprägung als auch mit höherer Comedy-Ausprägung zu mögen. Wird der Geschmack zunächst als binäres Merkmal aufgefasst, so kann festgehalten werden, ob eine Person Comedy beziehungsweise Action mag (=1) oder nicht (=0). Diese Informationen werden in der Usermatrix  $U \in \mathbb{R}^{4 \times 2}$

$$U = \begin{matrix} & \begin{matrix} A & C \end{matrix} \\ \begin{matrix} \text{👤}_1 \\ \text{👤}_2 \\ \text{👤}_3 \\ \text{👤}_4 \end{matrix} & \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} \end{matrix} \quad (16.4)$$



**Abb. 16.1:** Darstellung von drei User- und drei Movievektoren im Eigenschaftsraum. Eigenschaften könnten beispielsweise Genres wie Action oder Comedy sein.

gespeichert (vgl. [Serrano, 2018](#)). In diesem Beispiel liefert das Produkt der Usermatrix  $U$  und der Moviematrix  $M$  gerade die Bewertungsmatrix  $R$ . Es gilt  $R = U \cdot M$ . Ist eine derartige Zerlegung der Bewertungsmatrix in  $U$  und  $M$  bekannt, so kann diese genutzt werden, um auch unbekannte Bewertungen durch die Berechnung des Skalarprodukts des entsprechenden Zeilenvektors der Usermatrix  $U$  und des Spaltenvektors der Moviematrix  $M$  vorherzusagen. Mit den Lernenden lässt sich anschaulich herleiten, wann beziehungsweise warum das Skalarprodukt größere oder kleinere vorhergesagte Bewertungen liefert. Dazu können die Zeilenvektoren der Usermatrix und die Spaltenvektoren der Moviematrix visualisiert werden (vgl. [Abb. 16.1](#)).

Bei der nachfolgenden Entwicklung eines Empfehlungssystems wird die Idee der Faktorisierung der Bewertungsmatrix weiter verfolgt. Dazu wird zunächst die Annahme verworfen, dass spezifische Eigenschaften der Filme (bspw. Action und Comedy) quantifiziert werden können. Stattdessen wird das Problem abstrakter betrachtet: Gesucht sind die Einträge von  $U$  und  $M$ , sodass das Produkt  $U \cdot M$  die gegebene Bewertungsmatrix  $R$  ausreichend gut beschreibt. Die (unspezifischen) Eigenschaften werden implizit modelliert. Auch wird im Folgenden nicht mehr davon ausgegangen, dass lediglich zwei Eigenschaften relevant sind. Stattdessen wird versucht, die Anzahl an Eigenschaften  $n_f$  so zu wählen, dass die Daten mit unserem Modell ausreichend gut beschrieben werden können.

Zusammengefasst sind im weiteren Verlauf die folgenden Fragen zu beantworten:

- Die Quantifizierung der Eigenschaften der Filme (bspw. bezüglich der Genres) und damit die Einträge der Matrix  $M$  sind nicht im Netflixdatensatz enthalten. Wie kann dennoch eine geeignete Faktorisierung der Bewertungsmatrix  $R$  in  $U$  und  $M$  bestimmt werden?

- Wie viele Eigenschaften sind relevant und sollten bei der Faktorisierung berücksichtigt werden?

Bevor im Folgenden ein Verfahren beschrieben wird, mit dem eine Zerlegung in zwei Matrizen  $U$  und  $M$  berechnet werden kann, wird zunächst ein Maß definiert, welches den Fehler zwischen der Bewertungsmatrix  $R$  und der Vorhersage gegeben durch das Produkt  $U \cdot M$  quantifiziert. Das gewählte Fehlermaß kann dann genutzt werden, um die Güte einer Zerlegung zu bewerten und damit zugleich verschiedene Zerlegungen vergleichen zu können.

### 16.2.2 Wahl einer Fehlerfunktion

Durchaus sind bei der Wahl eines Fehlermaßes sowohl in der Praxis als auch seitens der Lernenden verschiedene Ansätze denkbar. Zum einen ließe sich die absolute Abweichung<sup>5</sup> zwischen den einzelnen Bewertungen und der Vorhersage gemäß

$$\sum_{(i,j) \in \kappa} |r_{ij} - \mathbf{u}_{i,\bullet} \mathbf{m}_{\bullet,j}| \quad (16.5)$$

aufsummieren. Hierbei bezeichnet  $\kappa$  die Menge der Paare  $(i, j)$ , für die das Rating  $r_{ij}$  bekannt ist,  $\mathbf{u}_{i,\bullet} \in \mathbb{R}^{1 \times n_f}$  bezeichnet den Zeilenvektor  $i$  der Matrix  $U$  (Uservektor) und  $\mathbf{m}_{\bullet,j} \in \mathbb{R}^{n_f \times 1}$  den Spaltenvektor  $j$  der Matrix  $M$  (Movievektor).

Alternativ ließe sich der Fehler als Summe der Fehlerquadrate

$$\sum_{(i,j) \in \kappa} (r_{ij} - \mathbf{u}_{i,\bullet} \mathbf{m}_{\bullet,j})^2 \quad (16.6)$$

definieren. Beide Formulierungen unterscheiden sich nicht nur im Hinblick auf die Differenzierbarkeit der erhaltenen Fehlerfunktion, auch der Einfluss von statistischen Ausreißern wird unterschiedlich gehandhabt: So gewichtet die Summe der Fehlerquadrate (16.6) Ausreißer stärker als es die 1-Norm-Formulierung (16.5) tut. Die Wahl einer passenden Fehlerfunktion stellt eine (mathematisch) interessante Modellentscheidung dar, die insbesondere bei dem nachfolgend beschriebenen Optimierungsverfahren erneut aufgegriffen werden kann. Für die weitere Modellierung wird die Summe der Fehlerquadrate gewählt. Diese Entscheidung wird in der Folge ausführlicher diskutiert.

Die Suche nach einer passenden Faktorisierung kann nun als nichtlineares Optimierungsproblem ohne Nebenbedingungen

$$\min_{U,M} \sum_{(i,j) \in \kappa} (r_{ij} - \mathbf{u}_{i,\bullet} \mathbf{m}_{\bullet,j})^2 \quad (16.7)$$

formuliert werden. Es wird leicht ersichtlich, dass dieses Problem keine eindeutige optimale Zerlegung in  $U$  und  $M$  besitzt, da der Fehler für jede invertierbare Matrix  $S \in \mathbb{R}^{n_f \times n_f}$  und eine Zerlegung in  $\hat{U} = US$  und  $\hat{M} = S^{-1}M$  genauso groß ist wie für die Zerlegung in  $U$  und  $M$ .

<sup>5</sup>In dieser Arbeit bezeichnet die absolute Abweichung die betragsmäßige Abweichung. Dies ist nicht mit dem in der Numerik verwendeten Begriff eines absoluten Fehlers zu verwechseln, der auch negativ sein kann.

### 16.2.3 Gewichtete Niedrigrang-Matrixapproximation

Das Problem (16.7) wird auch als *gewichtetes Niedrigrang-Matrixapproximationsproblem* (engl. *weighted Low-Rank Approximation*, kurz WLRA) bezeichnet. Die Problemstellung eines WLRA besteht darin, für eine gegebene Matrix  $R$  und eine Gewichtsmatrix  $W \in \mathbb{R}^{n_1 \times n_2}$  mit nichtnegativen Einträgen, zwei Matrizen  $A$  und  $B$  zu finden, deren Produkt  $A \cdot B$  höchstens Rang  $n_f$  besitzt und für die

$$\|W \otimes (R - AB)\|_F^2 \quad (16.8)$$

minimal wird (vgl. Ban et al., 2019). Hier bezeichnet  $\otimes$  die elementweise Multiplikation und  $\|\cdot\|_F$  die Frobenius-Norm. Im Falle unseres Empfehlungssystems lautet das gewichtet Approximationsproblem

$$\min_{U, M} \|W \otimes (R - UM)\|_F^2 = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} (w_{ij} (r_{ij} - \mathbf{u}_{i,\bullet} \mathbf{m}_{\bullet,j}))^2 \quad (16.9)$$

$$= \sum_{(i,j) \in \kappa} (r_{ij} - \mathbf{u}_{i,\bullet} \mathbf{m}_{\bullet,j})^2, \quad (16.10)$$

wobei die Einträge  $w_{ij}$  der Matrix  $W$  gegeben sind durch

$$w_{ij} = \begin{cases} 1 & \text{falls } (i, j) \in \kappa \\ 0 & \text{sonst.} \end{cases}$$

Das klassische, ungewichtete Niedrigrang-Approximationsproblem ist ein Spezialfall des gewichteten Problems bei dem alle Gewichte 1 sind. Srebro und Jaakkola (2003) diskutieren, dass das gewichtete Problem nichtkonvex ist und zahlreiche (nicht globale) lokale Minimalpunkte aufweisen kann. Dort wird zudem gezeigt, dass im ungewichteten Fall, obgleich ebenfalls nichtkonvex, alle lokalen Minimalpunkte zugleich globale Minimalpunkte sind. Im ungewichteten Fall, das heißt für Matrizen ohne fehlende Einträge, stellt die (abgeschnittene) Singulärwertzerlegung (SVD) eine effiziente Methode dar, um eine Niedrigrang-Approximation und damit Zerlegung in  $U$  und  $M$  zu bestimmen.

Da im Falle unseres Empfehlungssystems jedoch zahlreiche Einträge der zu approximierenden Matrix  $R$  fehlen, ist dieses Vorgehen nicht direkt anwendbar. Die SVD ist für Matrizen mit fehlenden Einträgen nicht definiert. Deswegen kommen beispielsweise Algorithmen zum Einsatz, bei denen die fehlenden Einträge iterativ aufgefüllt werden, sodass die SVD anwendbar wird. Derartige Strategien werden bei Hastie et al. (2015) beschrieben.

Dieser Ansatz wird hier nicht weiter verfolgt. Stattdessen werden Algorithmen verwendet, bei denen nur die tatsächlich bekannten Einträge der Bewertungsmatrix  $R$  berücksichtigt werden. Gängige Lösungsverfahren, die zur Lösung des nichtlinearen Optimierungsproblems (16.7) Einsatz finden, sind stochastische Gradientenverfahren (engl. *Stochastic Gradient Descent*, SGD) und die alternierende Methode der kleinsten Fehlerquadrate (engl. *Alternating Least Squares*, ALS) (vgl. Koren et al.,

2009, S. 33). Im Folgenden wird das ALS-Verfahren, welches verschiedene Anknüpfungspunkte an schulmathematische Inhalte bietet, ausführlicher beschrieben und eine mögliche didaktische Reduktion dessen skizziert. Das SGD-Verfahren wurde nicht mit Schülerinnen und Schülern erprobt. Da es jedoch ebenfalls verschiedene Anknüpfungspunkte an schulmathematische Inhalte aufweist und auch bei diversen anderen ML-Methoden (unter anderem Neuronale Netze) zentral ist, wird es in Anhang E.1 kurz erläutert.

### 16.2.4 Alternierende Methode der kleinsten Fehlerquadrate

Beim ALS-Verfahren wird ausgenutzt, dass sich ein quadratisches, konvexes Optimierungsproblem ergibt, sobald entweder die Usermatrix  $U$  oder die Moviematrix  $M$  fest gewählt wird. Das Problem (16.7) wird deswegen auch als bi-konvex bezeichnet (vgl. Hastie et al., 2015, S. 3369).

Wie der Name verrät, wird beim ALS-Verfahren in jedem Iterationsschritt ein lineares Ausgleichsproblem gelöst. Der Algorithmus besteht konkret aus folgenden Schritten:

1. Die beiden Matrizen  $U$  und  $M$  werden zu Beginn zufällig initialisiert.
2. Die Uservektoren  $\mathbf{u}_{i,\bullet}$  mit  $i = 1, \dots, n_1$  werden festgehalten und für jeden Movievektor  $\mathbf{m}_{\bullet,j}$  mit  $j = 1, \dots, n_2$  wird ein lineares Ausgleichsproblem (und damit ein Minimierungsproblem) gelöst.
3. Die durch Lösen der Ausgleichsprobleme gefundenen Movievektoren werden festgehalten. Anschließend wird für jeden Uservektor  $\mathbf{u}_{i,\bullet}$  mit  $i = 1, \dots, n_1$  ein lineares Ausgleichsproblem gelöst.
4. Die Schritte 2 und 3 werden solange wiederholt, bis entweder eine vorgegebene Anzahl an Iterationen erreicht ist, ein ausreichend kleiner Fehler bezüglich der Testdaten erzielt wurde oder die Optimierung auskonvergiert ist; wobei die Konvergenz gegen eine global optimale Faktorisierung nicht gesichert ist (vgl. Strang, 2009, S. 99).

An dem folgenden Minimalbeispiel wird deutlich, wie das Verfahren mit Lernenden erarbeitet werden kann:

#### Beispiel 16.2.1.

Gesucht ist eine Zerlegung der Bewertungsmatrix  $R$  mit<sup>6</sup>

$$R = \begin{pmatrix} 2 & 3 \end{pmatrix} \quad (16.11)$$

in eine Usermatrix  $U \in \mathbb{R}^{1 \times 2}$  und Moviematrix  $M \in \mathbb{R}^{2 \times 2}$ . Die Moviematrix kann im ersten Iterationsschritt beliebig festgelegt werden. In diesem Beispiel wird sie auf

$$M = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \quad (16.12)$$

<sup>6</sup>In diesem Fall wäre  $U = R$  und  $M = \mathbb{I}$  eine triviale Zerlegung. Die hier durchgeführte Rechnung dient daher nur als Erklärung der allgemeinen Vorgehensweise. Die Dimensionalität ist bewusst klein gewählt, damit die Zielfunktion des ersten Iterationsschritts visualisiert werden kann.

gesetzt und das Minimum der Funktion  $f$  mit

$$f(u_{11}, u_{12}) = (2 - (u_{11} \cdot 1 + u_{12} \cdot 1))^2 + (3 - (u_{11} \cdot 1 + u_{12} \cdot 1))^2 \quad (16.13)$$

bestimmt. Dazu bestimmen die Lernenden zunächst die partiellen Ableitungen und damit den Gradienten. Die Berechnung des Minimalpunkts wird dann über die Nullstellen des Gradienten

$$\nabla f = -2 \cdot \begin{pmatrix} 2 - (u_{11} + u_{12}) + 3 - (u_{11} + u_{12}) \\ 2 - (u_{11} + u_{12}) + 3 - (u_{11} + u_{12}) \end{pmatrix} \stackrel{!}{=} 0 \quad (16.14)$$

angegangen. Dies liefert ein lineares Gleichungssystem, welches von den Lernenden händisch gelöst werden kann. Die Lösung ist in diesem Beispiel nicht eindeutig. Es existieren unendlich viele Lösungen. Durch die interaktive Betrachtung der dreidimensionalen graphischen Darstellung der Zielfunktion (16.13) (vgl. Abb. 16.2) kann anschaulich erkundet werden, dass die Funktion keinen eindeutigen Minimalpunkt besitzt. Es wird eine Lösung des Gleichungssystems (z. B.  $u_{11} = 1$  und  $u_{12} = 1.5$ ) gewählt und damit die Werte für die Einträge der Usermatrix im nächsten Iterationsschritt festgelegt. Dann kann analog die Optimierung der Einträge der Moviematrix durchgeführt werden. Wir erhalten die Zielfunktion für diesen Optimierungsschritt, bestimmen den Gradienten und berechnen dessen Nullstellen durch erneutes Lösen eines linearen Gleichungssystems. In diesem kleinen Beispiel führt bereits der zweite Iterationsschritt zu einer exakten Zerlegung der Bewertungsmatrix  $R$ .

Bei der Entwicklung des Optimierungsverfahrens kann problemorientiert diskutiert werden, welche Vorteile die Wahl der Summe der Fehlerquadrate gegenüber der Summe der absoluten Abweichungen für das Lösen des Optimierungsproblems in den einzelnen Iterationen liefert. Wurde die Summe der absoluten Abweichungen zur Berechnung des Fehlers gewählt, wird die Zielfunktion (16.13) zu

$$f(u_{11}, u_{12}) = |2 - (u_{11} \cdot 1 + u_{12} \cdot 1)| + |3 - (u_{11} \cdot 1 + u_{12} \cdot 1)|. \quad (16.15)$$

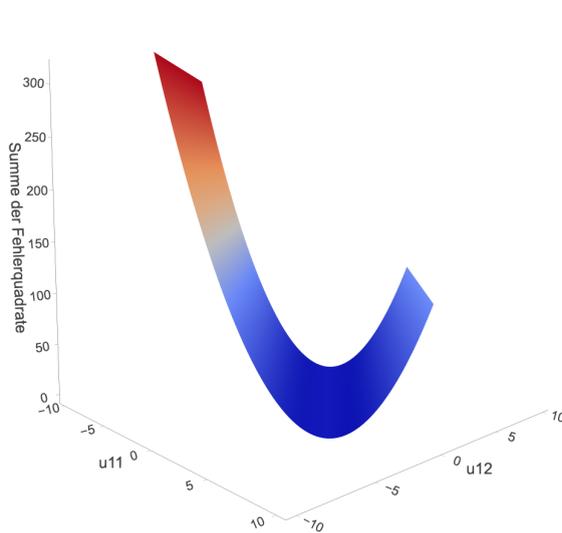
Am Graphen dieser Funktion (vgl. Abb. 16.3) wird erkenntlich, dass die Funktion nicht überall differenzierbar ist, womit die Berechnung des Minimalpunkts komplexer wird.

Fernab von der didaktisch-reduzierten Formulierung des Optimierungsverfahrens kann das Problem in den einzelnen Iterationsschritten in Matrix-Vektor-Notation überführt werden. Exemplarisch wird dies an dem Iterationsschritt beschrieben, in dem die Moviematrix  $M$  optimiert wird. Das Problem kann in diesem Fall als

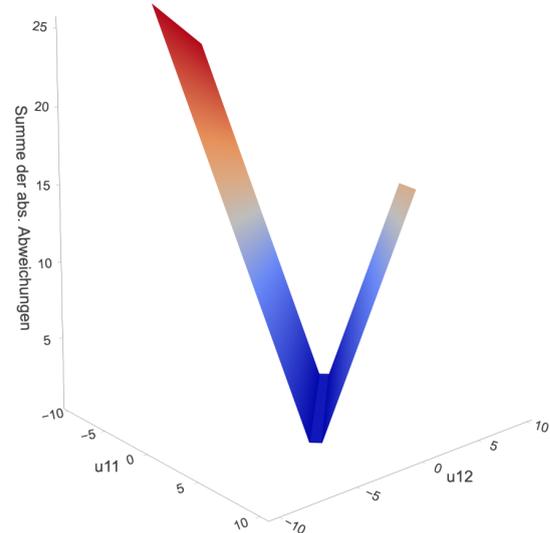
$$\min_{\mathbf{m}_{\bullet,j}} \frac{1}{2} \|W_j (U \mathbf{m}_{\bullet,j} - \mathbf{r}_{\bullet,j})\|^2 \quad (16.16)$$

mit fester Usermatrix  $U$  geschrieben werden, wobei  $\|\cdot\|$  für die euklidische Norm steht. Zudem ist  $W_j \in \mathbb{R}^{n_1 \times n_1}$  definiert als

$$W_j = \text{diag}(d_1, \dots, d_{n_1}) \text{ mit } d_i = \begin{cases} 1 & \text{falls } (i, j) \in \kappa \\ 0 & \text{sonst.} \end{cases}$$



**Abb. 16.2:** Darstellung der Zielfunktion (16.13). Es existiert kein eindeutiger globaler Minimalpunkt.



**Abb. 16.3:** Darstellung der Zielfunktion (16.15) basierend auf der Wahl der Summe der absoluten Abweichungen

Die Multiplikation mit der Matrix  $W_j$  stellt sicher, dass in dem Minimierungsproblem nur Einträge von  $\mathbf{r}_{\bullet,j}$  berücksichtigt werden, die bekannt sind. Das Problem wird in dieser Form auch als gewichtetes lineares Ausgleichsproblem bezeichnet, welches sich mit  $\tilde{U}_j = W_j \cdot U$  wiederum in Standardform überführen lässt

$$\min_{\mathbf{m}_{\bullet,j}} \frac{1}{2} \|\tilde{U}_j \mathbf{m}_{\bullet,j} - \mathbf{r}_{\bullet,j}\|^2. \quad (16.17)$$

Bei dem Optimierungsproblem (16.17) handelt es sich um ein quadratisches, konvexes Optimierungsproblem. Dies wird leicht ersichtlich, wenn die Zielfunktion geschrieben wird als

$$f(\mathbf{m}_{\bullet,j}) = \frac{1}{2} \left( \mathbf{m}_{\bullet,j}^T \tilde{U}_j^T \tilde{U}_j \mathbf{m}_{\bullet,j} - 2 \mathbf{r}_{\bullet,j}^T \tilde{U}_j \mathbf{m}_{\bullet,j} + \mathbf{r}_{\bullet,j}^T \mathbf{r}_{\bullet,j} \right) \quad (16.18)$$

mit Gradient

$$\nabla f(\mathbf{m}_{\bullet,j}) = \tilde{U}_j^T \tilde{U}_j \mathbf{m}_{\bullet,j} - \tilde{U}_j^T \mathbf{r}_{\bullet,j}$$

und Hesse-Matrix

$$\nabla^2 f(\mathbf{m}_{\bullet,j}) = \tilde{U}_j^T \tilde{U}_j.$$

Die Hesse-Matrix ist positiv semidefinit, da für beliebige Vektoren  $\mathbf{x} \in \mathbb{R}^{n_f}$

$$\mathbf{x}^T \tilde{U}_j^T \tilde{U}_j \mathbf{x} = \|\tilde{U}_j \mathbf{x}\|^2 \geq 0$$

gilt. Die Funktion  $f$  ist nach Satz 7 aus Abschnitt 2.1.1 somit konvex. Folglich ist jeder lokale Minimalpunkt von  $f$  zugleich globaler Minimalpunkt. Die Eindeutigkeit

des Minimalpunktes ist hingegen nicht garantiert, da die Zielfunktion nicht strikt konvex sein muss.

Die Bestimmung eines Minimalpunktes von (16.17) kann über die Berechnung der Nullstellen des Gradienten angegangen werden. Dies führt auf das Normalgleichungssystem

$$(\tilde{U}_j^T \tilde{U}_j) \mathbf{m}_{\bullet,j} = \tilde{U}_j^T \mathbf{r}_{\bullet,j}. \quad (16.19)$$

Hat die Matrix  $\tilde{U}_j$  vollen Spaltenrang und gilt  $n_1 \geq n_f$ , so ist  $\tilde{U}_j^T \tilde{U}_j$  invertierbar. Die Lösung für einen beliebigen Movievektor  $\mathbf{m}_{\bullet,j}$  kann dann geschrieben werden als (vgl. Strang, 2009, S. 124)

$$\mathbf{m}_{\bullet,j} = (\tilde{U}_j^T \tilde{U}_j)^{-1} \tilde{U}_j^T \mathbf{r}_{\bullet,j}. \quad (16.20)$$

Zur Lösung des Normalgleichungssystems existieren verschiedene Strategien. Beispielsweise kann das Problem über eine Cholesky-Zerlegung, eine QR-Zerlegung oder über die Singulärwertzerlegung angegangen werden (vgl. Nocedal und Wright, 2006, S. 251 ff.). Um das lineare Ausgleichsproblem für Matrizen  $\tilde{U}_j$  ohne vollen Spaltenrang zu lösen, kann die Pseudoinverse von  $\tilde{U}_j$  berechnet werden. Dies erübrigt sich jedoch bei Einbau der im folgenden Abschnitt beschriebenen Regularisierungsterme. Lösen des Optimierungsproblems (16.7) basierend auf den Bewertungsdaten  $r_{ij}$  des Trainingsdatensatzes, liefert eine Zerlegung der Bewertungsmatrix  $R$ . Diese kann genutzt werden, um Bewertungen des Testdatensatzes vorherzusagen und die Generalisierbarkeit des Modells auf unbekannte Daten zu validieren.

Ein Nachteil des ALS-Verfahrens ist, dass in den einzelnen Update-Schritten zur Lösung des Systems (16.19) (und analog für die Uservektoren) das recht aufwendige Lösen von linearen Gleichungssystemen notwendig ist. Dies wird gerne vermieden, wenn mit großen Datensätzen gearbeitet wird. Eine Möglichkeit ist, die Updates nicht vektoriell, sondern elementweise, das heißt für jeden Eintrag der Usermatrix  $U$  beziehungsweise der Moviematrix  $M$  einzeln, durchzuführen. Diese Vorgehensweise wird auch als *Coordinate Descent*-Verfahren bezeichnet (vgl. Feuerverger et al., 2012, S. 211; Aggarwal, 2016, S. 106). Ein Vorteil des ALS-Verfahrens ist, dass die Berechnungen parallelisiert werden können: In den einzelnen Iterationen kann gemäß (16.23) ein beliebiger Movievektor  $\mathbf{m}_{\bullet,j}$  unabhängig von den übrigen Movievektoren berechnet werden. Analoges gilt für die Uservektoren.

Im folgenden Beispiel wird das ALS-Verfahren zur Berechnung einer Faktorisierung einer kleinen Bewertungsmatrix eingesetzt.

**Beispiel 16.2.2.**

Für die Ratingmatrix  $R$  mit

$$R = \begin{matrix} & \begin{matrix} \text{🎬}_1 & \text{🎬}_2 & \text{🎬}_3 & \text{🎬}_4 & \text{🎬}_5 & \text{🎬}_6 \end{matrix} \\ \begin{matrix} \text{👤}_1 \\ \text{👤}_2 \\ \text{👤}_3 \\ \text{👤}_4 \end{matrix} & \begin{pmatrix} 3 & 1 & ? & 2 & 1 & ? \\ 1 & ? & 4 & 1 & 3 & 1 \\ ? & 1 & 1 & ? & 1 & 4 \\ ? & ? & 5 & ? & 4 & ? \end{pmatrix} \end{matrix}$$

wird mit dem ALS-Verfahren eine Zerlegung in  $U \in \mathbb{R}^{4 \times 2}$  und  $M \in \mathbb{R}^{2 \times 6}$  ( $n_f = 2$ ) berechnet. Nach 20 Iterationen ergibt sich folgende Vorhersagematrix  $P$  als Produkt der Usermatrix  $U$  und der Moviematrix  $M$

$$P = \begin{matrix} & \begin{matrix} \text{🎬}_1 & \text{🎬}_2 & \text{🎬}_3 & \text{🎬}_4 & \text{🎬}_5 & \text{🎬}_6 \end{matrix} \\ \begin{matrix} \text{👤}_1 \\ \text{👤}_2 \\ \text{👤}_3 \\ \text{👤}_4 \end{matrix} & \begin{pmatrix} 3.0 & 1.0 & 1.336 & 2.0 & 1.005 & 11.525 \\ 1.0 & 3.944 & 3.958 & 1.0 & 3.052 & 1.0 \\ 1.166 & 1.0 & 1.114 & 0.834 & 0.85 & 4.0 \\ -12.416 & 6.692 & 5.007 & -7.278 & 3.991 & -56.223 \end{pmatrix} \end{matrix},$$

wobei

$$U = \begin{matrix} \text{👤}_1 \\ \text{👤}_2 \\ \text{👤}_3 \\ \text{👤}_4 \end{matrix} \begin{pmatrix} 10.249 & -2.964 \\ 1.205 & 5.753 \\ 3.611 & -0.011 \\ -49.052 & 32.487 \end{pmatrix}, \quad M = \begin{matrix} \text{🎬}_1 & \text{🎬}_2 & \text{🎬}_3 & \text{🎬}_4 & \text{🎬}_5 & \text{🎬}_6 \\ \begin{pmatrix} 0.323 & 0.279 & 0.311 & 0.231 & 0.237 & 1.108 \\ 0.106 & 0.627 & 0.623 & 0.125 & 0.481 & -0.058 \end{pmatrix} \end{matrix}.$$

Zwar ist der Fehler bezüglich der bekannten Daten in  $R$  klein, der RMSE beträgt 0.052, jedoch wird sogleich ersichtlich, dass die Vorhersagen der fehlenden Daten nicht ansatzweise in dem Bereich von 1 bis 5 liegen und zum Teil sogar negativ sind.

### 16.2.5 Modellerweiterung: Regularisierung

Wie das vorangegangene Beispiel deutlich macht, führt die bisherige Formulierung des Optimierungsproblems dazu, dass die Trainingsdaten *zu gut* repräsentiert werden. Dies ist ein Effekt, der im Bereich ML vielfach beobachtet wird: die Überanpassung (engl. *overfitting*) an gegebene Trainingsdaten. Die Trainingsdaten werden nahezu exakt repräsentiert, zugleich ist das Generalisierungsvermögen auf ungesehene Daten (Testdaten) schlecht.

Der Effekt der Überanpassung an bekannte Daten tritt nicht nur im Falle unseres Empfehlungssystems, sondern bei zahlreichen weiteren Anwendungen auf, die mithilfe von ML-Methoden gelöst werden. Ein Beispiel, zu dem bereits Lernmaterial entwickelt und erprobt wurde, ist die Klassifizierung von Bildern basierend auf der Stützvektormethode (vgl. [Schönbrodt et al., 2021](#); [Schmidt, 2019](#)).

Um eine Überanpassung an die Trainingsdaten zu vermeiden, wird die Zielfunktion um einen Regularisierungsterm erweitert. Das Optimierungsproblem (16.7) wird zu

$$\min_{U, M} \sum_{(i,j) \in \kappa} (r_{ij} - \mathbf{u}_{i,\bullet} \mathbf{m}_{\bullet,j})^2 + \lambda \left( \sum_{i=1}^{n_1} \|\mathbf{u}_{i,\bullet}\|^2 + \sum_{j=1}^{n_2} \|\mathbf{m}_{\bullet,j}\|^2 \right) \quad (16.21)$$

mit Regularisierungsparameter  $\lambda > 0$  umformuliert. Der Regularisierungsterm soll letztlich verhindern, dass die Einträge der Matrizen  $U$  und  $M$  betragsmäßig sehr groß

werden.<sup>7</sup> Der hier gewählte Regularisierungsansatz wird auch als 2-Norm-Regularisierung beziehungsweise Tikhonov-Regularisierung bezeichnet. Durchaus wären andere Regularisierungsterme denkbar. Vielfach wird beispielsweise die 1-Norm anstelle der 2-Norm gewählt. Der 1-Norm-Regularisierungsansatz hat den Vorteil, dass der Fokus auf dünnbesetzte Lösungen gelegt wird, was für viele Anwendungen (bspw. Compressed Sensing) wünschenswert ist (vgl. Sra et al., 2011, S. 8). Nachteil ist jedoch, dass dies zwangsläufig zu einer nicht überall differenzierbaren Zielfunktion führt.

Die Normalengleichungen (16.19) werden nun zu

$$\left(\tilde{U}_j^T \tilde{U}_j + \lambda \mathbb{I}\right) \mathbf{m}_{\bullet,j} = \tilde{U}_j^T \mathbf{r}_{\bullet,j} \quad (16.22)$$

erweitert, wobei  $\mathbb{I} \in \mathbb{R}^{n_f \times n_f}$  der Einheitsmatrix entspricht. Die Matrix  $\left(\tilde{U}_j^T \tilde{U}_j + \lambda \mathbb{I}\right)$  ist invertierbar und die Lösung für  $\mathbf{m}_{\bullet,j}$  ist gegeben durch (vgl. Strang, 2009, S. 124)

$$\mathbf{m}_{\bullet,j} = \left(\tilde{U}_j^T \tilde{U}_j + \lambda \mathbb{I}\right)^{-1} \tilde{U}_j^T \mathbf{r}_{\bullet,j}. \quad (16.23)$$

Die Hesse-Matrix  $\left(\tilde{U}_j^T \tilde{U}_j + \lambda \mathbb{I}\right)$  der Zielfunktion mit Regularisierungsparameter ist positiv definit, denn für beliebige Vektoren  $\mathbf{x} \in \mathbb{R}^{n_f}$  mit  $\mathbf{x} \neq \mathbf{0}$  gilt

$$\mathbf{x}^T \left(\tilde{U}_j^T \tilde{U}_j + \lambda \mathbb{I}\right) \mathbf{x} = \|\tilde{U}_j \mathbf{x}\|^2 + \lambda \|\mathbf{x}\|^2 > 0,$$

da  $\lambda \|\mathbf{x}\|^2 > 0$ . Damit ist die Zielfunktion, die sich in den Iterationsschritten ergibt, strikt konvex. Dies führt dazu, dass das System (16.22) eindeutig lösbar ist. Das wird auch anschaulich am Graph der Zielfunktion aus Beispiel 16.2.1 ersichtlich, wenn die Funktion um den Regularisierungsterm erweitert wird (vgl. Abb. 16.4). Analog zu der hier beschriebenen Optimierung der Movievektoren kann bei der Optimierung der Uservektoren vorgegangen werden.

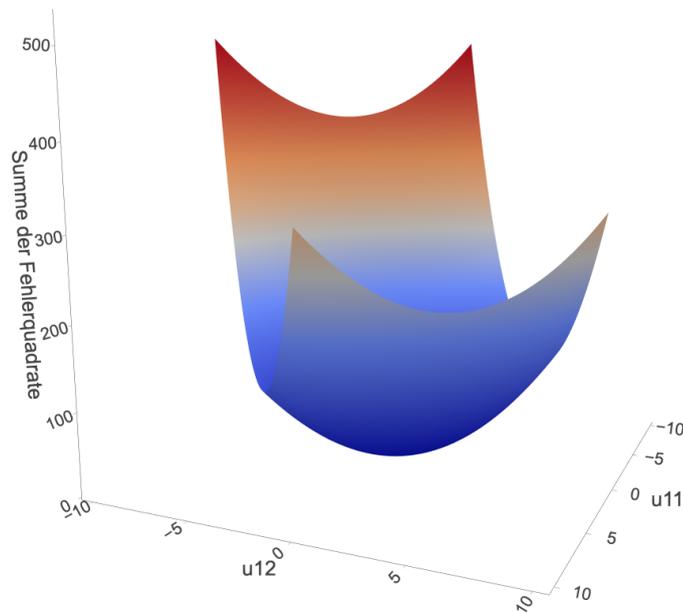
Der Regularisierungsparameter liefert die gewünschte Möglichkeit, eine Überanpassung der gesuchten Faktorisierung an gegebene Trainingsdaten zu regulieren. Wird ein großer Wert für  $\lambda$  gewählt, so führt dies zu einem größeren Fehler bezüglich der Trainingsdaten. Der Fehler bezüglich der Testdaten wird gegebenenfalls kleiner, da eine Überanpassung des Modells an die Trainingsdaten vermieden wird. Dies wird bei der Diskussion exemplarischer Ergebnisse in Abschnitt 16.2.6 und ausführlicher im Anhang E.3 aufgegriffen.

Die folgende Fortsetzung von Beispiel 16.2.2 macht den Einfluss des Regularisierungsparameters exemplarisch deutlich.

### Beispiel 16.2.3 (Fortsetzung von Beispiel 16.2.2).

Mit einem Regularisierungsparameter von  $\lambda = 0.4$  ergibt sich mit dem ALS-Verfahren

<sup>7</sup>Das regularisierte Problem lässt sich auch als Skalarisierung eines Mehrzielproblems betrachten, bei dem die beiden Ziele a) *Faktorisierung mit minimalem Fehler finden* und b) *Summe der quadrierten Einträge von  $U$  bzw.  $M$  minimieren*, verfolgt werden.



**Abb. 16.4:** Darstellung der Zielfunktion (16.13) erweitert um einen Regularisierungsterm  $f(u_{11}, u_{12}) + \lambda \cdot (u_{11}^2 + u_{12}^2)$  mit  $\lambda = 2$ . Es existiert ein eindeutiger globaler Minimalpunkt.

die Vorhersagematrix  $P$  ( $n_f = 2$ , 20 Iterationen)

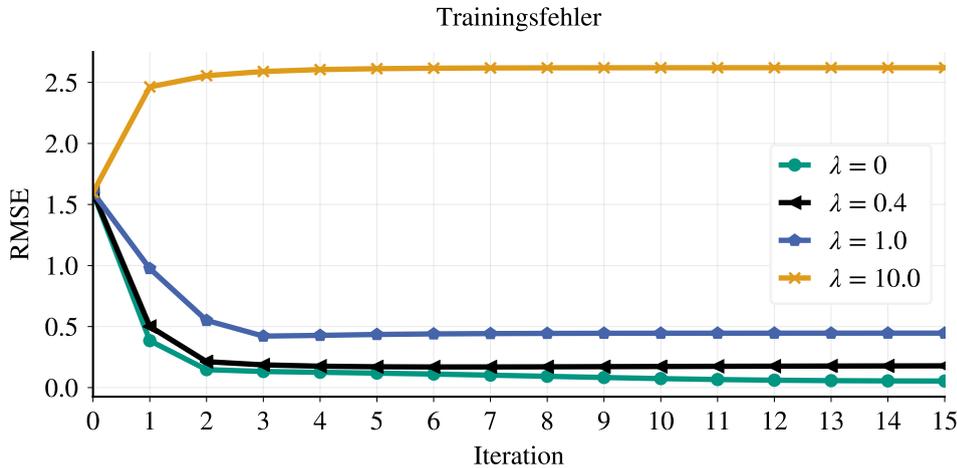
$$P = \begin{matrix} & \begin{matrix} \text{🎬}_1 & \text{🎬}_2 & \text{🎬}_3 & \text{🎬}_4 & \text{🎬}_5 & \text{🎬}_6 \end{matrix} \\ \begin{matrix} \text{👤}_1 \\ \text{👤}_2 \\ \text{👤}_3 \\ \text{👤}_4 \end{matrix} & \begin{pmatrix} 2.672 & 0.915 & 1.128 & 1.795 & 1.032 & 3.379 \\ 0.994 & 0.317 & 3.751 & 0.952 & 2.937 & 1.009 \\ 2.841 & 0.974 & 1.026 & 1.894 & 0.964 & 3.605 \\ 1.091 & 0.343 & 4.8 & 1.104 & 3.746 & 1.058 \end{pmatrix} \end{matrix}$$

als Produkt der Usermatrix  $U$  und der Moviematrix  $M$

$$U = \begin{matrix} \begin{matrix} \text{👤}_1 \\ \text{👤}_2 \\ \text{👤}_3 \\ \text{👤}_4 \end{matrix} & \begin{pmatrix} 1.8 & 0.014 \\ 0.628 & 1.624 \\ 1.916 & -0.069 \\ 0.682 & 2.115 \end{pmatrix} \end{matrix}, \quad M = \begin{matrix} \begin{matrix} \text{🎬}_1 & \text{🎬}_2 & \text{🎬}_3 & \text{🎬}_4 & \text{🎬}_5 & \text{🎬}_6 \end{matrix} \\ \begin{pmatrix} 1.484 & 0.508 & 0.61 & 0.996 & 0.56 & 1.877 \\ 0.038 & -0.001 & 2.073 & 0.201 & 1.591 & -0.105 \end{pmatrix} \end{matrix}.$$

Zwar ist der Fehler bezüglich der bekannten Daten etwas größer, der RMSE beträgt 0.180, jedoch wird sogleich ersichtlich, dass die Werte für die unbekannt Daten nun fast alle im Bereich von 1 bis 5 liegen. Die Einträge von  $U$  und  $M$  sind, wie zu erwarten, betragsmäßig kleiner.

Der Verlauf des Fehlers bezüglich der Trainingsdaten über die Iterationen des Verfahrens ist in Abbildung 16.5 für verschiedene Regularisierungsparameter dargestellt. Es wird ersichtlich, dass die Trainingsdaten schlechter approximiert werden, wenn der Regularisierungsparameter größer gewählt wird. Bei der Anwendung auf einen



**Abb. 16.5:** Fehler bezüglich der Trainingsdaten für unterschiedliche Werte von  $\lambda$ . Die Anzahl der Eigenschaften beträgt in allen Fällen  $n_f = 2$ .

Teil des Netflixdatensatzes in Anhang E.3 wird jedoch ersichtlich, dass der umgekehrte Effekt für die Testdaten zu beobachten ist.

Abbildung 16.5 macht zudem deutlich, dass der Regularisierungsterm die Zielfunktion dominiert, wenn der Regularisierungsparameter für eine geringe Anzahl an Eigenschaften  $n_f$  sehr groß gewählt wird ( $\lambda = 10$ ). Die Trainingsdaten werden gar nicht mehr approximiert. Die Einträge der Matrizen  $U$  und  $M$  gehen gegen Null.

### Exkurs: Alternative Formulierung des regularisierten Problems

Beim Einbau eines Regularisierungsterms sind hinsichtlich dessen, wie stark die einzelnen Vektoren  $\mathbf{m}_{\bullet,j}$  und  $\mathbf{u}_{i,\bullet}$  regularisiert werden, durchaus verschiedene Ansätze denkbar (und finden in der Praxis Einsatz). Hier wird kurz auf eine alternative Formulierung des regularisierten Problems eingegangen, die sich bei Zhou et al. (2008) wiederfindet. Bei dieser werden die User- und Movievektoren durch den Einbau von Gewichten *unterschiedlich stark regularisiert*. Das Optimierungsproblem lautet

$$\min_{U,M} \frac{1}{2} \sum_{(i,j) \in \kappa} (r_{ij} - \mathbf{u}_{i,\bullet} \mathbf{m}_{\bullet,j})^2 + \frac{\lambda}{2} \left( \sum_{i=1}^{n_1} n_i^{\text{user}} \|\mathbf{u}_{i,\bullet}\|^2 + \sum_{j=1}^{n_2} n_j^{\text{movie}} \|\mathbf{m}_{\bullet,j}\|^2 \right) \quad (16.24)$$

mit Regularisierungsparameter  $\lambda > 0$ . Hierbei entspricht  $n_i^{\text{user}}$  der Anzahl an Bewertungen der Person  $i$  und  $n_j^{\text{movie}}$  der Anzahl an Bewertungen von Film  $j$ . Dieses Problem ist äquivalent zu dem folgenden gewichteten Niedrigrang-Approximationsproblem mit Tikhonov-Regularisierung (vgl. Zhou et al., 2008, S. 339)

$$\min_{U,M} \frac{1}{2} \|W \otimes (R - U \cdot M)\|_F^2 + \frac{\lambda}{2} \left( \|U^\top \Gamma_u\|_F^2 + \|M \Gamma_m\|_F^2 \right), \quad (16.25)$$

wobei es sich bei  $\Gamma_u = \text{diag}(\sqrt{n_1^{\text{user}}}, \dots, \sqrt{n_{n_1}^{\text{user}}})$  und  $\Gamma_m = \text{diag}(\sqrt{n_1^{\text{movie}}}, \dots, \sqrt{n_{n_2}^{\text{movie}}})$  um Tikhonov-Matrizen handelt. Das Normalgleichungssystem, welches wir beim Update eines beliebigen Movievektors erhalten, lautet für dieses Problem

$$\left( \tilde{U}_j^\top \tilde{U}_j + \lambda n_j^{\text{movie}} \mathbb{I} \right) \mathbf{m}_{\bullet,j} = \tilde{U}_j^\top \mathbf{r}_{\bullet,j}. \quad (16.26)$$

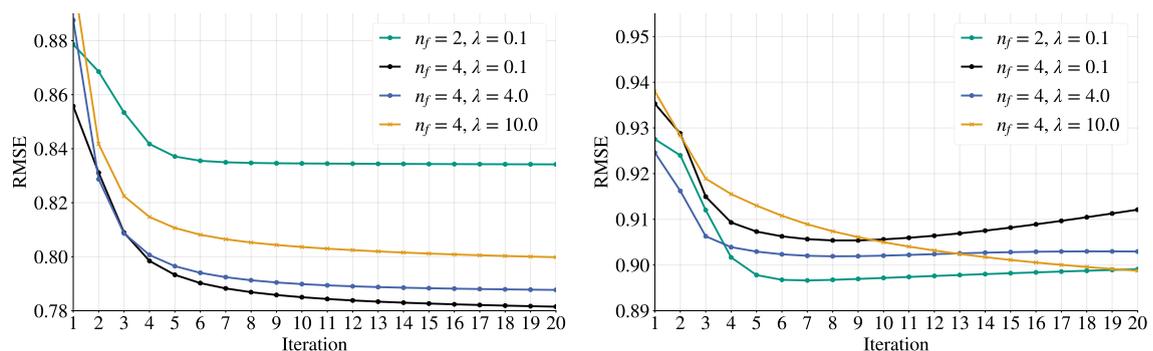
Unter der Annahme, dass von jedem Film mindestens eine Bewertung bekannt ist, also  $n_j^{\text{movie}} > 0$  gilt, ist  $(\tilde{U}_j^T \tilde{U}_j + \lambda n_j^{\text{movie}} \mathbb{I})^{-1}$  invertierbar und die Lösung des Normalgleichungssystems eindeutig. In diesem Fall kann ein beliebiger Movievektor berechnet werden durch

$$\mathbf{m}_{\bullet,j} = (\tilde{U}_j^T \tilde{U}_j + \lambda n_j^{\text{movie}} \mathbb{I})^{-1} \tilde{U}_j^T \mathbf{r}_{\bullet,j}. \quad (16.27)$$

Die gewichtete Formulierung des Problems reguliert die Vektoren  $\mathbf{m}_{\bullet,j}$  beziehungsweise  $\mathbf{u}_{i,\bullet}$  stärker, die in die Modellierung dichter besetzter Spalten oder Zeilen der Ratingmatrix eingehen. In der Diskussion numerischer Experimente in Anhang E.3.3 wird deutlich, dass der Regularisierungsansatz zu einer geringeren Überanpassung an die Trainingsdaten führt.

### 16.2.6 Anwendung auf Testdaten

Bisher haben wir lediglich quantifiziert, welchen Einfluss die unterschiedlichen Formulierungen des Optimierungsproblems auf den Trainingsfehler haben. Um zudem quantifizieren zu können, wie gut die Vorhersage für noch ungesehene Bewertungsdaten funktioniert, wird sich der bereits in Abschnitt 14.1 beschriebenen Strategie der Validierung auf einem Testdatensatz zunutze machen. Dazu werfen wir einen Blick auf numerische Experimente, die auf einem Teil des Netflixdatensatzes ausgeführt wurden. Hier werden lediglich exemplarische Ergebnisse gezeigt. Ziel ist es, wesentliche Beobachtungen darstellen zu können, die auch bei der Umsetzung im Lernmaterial zentral sind. Eine ausführliche Diskussion numerischer Experimente findet sich in Anhang E.3.



(a) Fehler bezüglich der Trainingsdaten      (b) Fehler bezüglich der Testdaten

**Abb. 16.6:** Ergebnisse basierend auf dem regularisierten Problem (16.21) für verschiedene Wahlen der Parameter  $n_f$  und  $\lambda$ . Die Ergebnisse werden ab der ersten Iteration (d. h. nach dem ersten Update von  $U$  und  $M$ ) dargestellt.

In Abbildung 16.6 sind Ergebnisse des ALS-Verfahrens für verschiedene Wahlen des Regularisierungsparameters  $\lambda$  und der Anzahl an betrachteten Eigenschaften  $n_f$  dargestellt. Es wird ersichtlich, ...

- dass die Überanpassung an die Trainingsdaten auch bei Anwendung auf größere Datensätze reguliert werden kann, indem der Regularisierungsparameter entsprechend erhöht wird.
- dass eine Erhöhung der Anzahl an Eigenschaften  $n_f$  bei gleichbleibendem Regularisierungsparameter wiederum zu einer größeren Überanpassung an die Trainingsdaten führt. Der Fehler bezüglich der Testdaten ist größer. Dies lässt sich dadurch erklären, dass eine größere Anzahl an Eigenschaften zu einem höherdimensionalen Optimierungsproblem und damit mehr Freiheitsgraden bei der Optimierung führt. Es bestehen vereinfacht gesagt mehr Möglichkeiten die bekannten Daten besonders gut darzustellen.

Es kann festgehalten werden, dass für einen gegebenen Datensatz eine geeignete Kombination aus Regularisierungsparameter  $\lambda$  und Anzahl Eigenschaften  $n_f$  gesucht werden muss. Dies ist beispielsweise über eine Kreuzvalidierung möglich.

### 16.2.7 Modellerweiterung: Verzerrungen in den Daten

Das Bewertungsverhalten einzelner Personen kann nicht nur hinsichtlich individueller Vorlieben für verschiedene Filme, sondern auch bezüglich deren Tendenz eher euphorisch oder reserviert zu bewerten, variieren. So ist es denkbar, dass eine Person A eher Bewertungen von 4 und 5 abgibt, wenn ihr ein Film gefällt, wohingegen eine eher kritische Person B, der die gleichen Filme gefallen, Bewertungen von 3 und 4 abgibt. Dies führt zu einer Verzerrung in den Bewertungen, die auch Bias genannt wird. Ähnliche Effekte können bei den Filmen auftreten. So ist es möglich, dass ein Film aufgrund eines besonders berühmten Schauspielers im Mittel um 0.5 Bewertungspunkte besser bewertet wird, als der Durchschnitt (vgl. [Aggarwal, 2016, S. 106](#)).

Der Bias sollte bei der Modellierung der Interaktion zwischen Nutzenden, Filmen und deren Eigenschaften über das Skalarprodukt  $\mathbf{u}_{i,\bullet} \mathbf{m}_{\bullet,j}$  nicht mit einbegriffen werden. Deswegen wird die Vorhersage  $\hat{r}_{ij}$  einer Bewertung  $r_{ij}$  unterteilt: in einen Term, der Effekte wie User- und Movie-Bias berücksichtigt, und einen Term, der die tatsächliche Interaktion von Nutzenden und Filmen hinsichtlich der Eigenschaften modelliert (vgl. [Koren et al., 2009, S. 33](#)). Dies liefert als Vorhersage  $\hat{r}_{ij}$

$$\hat{r}_{ij} = \mathbf{u}_{i,\bullet} \mathbf{m}_{\bullet,j} + \mu + o_i + p_j. \quad (16.28)$$

Hierbei entspricht  $\mu$  dem Mittelwert über alle bekannten Bewertungen im Trainingsdatensatz,  $o_i$  beschreibt den Bias von Person  $i$  und  $p_j$  den Bias eines beliebigen Filmes  $j$ . Für die Festlegung der Parameter  $o_i$  und  $p_j$  sind unterschiedliche Ansätze denkbar. Zum einen können die Bias-Parameter vor dem Lernen der eigentlichen Faktorisierung bestimmt werden und als Vorverarbeitung der Daten berücksichtigt werden. Dazu können die Parameter  $o_i$  und  $p_j$  als Minimalpunkte des konvexen Optimierungsproblems

$$\min_{\mathbf{o}, \mathbf{p}} \sum_{(i,j) \in \kappa} (r_{ij} - \mu - o_i - p_j)^2 + \lambda \left( \sum_{i=1}^{n_1} o_i^2 + \sum_{j=1}^{n_2} p_j^2 \right) \quad (16.29)$$

bestimmt werden (vgl. [Koren und Bell, 2011](#), S. 149). Eine andere Möglichkeit ist es, den User- und Movie-Bias aus den gegebenen Daten zu *lernen*, indem die Bias-Parameter als weitere Variablen in das Optimierungsproblem (16.21) einbezogen werden. Das Problem lautete nun

$$\begin{aligned} \min_{U, M, \mathbf{o}, \mathbf{p}} \sum_{(i,j) \in \kappa} (r_{ij} - \mathbf{u}_{i,\bullet} \mathbf{m}_{\bullet,j} - \mu - o_i - p_j)^2 \\ + \lambda \left( \sum_{i=1}^{n_1} (\|\mathbf{u}_{i,\bullet}\|^2 + o_i^2) + \sum_{j=1}^{n_2} (\|\mathbf{m}_{\bullet,j}\|^2 + p_j^2) \right). \end{aligned} \quad (16.30)$$

Um das ALS-Verfahren auch zum Lösen dieses Optimierungsproblems anwenden zu können, müssen lediglich kleine Änderungen vorgenommen werden. Diese können in Anhang E.2 eingesehen werden.

## 16.3 Nachbarschaftsmethoden – Ähnlichkeiten modellieren

Neben der Entwicklung von Empfehlungssystemen über eine Matrix-Faktorisierung finden vielfach auch Nachbarschaftsmethoden Einsatz. Die Grundidee von Nachbarschaftsmethoden ist es, Ähnlichkeiten zwischen Nutzenden (nutzerbasierter Ansatz) oder zwischen Filmen (filmbasierter Ansatz) über Ähnlichkeitsmaße zu modellieren. Basierend auf gefundenen Ähnlichkeiten zwischen den Nutzerinnen und Nutzern / den Filmen, werden die Filme festgelegt, die einer Person sehr wahrscheinlich gefallen. Die wesentlichen Elemente von Nachbarschaftsmethoden werden hier kurz beschrieben, da sich das Modell, welches eine Lerngruppe vollkommen unangeleitet auf einer Modellierungswoche erarbeitet hat, diesen Methoden zuordnen lässt (vgl. Abschn. 18.2.2).

Ein gängiges Ähnlichkeitsmaß ist die Verwendung des Pearson-Korrelationskoeffizienten. Die Ähnlichkeit zwischen zwei beliebigen Nutzenden  $u_1$  und  $u_2$  wird bestimmt über

$$\text{sim}(u_1, u_2) = \frac{\sum_{j \in I} (r_{u_1 j} - \mu_{u_1}) \cdot (r_{u_2 j} - \mu_{u_2})}{\sqrt{\sum_{j \in I} (r_{u_1 j} - \mu_{u_1})^2} \cdot \sqrt{\sum_{j \in I} (r_{u_2 j} - \mu_{u_2})^2}},$$

wobei  $\mu_{u_1}$  das mittlere Rating der Person  $u_1$  bezüglich der bekannten Bewertungen und analog  $\mu_{u_2}$  das der Person  $u_2$  bezeichnet.  $I$  stellt die Menge an Filmindizes dar, die beide Personen bewertet haben. Die Ähnlichkeit wird damit lediglich über die gemeinsamen Bewertungen berechnet (vgl. [Aggarwal, 2016](#), S. 35).

Basierend auf dem gewählten Ähnlichkeitsmaß werden die Ähnlichkeiten jedes User-User-Paars (oder jedes Film-Film-Paars) bestimmt. Um nun für eine beliebige Person  $u_i$ , die den Film  $j$  noch nicht gesehen hat, eine Vorhersage zu bestimmen, können die berechneten Ähnlichkeiten ausgenutzt werden. Eine Möglichkeit ist es, die Vorhersage als gewichtete Summe basierend auf den  $k$  ähnlichsten Nutzenden zu bestimmen.

Dies wird unter anderem bei [Aggarwal \(2016, S. 36\)](#) diskutiert.

Ein weiteres Ähnlichkeitsmaß stellt die sogenannte Kosinus-Ähnlichkeit dar:

$$\text{sim}(u_1, u_2) = \frac{\sum_{j \in I} r_{u_1, j} \cdot r_{u_2, j}}{\sqrt{\sum_{j \in I} r_{u_1, j}^2} \cdot \sqrt{\sum_{j \in I} r_{u_2, j}^2}} = \frac{\mathbf{r}_{u_1, \bullet}^{\text{red}} \cdot \mathbf{r}_{u_2, \bullet}^{\text{red}}}{\|\mathbf{r}_{u_1, \bullet}^{\text{red}}\| \cdot \|\mathbf{r}_{u_2, \bullet}^{\text{red}}\|}. \quad (16.31)$$

Dieses Ähnlichkeitsmaß lässt sich geometrisch als Kosinus des Winkels zwischen den beiden auf die gemeinsamen Bewertungen reduzierten Vektoren  $\mathbf{r}_{u_1, \bullet}^{\text{red}}$  und  $\mathbf{r}_{u_2, \bullet}^{\text{red}}$  interpretieren. Dieses Ähnlichkeitsmaß greift zentrale schulmathematische Inhalte aus dem Bereich der linearen Algebra, konkret das Skalarprodukt und Vektoren, auf und scheint damit besonders für die Diskussion mit Lernenden geeignet. Ausgehend von Beispielen des zwei- und dreidimensionalen Anschauungsraums bietet dieses Ähnlichkeitsmaß die Möglichkeit den geometrischen Winkelbegriff auf höherdimensionale euklidische Vektorräume zu abstrahieren. Für eine ausführliche Diskussion von Nachbarschaftsmethoden, weiteren Ähnlichkeitsmaßen und verschiedenen Möglichkeiten, diese zur Berechnung einer Vorhersage zu kombinieren sei auf [Aggarwal \(2016, Kap. 2\)](#) und [Sarwar et al. \(2001\)](#) verwiesen.

Latente Faktormodelle, wie die in [Abschnitt 16.2](#) beschriebene Matrix-Faktorisierung, weisen vielfach eine höhere Vorhersagegenauigkeit auf, als Nachbarschaftsmethoden (vgl. [Koren, 2010, S. 2](#)). Dennoch haben Nachbarschaftsmethoden zentrale Vorteile, die oft dazu führen, dass in der Praxis auf diese zurückgegriffen wurde (vgl. [Linden et al., 2003](#)). Ein wesentlicher Vorteil von Nachbarschaftsmethoden ist die Erklärbarkeit der Vorhersagen. Es ist möglich Aussagen darüber zu treffen, welche vergangenen Nutzerverhaltensweisen welchen Einfluss auf die berechnete Vorhersage haben (vgl. [Koren, 2010, S. 16](#)). Ein weiterer Vorteil ist, dass bei den filmbasierten Nachbarschaftsmethoden nicht das gesamte Modell neu gelernt werden muss, wenn neue Bewertungen eines Users hinzugenommen werden. Werden hingegen bei dem oben beschriebenen Modell einer Matrix-Faktorisierung neue Bewertungen hinzugefügt, so muss die gesamte Faktorisierung neu berechnet werden (vgl. [Koren, 2010, S. 16](#)).

Zu den Nachbarschaftsmethoden wurde im Rahmen der von mir co-betreuten Bachelorarbeit von [Rantzau \(2021\)](#) digitales Lernmaterial für die Oberstufe entwickelt. In diesem definieren die Lernenden zunächst eigene Ähnlichkeitsmaße. Zudem lernen sie gängige Maße wie die Pearson-Korrelation und die Kosinusähnlichkeit kennen und wenden diese auf einen Teil des originalen Netflixdatensatzes an.

## 16.4 Zwischenfazit und Konsequenzen für die Modellierung mit Lernenden

Die Modellierung des Empfehlungssystems über die Matrix-Faktorisierung erlaubt es, zentrale Strategien des überwachten maschinellen Lernens kennenzulernen. Dabei wird eindrücklich ersichtlich, dass Methoden aus diesem Bereich äußerst mathemathikhaltig sind. Durchaus ist bei der obigen Modellierung jedoch auch deutlich

geworden, dass das Optimierungsproblem, welches sich bei der Modellierung des Empfehlungssystems ergibt, als komplexer anzusehen ist als die Optimierungsprobleme, die im Falle des solarthermischen Kraftwerkes diskutiert wurden (vgl. Teil III). Gründe für die erhöhte Komplexität sind:

- Bei der Modellierung des Empfehlungssystems ergeben sich die Optimierungsfrage und insbesondere die Entscheidungsvariablen nicht in direkter, offensichtlicher Weise wie beim solarthermischen Kraftwerk. Bei der Problemstellung zur Solarthermie war es bereits vor der Entwicklung des mathematischen Modells möglich, festzuhalten, welche Parameter als Entscheidungsvariablen (bspw. die Absorberrohrhöhe oder die Spiegelpositionen) in Frage kommen. Die Entscheidungsvariablen sind „greifbar“ und haben auch in der Realität eine vorstellbare Bedeutung. Im Falle des Empfehlungssystems ergeben sich die Entscheidungsvariablen hingegen erst, nachdem das mathematische Modell entwickelt wurde. Zudem beschreiben die Entscheidungsvariablen abstrakte Einträge der User- und Moviematrix.
- Die Zielfunktion des Optimierungsproblems ist im Falle des Empfehlungssystems nicht sinnvoll in ein oder zwei Dimensionen (und damit im Anschauungsraum) darstellbar ist.

Aufgrund der erhöhten Komplexität wurde entschieden, im Lernmaterial rund um das Lösen des Optimierungsproblems bei unserem Empfehlungssystem einen stärker angeleiteten Ansatz zu verfolgen. Dennoch wurde die Problemstellung auch im Rahmen einer Modellierungswoche eingesetzt, um zu evaluieren, inwieweit sich derartig datenlastige Probleme – unabhängig davon, ob sie auf ein Optimierungsproblem hinauslaufen – für das offene, unangeleitete Modellieren mit Lernenden eignen (vgl. Abschn. 18.2.2).

# 17 Umsetzung im Lernmaterial

Im entwickelten Lernmaterial setzen sich die Schülerinnen und Schüler problemorientiert mit Empfehlungssystemen und dem Netflix Preis auseinander. Auf digitalen Arbeitsblättern (realisiert als Jupyter Notebooks) erkunden sie den Datensatz und sammeln Ideen für die Entwicklung eines Empfehlungssystems. Sie erarbeiten ein mathematisches Modell bei dem das Lösen eines Optimierungsproblems zentraler Bestandteil ist. Anschließend wenden sie das entwickelte Modell auf einen Teil des Netflixdatensatzes an. Das entwickelte Lernmaterial zeigt exemplarisch, wie datenlastige Probleme aus dem Alltag der Lernenden aufbereitet und im Rahmen von Modellierungsprojekten, sowohl im Distanzlernen als auch in Präsenz, durchgeführt werden können.

Tabelle 17.1 liefert eine Übersicht über Aufbau und Inhalte des Lernmaterials für die Durchführung im Rahmen angeleiteter Modellierungsprojekte.

## 17.1 Voraussetzungen und Ziele

Für die Bearbeitung des Lernmaterials wird vorausgesetzt, dass die Lernenden mit Lage- und Streumaßen (Mittelwert und Standardabweichungen) vertraut sind. Zudem sollten die Lernenden lineare Gleichungssysteme kennen und ohne digitale Hilfsmittel lösen können. Zentral ist zudem ein Verständnis von funktionalen Zusammenhängen. Für die Bearbeitung der Langversion von Arbeitsblatt 4 und des Zusatzmaterials zu den Fehlermaßen müssen die Lernenden die Differentialrechnung, insbesondere die Bestimmung von Extremstellen über die Nullstellen der Ableitung, beherrschen. Sie sollten die Kettenregel kennen. Für die Bearbeitung des Zusatzblattes zu den Fehlerfunktionen ist die Kenntnis über den Graph der Betragsfunktion günstig.

Mit der Entwicklung des Lernmaterials wurden verschiedene inhaltliche Ziele verfolgt, die im Folgenden beschrieben werden. Dabei wird lediglich auf die Hauptziele eingegangen. Die jeweiligen inhaltlichen Teilziele der Arbeitsblätter werden bei der Darstellung des Materials deutlich. Ziele, die unabhängig von mathematisch-inhaltlichen Lernzielen mit den Modellierungsaktivitäten verfolgt werden, wurden in Abschnitt 6.1 beschrieben. Folgende Ziele werden mit dem Lehr- und Lernmaterial verfolgt:

Die Lernenden ...

- setzen sich mit den Daten des Netflix Preises auseinander. Sie nutzen verschiedene Darstellungsformen der Daten, um diverse Fragen rund um den Datensatz zu beantworten.

**Tab. 17.1:** Übersicht über das Lernmaterial zu Empfehlungssystemen

Arbeitsblatt	Inhalt	Zusatzblätter
AB 1: Datenanalyse	Untersuchung des Netflixdatensatzes und erstes Brainstorming zu Ideen für die Entwicklung eines mathematischen Modells	
AB 2: Erste Modellentwicklung	Erarbeitung der Faktorisierungsidee	1 Zusatzblatt: Skalarprodukt
AB 3: Das Fehlermaß	Händische Berechnung von Faktorisierungen und Definition eines Fehlermaßes zur Bewertung der Vorhersagegüte	
AB 4 - Langversion: Entwicklung des ALS-Verfahrens	Schrittweise Entwicklung des ALS-Verfahrens an einem kleinen Beispiel und Untersuchung des Einflusses der Regularisierung	2 Zusatzblätter: 1) Einfluss der Fehlermaße, 2) Einfluss der Regularisierung
AB 4 - Kurzversion: Anwendung des ALS-Verfahrens	Anwendung des ALS-Verfahrens und Untersuchung des Einflusses der Regularisierung	
AB 5: Anwendung auf den Netflixdatensatz	Anwendung des ALS-Verfahrens auf den Netflixdatensatz und Parameterstudie	1 Zusatzblatt: eigene Empfehlungen

- untersuchen Zusammenhänge in einer kleinen Bewertungsmatrix, erklären diese im Hinblick auf die reale Situation und sammeln Ideen für die Entwicklung eines Empfehlungssystems.
- entwickeln angeleitet durch kleinere Aufgaben die Idee der Faktorisierung der Bewertungsmatrix. Sie interpretieren die Einträge von User- und Moviematrix im Hinblick auf den Geschmack der Nutzerinnen und Nutzer beziehungsweise auf die Eigenschaften der Filme. Sie stellen eine Formel für die Berechnung von Bewertungen basierend auf den Einträgen von User- und Moviematrix auf (Skalarprodukt).
- bestimmen händisch eine Zerlegung für eine kleine Bewertungsmatrix. Sie definieren eigene Fehlermaße, um die Güte der Vorhersage zu bewerten. Sie vergleichen ihre Fehlermaße mit denen ihrer Mitlernenden und diskutieren Vor- und Nachteile.
- lernen in der Kurzversion von Arbeitsblatt 4 das ALS-Verfahren kennen. Sie wenden dieses auf eine kleine Bewertungsmatrix an. Sie diskutieren und validieren die erhaltenen Vorhersagen für die bekannten und die unbekannt

Daten. Sie führen die Optimierung basierend auf der regularisierten Zielfunktion aus und diskutieren den Einfluss des Regularisierungsparameters.

- erarbeiten in der Langversion von Arbeitsblatt 4 die Schritte des ALS-Verfahrens, indem sie für ein kleines Beispiel die partiellen Ableitungen und die Nullstellen des Gradienten berechnen. Sie diskutieren und validieren die erhaltenen Vorhersagen für die bekannten und die unbekannt Daten. Sie führen die Optimierung basierend auf der regularisierten Zielfunktion aus und diskutieren den Einfluss des Regularisierungsparameters.
- erfassen die Idee von Trainings- und Testdaten. Anschließend wenden sie das ALS-Verfahren auf einen Teil des Netflixdatensatzes an und führen eine Parameterstudie für den Regularisierungsparameter  $\lambda$  und die Anzahl an Eigenschaften  $n_f$  durch. Sie diskutieren den Einfluss der Parameter auf die Ergebnisse.
- arbeiten mit realen Datensätzen und erkennen dabei, dass Daten das Herzstück für die Entwicklung und den Erfolg von ML-Methoden sind.
- reflektieren verschiedene Anwendungsmöglichkeiten der beschriebenen Modelle im Hinblick auf ethisch bedenkliche Anwendungen und deren gesellschaftliche Auswirkungen kritisch. Sie diskutieren Grenzen des entwickelten Modells und sammeln Ideen für Modellverbesserungen.

## 17.2 Darstellung des Lernmaterials

In diesem Abschnitt wird das Lernmaterial für die Durchführung im Rahmen von angeleiteten Projekten mit kürzerem zeitlichen Rahmen (Modellierungstage, Unterrichtsreihen) beschrieben. Im Zuge dessen werden die für die Bearbeitung des Materials notwendigen inhaltlichen Kompetenzen herausgearbeitet. Dabei wird darauf verzichtet im Detail auf einzelne Aufgabenstellungen einzugehen. Stattdessen sei auf das Lernmaterial im Anhang **F** und erneut auf die Möglichkeit verwiesen auf das Material über die Workshop-Plattform zuzugreifen (vgl. **Vorwort**).

Abschließend wird das Material für die Durchführung des Projektes im Rahmen einer Modellierungswoche kurz beschrieben.

### 17.2.1 Lernmaterial für angeleitete Modellierungsprojekte

Für die Entwicklung des Lernmaterials wurden die in Teil **II** beschriebenen digitalen Werkzeuge (vgl. Kap. 5) und methodischen Bausteine (vgl. Kap. 6) ein- und umgesetzt.

#### **Folien für die Einstiegspräsentation**

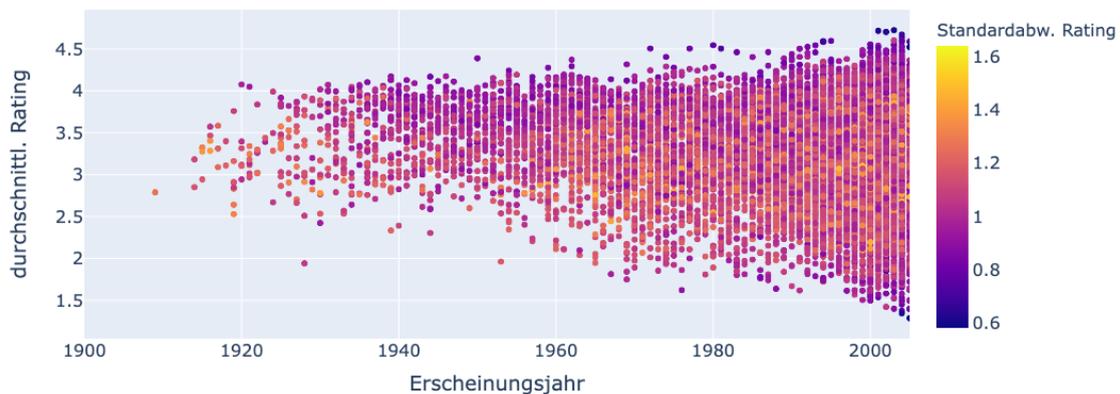
In der Einstiegspräsentation (vgl. Anh. **F.9.1**) wird zunächst die Problemstellung, will heißen die Entwicklung von guten Empfehlungssystemen, vorgestellt. Dabei werden die Lernenden mithilfe verschiedener Fragen in die Diskussion eingebunden:

- Welche Ziele verfolgen die Unternehmen mit dem Einsatz von Empfehlungssystemen?
- Welche Informationen können hilfreich sein, um Nutzerinnen und Nutzern neue Produkte (z. B. Filme, Bücher etc.) zu empfehlen?

Anschließend wird der Netflix Preis, der zugehörige Datensatz und das Ziel, fehlende Bewertungen bestmöglich vorherzusagen, vorgestellt. Dabei wird vereinfachend angenommen, dass a) die vorhandenen Bewertungsdaten wahr sind und b) eine fehlende Bewertung bedeutet, dass die Nutzerin / der Nutzer den entsprechenden Film noch nicht gesehen hat. Die Problemstellung und die getroffenen Vereinfachungen werden in den vierschriftigen Modellierungskreislauf (vgl. Abschn. 1.1) eingebettet. Abschließend werden das Ziel des Workshops (die Entwicklung eines eigenen Empfehlungssystems) und dessen Ablauf transparent gemacht.

### Antwortblätter

Zu Beginn der ersten Arbeitsphase erhalten die Lernenden Antwortblätter zu allen fünf Arbeitsblättern. Diese können bei Präsenzdurchführungen ausgedruckt und bei der virtuellen Durchführung als bearbeitbares pdf ausgefüllt werden. Ziel der Antwortblätter ist es, die Dokumentation der Ergebnisse durch die Lernenden bei der Bearbeitung des Lernmaterials zu erleichtern und zu strukturieren.



**Abb. 17.1:** Interaktive Visualisierung der Daten auf dem ersten Arbeitsblatt (Streudiagramm)

### Arbeitsblatt 1: Datenanalyse

Ausgangspunkt des Lernmaterials ist der Netflixdatensatz. Die Lernenden untersuchen diesen mithilfe verschiedener Darstellungsformen wie der Ratingmatrix beziehungsweise Ratingtabelle (16.1) oder Streudiagrammen und tabellarischen Darstellungen (vgl. Abb. 17.1 und 17.2). Dabei beantworten sie Fragen wie „Bei welchem Film weichen die Bewertungen am wenigsten vom Mittelwert ab?“ und „Welcher ist der beliebteste, welcher der älteste Film im Datensatz?“ Die Lernenden entscheiden, welche Darstellungsform für die Beantwortung welcher Frage geeignet ist. Bei

der Einführung der tabellarischen Darstellung der Bewertungsdaten (16.1) wird der Begriff einer Matrix zunächst vermieden, da dieser keine Voraussetzung für die verständige Bearbeitung des Lernmaterials sein sollte.

Anschließend erarbeiten die Lernenden eigene Ideen, um mittels der bekannten Bewertungen in der Ratingtabelle die fehlenden Bewertungen vorherzusagen. Ansätze, die in bisherigen Durchführungen von den Lernenden bereits genannt wurden, werden in Kapitel 18 diskutiert.

Filmtitel	Anzahl Ratings	durchschnittl. Rating	Standardabweichung	Erscheinungsjahr
Avia Vampire Hunter	132	1.2879	0.5843	2005
Zodiac Killer	289	1.3460	0.6209	2004
Alone in a Haunted House	205	1.3756	0.7589	2004
Vampire Assassins	247	1.3968	0.8275	2005
Absolution	125	1.4000	0.8854	2003
...	...	...	...	...
Battlestar Galactica: Season 1	1747	4.6388	0.6763	2004
Lost: Season 1	7249	4.6710	0.6825	2004
Lord of the Rings: The Two Towers: Extended Ed...	74912	4.7026	0.6289	2002

**Abb. 17.2:** Tabellarische Darstellung der Filme aus dem Netflixdatensatz. Die Lernenden können die Tabelle auf dem Arbeitsblatt nach beliebigen Spalten sortieren.

### Folien für die Zwischendiskussion

In der ersten Zwischendiskussion werden bisherige Ergebnisse der Lernenden im Plenum diskutiert (vgl. Anh. F.9). Bei der Diskussion der Fragen rund um den Datensatz wird aufgegriffen, welche Darstellungsform die Lernenden verwendet haben und warum. Anschließend werden die Ideen der Lernenden für die Entwicklung eines Empfehlungssystems basierend auf der Ratingmatrix gesammelt. Zudem werden die an einem kleinen Beispiel untersuchten Zusammenhänge zwischen den Zeilen beziehungsweise Spalten der Ratingmatrix diskutiert und von den Lernenden im Hinblick auf deren Bedeutung für das reale Problem eingeordnet. Abschließend wird die Problemstellung erneut in den vierschrittigen Modellierungskreislauf eingebettet. Ausgangspunkt für die weitere Modellierung ist die Betrachtung von wenigen Nutzerinnen und Nutzern (vereinfachtes Problem).

### Arbeitsblatt 2: Modell zur Vorhersage von Bewertungen

Die Lernenden steigen an kleinen Ratingtabellen in die Entwicklung des ausgewählten mathematischen Modells ein: die Zerlegung der Ratingtabelle in eine User-Eigenschafts-Tabelle (16.4) und eine Film-Eigenschafts-Tabelle (16.3). Sie erarbeiten ausgehend von einer gegebenen Beispielzerlegung (mit konkreten Zahlenwerten) wie sich die bekannten Bewertungen aus den Zeilen der Usertabelle und den Spalten der Movietabelle ergeben. Den gefundenen Zusammenhang wenden sie zur Berechnung fehlender Bewertungen an. Anschließend verallgemeinern sie ihre Berechnungen, indem sie eine Formel angeben, mit der sich basierend auf den Einträgen von  $U$

und  $M$  die Bewertungen bestimmen lassen. Die Berechnung ist nichts anderes als das Skalarprodukt aus Zeilenvektor der Usertabelle und Spaltenvektor der Movietabelle. Der Kern des beschriebenen Modells ist die in Abschnitt 16.2 beschriebene Matrix-Faktorisierung, die auch wesentlicher Bestandteil des Gewinnermodells des Netflix Preises war (vgl. [Koren et al., 2009](#)). Abschließend wird erstmalig der Begriff einer Matrix eingeführt. Eine in Form eines Erklärvideos umgesetzte Einführung der Faktorisierungsidee finden interessierte Leserinnen und Leser bei [Serrano \(2018\)](#).

### Zusatzblatt: Das Skalarprodukt

Auf diesem Zusatzblatt, welches auf Arbeitsblatt 1 verlinkt ist, setzen die Lernenden sich mit der geometrischen Interpretation von User- und Movievektoren auseinander. Sie diskutieren, inwieweit sich anhand einer gegebenen Darstellung der User- und Movievektoren (analog zu Abb. 16.1) ablesen lässt, ob eine größere oder kleinere Bewertung zu erwarten ist. Sie nutzen zur Argumentation die Definition des Skalarprodukts gemäß

$$\cos(\angle(\mathbf{m}, \mathbf{u})) = \frac{\mathbf{m} \cdot \mathbf{u}}{\|\mathbf{m}\| \cdot \|\mathbf{u}\|}. \quad (17.1)$$

### Folien für die Zwischendiskussion

Die Ergebnisse von Arbeitsblatt 2 werden im Plenum diskutiert und gesichert. Dabei wird zunächst die Interpretation der Einträge der User- und Moviematrix diskutiert. Die Lernenden erläutern erst an einem konkreten Zahlenbeispiel und anschließend basierend auf Variablen für die Einträge von  $U$  und  $M$ , wie sich die Bewertungen aus den Zeilen der Usermatrix und den Spalten der Moviematrix ergeben. In Lerngruppen, in denen das Skalarprodukt bekannt ist, wird diese Berechnung entsprechend eingeordnet. Die Bezeichnung der Tabellen als User- und Moviematrix wird erneut aufgegriffen. Abschließend wird der aktuelle Stand der Modellierung in den vierstufigen Modellierungskreislauf eingeordnet. Das nächste Teilziel für den weiteren Verlauf des Workshops wird benannt: die Bestimmung einer geeigneten Zerlegung in  $U$  und  $M$  ausgehend von der Ratingmatrix  $R$ . Zudem wird die Vereinfachung getroffen, dass es ausreicht zwei Eigenschaften ( $n_f = 2$ ) zu betrachten.

### Arbeitsblatt 3: Berechnung einer Zerlegung in $U$ und $M$

Tatsächlich enthält der Netflixdatensatz keine Informationen zu den Ausprägungen der Filme hinsichtlich gewisser Eigenschaften (bspw. Genres). Die Lernenden sollen deswegen zunächst händisch verschiedene Zerlegungen einer kleinen  $2 \times 2$  Bewertungsmatrix berechnen und mit den Zerlegungen ihrer Mitlernenden vergleichen. Dabei soll ersichtlich werden, dass die Zerlegung nicht eindeutig ist. Haben die Lernenden Schwierigkeiten, eine Zerlegung zu bestimmen, bei der die zuvor hergeleitete Formel (das Skalarprodukt) jeweils die bekannten Bewertungen liefert, so können sie auf zweistufige *Tippkarten* zugreifen.

Da die händische Bestimmung einer Zerlegung aufwendig und für große Ratingmatrizen nicht denkbar ist, sollen die Berechnungen dem Computer überlassen werden.

Dazu definieren die Lernenden ein Fehlermaß, mit dem sie (und das Computerverfahren) bewerten können, ob eine Zerlegung ausreichend gut ist. Sie vergleichen ihr eigenes Fehlermaß mit der Summe der Fehlerquadrate, die im weiteren Verlauf des Lernmaterials Anwendung findet.

### Folien für die Zwischendiskussion

Die Lernenden erläutern, wie sie bei der Berechnung einer Zerlegung per Hand vorgegangen sind. Dabei wird aufgegriffen, inwieweit die gefundene Zerlegung eindeutig ist. Es wird diskutiert, dass die Berechnung per Hand aufwendig und für größere Matrizen nicht praktikabel ist. Anschließend benennen die Lernenden ihre Ideen für die Berechnung des Fehlers zwischen der Ratingmatrix und der Vorhersagematrix. Die Ideen der Lernenden werden diskutiert. Zudem vergleichen und bewerten die Lernenden die folgenden Ansätze zur Berechnung des Fehlers: a) die orientierte Summe der Abweichungen, b) die Summe der absoluten Abweichungen und c) die Summe der quadratischen Abweichungen. Als Ausgangspunkt für die anstehende Optimierung wird die Summe der quadratischen Abweichungen als Fehlermaß festgehalten. Die Lehrkraft deutet bereits an, dass dieses Fehlermaß Vorteile für die Entwicklung eines Optimierungsverfahrens mit sich bringt, die im späteren Verlauf erneut aufgegriffen werden.

Anschließend werden die Schritte des ALS-Verfahrens an einem kleinen Beispiel erläutert. Die Bestimmung der *besten* Einträge von  $U$  oder  $M$  in den Iterationen des ALS-Verfahrens wird in Bezug zur Berechnung von Extremstellen von Funktionen einer Variablen gesetzt. Es wird deutlich, dass sich das aus dem Mathematikunterricht bekannte Vorgehen („Ableitung gleich Null“) auf die Berechnung der optimalen Einträge von  $U$  beziehungsweise  $M$  in den Iterationen des ALS-Verfahrens übertragen lässt. Auch hier wird auf ein kleines Zahlenbeispiel und die Visualisierung einer dreidimensionalen Zielfunktion (vgl. Abb. 16.2) zurückgegriffen.

### Arbeitsblatt 4 (Langversion): Das Optimierungsverfahren

Ziel des Arbeitsblattes ist die schrittweise Entwicklung des ALS-Verfahrens anhand eines kleinen Beispiels. Bei der Entwicklung des Verfahrens wird an schulmathematisches Wissen aus dem Bereich der Analysis (Ableitung, Extremwertbestimmung) angeknüpft. Die Erarbeitung des Verfahrens zielt darauf ab, einen problemorientierten Einstieg in mehrdimensionale Funktionen, Gradienten und die Berechnung von Minimalpunkten zu geben. Zugleich wird das Anwenden von Ableitungsregeln bei der Bestimmung der Gradienten und das Lösen von linearen Gleichungssystemen an einer authentischen Anwendung geübt. Die Grundidee des ALS-Verfahrens kann von den Lernenden bei Bedarf in einem verlinkten *Erklärvideo* erneut angeschaut werden.

Vor der Entwicklung des Verfahrens erhalten die Lernenden vertiefendes Informationsmaterial zu mehrdimensionalen Funktionen, welches die Übertragung von bekannten Konzepten aus der Differentialrechnung für Funktionen einer Variablen auf Funktionen mehrerer Variablen unterstützt. Es sei hier erwähnt, dass dieser Teil des Lernmaterials für leistungsstarke Schülerinnen und Schüler konzipiert wurde. Anschließend erarbeiten die Lernenden das Optimierungsverfahren analog zu Bei-

spiel 16.2.1. Bei der Entwicklung des Verfahrens kann problemorientiert diskutiert werden, welche Vorteile die Wahl der Summe der Fehlerquadrate gegenüber der Summe der absoluten Abweichungen im Hinblick auf die Eigenschaften der Zielfunktion liefert. Diese Diskussion wurde auf das im Folgenden beschriebene Zusatzblatt ausgelagert.

Nachdem die Lernenden die Berechnungen von zwei Iterationsschritten des ALS-Verfahrens händisch durchgeführt haben, können sie das Verfahren computergestützt auf eine größere Ratingmatrix anwenden und die erhaltenen Vorhersagen im Hinblick auf das reale Problem validieren. Dabei wird ersichtlich, dass die Vorhersage der bekannten Daten gut funktioniert, wohingegen die Vorhersagen der fehlenden Bewertungen zum Teil nicht ansatzweise im gewünschten Bereich zwischen 1 und 5 liegen. Die Grundidee der Regularisierung wird als Modellverbesserung eingeführt und in einem Erklärvideo erläutert. Die Lernenden untersuchen den Einfluss des Regularisierungsparameters auf zwei Arten: Zum einen experimentell, indem sie den Wert des Parameters variieren und den Einfluss auf die erhaltenen Ergebnisse untersuchen, zum anderen, indem sie Bezug auf den modifizierten Funktionsterm nehmen (*Wie müssen die Einträge von  $U$  und  $M$  gewählt werden, damit die Funktionswerte trotz hohem Regularisierungsparameter möglichst klein sind?*) Optional kann der Einfluss der Regularisierung auf die Gestalt der Zielfunktion anschaulich untersucht werden (vgl. Zusatzblatt *Regularisierung*).

### Zusatzblatt: Die Wahl der Fehlerfunktion

Auf diesem Zusatzblatt können die Lernenden den Einfluss der unterschiedlichen Fehlermaße auf die Zielfunktion sowohl grafisch als auch analytisch erkunden. Dazu wird erneut das Beispiel 16.2.1 aufgegriffen.

Wurde die Summe der absoluten Abweichungen zur Berechnung des Fehlers gewählt, so wird die Zielfunktion (16.13) zu:

$$f(u_{11}, u_{12}) = |2 - (u_{11} \cdot 1 + u_{12} \cdot 1)| + |3 - (u_{11} \cdot 1 + u_{12} \cdot 1)|. \quad (17.2)$$

Mit den Lernenden kann am Graph dieser Funktion (vgl. Abb. 16.3) diskutiert werden, dass diese nicht überall differenzierbar ist, womit die Berechnung des Minimalpunktes komplexer wird. Zudem wird die Zielfunktion basierend auf der Summe der orientierten Abstände

$$f(u_{11}, u_{12}) = 2 - (u_{11} \cdot 1 + u_{12} \cdot 1) + 3 - (u_{11} \cdot 1 + u_{12} \cdot 1) \quad (17.3)$$

graphisch dargestellt. Dabei wird anschaulich ersichtlich, dass diese Funktion keinen Minimalpunkt besitzt und als Fehlermaß nicht sinnvoll ist.

### Zusatzblatt: Regularisierung

Die Lernenden untersuchen den Einfluss des Regularisierungsterms auf den Graph der Zielfunktion  $f$  anschaulich in dem Minimalbeispiel 16.2.1. Auf dem Zusatzblatt wird der Graph der Zielfunktion wie in Abbildung 16.4 interaktiv visualisiert. Es ist zu erkennen, dass die Regularisierung dazu führt, dass die Zielfunktion einen eindeutigen Minimalpunkt aufweist.

### Arbeitsblatt 4 (Kurzversion): Das Optimierungsverfahren

Bei Bedarf kann die Grundidee des ALS-Verfahrens in einem verlinkten *Erklärvideo* erneut von den Lernenden angeschaut werden. Anschließend wird das ALS-Verfahren von den Lernenden nicht selbst entwickelt, sondern als Blackbox angewandt. Die anschließende Diskussion der Ergebnisse und des Einflusses des Regularisierungsterms erfolgt analog zur Langversion des Arbeitsblattes 4.

### Folien für die Zwischendiskussion

Zunächst wird erneut der vierschriftige Modellierungskreislauf aufgegriffen und die bereits absolvierten Modellierungsschritte in diesen eingeordnet. Die Lernenden erläutern ihre Einschätzung zur Güte der erhaltenen Ergebnisse des ALS-Verfahrens ohne Regularisierung. Anschließend beschreiben sie den Einfluss des Regularisierungsterms und begründen, wie sich die Wahl des Regularisierungsparameters  $\lambda$  auf die Einträge von  $U$  und  $M$  und die Vorhersage von bekannten und von fehlenden Bewertungen auswirkt.

Da bisher nur quantifiziert wurde, wie gut das Modell zur Repräsentation bekannter Bewertungen geeignet ist, aber unbekannte Bewertungen vorhergesagt werden sollen, wird eine Strategie des überwachten MLs vorgestellt: Die Unterteilung der bekannten Bewertungsdaten in solche, die zur Berechnung einer Zerlegung verwendet werden (Trainingsdaten) und solche, die genutzt werden, um zu validieren, wie gut die Vorhersage auf „unbekannten“ Daten (Testdaten) funktioniert.

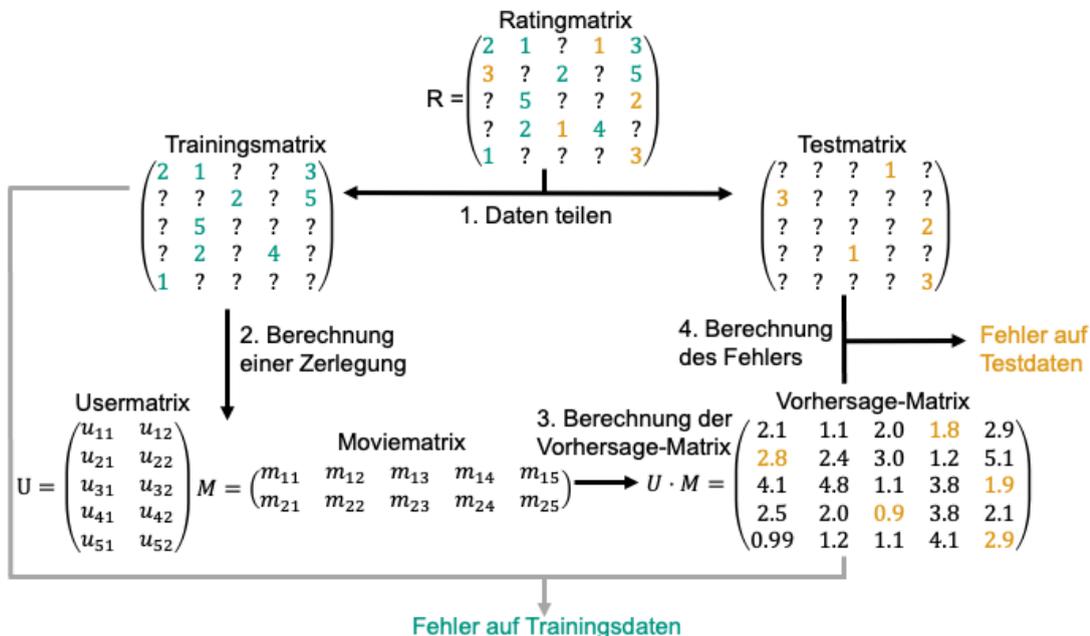


Abb. 17.3: Unterteilung der bekannten Daten in Trainings- und Testdaten

### Arbeitsblatt 5: Anwendung auf den Netflixdatensatz

Ziel ist die Anwendung des erarbeiteten Modells und des Optimierungsverfahrens auf den Netflixdatensatz. Dazu wird dieser zunächst geladen und ein kleinerer Teildatensatz ausgewählt. Anschließend werden die Daten in Trainings- und Testdaten

unterteilt. Die Idee dieser Unterteilung wird als Gif, welches die Schritte von Abbildung 17.3 nacheinander einblendet, auf dem Arbeitsblatt erneut veranschaulicht. Anschließend wenden die Lernenden das ALS-Verfahren auf den Trainingsdatensatz an, berechnen eine Zerlegung und lassen sich den quadratisch gemittelten Fehler für die Trainings- und Testdaten ausgeben.

Die Lernenden führen eine Parameterstudie für  $\lambda$  und  $n_f$  durch. Ein Teil der in der Parameterstudie einzustellenden Werte ist auf dem Antwortblatt vorgegeben. Basierend auf den dokumentierten Ergebnissen für den Fehler bezüglich der Test- und Trainingsdaten beschreiben und begründen die Lernenden den Einfluss der beiden Parameter. Den Lernenden steht es frei für beliebige weitere Parameterwahlen eine Zerlegung der Ratingmatrix zu bestimmen.

### **Zusatzblatt: Eigene Empfehlungen**

Dieses Zusatzblatt können die Lernenden am Ende von Arbeitsblatt 5 bearbeiten. Ein Teil des Netflixdatensatzes wird erneut geladen. Die Lernenden geben für zehn vorgegebene (populäre, dennoch alte) Filme eine Bewertung ab. Die Bewertungen der Lernenden werden als weitere Zeile in die Trainingsmatrix aufgenommen. Eine Faktorisierung der erweiterten Trainingsmatrix wird bestimmt. Den Lernenden wird dann eine individuelle Filmempfehlung ausgegeben.

### **Folien für die Abschlussdiskussion**

Zunächst werden die Ergebnisse der Lernenden aus der Parameterstudie gesammelt („Wer hat das beste Ergebnis auf den Testdaten erzielt?“). Anschließend erläutern die Lernenden ihre Beobachtungen bezüglich des Einflusses der beiden Parameter auf den Trainings- und Testfehler.

Die Lernenden reflektieren verschiedene Anwendungsmöglichkeiten der beschriebenen Modelle kritisch. Es werden ethisch bedenkliche Anwendungen und deren gesellschaftliche Auswirkungen aufgezeigt und die Erfahrungen und persönlichen Meinungen der Lernenden eingebunden. Grenzen der beschriebenen Modelle werden diskutiert und Ideen der Lernenden für mögliche Modellverbesserungen gesammelt. Konkret werden folgende Punkte diskutiert:

- Das Kaltstartproblem
- Die Einschätzung der Lernenden, ob sie Empfehlungssysteme für sich persönlich als sinnvoll und hilfreich empfinden.
- Die Manipulation von Empfehlungssystemen
- Die kritische Diskussion, inwieweit eine De-Anonymisierung des Datensatzes für Nutzerinnen und Nutzer problematisch sein kann.

Abschließend werden zentrale Schritte des Modellierungsprojektes resümiert.

## **17.2.2 Lernmaterial für Modellierungswochen**

Neben dem Lernmaterial für die stärker angeleiteten Modellierungsprojekte wurde zudem Material für offenes Modellieren im Rahmen einer Modellierungswoche konzi-

piert. Das Material sollte die unangeleitete Entwicklung eines Empfehlungssystems durch die Lernenden erlauben. Das Lernmaterial ist in Anhang F.10 zu finden.

### Die Problembeschreibung

Die Lernenden erhalten zu Beginn des Projektes eine kurze Beschreibung der Problemstellung: die Entwicklung eines Empfehlungssystems basierend auf den originalen Daten des Netflix Preises. Wie bei der Ausschreibung des Netflix Preises wird das Fehlermaß (der RMSE), mit dem das System validiert werden soll, vorgegeben. Die Problemstellung wird zu Beginn der Modellierungswoche in einem kurzen fünf-minütigen Vortrag von der Betreuungsperson der Gruppe vorgestellt.

### Die Teildatensätze

Da den Lernenden eingeschränkte Rechenressourcen zur Verfügung stehen, soll das Empfehlungssystem auf kleineren Teildatensätzen entwickelt und getestet werden. Konkret werden drei Trainings- und Testdatensätze bereitgestellt, die sich in ihrer Größe unterscheiden. Die Datensätze sind gleichermaßen dünnbesetzt, wie der gesamte Netflixdatensatz. Neben den Bewertungen der ausgewählten Nutzerinnen und Nutzer sowie Filme in den Teildatensätzen erhalten die Lernenden zudem die Zuordnung der Filme zu Titel und Erscheinungsjahren.

### Die Jupyter Notebooks

Den Lernenden wird Zugriff auf ein CoCalc-Projekt (vgl. Abschn. 5.3) gegeben, in dem sich drei Jupyter Notebooks befinden:

- Ein Notebook<sup>1</sup>, in dem die Lernenden Hinweise und Beispiele zur Verwendung von Python<sup>2</sup> finden.
- Ein Notebook, in dem der gesamte Datensatz des Netflix Preises eingelesen und verschiedene Visualisierungen bereitgestellt werden.
- Ein Notebook, in dem die Lernenden drei verschieden große Teile des originalen Netflixdatensatzes einlesen und visualisieren können. Dieses Notebook dient als Ausgangspunkt für die Implementierung des eigenen Empfehlungssystems.

## 17.3 Lehrplananbindung

Im Folgenden wird die Anbindung des Lernmaterials für die angeleiteten Projekte an die Mathematiklehrpläne der Bundesländer<sup>3</sup> Nordrhein-Westfalen (NRW), Baden-

---

<sup>1</sup>Dieses Notebook wurde von wissenschaftlichem Personal des CAMMP Teams und nicht explizit im Rahmen dieser Arbeit entwickelt.

<sup>2</sup>Den Betreuungspersonen der Modellierungswochen ist es freigestellt, welche der verfügbaren Programmiersprachen (Python, Julia, Octave) die betreute Lerngruppe nutzen soll. Die Gruppenbetreuerin der ersten Erprobung (wissenschaftlichen Mitarbeiterin des KIT) entschied sich für Python.

<sup>3</sup>Die genannten Bundesländer wurden gewählt, da die Mehrheit der durchgeführten Modellierungsveranstaltungen mit Lernenden dieser Ländern stattgefunden hat.

Württemberg (BW) und Rheinland-Pfalz (RLP) sowie übergeordnet an die Bildungsstandards der Kultusministerkonferenz (KMK) beschrieben. Es wird ersichtlich, dass das Material zahlreiche Inhalte aufgreift, die in den Lehrplänen verankert sind, was die innermathematische Reichhaltigkeit des Lernmaterials erneut hervorhebt. Wie bereits das Lernmaterial zu den Solarkraftwerken ist auch das Material rund um Empfehlungssysteme zudem außermathematisch reichhaltig. Insbesondere an das Fach Informatik gibt es zahlreiche Anknüpfungspunkte, die ebenfalls in diesem Abschnitt aufgezeigt werden.

### Mathematiklehrpläne

In den Bildungsstandards wird **Daten und Zufall** als eine von fünf Leitideen, die das mathematische Curriculum spiralförmig durchziehen sollen, benannt (vgl. [KMK Mathe Sek. I, 2003](#), S. 9). Dort und auch in den Lehrplänen der untersuchten Bundesländer wird gefordert, dass die Lernenden mit unterschiedlichen Darstellungsformen von Daten umgehen lernen und Daten unter Verwendung von Kenngrößen (z. B. Mittelwert und Standardabweichung) interpretieren können. Im Lehrplan BWs heißt es bereits für die Klassenstufen 5/6:

„Die Schülerinnen und Schüler planen Datenerhebungen und sammeln systematisch Daten. Sie beherrschen unterschiedliche Darstellungsformen, um aus ihnen die für eine Fragestellung relevanten Daten zu entnehmen oder um selbst Daten zu visualisieren. Sie sind in der Lage, verschiedene Darstellungsformen derselben Datenmenge untereinander vergleichend – auch hinsichtlich möglicher Irreführung – zu beurteilen und die vorliegenden Daten auszuwerten“ ([Bildungsplan BW Mathe Sek. I und II, 2016](#), S. 22).

Durch die aktive, reflektierte Auseinandersetzung mit realen Daten bietet das Lernmaterial die Möglichkeit Kompetenzen im (kritischen) Umgang mit authentischen Datensätzen zu schulen.

Neben der Leitidee Daten und Zufall greift das Lernmaterial insbesondere Inhalte aus dem Bereich der **linearen Algebra** auf. Das **Skalarprodukt** wie auch **Vektoren** sind zentral. Beides ist in den betrachteten Lehrplänen für die Sekundarstufe II enthalten. Die Inhalte des Lernmaterials bieten darüber hinaus eine gute Möglichkeit, um den geometrischen Winkelbegriff durch die Betrachtung des Skalarprodukts und das Arbeiten mit (höherdimensionalen) Vektoren zu verallgemeinern. Matrizen tauchen in den Lehrplänen BWs und NRWs lediglich im Rahmen der Darstellung von **linearen Gleichungssystemen** in Matrix-Vektorschreibweise auf (vgl. [Lehrplan NRW Mathe Sek. II, 2014](#); [Bildungsplan BW Mathe Sek. I und II, 2016](#)).

Im Lehrplan RLPs sind Matrizen in praktischen Anwendungen zudem als mögliches Wahlpflichtgebiet verankert. Neben dem Aufstellen und Lösen von linearen Gleichungssystemen in Matrix-Vektorschreibweise wird dort die Matrix-Multiplikation und das Invertieren von Matrizen benannt. Es wird zudem eine verstärkte Anwendungsorientierung betont:

„Der Schwerpunkt bei diesem Wahlpflichtgebiet liegt auf dem Mathematisieren von Sachproblemen und nicht auf dem Erlernen und Ein-

üben des Matrizenkalküls. Die einzelnen Matrizenoperationen werden anhand konkreter Sachaufgaben, zum Beispiel aus der Industrie oder Wirtschaft, erarbeitet und dann in komplexeren Problemstellungen angewendet“ ([Lehrplan RLP Mathe Sek. II, 2015](#), S. 28).

Darüber hinaus bietet das Lernmaterial die Möglichkeit das Verständnis **funktionaler Zusammenhänge** (ebenfalls eine Leitidee in den Bildungsstandards) beim Aufstellen und Untersuchen der Fehlerfunktion zu schulen. Zudem wird bei der eigenständigen Entwicklung des Optimierungsverfahrens (Langversion von AB 4) an Inhalte aus dem Bereich **Differentialrechnung**, unter anderem die Kettenregel, angeknüpft. Auch werden Grundlagen der Extremwertbestimmung von Funktionen einer Veränderlichen aufgegriffen (Nullstellen der Ableitung) und auf mehrere Dimensionen erweitert (Nullstellen des Gradienten), dabei sind sogleich Kenntnisse zum Lösen linearer Gleichungssysteme erforderlich. Die hier genannten Inhalte sind in den Lehrplänen der Sekundarstufe II in allen drei Bundesländern enthalten.

Die Erarbeitung und Anwendung des alterierenden Optimierungsverfahrens bietet zudem die Möglichkeit **algorithmisches Denken** zu fördern. Algorithmen sind unter der Leitidee **Algorithmus und Zahl** in den Bildungsstandards verankert. Diese Leitidee umfasst „die Kenntnis, das Verstehen und das Anwenden mathematischer Verfahren, die prinzipiell automatisierbar und damit einer Rechnernutzung zugänglich sind“ ([KMK Mathe Sek. II, 2012](#)).

### **Informatiklehrpläne**

Den kritischen Umgang mit Daten zu schulen stellt einen zentralen Bestandteil des Informatikunterrichts in NRW dar. Das Lernmaterial bietet sich insbesondere durch die integrierten kritischen Diskussionen, sowohl der Daten als auch von Empfehlungssysteme an sich, für fächerübergreifende Projekte mit der Informatik an. Bereits im Informatikunterricht der Klassen 5 und 6 sollen die Lernenden in NRW „anhand von ausgewählten Beispielen die Verarbeitung und Nutzung personenbezogener Daten“ beschreiben und „anhand von Beispielen aus ihrer Lebenswelt Nutzen und Risiken beim Umgang mit eigenen und fremden Daten auch im Hinblick auf Speicherorte“ erläutern ([Lehrplan NRW Info Sek. I, 2021](#), S. 19). Diskussionen zu einem kritischen Umgang mit personenbezogenen Daten und zu Chancen und Risiken von informationstechnischen Systemen sind zudem im Lehrplan RLPs verankert (vgl. [Lehrplan RLP Info Sek. II, o. D.](#)) und werden auch im Lehrplan der Sekundarstufe II NRWs expliziert:

„Informatiksysteme stehen in intensiver Wechselwirkung mit Individuum und Gesellschaft. Ihr Einsatz hat weitreichende Konsequenzen für unsere Lebens- und Arbeitswelt. Handlungsspielräume müssen im Spannungsfeld von Rechten und Interessen des Individuums, gesellschaftlicher Verantwortung und möglichen Sicherheitsrisiken wahrgenommen werden“ ([Lehrplan NRW Info Sek. II, 2014](#), S. 18).

Dass sich die grundlegenden Strategien des überwachten MLs, die im Lernmaterial aufgegriffen werden, auch für eine Anbindung an den Informatikunterricht eignen, verdeutlicht der folgende Auszug aus dem Lehrplan RLPs:

„Computerunterstützte Informationsverarbeitung läuft immer in drei Schritten ab: Information muss zunächst geeignet in Form von Daten dargestellt werden; erst diese Daten können maschinell zu neuen Daten verarbeitet werden; durch Interpretation der erzeugten Daten gewinnt man schließlich neue Information. Um diesen Prozess zu verstehen, sollen Schülerinnen und Schüler sich mit verschiedenen Aspekten dieser Schritte beschäftigen“ (Lehrplan RLP Info Sek. II, o. D., S. 16).

### **Anbindung an weitere Schulfächer**

Mit dem rasanten Fortschritt im Bereich KI und den zahlreichen Möglichkeiten und Chancen von KI-Systemen, die unser Leben einfacher machen (sollen), entstehen sogleich zentrale ethische, gesellschaftliche und soziale Herausforderungen. Durchaus gibt es eben auch negative Schlagzeilen und bedenkliche Auswirkungen von KI-Systemen, wie die Diskriminierung einzelner Bevölkerungsgruppen, die gesellschaftskritisch und ethisch diskutiert werden müssen. Mittlerweile gibt es ganze Forschungsgruppen an Universitäten,<sup>4</sup> Abteilungen in großen Technologieunternehmen oder gar Studiengänge,<sup>5</sup> die sich dem Thema Ethik und KI widmen.

Welchen Einfluss die Entwicklung von KI-Systemen auf großen Datensätzen hat und inwieweit diese bei einem unreflektierten Einsatz Diskriminierungen und soziale wie auch finanzielle Benachteiligung verstärken können, diskutiert unter anderem O’Neil (2016) an zahlreichen Beispielen (Vergabe von Krediten, Zulassungen zum College etc.). Solche Diskussionen auch mit Lernenden zu führen scheint in der heutigen Zeit wesentlich. Das Lernmaterial bietet die Möglichkeit Gefahren und Chancen von Empfehlungssystemen, als eine spezielle KI-Anwendung, zu diskutieren. Das Material kann als Ausgangspunkt dienen, um tiefer in die ethische Diskussion rund um KI-Systeme im Rahmen von fächerübergreifenden Projekten mit dem Schulfach Ethik oder Philosophie einzusteigen. Exemplarisch sei hier auf den Lehrplan BWs verwiesen, in dem für den Ethikunterricht der Klassen 11/12 sogar explizit gefordert wird, dass die Lernenden „Grundzüge der digitalen Revolution beschreiben und ethisch relevante Probleme der Digitalisierung identifizieren (zum Beispiel Künstliche Intelligenz, Big Data, Cyberwar, Post-Privacy) [können]“ (Bildungsplan BW Ethik Sek. I und II, 2016). Themen wie Privatsphäre, Fairness, Überwachung, Fehlinformationen oder Bias könnten in einem solchen Projekt diskutiert werden.

<sup>4</sup>Zum Beispiel das *Institute for Ethics in Artificial Intelligence* an der TU München, weitere Informationen unter: <https://ieai.mcts.tum.de>, letzter Aufruf: 18.12.2021.

<sup>5</sup>Zum Beispiel der Masterstudiengang *AI Ethics and Society* an der Universität in Cambridge, weitere Informationen unter: [www.ice.cam.ac.uk/course/mst-artificial-intelligence-ethics-and-society](http://www.ice.cam.ac.uk/course/mst-artificial-intelligence-ethics-and-society), letzter Aufruf: 18.12.2021.

# 18 Erfahrungswerte und didaktische Reflexion

Um eine erste Einschätzung vornehmen zu können, inwieweit das Lehr- und Lernmaterial für den Einsatz in mathematischen Modellierungsprojekten geeignet ist, wurde dieses in diversen Modellierungsveranstaltungen mit Lernenden erprobt und basierend auf den Erfahrungen der einzelnen Durchführungen iterativ verbessert. Ziel der Erprobungen war es, das Material im Hinblick auf Interesse und Motivation der Lernenden, auf Lernzuwachs und Angemessenheit des Schwierigkeitsgrades sowie auf die Einsatzmöglichkeiten in heterogenen Lerngruppen bewerten zu können.

## 18.1 Durchgeführte Modellierungsveranstaltungen

Die durchgeführten Veranstaltungen zum Thema Empfehlungssysteme sind Tabelle 18.1 zu entnehmen. Insgesamt haben 120 Lernende davon 77 männlich (m), 42 weiblich (w) und 1 divers (d), an den angeleiteten Modellierungsprojekten teilgenommen. An dem offenen Modellierungsprojekt im Rahmen einer Modellierungswoche haben sechs Lernende teilgenommen. Fünf Veranstaltungen (Nr. 1, 2, 5, 7, 8, vgl. Tab. 18.1) haben mit Lernenden stattgefunden, die sich explizit für die Veranstaltung angemeldet haben. Es ist davon auszugehen, dass diese Lernende Interesse an Mathematik beziehungsweise mathematischer Modellierung hatten. Die übrigen vier Veranstaltungen haben mit ganzen Oberstufenkursen und damit auch heterogen interessierten Lernenden stattgefunden.

## 18.2 Evaluation

Die Auswertung der Erprobungen erfolgt basierend auf den Beobachtungen, die die Durchführenden während der Modellierungsveranstaltungen gemacht und dokumentiert haben. Zudem füllten die Lernenden nach den Veranstaltungen eine schriftliche Befragung aus. Diese diente dazu, die persönlichen Erfahrungen der Lernenden, ihre Meinung zum Arbeitsmaterial, zum Ablauf des Workshops sowie zu dessen Thema abzufragen. Die Ergebnisse der schriftlichen Befragung aller Erprobungen zum Thema Empfehlungssysteme im Rahmen von angeleiteten Modellierungsprojekten sind Anhang G.2.1 und die Ergebnisse der Modellierungswoche Anhang G.2.2 zu entnehmen.

**Tab. 18.1:** Durchgeführte Modellierungsveranstaltungen zum Thema Empfehlungssysteme. Alle Veranstaltungen haben digital stattgefunden. In der mit \* markierten Veranstaltung haben die Lernenden mit dem Material für offene Modellierungsprojekte gearbeitet, in den übrigen kam das Material für angeleitete Projekte zum Einsatz.

Nr.	Veranstaltung	Anzahl Lernende	Jgst.	Zeitlicher Rahmen
1	Modellierungs-AG RWTH (NRW)	7 (6m/1w)	11–12	01–02/2021 8h (4 x 2h)
2	Modellierungs-AG KIT (BW)	5 (2m/3w)	8–12	01–02/2021 8h (4 x 2h)
3	U-Reihe Leistungskurs Mathematik Gym. (RLP)	13 (4m/9w)	12	02–03/2021 6h 24Min (3.5 x 110Min)
4	U-Reihe Vertiefungskurs Mathematik Gym. (BW)	8 (6m/2w)	11	02–03/2021 4.5h (3 x 1.5h)
5	Schnupperkurs Mathematik des KIT (BW)	22 (17m/4w)	10–12	04–05/2021 9h (6 x 1.5h)
6	Modellierungstag Mathematikkurs Gym. (NRW)	24 (21m/3w)	10	06/2021 4h 10Min (1 Tag)
7	CAMMP week Modellierungswoche (NRW, BW)*	6 (4m/2w)	10–11	06/2021 5 Tage
8	Schüleruni Mathematik RWTH Aachen (NRW)	12 (5m/6w/1d)	10–12	07/2021 5h 30Min (1 Tag)
9	Modellierungstag Leistungskurs Mathematik Gym. (Niedersachsen)	29 (16m/13w)	12	07/2021 4h 20Min (1 Tag)

### 18.2.1 Ergebnisse der angeleiteten Modellierungsprojekte

#### Zentrale Beobachtungen

Zunächst werden zentrale Beobachtungen, die während der Modellierungsveranstaltungen von den Durchführenden gesammelt wurden, zusammenfassend dargelegt. Zur besseren Übersicht werden die Beobachtungen entlang des Ablaufs des Workshops beschrieben. Bei der Darstellung der Beobachtungen wird insbesondere auf Schwierigkeiten der Lernenden bei der Bearbeitung des Lernmaterials eingegangen. Es wird erläutert, welche Änderungen am Material vorgenommen wurden, um den Schwierigkeiten zu begegnen.

### **Einstieg in die Problemstellung und potentielle Datenakquise**

In der Diskussion zu Beginn der Modellierungsprojekte war zu beobachten, dass die Lernenden sich zahlreich und mit vielfältigen Ideen einbrachten. Als mögliche Ziele, die mit Empfehlungssystemen verfolgt werden, nannten die Lernenden unter anderem Kundenbindung, Kundenwerbung, Gewinnmaximierung und Konsumerhöhung. Auch die Ideen bezüglich der Informationen und Daten, die ausgenutzt werden könnten, um Empfehlungssysteme zu entwickeln, waren facettenreich. Einige Beispiele, die von Lernenden wiederholt genannt wurden, lauten:

- Informationen über Kauf- und Nutzerverhalten auszunutzen und Produkte vorzuschlagen, die ähnlich zu Produkten sind, die der Nutzerin oder dem Nutzer bisher gefallen haben
- Bewertungen nutzen, die die Nutzerinnen und Nutzer für Filme abgegeben haben, und dann ähnliche Filme wie die, die von den Nutzenden gut bewertet wurden, vorschlagen
- Anhand von Nutzerinnen und Nutzern mit ähnlichem Kauf- oder Konsumverhalten Vorschläge aussprechen
- Informationen, wie den Wohnort, das Alter, das Geschlecht oder andere Interessen, der Nutzerinnen und Nutzer ausnutzen
- Unterschiedliche Datenquellen kombinieren (bspw. das Nutzerverhalten auf Netflix mit den Suchverläufen auf Google)
- Produkte in Kategorien einteilen und Interessenbereiche einer Nutzerin / eines Nutzers feststellen
- Die Kategorien der Filme mit den Eigenschaften der Nutzerinnen und Nutzer kombinieren
- Untersuchen, was der Mehrheit der Nutzerinnen und Nutzern gefällt, um Mehrheitsmeinungen dann auf andere Personen zu übertragen

Im Hinblick auf die in Abschnitt 15.1 vorgestellten Ansätze für Empfehlungssysteme, die in der Praxis Einsatz finden, wird ersichtlich, dass die Lernenden sowohl die Grundgedanken des kollaborativen Filterns als auch die Datengrundlage von inhaltsbasierten sowie hybriden Methoden benannt haben.

### **Datenanalyse**

Bei der Bearbeitung des Arbeitsblattes zur Datenanalyse waren vereinzelt Startschwierigkeiten bei der Bedienung der Jupyter Notebooks zu beobachten. Die Lernenden wussten zum Teil (trotz Beschreibung auf dem Arbeitsblatt) nicht, wie sie Codefelder ausführen. Um dies zu vermeiden, wurde bei den nachfolgenden Erprobungen eine längere Einführung in die Workshopumgebung gegeben. Es wurde explizit vorgeführt, wie Codefelder ausgeführt werden. Im Hinblick auf die Bearbeitung der Aufgaben waren kaum Schwierigkeiten zu beobachten. Lediglich einzelne Lernende hatten Probleme, für die Beantwortung der Frage zur *Änderung der Verteilung der Bewertungen im Datensatz über die Zeit* eine geeignete Darstellungsform zu

wählen. Sie versuchten die Frage mithilfe der Tabelle 17.2 zu beantworten. Sowohl beim Rundgang durch die Kleingruppen als auch bei der anschließenden Plenumsdiskussion zeigte sich, dass die Mehrheit der Lernenden in der Lage war, gefragte Informationen aus den dargebotenen Darstellungsformen des Netflixdatensatzes abzulesen. Zudem zeigte sich, dass die Mehrheit der Lernenden erkannte, dass einzelne Spalten beziehungsweise Zeilen in der Bewertungsmatrix identisch sind oder sich als Linearkombinationen anderer Spalten beziehungsweise Zeilen ergeben. Sie interpretierten dies als Ähnlichkeiten zwischen Nutzenden oder Filmen.

In den Plenumsdiskussionen ergab sich ein vielfältiges Bild hinsichtlich der Ideen, die die Lernenden zur Entwicklung eines Empfehlungsmodells basierend auf der Ratingtabelle äußerten. Exemplarische Ansätze waren:

- Unterscheiden, ob eine Person neu ist oder nicht. Ist sie neu, so schlagen wir einen der aktuell beliebtesten Filme vor. Hat sie schon Bewertungen abgegeben, so schauen wir, welche Nutzerinnen und Nutzer ähnlich bewertet haben und schlagen Filme vor, die diese Nutzerinnen und Nutzer gut fanden.
- Ausnutzen, dass es Filme mit ähnlichen Bewertungsmustern gibt
- Untersuchen, welches Genre eine Nutzerin oder ein Nutzer gut bewertet hat, und Vorschläge aus dem selben Genre oder einem ähnlichen Genre unterbreiten
- Filme in Gruppen einteilen, beispielsweise nach Genres oder Filmreihen
- Notfalls (bspw. wenn zu wenige Daten vorliegen) die durchschnittliche Bewertung jeder Person oder jedes Films ausrechnen und berücksichtigen.
- Verteilung der Bewertungen untersuchen und ausnutzen. Manche Personen haben eher hoch, andere eher tief bewertet. Das sollte berücksichtigt werden.

### Erste Modellentwicklung

Bei der Bearbeitung von Arbeitsblatt 2 waren kaum Schwierigkeiten zu beobachten. Auch in den Sicherungsphasen im Plenum wurden die Ergebnisse zu den Aufgaben in allen Durchführungen korrekt erläutert. Lediglich bei einzelnen Erprobungen wurde der Zusammenhang zum Skalarprodukt direkt erkannt. Einige Lernende kannten Matrizen bereits aus dem Mathematikunterricht. Sie erfassten die Berechnung der Vorhersage als Produkt zweier Matrizen frühzeitig und argumentierten auch in der Folge damit.

### Händisches Berechnen einer Zerlegung

Bei den ersten Erprobungen sollten die Lernenden für **mehrere** Bewertungsmatrizen händisch eine Zerlegungen berechnen. Dies stellte sich als äußerst zeitaufwendig heraus. Deswegen wurde das Lernmaterial auf die Berechnung **einer** Zerlegung reduziert. Zudem wurde eine mehrstufige Tippkarte entwickelt, auf die die Lernenden bei der händischen Berechnung einer Zerlegung zugreifen konnten. Auf dieser wird ein möglicher Ansatz zur Berechnung einer Zerlegung an einem kleinen Beispiel veranschaulicht. Bei den nachfolgenden Erprobungen war zu beobachten, dass die

Lernenden auf diese Tippkarte zugegriffen.

Bei der Berechnung einer Zerlegung waren unterschiedliche Strategien zu beobachten. Diese reichten vom einfachen Ausprobieren bis hin zum Festlegen einzelner Werte und dem anschließenden Lösen eines Gleichungssystems:

„Wir haben einfach Werte festgelegt und dann bei den anderen Werten geschaut, dass das passt. Dazu haben wir dann Gleichungen aufgeschrieben, in denen vier Unbekannte waren. Da haben wir die festgelegten Werte eingesetzt und dann die anderen mithilfe des Einsetzungsverfahrens bestimmt“ (Zitat einer teilnehmenden Person).

### Definition eines Fehlermaßes

Bei der eigenständigen Definition eines Fehlermaßes durch die Lernenden wurden in den Erprobungen diverse Ansätze genannt:

- **Summe der einfachen (orientierten) Abweichungen.** In den Plenumsdiskussionen entgegneten andere Lernende auf diesen Vorschlag, dass sich in der Summe negative und positive Terme ergeben, die sich aufheben. Sie schätzten diesen Ansatz als nicht sinnvoll ein.
- Das Fehlermaß, was am häufigsten benannt wurde, war die **Summe der absoluten Abweichungen.** Einzelne Lernende schlugen zudem vor, die Summe durch a) die Anzahl bekannter Bewertungen oder b) die Anzahl aller Einträge in der Bewertungsmatrix zu dividieren, um das Ergebnis vergleichbar zu machen.
- Von einzelnen Lernenden wurde die **Summe der quadratischen Abweichungen** genannt.

In der Diskussion zu Vor- und Nachteilen der unterschiedlichen Fehlermaße nannten die Lernenden, dass die (gemittelte) Summe der absoluten Abweichungen besser zu interpretieren sei (als *Durchschnitt über alle Abweichungen*). Andere Lernende merkten an, dass die Summe der Fehlerquadrate den Vorteil hat, dass größere Abweichungen stärker ins Gewicht fallen. Sie stellten Bezug zum Physikunterricht her, aus dem sie dieses Fehlermaß beim Anpassen von experimentellen Daten kennengelernt hatten. Vereinzelt wurde von Lernenden kritisiert, dass die Entscheidung, warum im Verlauf des Workshops die Summe der Fehlerquadrate beibehalten wird, nicht transparent nachvollziehbar war. Aus diesem Grund wurde diese Diskussion in nachfolgenden Erprobungen nach der Entwicklung des Optimierungsverfahrens erneut aufgegriffen.

### Entwicklung und Anwendung des Optimierungsverfahrens

Bei der Vorstellung des ALS-Verfahrens, wurde Bezug zum Lösen von Extremwertproblemen aus der Schule genommen. Der von den Lernenden genannte Ansatz war stets das Bestimmen und Nullsetzen der ersten Ableitung. Bei Projekten, die über mehrere Wochen hinweg stattfanden (Unterrichtsreihen), war zu beobachten, dass die Lernenden Schwierigkeiten hatten, sich das ALS-Verfahren ins Gedächtnis zu

rufen. Deswegen wurde ein entsprechendes Erklärvideo erstellt und in die Arbeitsblätter eingebunden.

Bei der Entwicklung des Optimierungsverfahrens (AB 4 Langversion) waren vermehrt Schwierigkeiten beim Anwenden der Kettenregel zur Bestimmung der partiellen Ableitungen zu beobachten. Deswegen wurde in späteren Durchführungen eine gestufte Tippkarte bereitgestellt.

In den Diskussionsphasen konnten die Lernenden den Einfluss des Regularisierungsparameters auf den Test- und Trainingsfehler beschreiben. Sie erfassten, dass der Regularisierungsterm dazu führt, dass die Einträge der beiden Matrizen  $U$  und  $M$  betragsmäßig nicht beliebig groß werden. Eine Schülerin / ein Schüler beschrieb den Einfluss eines großen Wertes für den Regularisierungsparameter in einer Diskussionsphase wie folgt:

„Dann wurden die Ergebnisse für die Einträge von  $U$  und  $M$  sehr klein. Also die werden ja jeweils quadriert und aufsummiert und damit der ganze Term klein sein kann, müssen die Einträge auch klein sein. Wenn man  $\lambda$  etwas größer als Null gewählt hat, dann lagen auch die Ergebnisse für die unbekanntenen Daten in einem realistischen Bereich.“

Bei der Anwendung des ALS-Verfahrens auf einen Teil des Netflixdatensatzes erkannten die Lernenden, dass der Fehler sowohl auf den Testdaten als auch auf den Trainingsdaten ansteigt, wenn der Regularisierungsparameter zu groß gewählt wird. Der Einfluss der Anzahl an Eigenschaften  $n_f$  wurde hingegen nicht immer erfasst. Eine Schülerin / ein Schüler begründete die Beobachtung, dass der Trainingsfehler bei einer großen Anzahl an Eigenschaften  $n_f$  sehr klein wird, wie folgt:

„Viele features führen dazu, dass der PC natürlich viel mehr Auswahlmöglichkeiten hat. Er kann die Trainingsdaten also beliebig genau darstellen.“

### Abschlussdiskussion

Es war zu beobachten, dass die Lernenden sich rege in die Diskussionen einbrachten und diverse Meinungen zu Nutzen, Gefahren und persönlichen Erfahrungen äußerten. Exemplarische Meinungen der Lernenden zu Empfehlungssystemen waren:

- Manipulation: „So Systeme und Algorithmen können künstlich Bedürfnisse erzeugen, so den Konsum ankurbeln und uns manipulieren“ (Zitat einer teilnehmenden Person).
- Eine Vielzahl der Lernenden empfand Empfehlungssysteme für Filme als hilfreich, äußerte jedoch Kritik daran, dass man sich gegebenenfalls in einer Blase befindet und nichts Neues (im Sinne von andersartig) vorgeschlagen wird. Die Filterblasen wurden insbesondere bei Nachrichtensystemen als problematisch eingeschätzt.

In den Diskussionen zu einer möglichen De-Anonymisierung der Daten schätzten einzelne Lernende es als unproblematisch ein, wenn ihr eigenes Nutzer- / Konsumverhalten auf Netflix öffentlich würde. Die Mehrheit der Lernenden fand es hingegen

äußerst problematisch, wenn die eigenen Daten weitergegeben und veröffentlicht würden. Zudem wurde von den Lernenden kritisiert, dass sich Unternehmen vielfach Hintertüren offen halten, um die eigenen Daten auf nicht offensichtliche Weise zu verwenden oder weiterzugeben.

Auch die Ideen der Lernenden, wer ein Interesse an der Manipulation von Empfehlungssystemen haben könnte, waren vielfältig und reichten von Filmproduzenten, die ihren Film mit Fakeaccounts besonders gut bewerten, bis hin zu Konkurrenzunternehmen, die die Empfehlungssysteme anderer Unternehmen durch gefälschte Bewertungen schädigen wollen.

Zum Lösen des Kaltstartproblems schlugen die Lernenden vor, bereits bei der Registrierung auf der Plattform Interessen abzufragen, entweder bezüglich der Genres oder im Hinblick auf konkrete Filme, die von der neuen Nutzerin / dem neuen Nutzer zu bewerten wären.

In der Diskussion zu weiteren Verbesserungen des entwickelten Modells warfen einzelne Lernende ein, dass die Annahme *fehlende Bewertung = Film nicht geschaut* nicht sinnvoll ist, da Nutzerinnen und Nutzer Filme gegebenenfalls anschauen, aber nicht bewerten. Zudem ist es möglich, dass sie Filme abbrechen. Man sollte somit Daten zur Dauer, die ein Film angeschaut wird, berücksichtigen. Zudem wurde angemerkt, dass es denkbar ist, dass Filme bewusst nicht geschaut werden und dass dies in das Modell einfließen könnte:

„Man könnte Gewichte einführen. Also für die fehlenden Einträge einen Faktor setzen. Weil man ja manchmal absichtlich einen Film nicht schaut. Also sollte da dann im Prinzip eine schlechte Bewertung stehen. Aber ich bin nicht sicher, welchen Faktor man wählen sollte, weil manchmal hat man ja wirklich einen Film nur noch nicht entdeckt und deswegen noch nicht gesehen“ (Zitat einer teilnehmenden Person).

Im Hinblick auf die zur Differenzierung entwickelten Lernmaterialien sei abschließend angemerkt, dass alle Zusatzblätter mindestens einmal zum Einsatz kamen. Auch war zu beobachten, dass die Lernenden von den Tippkarten Gebrauch machten. Zudem wurde wiederholt nach Arbeitsblatt 3 differenziert, indem ein Teil der Lernenden die Langversion von Arbeitsblatt 4 bearbeitete.

## Ergebnisse der schriftlichen Befragung

Die Ergebnisse der schriftlichen Befragung werden hier zusammengefasst. Es sei angemerkt, dass a) nicht alle Lernenden überhaupt die Befragung ausgefüllt haben (daher die Abweichung der Anzahl an Antworten aus der Evaluation zu den in Tabelle 18.1 dargestellten Schülerzahlen) und dass b) nicht alle Lernenden, die an der Befragung teilgenommen haben, alle Fragen beantwortet haben.

Die Mehrheit der Lernenden gab an, dass sie durch den Workshop ein *besseres* oder *eher besseres* Verständnis von mathematischer Modellierung gewonnen hat (92/103). Dies wird auch wiederholt auf die Frage genannt, was die Lernenden durch den Workshop gelernt haben.

Auch hoben die Lernenden wiederholt hervor, dass der Workshop die Bedeutung

von Mathematik für den Alltag erfahrbar gemacht hat. Exemplarisch sei hier eine Auswahl an Schüleraussagen<sup>1</sup> zitiert, die dies als Antworten auf die Frage *Was hast du für dich persönlich durch die Teilnahme am Workshop gelernt?* eindrücklich unterstreichen:

- „Dass Mathematik sehr vielfältig ist und dass man dadurch sehr viel Neues lernen kann und vieles gut erklären kann.“
- „Dass Mathematik wichtiger ist, als ich anfangs annahm.“
- „Dass Mathe in alltäglichen Dingen wie sogar Netflix enthalten ist.“
- „Dass Matrizen auch Anwendung in alltäglichen Dingen finden und dort eine wichtige Rolle spielen. Dass die Grundlagen des maschinellen Lernens relativ gut zu verstehen sind und sich mithilfe von vereinfachten Beispielen erklären lassen.“

Das Thema des Workshops stuften 81/89 Lernende als *interessant* oder *eher interessant* ein. Dies wurde zudem vielfach auf die Frage *Hat dir etwas an dem Workshop besonders gut gefallen?* genannt. Auch betonten die Lernenden wiederholt, dass sie das Arbeiten in Kleingruppen und zudem den Wechsel aus Gruppenarbeit und Diskussionsphasen als sinnvoll empfunden haben.

„Ich fand den Kurs sehr schön und alles erklärt. Die Gruppeneinteilung war sehr hilfreich, sodass man mit Personen, die ein ähnliches Wissen teilen, zusammengearbeitet hat, was für eine angenehme Atmosphäre gesorgt hat. Die Diskussionen und anfänglichen Wiederholungen waren auch super.“

Zudem wurde mehrfach das „selbstständige Erarbeiten der Aufgaben“ beziehungsweise „eigenes Erproben“ und „dass man viele Dinge selber durchführen konnte“ positiv hervorgehoben. Obgleich das Lernmaterial (insbesondere im Vergleich zur Modellierungswoche) deutlich angeleiteter konzipiert ist, so war die wahrgenommene Eigentätigkeit angemessen.

Der Workshop hat zum Verständnis der Verwendung und Verarbeitung großer Datenmengen beigetragen, was folgende Schülerzitate auf die Frage *Was hast du für dich persönlich durch die Teilnahme am Workshop gelernt?* unterstreichen:

- „Ich habe einen interessanten Einblick erhalten, wie aus vielen Daten weitere Daten vorhergesagt werden können.“
- „Wie große Datenmengen mit Matrizen analysiert werden.“
- „Wo überall maschinelles Lernen vorhanden ist und wie es in groben Zügen funktioniert.“
- „Ich habe gelernt, kleinschrittig und logisch an große mathematische Probleme heranzugehen sowie das Rechnen mit Matrizen und allgemein den Umgang mit großen Datensätzen.“

<sup>1</sup>Rechtschreib- und Interpunktionsfehler wurden zur besseren Lesbarkeit der Schülerzitate korrigiert.

Im Hinblick auf die Gestaltung der Arbeitsblätter gaben 92/111 Befragte an, dass ihnen diese *eher gut* oder *gut* gefallen hat. Dies unterstreichen auch zahlreiche Freitextantworten, auf die Frage *Hat dir etwas an dem Workshop besonders gut gefallen?*

- „Die Aufgabenblätter und besonders das Verändern der Parameter im Code, sodass man deren Einfluss nachvollziehen konnte.“
- „Die Strukturierung mit den Code-Feldern war sehr gelungen und brachte Dynamik.“
- „Die informative und reaktionelle [sic; direkte; Anmerk. der Verf.] Kontrolle der Antworten.“

Die automatischen Rückmeldungen nach den Eingaben im Code bewertete die Mehrheit der Lernenden als *hilfreich* oder *eher als hilfreich* (99/112). Die Tippkarten wurden von 89/111 Lernenden als *hilfreich* oder *eher hilfreich* eingestuft. Zehn Lernende konnten diese Aussage nicht beurteilen, was vermutlich daran liegt, dass sie die Tippkarten nicht genutzt haben. Lediglich zwölf Lernende stimmten dieser Aussage *nicht* oder *eher nicht* zu.

Die Antworten auf die Aussage *Der Umgang mit der Programmiersprache Julia fiel mir schwer.* zeichnen ein gemischtes Bild. So gaben 36/111 Befragte an, dass dies *zutrifft* oder *eher zutrifft*. Dies lässt sich womöglich in Teilen darauf zurückführen, dass bei den ersten Erprobungen einzelne Hinweise zur Eingabe der Lösungen (bspw. die Verwendung von Punkten statt Kommata bei der Eingabe von Dezimalzahlen oder die Verwendung des Zeichens  $\wedge$  zum Potenzieren von Zahlen) noch nicht auf den Arbeitsblättern gegeben waren. Hingegen gaben 55/111 Lernende an, dass ihnen der Umgang mit der Programmiersprache *nicht* oder *eher nicht* schwerfiel. 20 Lernende gaben *nicht beurteilbar* an.

Als negativ wurde von einzelnen Befragten die Technik beziehungsweise der Server angemerkt. Dies lässt sich darauf zurückführen, dass bei einzelnen Erprobungen Instabilitäten eben dessen auftraten.

Im Hinblick auf den Schwierigkeitsgrad der Aufgaben gaben 31/111 Lernende an, dass sie diese *eher zu einfach* oder *zu einfach* fanden. Auf die Aussage *Die Aufgaben waren zu schwer* antworteten 36/112 mit *trifft eher zu* oder *trifft voll zu*. Lediglich 19/111 Lernende gaben an, dass sie die Aufgaben *eher nicht* eigenständig bearbeiten konnten und nur eine Person gab an, dass sie die Aufgaben *gar nicht* eigenständig bearbeiten konnte. Der Schwierigkeitsgrad ist im Schnitt als angemessen zu beurteilen. Es besteht jedoch Potential sowohl einzelne stärker herausfordernde Zusatzaufgaben als auch Aufgaben mit geringerem Anforderungsniveau bereitzustellen. Letztgenanntes scheint insbesondere im Hinblick auf die Durchführung in heterogenen Lerngruppen wesentlich.

Auf die Frage *Hast du im Workshop irgendetwas gar nicht verstanden?* antwortete die Mehrheit der Lernenden entweder mit *Nein* oder sie gaben an, dass sich dies spätestens in den Diskussionsphasen klären konnte. Die Antworten der Lernenden, für die etwas unverständlich geblieben ist, bezogen sich fast ausschließlich auf das Arbeitsblatt 4 in der Langversion (Bestimmung von Nullstellen des Gradienten). Zweimal wurde zudem angegeben, dass der Regularisierungsparameter nicht klar

geworden ist. Insgesamt gab die Mehrheit der Befragten an, dass die Inhalte *klar* oder *eher klar* vermittelt wurden (92/101).

Die Gesamtbewertung der Workshops fiel positiv aus. Die Workshops wurden im Durchschnitt mit der Schulnote 1.99 bewertet. Dies unterstreicht auch das Ergebnis auf die Aussage *Alles in allem hat mit der Workshop gut gefallen.*, worauf 105/110 Befragte mit *trifft eher zu* oder *trifft voll zu* antworteten.

### 18.2.2 Ergebnisse der Modellierungswoche

Sechs Lernende der Klassen 10 – 11 verschiedener Gymnasien aus BW und NRW haben im Rahmen einer mathematischen Modellierungswoche (vgl. Abschn. 4.2) an der Entwicklung eines Empfehlungssystems gearbeitet. Zu Beginn der Woche erhielten die Lernenden die in Abschnitt 17.2.2 beschriebenen Materialien. Die Modellierungswoche fand im Juni 2021 online statt. Die Problemstellung wurde Montagmorgen an die Lernenden ausgegeben. Die Lernenden hatten bis zu den Präsentationen am Freitagmorgen Zeit an der Problemstellung zu arbeiten.

Im Folgenden wird der Modellierungsprozess der Lernenden skizziert und wesentliche Beobachtungen im Hinblick auf die Eignung der Problemstellung für offene Modellierungsprojekte dargelegt. Zudem werden die Ergebnisse der schriftlichen Befragung am letzten Tag der Modellierungswoche einbezogen (vgl. Anh. G.2.2).

Die Lernenden versuchten zunächst die Daten zu verstehen und diese mithilfe der gegebenen Visualisierungen auf dem Jupyter Notebook zu analysieren. Anschließend betrachteten sie verschiedene Kennzahlen des Datensatzes, wie das mittlere Rating der Filme oder der Nutzenden. Die Lernenden entschieden jedoch schnell, dass die Berücksichtigung von einfachen statistischen Kennzahlen nicht ausreicht, um den Geschmack individueller Nutzerinnen und Nutzer abbilden zu können:

„Unsere ersten Ideen basierten darauf, Ähnlichkeiten und Regelmäßigkeiten zu finden. So könnten einfache Parameter, wie die durchschnittliche Bewertung eines Films, eine Rolle spielen. Jedoch wurde recht schnell klar, dass die alleinige Berücksichtigung solcher und ähnlicher Parameter nicht ausreichen würde, um eine individuelle Empfehlung zu ermöglichen“ (Zitat aus dem Bericht der Lerngruppe).

Die Lernenden verfolgten im Laufe der Modellierungswoche den Ansatz, Ähnlichkeiten zwischen Nutzenden basierend auf den bisherigen Bewertungen zu modellieren. Sie entwickelten im Kern eine nutzerbasierte Nachbarschaftsmethode, die auf dem Ähnlichkeitsmaß **gemittelte Summe der absoluten Abweichungen** beruhte. Folgende Schritte waren bei dem Modell der Lernenden zentral:

1. Die Ähnlichkeiten zwischen den Nutzerinnen und Nutzern, die mindestens  $p$  gemeinsam bewertete Filme haben, werden paarweise berechnet.
2. Für eine betrachtete Person wird eine Nachbarschaft aus ähnlichen Nutzerinnen und Nutzern definiert. In diese Nachbarschaft werden lediglich diejenigen aufgenommen, bei denen die gemittelte Summe der absoluten Abweichungen unter einem fest gewählten Grenzwert lag.

3. Für eine Person  $i$  und einen von dieser noch ungesehenen Film  $j$  wird basierend auf der Nachbarschaft der Person  $i$  eine Vorhersage als gewichtete Summe der Bewertungen, die die Nachbarinnen und Nachbarn für Film  $j$  abgegeben haben, berechnet. Hierbei werden nur die Nachbarinnen und Nachbarn berücksichtigt, die auch tatsächlich eine Bewertung für Film  $j$  abgegeben haben.

Die Lernenden beschäftigten sich gegen Ende der Modellierungswoche zudem mit weiterführenden Fragestellungen (Kaltstartproblem, Manipulation von Empfehlungssystemen, Empfehlungen bei Systemen mit binären Bewertungsskalen) und entwickelten erste Ideen, wie diesen Problemen begegnet werden könnte.

Im Rahmen des Modellierungsprozesses kamen die Lernenden zu einer im Bereich ML zentralen Erkenntnis:

„Des Weiteren sind wir zu der Erkenntnis gekommen, je mehr Daten wir bei dem Algorithmus benutzen, desto genauer wird unser Ergebnis. Ein Problem, das dabei auftrat, war die lange, sehr schnell ansteigende, Rechenzeit, die das Programm benötigt, um bei den vielen Daten zu Ergebnissen zu kommen“ (Zitat aus dem Bericht der Lerngruppe).

Die Lernenden arbeiteten während der Modellierungswoche weitgehend ausdauernd und konzentriert an der Problemstellung. Auch aus der abschließenden schriftlichen Befragung wird ersichtlich, dass die Lernenden Interesse an der Problemstellung hatten und dass der Praxisbezug für alle Lernenden klar erkennbar war. Zudem gaben die Lernenden an, dass sie durch die Modellierungswoche ein besseres Verständnis der mathematischen Modellierung gewonnen haben. Der Schwierigkeitsgrad der Aufgabenstellung schien für die Lernenden angemessen:

„Die größte Schwierigkeit bestand darin, mit den stark begrenzten Daten, die uns zur Verfügung standen, umzugehen. Dadurch war es jedoch auch recht übersichtlich und man konnte sich auf weniger Teilaspekte des Problems beschränken. Insgesamt fand ich die Aufgabenstellung recht herausfordernd, aber keineswegs überfordernd“ (Zitat aus der Evaluation einer teilnehmenden Person).

Keine Person aus der Lerngruppe gab an, dass sie die Aufgabe zu schwierig oder zu einfach fand. Schwierigkeiten im Modellierungsprozess traten insbesondere beim Umgang mit der für die Lernenden neuen Programmierumgebung CoCalc auf. CoCalc setzte zudem wiederholt den Arbeitsstand des Codes auf ein paar Minuten zurück. Änderungen, die in dieser Zeit gemacht wurden, waren gelöscht. Dies bemängelten die Lernenden zurecht.

Zwei Lernende hatten zudem vor der Modellierungswoche noch keine Programmiererfahrungen gesammelt, womit das Schreiben von Programmcode für diese Lernende zu Beginn der Modellierungswoche eine abschreckende Herausforderung darstellte. Besagte Lernende brachten sich während der Woche verstärkt bei der Entwicklung neuer mathematischer Ansätze, dem Schreiben des Berichtes und der Vorbereitung der Präsentation ein. Obgleich die Gruppenarbeit durchaus arbeitsteilig ablief, funktionierte die Online-Absprache durch wiederholte Diskussionen in gemeinsamen Videokonferenzen und dem dortigen Austausch in der gesamten Gruppe gut.

Insgesamt bewerteten die Lernenden die Modellierungswoche mit der durchschnittlichen Schulnote 1.33 und damit durchaus positiv.

- „Ich habe gelernt, dass es möglich ist, in einem kurzen Zeitraum sehr viel Neues zu lernen und dass man auch mit vielen fremden Personen in kurzer Zeit sehr gut zusammenarbeiten kann und sie auch kennenlernt“ (Zitat aus der Evaluation einer teilnehmenden Person).
- „Ich habe für mich persönlich gelernt, dass mir wissenschaftliches Modellieren und Problemlösen wirklich Spaß macht. Das hat mich in meinem Wunsch gestärkt, eine Naturwissenschaft (wahrscheinlich Mathematik oder Physik) zu studieren. Außerdem war es eine tolle Erfahrung, in einer Gruppe von interessierten Menschen zu arbeiten, bei der jeder und jede seine/ihre Stärken mit eingebracht hat (z. B. Personen, die (nicht) programmieren können) [...]“ (Zitat aus der Evaluation einer teilnehmenden Person).

## 18.3 Didaktische Reflexion

Hier wird zunächst die gewählte Problemstellung insbesondere im Hinblick auf die in Abschnitt 1.3 diskutierten Kriterien für gute Modellierungsprobleme reflektiert. Zudem wird das stärker angeleitete Lernmaterial für Modellierungstage / Unterrichtsreihen bezugnehmend auf die in Abschnitt 6.1 genannten übergeordneten Ziele reflektiert. Abschließend werden die Erfahrungen aus der Durchführung einer Modellierungswoche kurz resümiert.

Empfehlungssysteme werden in zahlreichen eCommerce Unternehmen eingesetzt, deren Angebote von vielen Jugendlichen und Erwachsenen tagtäglich genutzt werden, sei es auf Instagram, YouTube, Amazon, Spotify oder eben Netflix. Die Problemstellung ist damit durchaus als **relevant** für die Gesellschaft, aber insbesondere für die jugendliche Zielgruppe einzuschätzen.

Auch die zugrundeliegenden Strategien aus dem Bereich ML beziehungsweise DS sind über die konkrete Problemstellung hinaus für zahlreiche Anwendungen wesentlich. Da die Relevanz von ML und DS in den kommenden Jahrzehnten eher noch zunehmen wird, ist es zentral, dass die Lernenden ein gewisses Urteilsvermögen im Hinblick auf Funktionsweise, Gefahren und auch Chancen dieser Bereiche ausbilden. Diese Arbeit hat exemplarisch gezeigt, dass Methoden des MLs zahlreiche Anknüpfungspunkte an schulmathematische Inhalte bieten, was zu einer gewissen Legitimation derartiger Inhalte für den Mathematikunterricht beiträgt.

Die Ergebnisse der schriftlichen Befragung haben unterstrichen, dass das Thema *Empfehlungssysteme und der Netflix Preis* wie auch dessen Umsetzung im Lernmaterial dazu beitragen konnte, die Bedeutung von Mathematik für Alltag, Gesellschaft und Wirtschaft erfahrbar zu machen. Die Erfahrungswerte deuten zudem darauf hin, dass die Lernenden ein besseres Verständnis davon erlangen konnten, was mathematische Modellierung ist und wo sie Einsatz findet. Durch das wiederholte Aufgreifen des vierschriftigen Modellierungskreislaufs wurde zudem transparent, welche

Schritte beim Modellieren idealtypisch durchlaufen werden. Insgesamt ermöglichte die Modellierung des Empfehlungssystems den Lernenden das Durchlaufen des gesamten Kreislaufs und zudem den Einbau von ersten Modellverbesserungen (Erweiterung der Zielfunktion um einen Regularisierungsterm).

Im Hinblick darauf, dass der Netflix Preis tatsächlich stattgefunden hat und auch jetzt noch intensiv an der Weiterentwicklung von Empfehlungssystemen gearbeitet wird, sowohl in den Unternehmen selbst als auch in der Forschung, ist das Problem als **authentisch** zu bewerten. Dies gilt nicht nur für die Problemstellung an sich, sondern auch für die zur Lösung des Problems eingesetzte Mathematik. So diskutieren die Gewinner des Netflix Preises [Koren et al. \(2009\)](#) in ihrem Artikel *Matrix Factorization Techniques for Recommender Systems* die Methode einer Matrix-Faktorisierung als Grundlage des Modells, welches ihnen (in Kombination mit diversen Erweiterungen / anderen Modellen) letztlich den Sieg beschert hat. Auch das ALS-Verfahren wird dort als mögliches Optimierungsverfahren aufgegriffen.

Wie bereits in Abschnitt 17.3 deutlich geworden ist, ist die Problemstellung sowohl inner- als auch außermathematisch **reichhaltig**. Dass der innermathematische Vernetzungsgedanke im Lernmaterial aufgegriffen wird, wurde unter anderem bei der Diskussion von Bezügen zu schulmathematischen Inhalten ersichtlich. Auch greift das Lernmaterial die **Interdisziplinarität** der Problemstellung auf, indem insbesondere zum Abschluss des Workshops gesellschaftskritische Diskussionen geführt werden. Darüber hinaus macht der Einsatz des Computers und die Verwendung einer modernen Programmiersprache zum Lösen des datenlastigen Problems die Vernetzung zur Informatik deutlich.

Ein Teilziel dieser Arbeit war es, zu untersuchen, inwieweit die Mathematik, die beim Formulieren und Lösen von Optimierungsproblemen in realen Problemlöseprozessen Anwendung findet, Schülerinnen und Schülern zugänglich gemacht werden kann. Das Optimierungsproblem, welches bei der Entwicklung des Empfehlungssystems über eine Matrix-Faktorisierung auftritt, ist durchaus als *schwierig* einzustufen. Das Problem hängt schon bei der Betrachtung kleiner Beispiel-Bewertungsmatrizen von mehr als zwei Variablen ab. Damit übersteigt das Problem bereits in der didaktisch reduzierten Form den zwei- oder dreidimensionalen Anschauungsraum. Die Zielfunktion ist nicht anschaulich darstellbar. Das Lösen des mehrdimensionalen Optimierungsproblems fordert somit ein höheres Maß an Abstraktion von den Lernenden, die Funktionen in der Schule i.d.R. nur in Abhängigkeit von einer Variablen kennenlernen. Demzufolge wurde im Lernmaterial ein stärker angeleiteter Ansatz beim Formulieren und Lösen des Optimierungsverfahrens gewählt als es im Lernmaterial zum Solarkraftwerk realisiert wurde. Es lässt sich im Hinblick auf die Optimierung festhalten, dass der Mehrheit der Lernenden ein authentisches Optimierungsverfahren durch entsprechendes Lernmaterial und stärker angeleitete Aufgabenstellungen zugänglich gemacht werden konnte. Zudem war es möglich, den Einbau eines Regularisierungsterms in die Zielfunktion (und damit die Änderung der Formulierung des Optimierungsproblems) mit den Lernenden zu diskutieren. Diese können den Einfluss der Regularisierung im Lernmaterial sowohl experimentell im Rahmen einer Parameterstudie erkunden als auch bezugnehmend auf den Funktionsterm algebra-

isch diskutieren.

Im Rahmen von zwei Zusatzblättern bietet das Lernmaterial zudem die Möglichkeit den Einfluss verschiedener Fehlermaße und der Regularisierung auf die Eigenschaften der Zielfunktion, die in den Iterationsschritten des ALS-Verfahren minimiert wird, anschaulich zu diskutieren. Aufgrund von Zeitmangel wurde besagtes Zusatzmaterial nur von wenigen Lernenden bearbeitet. Um eine Aussagen über die Eignung der Zusatzblätter für eine breitere Schülerschaft treffen zu können, sind weitere Erprobungen notwendig.

Obleich das Lernmaterial die Modellierung des Empfehlungssystems und insbesondere das Formulieren und Lösen des Optimierungsproblems durch kleinere Aufgabenstellungen anleitet, werden kreative Ansätze der Lernenden in verschiedenen Phasen des Workshops eingebunden. So werden die Ideen der Lernenden zur Entwicklung eines Modells basierend auf der Ratingmatrix ausgiebig im Plenum diskutiert und auch die unterschiedlichen Ansätze der Lernenden bei der Definition eines Fehlermaßes aufgegriffen und gegeneinander abgewogen.

Das Lernmaterial ist **problemorientiert** aufgebaut. Für die Lernenden neue mathematische Inhalte, beispielsweise das partielle Ableiten, werden an den Stellen eingeführt, an denen sie im Modellierungsprozess benötigt werden. Durch verschiedene Differenzierungselemente (Tippkarten und Zusatzmaterial), durch die Rückmeldefunktionen und die auf eine eigenständige Bearbeitung ausgelegten Arbeitsblätter erlaubt das Lernmaterial ein hohes Maß an **Schüleraktivität**.

Der Schwierigkeitsgrad der Aufgaben zeigte sich im Bezug auf die Evaluationsergebnisse für die Mehrheit der Lernenden als angemessen. Lediglich bei der Bearbeitung von Arbeitsblatt 4 in der Langversion traten wiederholt Verständnisschwierigkeiten auf. Dies eröffnet im Hinblick auf die Bereitstellung von zusätzlichem Unterstützungsmaterial Potential für Weiterentwicklungen des Lernmaterials. Auf mögliche Erweiterungen des Materials wird im Ausblick eingegangen.

Aus der einmaligen Erprobung der unangeleiteten Entwicklung eines Empfehlungssystems im Rahmen der Modellierungswoche ist ersichtlich geworden, dass die Modellierung über Nachbarschaftsmethoden naheliegender für die Lernenden war, als das Optimieren der Parameter eines gewählten Modells. Grundsätzlich hat sich die Problemstellung durchaus als geeignet für offene Modellierungsprojekte erwiesen und den Lernenden ermöglicht, Erfahrungen im Umgang mit großen Datenmengen zu sammeln. Die Software CoCalc, die zur Bereitstellung und zum kollaborativen Arbeiten am gleichen Code eingesetzt wurde, erwies sich aufgrund von Instabilität und Datenverlusten (Zurücksetzen des Arbeitsstandes) als nur bedingt geeignet.

## 18.4 Ausblick

Durchaus bestehen zahlreiche Möglichkeiten, das Lernmaterial didaktisch-methodisch weiterzuentwickeln. Auch bieten sich verschiedene weitere DS-Probleme für die Entwicklung von Lehr- und Lernmaterial zur Modellierung realer, alltagsnaher Problemstellungen an. Hier seien exemplarische Erweiterungs- und Verbesserungsmöglichkeiten genannt:

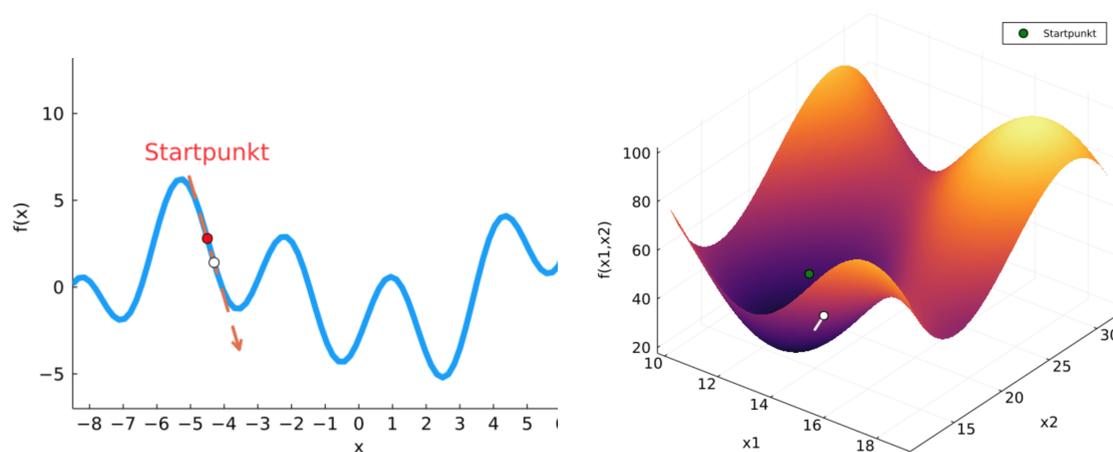


Abb. 18.1: Gradientenverfahren visualisiert im zwei- und dreidimensionalen Fall

- Fächerübergreifendes Projekt – von den Daten zum Produkt:** Durchaus interessant wäre die Konzipierung eines fächerübergreifenden Projektkurses (Informatik, Mathematik) zur Entwicklung eines Empfehlungssystems, bei dem die Lernenden alle sechs Phasen des CRISP-DM Modells (vgl. Abschn. 14) eigenständig durchlaufen. Die Lernenden könnten dazu entweder den Netflixdatensatz oder gar einen selbst erhobenen Datensatz (aufgenommen durch eine Befragung ihrer Mitlernenden) verwenden. Projektziel könnte sein, dass die Lernenden ihr selbst entwickeltes Empfehlungssystem auf einer eigenen Webseite bereitstellen.
- Erprobung anderer digitaler Werkzeuge:** Denkbar wäre es den Netflixdatensatz (bzw. einen Teil davon) über eine Software bereitzustellen, die weniger Programmiererfahrung seitens der betreuenden Lehrkräfte bedarf. Zu Erkundung des Datensatzes wären beispielsweise CODAP<sup>2</sup> oder Excel einsetzbar.
- Weiterentwicklung des Lernmaterials zur Optimierung:** Nicht nur das ALS-Verfahren, sondern auch das Gradientenverfahren ließe sich didaktisch reduzieren und im Rahmen von interaktiven Jupyter Notebooks aufbereiten. Ausgehend von einer eindimensionalen Funktion ließe sich die Idee von Abstiegsrichtung, Schrittweite und Startpunkt einführen. Diese Konzepte könnten anschließend auf zweidimensionale Funktionen übertragen werden;<sup>3</sup> jeweils visualisiert durch entsprechende interaktive Plots (vgl. Abb. 18.1). Anschließend wird diese Grundidee auf beliebig-dimensionale Funktionen abstrahiert. Möglich wäre, die Lerngruppe aufzuteilen und einen Teil das ALS-Verfahren, den anderen Teil das Gradientenverfahren erarbeiten zu lassen. Im Plenum könnten beide Verfahren vorgestellt, verglichen und diskutiert werden.
- Verzerrungen in den Daten diskutieren:** Die Diskussion von Verzerrungen im Datensatz wird im bisherigen Lernmaterial nicht aufgegriffen. Anhand von geeigneten Beispielen und unterstützt durch Visualisierungen könnten die

<sup>2</sup>Weitere Informationen unter: <https://codap.concord.org>, letzter Aufruf: 19.11.2021.

<sup>3</sup>Lernmaterial, welches dieser Grundidee folgt, wurde bereits entwickelt. Da es jedoch noch nicht unterrichtspraktisch erprobt wurde, wurde es in Abschnitt 17.2.1 nicht beschrieben.

Verzerrungen in den Daten auch mit Schülerinnen und Schülern erarbeitet werden.

- **Eigenständiges Programmieren ermöglichen:** Um programmieraffinen Lernenden die Möglichkeit zu bieten, im Rahmen der stärker angeleiteten Modellierungstage eigenständigen Programmcode zu schreiben, kann ein Jupyter Notebook bereitgestellt werden, in dem die Lernenden die Kreuzvalidierung des Regularisierungsparameters  $\lambda$  und der Anzahl der Eigenschaften  $n_f$  selbstständig implementieren. Dies wäre über zwei verschachtelte for-Schleifen möglich.
- **Modellierungsprojekte zu weiteren ML-Methoden:** Nicht nur die hier beschriebene Matrix-Faktorisierung, sondern zahlreiche weitere ML-Methoden beruhen im Kern auf mathematischen Modellen, die didaktisch reduziert, durchaus mit schulmathematischem Wissen zugänglich gemacht werden können. Ein Beispiel ist die bereits in der Einleitung erwähnte Stützvektormethode, die im Grunde auf der Minimierung von Abständen zwischen Punkten und Hyperebenen beruht. Sowohl Vektoren, Geraden und Ebenen wie auch das Skalarprodukt können eine authentische Anwendung finden (vgl. Schönbrodt et al., 2021). Diverse Anknüpfungspunkte an schulmathematische Inhalte bieten auch Entscheidungsbäume, die unter anderem für Klassifizierungsprobleme eingesetzt werden und zu denen von Biehler und Fleischer (2021) digitales Lernmaterial (umgesetzt via CODAP und Jupyter Notebook) entwickelt wurde. Ein drittes Beispiel stellen sogenannte N-Gramm-Modelle dar, die beispielsweise zur Vorhersage von Wörtern beim Tippen einer Nachricht auf dem Smartphone Einsatz finden und bei denen bedingte Wahrscheinlichkeiten sowie absolute und relative Häufigkeiten eine zentrale Rolle spielen (vgl. Hofmann, im Druck). Die Bedeutung von KI für Technik, Wirtschaft und Kultur, und damit insbesondere den Alltag der Lernenden, wird in den kommenden Jahrzehnten eher noch zunehmen. Es ist Aufgabe der Mathematik(-didaktik) ihrerseits die Bedeutung von Mathematik für derartige Methoden für Lernende erfahrbar zu machen – eben gerade weil ein verständiger Zugang möglich ist. Die Gestaltung von geeignetem Unterrichtsmaterial und der Einbau dessen in den schulischen (Projekt-)Unterricht kann dazu beitragen, Datenkompetenz im Sinne einer umfassenden Data Literacy zu schulen und die Lernenden dazu zu befähigen, die Bedeutung von Mathematik für ML-Methoden zu erfassen und ihren von KI-Anwendungen durchzogenen Alltag im Sinne Winters (1985) „in einer spezifischen Art und Weise wahrzunehmen.“
- **Verbesserung des Lernmaterials zum ALS-Verfahren:** In den bisherigen Erprobungen des Lernmaterials waren Schwierigkeiten bei der Erarbeitung einzelner Schritte des ALS-Verfahrens zu beobachten. Um dem entgegenzuwirken, könnte dieser Teil des Workshops wiederholt durch kleinere Sicherungsphasen unterbrochen und Aufgaben zum Üben der eingesetzten mathematischen Inhalte (insbesondere zur Kettenregel) integriert werden. Im Hinblick auf die Optimierung wäre interessant zu erproben, wie die Lernenden komplett unangeleitet an die Minimierung der Zielfunktion (16.6) herangehen würden.

Auch im Hinblick auf die (Weiter-)Entwicklung des beschriebenen mathematischen Modells des Empfehlungssystems wären zahlreiche Ansätze denkbar. Hier seien exemplarisch zwei Ansätze benannt, die in der Forschung rund um Empfehlungssysteme bereits umgesetzt werden:

- Implizite Bewertungen im bisherigen Modell berücksichtigen. [Koren und Bell \(2011, S. 150\)](#) argumentieren dazu:

„The dataset does not only tell us the rating values, but also which movies users rate, regardless of how they rated these movies. In other words, a user implicitly tells us about her preferences by choosing to voice her opinion and vote a (high or low) rating. This creates a binary matrix, where “1” stands for “rated”, and “0” for “not rated”. While this binary data may not be as informative as other independent sources of implicit feedback, incorporating this kind of implicit data does significantly improve prediction accuracy. The benefit of using the binary data is closely related to the fact that ratings are not missing at random“.

Es ist durchaus denkbar, dass Bewertungen entgegen unserer Annahme nicht fehlen, weil eine Person den Film bisher nicht entdeckt hat, sondern weil sie sich bewusst gegen diesen entschieden hat. Diese Beobachtung könnte wie von [Koren und Bell \(2011\)](#) vorgeschlagen in das bisherige Modell integriert werden.

- Auch könnte der Ansatz einer Nichtnegativen Matrix-Faktorisierung (NNMF) verfolgt werden. Wie der Name verrät, dürfen die Einträge der Faktormatrizen in diesem Fall nicht negativ werden. Die NNMF hat den Vorteil, dass die Einträge von User- und Moviematrix womöglich leichter zu interpretieren sind. Nachteil ist jedoch, dass zur Bestimmung einer NNMF ein Optimierungsproblem unter Nebenbedingungen zu lösen ist. Das Problem wird schwieriger.

**Teil V**

**Abschließende Kommentare**



# 19 Optimierungsprobleme in Modellierungsprojekten – wenig bis nicht angeleitet?

Ein langfristiges Ziel des mathematischen Modellierungsunterrichts sollte es unserer Meinung nach sein, Schülerinnen und Schüler dazu zu befähigen reale, authentische Problemstellungen möglichst selbstständig und unangeleitet zu bearbeiten. Dass dies möglich ist, zeigen zahlreiche Erfahrungen aus mathematischen Modellierungswochen (vgl. Abschn. 4.2), die im Rahmen von außerschulischen Veranstaltungen stattgefunden haben. Andere Beispiele, in denen offenes, unangeleitetes Modellieren seitens der Lernenden möglich gemacht wird, sind mathematische Modellierungswettbewerbe, wie die *International Mathematical Modeling Challenge*<sup>1</sup> oder der *COMAP's Mathematical Contest in Modeling*<sup>2</sup>.

Im Hinblick auf die Zielsetzung eines stärkeren Einbezugs möglichst unangeleiteter Modellierungsaktivitäten zu authentischen, offenen Problemen in den Mathematikunterricht, soll hier diskutiert werden, ...

- inwieweit sich insbesondere Optimierungsprobleme, die bei der Bearbeitung realer Problemstellungen auftreten, für (unangeleitete) kreative mathematische Modellierungsaktivitäten eignen. Dazu wird herausgestellt, in welchen Phasen des Formulierens und Lösens der Probleme individuelle Modellentscheidungen (seitens der Lernenden) denkbar sind.
- welche exemplarischen Optimierungsansätze und Strategien Schülerinnen und Schüler bereits in diversen offenen Modellierungsprojekten (Modellierungswochen) verfolgt haben und wie sich die Ansätze und Strategien der Lernenden einordnen lassen.
- welche leitenden Fragestellungen einer Lehrkraft bei der Betreuung offener (Optimierungs-)Probleme zur Orientierung gegeben werden können. Diese Leitfragen sollen helfen, mögliche Optimierungsstrategien und sinnvolle Formulierungen der Probleme abzuschätzen. Zugleich können diese bei der Betreuung der Modellierungsprojekte herangezogen werden, um Impulse zu geben, welche weiterführenden Fragen beim Optimieren durchdacht werden könnten (Stichworte: Existenz, Eindeutigkeit, Güte und Rechenzeit).

---

<sup>1</sup>Weitere Informationen unter: <https://immchallenge.org>, letzter Aufruf: 19.11.2021.

<sup>2</sup>Weitere Informationen unter: [www.comap.com/undergraduate/contests](http://www.comap.com/undergraduate/contests), letzter Aufruf: 19.11.2021.

## 19.1 Individuelle Modellentscheidungen rund um Optimierungsprobleme

Das Formulieren und Lösen von Optimierungsproblemen erlaubt an diversen Stellen des Modellierungsprozesses individuelle Entscheidungen zu treffen. Dies unterstreicht eindrücklich, dass mathematisches Modellieren keineswegs ein deterministischer Prozess ist, sondern zahlreiche Handlungsspielräume bestehen, die in unterschiedliche mathematische Formulierungen und Eigenschaften des Optimierungsproblems münden. Im Folgenden werden verschiedene Modellentscheidungen beim Bearbeiten von Optimierungsproblemen diskutiert und veranschaulicht. Es wird deutlich, dass die Optimierung zahlreiche Möglichkeiten bietet, um Modellierungsprozesse offen und unangeleitet zu gestalten. Zugleich wird bei der Beschreibung der Modellentscheidungen erneut das Wechselspiel zwischen innermathematischen Überlegungen hinsichtlich Eigenschaften des formulierten mathematischen Problems und Anwendungsfragen herausgehoben.

### Das Ziel

Die erste Entscheidung kann oft bereits bei der Festlegung des eigentlichen Ziels getroffen werden: Was möchte ich eigentlich erreichen? Am Beispiel der Optimierung des Solarkraftwerks ließe sich fragen: Soll der absolute Energieumsatz optimiert werden? Über einen Tag oder mehrere Jahre? Soll vielmehr die Effizienz (Energieumsatz bezogen auf die Flächenausnutzung des Kraftwerkareals) maximiert werden? Oder gar die gesamte Wirtschaftlichkeit des Kraftwerks? Die Entscheidung, welches Ziel letztlich verfolgt wird,<sup>3</sup> findet somit oft bereits vor der mathematischen Formulierung des Optimierungsproblems statt.

### Mathematische Formulierung des Ziels

Bei der mathematischen Formulierung des Ziels bestehen je nach Anwendungsfrage ebenfalls Handlungsspielräume. Bei der Entscheidung, wie genau die Zielfunktion letztlich formuliert wird, spielen in der Praxis meist Kenntnisse über die mathematischen Eigenschaften der Zielfunktion und der damit einhergehenden Lösbarkeit des Optimierungsproblems eine wesentliche Rolle. Im Workshop zum Thema Empfehlungssysteme wurde die Wahl unterschiedlicher Fehlermaße und damit verbundener Eigenschaften der Zielfunktion mit den Lernenden erarbeitet und auf anschaulicher Ebene diskutiert (vgl. Abschn. 17.2.1). Zwei weiterführende Beispiele, die dies verdeutlichen und die sich durchaus ebenfalls für die Diskussion mit Lernenden eignen, werden hier aufgegriffen:

#### Beispiel 19.1.1 (Arithmetisches Mittel und Median).

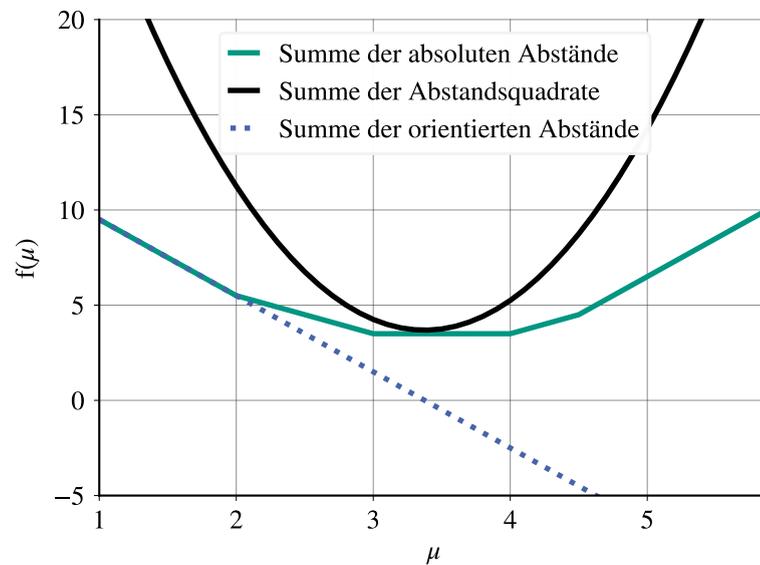
In diesem Beispiel, welches unter anderem von Winter (1985) und Vogel (2010b) fachdidaktisch diskutiert wurde, geht es um die Frage, welche Zahl  $\mu$  eine gegebene

---

<sup>3</sup>Im entwickelten Lernmaterial wurde die Festlegung des Ziels meist vorweggenommen, indem festgelegt wurde, welche Größen optimiert werden sollen (die Leistung / der Energieumsatz im Falle der Solarkraftwerke).

Menge an Werten  $\{x_1, \dots, x_N\}$  am besten beschreibt. Seitens der Lernenden sind bei der Definition eines Fehlermaßes verschiedene Strategien und Formulierungen denkbar:

- die Summe der orientierten Abstände<sup>4</sup>  $\sum_{i=1}^N (x_i - \mu)$
- die Summe der absoluten Abstände  $\sum_{i=1}^N |x_i - \mu|$
- die Summe der quadratischen Abstände  $\sum_{i=1}^N (x_i - \mu)^2$



**Abb. 19.1:** Arithmetisches Mittel und Median als Minimalpunkt(e) von Optimierungsproblemen. Die Summe der orientierten Abstände eignet sich nicht als Fehlermaß.

Auf anschaulicher Ebene kann diskutiert werden, welchen Einfluss die verschiedenen Formulierungen der Zielfunktion auf die Existenz und Eindeutigkeit eines Minimalpunktes haben. Dazu werden exemplarisch die Werte  $x_1 = 2, x_2 = 3, x_3 = 4$  und  $x_4 = 4.5$  betrachtet. In Abbildung 19.1 wird für dieses Beispiel deutlich, dass sich bei Wahl der Summe der Abstandsquadrate eine differenzierbare, strikt konvexe Funktion ergibt, die einen eindeutigen globalen Minimalpunkt besitzt. Dieser entspricht gerade dem arithmetischen Mittel (vgl. Vogel, 2010b, S. 51).

Zudem wird ersichtlich, dass die Summe der orientierten Abstände auf eine Zielfunktion führt, die keinen Minimalpunkt besitzt. Anschaulich wird klar, dass dies kein sinnvolles Gütemaß darstellt.

Wird die Optimierung hingegen basierend auf der Summe der absoluten Abstände durchgeführt, so weist der Graph der Zielfunktion Knickstellen auf. Die Funktion ist damit nicht überall differenzierbar. Weiterhin ist zu erkennen, dass kein eindeutiger

<sup>4</sup>Dieser Ansatz ist nicht als geeignetes Fehlermaß zu bezeichnen, dennoch wird er hier aufgeführt, da dieser von Lernenden in durchgeführten Modellierungsprojekten vielfach genannt wurde und aus didaktischer Sicht Raum für Diskussionen zu gewünschten Eigenschaften eines Fehlermaßes bietet.

globaler Minimalpunkt existiert. Im betrachteten Beispiel sind alle Werte, die zwischen  $x_3$  und  $x_4$  liegen, globale Minimalpunkte (darunter auch der Median). Dies ist stets der Fall, wenn eine gerade Anzahl an Werten vorliegt.

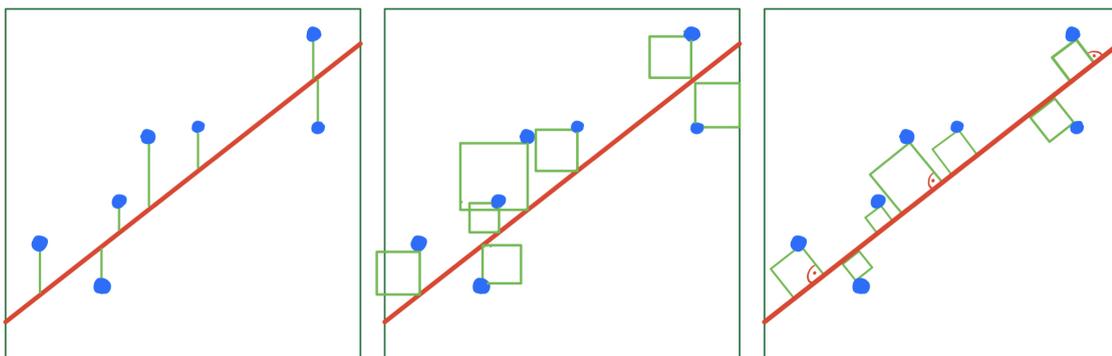
Schön an diesem Beispiel ist, dass die Diskussion von Eigenschaften der Zielfunktion und damit einhergehend der Existenz und Eindeutigkeit von Minimalpunkten auf anschaulicher Ebene möglich ist. Zugleich kann die Betrachtung und Diskussion des Optimierungsproblems zu einer tragfähigen Begriffsbildung (arithmetisches Mittel und Median) beitragen (vgl. Vogel, 2010a, S. 5).

**Beispiel 19.1.2** (Anpassung an Messdaten).

Ein weiteres Problem, welches bei verschiedenen Anwendungsfragen auftritt und zu dem Diskussionen mit Lernenden rund um den Einfluss des Fehlermaßes auf die Eigenschaften der Zielfunktion denkbar sind, ist die Anpassung einer Funktion an gegebene Messdaten. Ziel ist es, gegebene Messdaten durch eine Funktion anzunähern. Dabei soll aus einer zuvor gewählten Klasse an Funktionen (bspw. lineare Funktionen) die Funktion gewählt werden, die die Daten *bestmöglich* beschreibt. Der (zu definierende) Fehler zwischen den Datenpunkten und der Beschreibung durch die gesuchte Funktion soll minimal werden.

Wir bleiben beim einfachsten Fall: der linearen Ausgleichsrechnung. Gegebene Datenpunkte  $(x_i, y_i) \in \mathbb{R}^2$  für  $i = 1, \dots, N$  sollen durch eine lineare Funktion  $f$  mit  $f(x) = mx + b$  angepasst werden. Der Fehler zwischen den Datenpunkten und der Beschreibung durch die lineare Funktion kann auf unterschiedliche Weise formuliert werden. Denkbare Ansätze sind

1. die Summe der absoluten Abweichungen entlang der  $y$ -Achse  $\sum_{i=1}^N |y_i - f(x_i)|$  (vgl. Abb. 19.2 links),
2. die Summe der quadratischen Abweichungen entlang der  $y$ -Achse  $\sum_{i=1}^N (y_i - f(x_i))^2$  (vgl. Abb. 19.2 Mitte) und
3. die Summe der absoluten oder quadratischen, tatsächlichen Abstände zwischen den Datenpunkten und der Geraden (Berechnung über das Lot) (vgl. Abb. 19.2 rechts).



**Abb. 19.2:** Skizze verschiedener Ansätze für die Anpassung von Messdaten durch Ausgleichsgeraden

Meist wird die Summe der Abstandsquadrate der womöglich intuitiven Wahl der Summe der absoluten Abstände vorgezogen. Ein Grund dafür ist wiederum innermathematischer Natur und beruht, wie bereits im Lernmaterial zu Empfehlungssystemen ersichtlich wurde, auf den Eigenschaften der erhaltenen Zielfunktion: Diese ist im Falle der Summe der Abstandsquadrate strikt konvex (sofern mindestens zwei verschiedene  $x_i$  vorliegen) und differenzierbar. Das Problem ist leicht mit Standardmethoden lösbar. Zudem garantiert die strikte Konvexität der Zielfunktion die Eindeutigkeit des Minimalpunktes. Nachteil dieses Ansatzes ist jedoch, dass „Ausreißer“ in den Messwerten stärker ins Gewicht fallen.

Die Summe der absoluten Abstände hingegen läuft auf eine nicht differenzierbare Zielfunktion hinaus und auch die Eindeutigkeit eines Minimalpunktes ist nicht gesichert. Dennoch spricht insbesondere die Robustheit gegenüber Ausreißern in den Daten in manchen Fällen für die Verwendung dieses Ansatzes, der auch als *Least Absolute Deviation* bezeichnet wird (vgl. Bloomfield und Steiger, 1980).

Die dritte Strategie, über die Summe der quadratischen orthogonalen Abstände (Stichwort: *orthogonale Regression*), hat den Nachteil, dass zur Bestimmung / Implementierung der „tatsächlichen Abstände“ aufwendigere Berechnungen nötig sind. Nicht nur die weniger aufwendigen Rechnungen, sondern auch messpraktische Gründe sprechen für die Abstände in  $y$ -Richtung: uns interessiert letztlich die Differenz zwischen tatsächlichem Messwert  $y_i$  und dem durch die Funktion zugewiesenen Wert  $f(x_i)$  (vgl. Schupp, 1992, S. 46).

Gallin (2010) diskutiert, dass anstelle der quadratischen vertikalen Abstände (in  $y$ -Richtung) und der orthogonalen Abstände auch die Abstände in  $x$ -Richtung beobachtete Vorschläge von Lernenden sind. An dieser Stelle ließen sich durchaus interessante tiefer gehende Ausführungen über weitere Vor- und Nachteile der unterschiedlichen Zielfunktionen und über statistische Charakteristika der verschiedenen Ansätze aufnehmen, die den Rahmen dieser Arbeit jedoch sprengen würden.

Ein Beispiel, in dem die lineare Ausgleichsrechnung im Rahmen von mathematischen Modellierungsprojekten eine authentische Anwendung findet, ist bei Hattebuhr (im Druck) zu finden. Konkret wird die lineare Regression dort zur Analyse von realen Temperaturdaten angewendet, um Fragen rund um Klimawandel und Erderwärmung zu beantworten.

Das Beispiel 19.1.2 macht erneut ersichtlich, dass das Formulieren und Lösen von Optimierungsproblemen oftmals ein Abwägen ist zwischen a) mathematisch gewünschten Eigenschaften des Problems (Existenz, Eindeutigkeit), b) rechnerischem Aufwand zur Lösung des Problems und c) (bspw. statistischen) Eigenschaften, die die gefundene Lösung erfüllen soll.

### Die Wahl der Variablen

Nicht nur das Ziel, sondern auch die Entscheidungsvariablen können je nach Problemstellung unterschiedlich gewählt werden. Die Frage danach, welche Parameter aus praktischer Sicht überhaupt variiert werden *können*, wird insbesondere am Beispiel des Solarkraftwerks ersichtlich: Sind die Maße der Spiegel durch das herstellende Unternehmen festgelegt oder können wir diese frei wählen? Wie sieht es mit der Form der Spiegel und der Breite des Kollektors aus? Bei welchen weiteren

Parametern haben wir Handlungsspielraum? Welche Parameter können wir somit als Entscheidungsvariablen betrachten, variieren und nach eigenen Wünschen (optimal) festlegen? Im Rahmen der stärker angeleiteten Workshops zum Solarkraftwerk wurden zwei Kraftwerksparameter als Variablen betrachtet und von den Lernenden optimiert: die Höhe des Absorberrohres (vgl. Abschn. 11.1) und die Positionen der Spiegel (vgl. Abschn. 11.2). In den offenen Modellierungsprojekten (vgl. 12.2.4) haben die Lernenden sich zudem mit weiteren Kraftwerksparametern (Spiegelbreite und Breite des Sekundärreflektors) beschäftigt.

### Die Festlegung der zulässigen Menge

Auch bei der Festlegung der zulässigen Menge können diverse Modellentscheidungen getroffen werden. Eine grundlegende Entscheidung ist, ob der Suchraum zunächst diskretisiert wird oder ob dieser als Kontinuum betrachtet wird. Diese Entscheidung wurde in Abschnitt 11.2 am Beispiel der Optimierung der Spiegelpositionen ausführlich diskutiert. Die Lernenden haben dabei sowohl diskrete als auch kontinuierliche Formulierungen des Problems entwickelt und untersucht.

Nicht nur bei der Festlegung der Variablen auf eine diskrete oder kontinuierliche zulässige Menge, sondern durchaus auch im Hinblick auf die Hinzunahme von weiteren Nebenbedingungen sind verschiedene Entscheidungen denkbar. Im Workshop zur Positionsoptimierung wurde ersichtlich, dass die Hinzunahme von Nebenbedingungen zum Teil notwendig ist, um realistische Optimalpunkte zu erhalten (Nebenbedingung für den Mindestabstand zwischen Spiegeln). In diesem Fall führte die Hinzunahme der Nebenbedingung jedoch sogleich dazu, dass das Optimierungsproblem nicht konvex war, da die Nebenbedingungen für den Mindestabstand keine konvexe Menge induzierten. Dies wird durch das folgende Beispiel anschaulich unterstrichen.

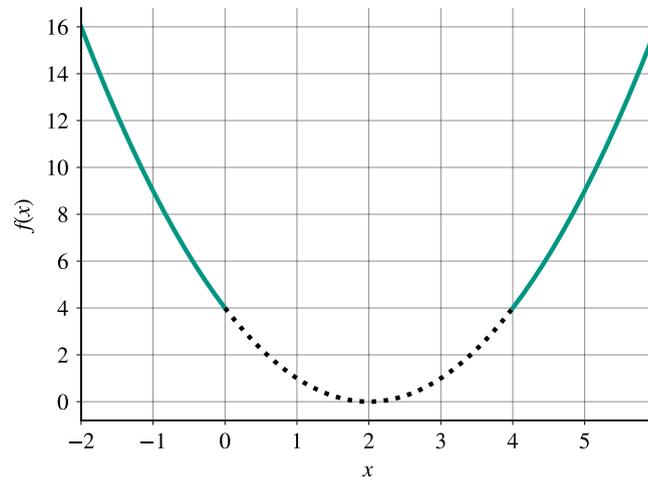
#### Beispiel 19.1.3.

Die in Abbildung 19.3 dargestellte quadratische Funktion  $f$  mit  $f(x) = (x - 2)^2$  ist strikt konvex. Betrachten wir als zulässige Mengen ganz  $\mathbb{R}$ , so finden wir einen eindeutigen Minimalpunkt (der Scheitelpunkt der Parabel). Wird die zulässige Menge hingegen auf alle Werte  $x$  eingeschränkt, für die  $|x - 2| \geq 2$  gilt, so wird ersichtlich, dass die Zielfunktion zwei globale Minimalpunkte besitzt. Die zulässige Menge ist in diesem Fall nicht mehr konvex. Wird die zulässige Menge jedoch auf alle  $x$  für die  $(x - 2) \geq 2$  gilt eingeschränkt, so ergibt sich erneut eine konvexe zulässige Menge und damit ein konvexes Optimierungsproblem mit eindeutigem Minimalpunkt.

Dass die schrittweise Hinzunahme von Nebenbedingungen sinnvoll sein kann und auch mit Lernenden denkbar ist, wurde im Unterrichtsmaterial und den durchgeführten Erprobungen zur Optimierung der Spiegelpositionen ersichtlich. Dort wurde neben der Forderung des Mindestabstandes zwischen den Spiegeln zudem eine auf- oder absteigende Sortierung der Spiegel gefordert.

### Die Wahl des Lösungsalgorithmus

Wurde das Optimierungsproblem mathematisch formuliert, so muss noch ein geeignetes Verfahren gefunden werden, um das Problem zu lösen. Seit vielen Jahrzehnten wird intensiv daran geforscht Algorithmen zur Lösung verschiedener Klassen



**Abb. 19.3:** Quadratische Zielfunktion eines Optimierungsproblems unter der Nebenbedingung  $|x - 2| \geq 2$ . Die zulässige Menge ist nicht konvex.

von Optimierungsproblemen zu entwickeln. Dabei werden die Eigenschaften der Optimierungsprobleme analysiert und versucht gute Software-Implementierungen zu entwickeln. Die Effektivität der Algorithmen variiert beträchtlich und hängt von verschiedenen Faktoren wie den Eigenschaften der Zielfunktion, der Anzahl der Variablen und der Gestalt der Nebenbedingungen ab (vgl. [Boyd und Vandenberghe, 2004](#), S. 3). Bei der Formulierung des Optimierungsproblems und bei der Wahl eines passenden Algorithmus für die jeweilige Problemklasse wird meist zwischen a) der Güte von gefundenen Lösungen beziehungsweise der Tatsache, ob mit dem gewählten Algorithmus eine optimale Lösung gefunden werden kann und b) der Laufzeit des gewählten Lösungsverfahrens abgewogen.

### Zwischenfazit

Beim Formulieren und Lösen von Optimierungsproblemen gibt es insbesondere bei realen, komplexen Problemen vielfach nicht den einen, richtigen Ansatz. Stattdessen existieren oft verschiedene Möglichkeiten, wie die Probleme formuliert und gelöst werden können, die wiederum mit unterschiedlichen (inner- sowie außermathematischen) Vor- und Nachteilen einhergehen. Dies wurde insbesondere bei der Optimierung der Spiegelpositionen, bei der die Lernenden unterschiedliche Formulierungen eines Optimierungsproblems diskutiert und verschiedene Algorithmen erarbeitet und evaluiert haben, ersichtlich (vgl. [Abschn. 11.2](#)).

## 19.2 Optimierungsprobleme im Rahmen mathematischer Modellierungswochen

In diesem Abschnitt wird eine kurze Übersicht über verschiedene Formulierungen und Lösungsansätze geboten, die Lernende unangeleitet im Rahmen von mathematischen Modellierungswochen bei der Bearbeitung von Optimierungsproblemen verfolgt haben. Die Ansätze der Lernenden wurden aus den Berichten abstrahiert, die die Lerngruppen während der Modellierungswochen verfasst haben. Es wird einmal mehr ersichtlich, dass das Optimieren eine Vielzahl an unterschiedlichen Ansätzen und Lösungsstrategien erlaubt, und dass sich diese Vielfalt auch in den Modellierungsentscheidungen der Lernenden widerspiegelt. Die Problemstellungen der Modellierungswochen sind in Anhang H zu finden.

### Problem 1: Optimale Energieversorgung basierend auf drei erneuerbaren Energiequellen

Das Problem wurde vom Lehrstuhl für Reaktorsicherheit und -technik der RWTH Aachen gestellt und auf einer Modellierungswoche im Jahr 2015 von einer Lerngruppe bearbeitet. Die Problemstellung bestand darin, die optimale Anzahl an Windenergieanlagen und Photovoltaikzellen zu finden, sodass die Energieversorgung der Stadt Aachen zu jeder Zeit gewährleistet ist (vgl. Anh. H.1). Die dritte Energiequelle stellten sogenannte Pumpspeicherkraftwerke dar. Bei hohem Strombedarf wird Wasser aus einem höher gelegenen Oberbecken in ein tiefer liegendes Unterbecken geleitet. Bei diesem Vorgang wird eine Turbine angetrieben, die Strom erzeugt. In Zeiten geringen Strombedarfs wird der mit Sonne und Wind erzeugte Strom genutzt, um Wasser in das obere Becken zu pumpen. Die Anzahl der Pumpspeicherkraftwerke war nicht variabel.

**Modellierungsansatz der Lerngruppe:** Die Lernenden entwickelten ein Modell zur Beschreibung des Energieumsatzes basierend auf allen drei Energiequellen und setzten dieses mit dem Energiebedarf der Stadt Aachen in Relation. Sie berechneten einen Mindestwert an Energie, der insgesamt von Wind- und Sonnenkraft zu erbringen ist, um den Energiebedarf erfüllen zu können.

Die Lernenden bearbeiteten ein **ganzzahliges Optimierungsproblem**. Ziel war die Kombination aus Anzahl Windanlagen  $n_W$  und Anzahl Photovoltaikzellen  $n_P$  zu bestimmen, die besonders kostengünstig ist und den Mindestenergieumsatz erfüllt. Zum Lösen des Problems entwickelten die Lernenden ein **Suchverfahren**. Konkret gingen sie wie folgt vor: Sie wählten Startwerte für  $n_W$  und  $n_P$ . Dann hielten sie eine der beiden Variablen fest und reduzierten die jeweils andere in einer festgelegten Schrittweite bis der Mindestumsatz nicht mehr erfüllt werden konnte. Die jeweiligen Kosten für die unterschiedlichen Kombinationen wurden gespeichert. Analog gingen sie mit der anderen Variable vor. Letztlich wurde die Kombination an Anzahl Windanlagen und Photovoltaikzellen gewählt, für die die geringsten Kosten gefunden wurden.

In bisher von CAMMP durchgeführten Modellierungswochen wurden zahlreiche weitere Problemstellungen, die auf ganzzahlige beziehungsweise kombinatorische Op-

timierungsprobleme hinauslaufen, von Lernenden bearbeitet. Ein zweites Beispiel stellt das folgende Problem dar:

### **Problem 2: Parkplatzwahl auf Automobilterminals**

Das Problem wurde von der Firma INFORM gestellt und auf einer Modellierungswoche im Jahr 2017 von einer Lerngruppe bearbeitet. Das Problem befasste sich mit riesigen Autoterminals, auf denen Autos von Fahrerinnen und Fahrern (die auf dem Terminal arbeiten) mehrfach umgeparkt werden müssen. Die Aufgabe bestand darin, die Zeit zu minimieren, die die Fahrerinnen und Fahrer insgesamt zum Umparken auf einen neuen Zwischenparkplatz und zum Gehen bis zum nächsten ihnen zugewiesenen, umzuparkenden Auto benötigen (vgl. Anh. H.2). Die Zuordnung der umzuparkenden Autos (und damit der Fahrenden, die den Autos zugewiesen waren) zu freien Zwischenparkplätzen sollte dazu bestmöglich gewählt werden. Sofern die Lerngruppe genug Zeit hat, sollte nicht nur die Zuweisung zu freien Zwischenparkplätzen, sondern zudem die Zuweisung der Fahrenden zu umzuparkenden Autos als Variable in das Modell integriert werden. Dazu ist die Lerngruppe jedoch nicht mehr gekommen.

**Modellierungsansatz der Lerngruppe:** Die Lernenden entwickelten basierend auf geometrischen Überlegungen ein Modell, welches die Zeit, die eine Fahrerin / ein Fahrer insgesamt braucht, um a) ein Auto von einem Standort aus auf einen Zwischenparkplatz zu fahren und b) anschließend zum nächsten umzuparkenden Auto zu gehen. Basierend auf diesem Modell entwickelten die Lernenden ein Verfahren, um zu entscheiden, welche Zwischenparkplätze verschiedenen Fahrenden zugewiesen werden. Konkret bestimmten sie für jede Fahrerin / jeden Fahrer die Zwischenparkplätze mit geringstem Zeitaufwand. Erhielten mehrere Personen den gleichen optimalen Zwischenparkplatz, so wurde anhand der Zeitdifferenz zum zweitbesten Fahrzeug entschieden:

„Wenn zwei oder mehrere Fahrer denselben optimalen Parkplatz haben, bestimmt man den zweit-optimalen Parkplatz aller Fahrer. Anschließend berechnet man für alle Fahrer die Differenz zwischen der Zeit des zweit-optimalen und des optimalen Parkplatzes. Der Fahrer, dessen Differenz am größten ist, bekommt seinen optimalen Parkplatz. Dieser wird als belegt markiert. Hierbei wird natürlich wieder auf die Terminaldaten zurückgegriffen, um bereits besetzte Parkplätze auszuschließen. Dieser Vorgang wird so oft wiederholt, bis allen Fahrern ein Parkplatz zugewiesen worden ist“ (Zitat aus dem Bericht der Lernenden).

Im Kern verfolgten die Lernenden somit einen **Greedy-Ansatz**, um bei mehreren Fahrenden zu entscheiden, welcher Zwischenparkplatz zugewiesen wird.

### **Problem 3: Optimierung eines Windparks**

Das Problem wurde von der Firma Vattenfall gestellt und auf einer Modellierungswoche im Jahr 2017 und im Jahr 2020 von je einer Lerngruppe bearbeitet. Die Problemstellung bestand im Kern darin, ein Modell für die Leistung eines Offshore-Windparks zu entwickeln und darauf aufbauend das Layout des Windparks (d. h. die

Positionen der einzelnen Windenergieanlagen, kurz: WEAs) unter Einhaltung von gewissen Lastkriterien zu optimieren (vgl. Anh. H.3). Anschließend sollte untersucht werden, wie die gegenseitige Windabschattung der WEAs bei dem selbstgewählten Layout reduziert werden könnte.

**Modellierungsansatz der Lerngruppe 2017:** Die Lernenden entwickelten basierend auf verschiedenen geometrischen und physikalischen Überlegungen ein Modell zur Beschreibung der Leistung des Windparks. Dabei flossen Überlegungen zu Windschatten und Windgeschwindigkeit ein.

Die Lerngruppe berechnete die Positionierung der WEAs basierend auf rationalen Argumenten: Sie legten Reihen auf dem Windparkareal fest. Diese Reihen hatten einen festen Abstand voneinander (der Abstand wurde nicht optimiert). Die Lernenden positionierten die WEAs auf diesen Reihen und zwar erneut in einem festen Abstand zueinander. Die Abstände legten die Lernenden basierend auf Argumenten zur Windabschattung fest. Sie nutzten zwar im Kern einen **musterbasierten Ansatz**, führten jedoch keine Optimierung der Reihenabstände durch.

**Modellierungsansatz der Lerngruppe 2020:** Die Lerngruppe der Modellierungswoche 2020 formulierte tatsächlich ein Optimierungsproblem. Die Zielfunktion entsprach der zuvor modellierten Leistung in Abhängigkeit der Koordinaten der WEAs auf dem zweidimensionalen Windparkareal. Die Lernenden wählten einen **diskreten Ansatz** zur Positionierung der WEAs. Konkret definierten sie ein Gitter auf dem Areal und setzten verschiedene Strategien zur Positionierung der WEAs auf den Gitterpunkten ein: a) eine zufällige Positionierung, b) einen **Brute-Force-Ansatz** und c) eine Methode, bei der die WEAs nach und nach auf die besten Positionen gesetzt werden (**Greedy-Ansatz**). Die Ansätze a) und b) setzten die Lernenden im Programmcode um und berechneten Ergebnisse für verschieden feine Diskretisierungen. Sie berücksichtigten dabei die Nebenbedingung, dass die WEAs im Bereich des Kraftwerks positioniert werden. Zur Umsetzung des Greedy-Ansatzes hatten die Lernenden keine Zeit mehr.

#### **Problem 4: Optimierung eines Fresnelkraftwerks**

Das Problem der Optimierung der Spiegelpositionen eines Fresnelkraftwerks wurde im Rahmen einer Modellierungswoche im Jahr 2011 von einer Lerngruppe bearbeitet (vgl. Anh. H.4).

**Modellierungsansatz der Lerngruppe:** Der Modellierungsprozess der Lernenden wurde von [Roeckerath \(2012\)](#) dokumentiert. Die Lernenden entwickelten zunächst eigenständig ein Modell für die gewünschte Ausrichtung der Spiegel, für Verschattungseffekte und darauf aufbauend den Jahresenergieumsatz des Kraftwerks. Dieser Umsatz sollte maximiert werden. Die Lernenden wählten als Variablen der Optimierung den äquidistanten Abstand zwischen den Spiegelreihen des Fresnelkraftwerks sowie die Absorberrohrhöhe. Damit wählten sie einen **musterbasierten Ansatz** und formulierten ein zweidimensionales Optimierungsproblem. Zur Lösung des Optimierungsproblems beschränkten sie die zulässige Menge auf **diskrete Werte**. Über einen **Brute-Force-Ansatz** bestimmten sie die beste Kombination aus Spiegelabstand und Rohrhöhe.

### Zwischenfazit

Bei der gänzlich unangeleiteten Bearbeitung der Problemstellungen, die auf Optimierungsprobleme hinausliefen, verfolgten die Lernenden diverse Formulierungsansätze. Diese lassen sich leicht auf die bereits zuvor diskutierten Problemklassen und Lösungsansätze aus der Modellierung des Solarkraftwerks in Teil III (musterbasiert, Brute-Force, Greedy) zurückführen. Es fällt zudem auf, dass die Lernenden zur Lösung der Probleme in allen Fällen eine Diskretisierung ansetzten, was nicht weiter verwunderlich scheint. Zum einen, weil mittels Programmen und Computern letztlich endlich viele Möglichkeiten durchprobiert werden können (auch wenn es sehr viele sind), und zum anderen, weil die Lernenden üblicherweise nicht vertraut mit mehrdimensionalen Optimierungsproblemen / Funktionen sind. Derartige Probleme kontinuierlich zu betrachten bedarf somit durchaus ein hohes Maß an Abstraktionsvermögen.

## 19.3 Leitfragen beim Formulieren und Lösen von Optimierungsproblemen

Hier sollen abschließend Leitfragen aufgeführt werden, welche bei der Betreuung von Modellierungsprojekten, die auf Optimierungsprobleme hinauslaufen, Orientierung geben können. Der folgende Fragenkatalog ist als Auswahl zentraler Fragen (ohne Anspruch auf Vollständigkeit) zu verstehen. Diese Fragen sind aus den Erfahrungen bei der Modellierung der in Teil III und IV beschriebenen Problemstellungen – einerseits aus den Erprobungen des Materials und andererseits aus der Fachliteratur – zusammengetragen worden.

### 1. Schritt: Formuliere das Optimierungsproblem

- Kann bereits bei der ersten Formulierung des Problems (Definition der Zielfunktion, Festlegung der Entscheidungsvariablen, Definition der zulässigen Menge) eine wohlgestellte, in der Praxis leicht lösbare Struktur erzeugt werden?
  - Kann das Problem beispielsweise konvex formuliert werden? Kann die Zielfunktion so gewählt werden, dass sie differenzierbar ist?
  - Ist die Zielfunktion gegebenenfalls durch ein physikalisch-technisches Modell / durch eine Simulation gegeben, sodass zunächst wenig Freiheit bei der Formulierung besteht?
- Lässt sich die Frage nach Optimalität sinnvoll auf wenige Variablen (ein oder zwei) reduzieren, sodass die Zielfunktion anschaulich untersucht werden kann? Falls nicht, inwieweit kann der Modellierungs- beziehungsweise Optimierungsprozess angeleitet werden, sodass Lernende dennoch auf verständige Art und Weise an dem Optimierungsproblem arbeiten können (vgl. Lernmaterial zu Empfehlungssystemen in Kap. 17)?

## 2. Schritt: Klassifiziere das Optimierungsproblem

- Welche Art von Optimierungsproblem liegt vor? Diskret oder kontinuierlich? Mit oder ohne Nebenbedingungen? Linear, quadratisch oder nichtlinear? Spielt der Zufall eine Rolle? Ist die Zielfunktion differenzierbar?
- Können Aussagen über Existenz und Eindeutigkeit eines Optimalpunktes getroffen werden? Ist das Problem beispielsweise konvex? Ist die Zielfunktion koerziv? Wird dies im ein- oder zweidimensionalen Fall durch die anschauliche Betrachtung des Zielfunktionsgraphen ersichtlich?

## 3. Schritt: Löse das Optimierungsproblem

Je nach Art des Problems wird ein geeignetes Optimierungsverfahren ausgewählt.

- Eignet sich ein lokales Verfahren (weil das Problem bspw. konvex ist) oder wäre ein globales Suchverfahren geeigneter?
- Kann eines der in der Praxis eingesetzten Verfahren didaktisch reduziert und von den Lernenden selbst erarbeitet werden?
- Ist es denkbar, dass die Lernenden unangeleitet ein problemspezifisches Verfahren / eine Heuristik entwickeln?
- Ist das Problem derart komplex, dass ein Verfahren als Blackbox eingesetzt werden sollte?

Existiert kein geeignetes Verfahren, so gehe zurück zum 1. Schritt (Formulieren des Problems):

- Gibt es eine Möglichkeit das Problem umzuformulieren, sodass es leichter handhabbar wird? Lässt sich die Zielfunktion beispielsweise linearisieren oder konvex formulieren? Können Nebenbedingungen weggelassen oder als Strafterm in die Zielfunktion integriert werden? Kann die Dimensionalität des Problems verringert werden (bspw. wie im Falle der Spiegelpositionsoptimierung durch Einschränkung auf gewisse Muster)?

## 4. Schritt: Evaluiere die gefundene Lösung

Die Evaluation der Lösung sollte auf unterschiedlichen Ebenen stattfinden:

*Validieren / Interpretieren der Lösung im Hinblick auf das reale Problem*

- Ist die gefundene Lösung (Lösungspunkt und zugehöriger Zielfunktionswert) im Hinblick auf das reale Problem sinnvoll? Erfüllt die Lösung alle Nebenbedingungen, die das reale Problem an diese stellt?
- Falls nicht, überdenke Modellannahmen / Vereinfachungen. Prüfe, ob sich beim Beschreiben der Zielfunktion oder der zulässigen Menge Fehler eingeschlichen haben.
- Prüfe, ob ein ungünstiger Startpunkt oder eine schlechte Wahl der Schrittweite zu nicht sinnvollen Ergebnissen geführt hat.

*Güte der gefundenen Lösung und Laufzeit des Verfahrens*

- Ist die Validierung der Lösungspunkte durch die Betrachtung des Zielfunktionsgraphen im anschaulichen Fall möglich?
- Kann die Optimalität der gefundenen Lösungspunkte über klassische Optimalitätskriterien erster und zweiter Ordnung (bspw.  $f'(x) = 0$  und  $f''(x) > 0$ ) überprüft werden?
- Findet das Verfahren in annehmbarer Zeit eine Lösung?
- Wie sind Güte der gefundenen Lösung und Effizienz des Verfahrens gegeneinander abzuwägen?
- Kann die Formulierung des Problems geändert werden, falls das Verfahren keinen zulässigen Punkt findet oder die Laufzeit zu hoch ist? Kann beispielsweise der Suchraum / die zulässige Menge durch Hinzunahme von Nebenbedingungen verkleinert werden?

*Eindeutigkeit und Robustheit*

- Ergeben sich unterschiedliche Optimalpunkte mit gleichem Zielfunktionswert?
- Können mehrfache Lösungen, die im Hinblick auf das reale Problem identisch sind, durch Hinzunahme von Nebenbedingungen ausgeschlossen werden?
- Ergibt sich für unterschiedliche Startpunkte stets dieselbe Lösung?

## 20 Zusammenfassung und Ausblick

Die vorliegende Arbeit liefert einen Überblick über die Optimierung und das mathematische Verständnis, welches notwendig ist, um eine Vielzahl an Optimierungsproblemen einordnen und lösen zu können. Zugleich liefert sie einen Einblick, wie derartige Probleme von Schülerinnen und Schülern bearbeitet werden können.

Im Rahmen dieser Dissertation ist **innovatives, computergestütztes Unterrichtsmaterial** zu zwei Problemstellungen entstanden, welches sich für den Einsatz unter unterschiedlichen Rahmenbedingungen und insbesondere für die Durchführung im Distanzunterricht eignet. Diese Arbeit liefert somit nicht nur eine theoretische Diskussion dessen, was in Modellierungsprojekten mit Lernenden möglich ist, sondern zugleich erprobtes, digitales Lehr- und Lernmaterial. Dieses erlaubt es Lernenden ab Klasse 9 aktiv, möglichst selbstständig, je nach Rahmenbedingung mehr oder weniger angeleitet, aber durchaus problemorientiert an die Modellierung und die sich ergebenden Optimierungsprobleme heranzugehen.

In dieser Arbeit und auch im entwickelten Lernmaterial wurden zwei **reale, hochaktuelle Problemstellungen aus den Bereichen KI und erneuerbare Energien** in den Blick genommen. Bei der Bearbeitung der Problemstellungen waren Optimierungsprobleme zentral: Bei dem ersten Problem, der Modellierung und Optimierung eines Fresnelkraftwerks, wurden die Höhe des Absorberrohres und die Positionen der Spiegel optimiert. Bei dem zweiten Problem, der Entwicklung eines Empfehlungssystems für Filme basierend auf einem realen Datensatz, wurden Einträge von Faktormatrizen bestmöglich gewählt. Dabei kamen Methoden aus dem Bereich des maschinellen Lernens zum Einsatz.

Beide Problemstellungen sind in Forschung, Technik und Wirtschaft von großer Relevanz und zudem in unserem Alltag vertreten. Die Modellierung der beiden Problemstellungen fand zudem, wenngleich elementarer, entlang in der Forschung tatsächlich eingesetzter Methoden statt. Damit liefert diese Arbeit Lehrenden und das entwickelte Lernmaterial Lernenden einen authentischen Einblick in aktuelle Problemlösestrategien von MINT-Berufen.

Bei der Diskussion verschiedener Optimierungsfragen ist deutlich geworden, dass beim Formulieren und Lösen von Optimierungsproblemen **verschiedene Ansätze und Lösungsstrategien** verfolgt werden können. Der Facettenreichtum der Optimierung und die damit verbundenen **Chancen für möglichst offen gestaltete Modellierungsaktivitäten** zu reichhaltigen Problemen wurden umfänglich herausgearbeitet. Dazu wurden die Probleme aus verschiedenen mathematischen Blickwinkeln (diskret, kontinuierlich, Einbezug von Unsicherheiten etc.) angegangen. Es wurde aufgezeigt, welche Ansätze und Formulierungen denkbar sind, und

---

unterschiedliche Modellentscheidungen basierend auf innermathematischen Überlegungen zu den Eigenschaften der Probleme diskutiert. Dabei ist insbesondere ersichtlich geworden, dass das Optimieren um einiges vielseitiger ist, als das aus dem Analysisunterricht bekannte „Nullsetzen der Ableitung“. Wie Lehr- und Lernmaterial gestaltet sein kann, welches den Facettenreichtum der Optimierung aufgreift und den Lernenden erlaubt mathematisch kreativ tätig zu sein, wurde insbesondere durch das entwickelte Unterrichtsmaterial rund um solarthermische Kraftwerke hervorgehoben.

Die Modellierung der Problemstellungen hat deutlich gemacht, dass **fundierte mathematische Überlegungen** beim Formulieren und Lösen von Optimierungsproblemen zentral sind, um **mathematisch wohlgestellte Probleme** (im Hinblick auf Existenz, Eindeutigkeit und praktische Lösbarkeit) zu erhalten. Daraus lässt sich mit Blick auf die Planung, Gestaltung und praktische Umsetzung mathematischer Modellierungsaktivitäten ableiten, dass Lehrkräfte entweder bereits über fundiertes mathematisches Wissen verfügen müssen oder in der Lage sein sollten sich dieses anzueignen. Dies lässt sich als weiteres Argument für die Einbindung des Modellierens in den Mathematikunterricht – und nicht ausschließlich in die Fächer und Disziplinen, in denen das Problem auftritt – ansehen: Denn den Anspruch über fundierte mathematische Kenntnisse zu verfügen oder sich diese aneignen zu können, darf / sollte wohl am ehesten an eine Mathematiklehrkraft gestellt werden. Von Lehrkräften anderer Fächer ist dies hingegen nicht zu verlangen.

Die Tatsache, dass fundierte mathematische Kenntnisse relevant sind, um Optimierungsprobleme geeignet zu formulieren, birgt auf der anderen Seite die Gefahr, dass es seitens der Lernenden bei der Bearbeitung der Probleme zu Überforderung kommt. Die Erfahrungen aus den Modellierungstagen haben jedoch gezeigt, dass es durch eine entsprechende Gestaltung des Lehr- und Lernmaterials durchaus möglich ist, (mathematisch-interessierten) Lernenden einen problemorientierten Zugang zu Optimierungsproblemen zu erlauben. Angeleitet durch kleinere Aufgaben konnten die Lernenden Merkmale sowie Vor- und Nachteile unterschiedlicher Optimierungsformulierungen erarbeiten, anwenden und vergleichend diskutieren.

Diverse Modellierungswochen zu offenen Problemen haben darüber hinaus gezeigt, dass die Lernenden auch gänzlich unangeleitet in der Lage sind Optimierungsprobleme auf verschiedene Weise anzugehen und zu lösen.

Für die Durchführung von Modellierungsaktivitäten im eigenen (Projekt-)Unterricht erhalten Lehrkräfte mit dieser Arbeit **umfassendes Hintergrundwissen** zur mathematischen Optimierung sowie zu zwei ausgewählten Fragestellungen. Zudem liefert diese Arbeit einen Überblick über Ansätze und Strategien, die Lernende bei unangeleiteten Modellierungsprojekten bereits verfolgt haben, und einen Katalog zentraler Fragen beim Formulieren und Lösen von Optimierungsproblemen. Dies soll Orientierung bei der Betreuung offener Modellierungsprojekte bieten.

## Ausblick

Das entwickelte digitale Lehr- und Lernmaterial wurde primär im Rahmen von Modellierungsaktivitäten eingesetzt, die von wissenschaftlichen Mitarbeitenden oder studentischen Hilfskräften – meist außerhalb des regulären Mathematikunterrichts – betreut wurden. Langfristiges Ziel ist es, dass auch (angehende) Lehrkräfte in die Lage versetzt werden diese oder ähnliche Projekte durchzuführen. Im Sinne Freudenthals (1973, S. 9) müssen wir

„beim Entwerfen von Unterrichtsmaterialien- und methoden [...] nicht nur daran denken, was lernbar und lernenswert ist, sondern auch daran, was von dem Stoff die Lehrer lehren lernen können, oder noch besser, was wir die Lehrer lehren können, die Schüler zu lehren.“

Um eine erste Einschätzung vornehmen zu können, inwieweit angehende Lehrkräfte das Lernmaterial für den Einsatz im eigenen Unterricht in Betracht ziehen, wurde dieses in verschiedenen Seminaren für Studierende des Lehramts Mathematik (Sek. II) am KIT, an der RWTH Aachen und an der TU Darmstadt eingesetzt (vgl. Tab. 20.1). In den Seminaren bearbeiteten die Studierenden das Lernmaterial in der Rolle eines Lernenden. Sie reflektierten in verschiedenen Plenumsdiskussionen und abschließend schriftlich, inwieweit sie das Material für den eigenen (zukünftigen) Unterricht als geeignet ansehen und in welchem Rahmen sie dieses einsetzen würden.

**Tab. 20.1:** Durchgeführte Modellierungsprojekte im Rahmen von Lehramtsseminaren. Die mit \* markierte Veranstaltung hat in Präsenz stattgefunden. Thema Solar I  $\hat{=}$  Optimierung der Rohrhöhe, Thema Solar II  $\hat{=}$  Optimierung der Spiegelpositionen

Veranstaltung	TN	Themen	Zeitlicher Rahmen
Workshop im Seminar <i>Anwendung und Modellierung</i> im Bachelorstudium Lehramt Mathematik (Gym/Ge), RWTH Aachen*	32	Solar I	12/2019 3h (2 x 1.5h)
Workshop im Seminar <i>Digitale Werkzeuge für den Mathematikunterricht</i> im Bachelorstudium Lehramt Mathematik (Gym/Ge), KIT	9	Solar II	07/2020 4h (1 Tag)
Workshop im Seminar <i>Anwendung und Modellierung</i> im Bachelorstudium Lehramt Mathematik (Gym/Ge), RWTH Aachen	28	Netflix	05/2021 5h (1 Tag)
Workshop im Rahmen der Veranstaltung <i>Anwendungsorientierter Mathematikunterricht</i> im Lehramtsstudium Mathematik, TU Darmstadt	5	Netflix	06/2021 3.5h (1 Tag)

---

Die Mehrheit der Studierenden gab an, dass sie derartiges Lernmaterial für sinnvoll erachtet und sich den Einsatz im Rahmen von Projekttagen/-wochen oder AGs vorstellen kann. Gleichzeitig wurde das Material wiederholt als zu umfangreich für den regulären Mathematikunterricht eingeschätzt. Im Hinblick darauf, welches weitere Wissen oder Material sie benötigen, damit sie sich die Durchführung im Unterricht zutrauen, wurden mehrfach vertiefte Kenntnisse im Umgang mit den eingesetzten digitalen Werkzeugen und der verwendeten Programmiersprache sowie Hintergrundwissen zur Problemstellung und der eingesetzten Mathematik genannt. Zudem wurde angemerkt, dass derartige Projekte Teil des Lehramtsstudiums sein sollten, um selbst ein besseres Verständnis für das Modellieren entwickeln zu können.

Aus diesen Rückmeldungen werden zentrale Herausforderungen deutlich, die auch von [Niss und Blum \(2020\)](#) aufgegriffen wurden. [Niss und Blum \(2020, S. 90–110\)](#) diskutieren detailliert, welche Herausforderungen der Einbindung von mathematischen Modellierungsaktivitäten in den Mathematikunterricht auf verschiedenen Ebenen gegenüberstehen. Auf Seiten der Lehrkraft nennen die Autoren unter anderem:

- Die Lehrkraft muss selbst über die Kompetenzen verfügen, die von den Lernenden erwartet werden bzw. zu entwickeln sind. Insbesondere benötigt sie a) die Fähigkeit selbst Modellierungen durchzuführen und b) ausreichend inner- sowie außermathematisches Wissen.
- Modellieren ist herausfordernd für die Lehrkraft (und für die Lernenden), weil es den Unterricht weniger vorhersehbar macht. Es ist unter anderem schwieriger abzusehen, welche Wege die Lernenden verfolgen werden, über welche Schwierigkeiten sie stolpern oder wie viel Zeit sie benötigen.
- Auch die Leistungsbewertung wird komplexer – sowohl mündlich als auch schriftlich. Es existiert nicht mehr nur ein *Richtig* oder *Falsch*.

Darüber hinaus gehen [Niss und Blum \(2020, S. 97\)](#) auch auf organisatorische / strukturelle Herausforderungen ein, wie der bereits von den Studierenden angemerkte zeitliche Druck, die Inhalte des Lehrplans durchzubringen.

Für einen verstärkten Einzug der hier beschriebenen Unterrichtsmaterialien und Konzepte in den schulischen (Projekt-)Unterricht lassen sich basierend auf den Rückmeldungen der Studierenden und den von [Niss und Blum \(2020\)](#) benannten Herausforderungen diverse Konsequenzen für zukünftige Forschung sowie die Weiterentwicklung und Erprobung des Materials ziehen:

Aufgrund des Umfangs der vorgestellten Projekte und im Hinblick auf den enormen lehrplanbedingten Zeitdruck im Mathematikunterricht scheint das vorgestellte Lernmaterial aktuell besser für den Einsatz in Fächern geeignet, in denen größere inhaltliche Freiheit besteht. Denkbar sind die Schulfächer IMP und NWT (beides Baden-Württemberg). Alternativ wäre der Einsatz im Rahmen von Fach-/Seminararbeiten, AGs oder Projektzeiten außerhalb des regulären Mathematikunterrichts denkbar. Um das Material stärker in den Mathematikunterricht zu bringen, scheint es in der jetzigen Situation notwendig, den **Umfang des Materials zu reduzieren**.

Auch sollten Konzepte für eine entsprechende **Aus- und Weiterbildung von (angehenden) Lehrkräften** zur mathematischen Modellierung und zur (gestaltenden)

Nutzung der digitalen Werkzeuge erprobt werden. Dabei folgen wir der von Blum (2013) formulierten These:

„Mathematisches Modellieren kann gelernt und gelehrt werden! Die Lehrpersonen müssen die für das Modellieren wichtigen und spezifischen Kompetenzen erwerben, so dass die Anwendung von Mathematik somit zu einem natürlichen Bestandteil des Unterrichts wird“ (Blum, zitiert nach [Borromeo Ferri et al., 2013, S. 2](#)).

Wenngleich in dieser Arbeit nicht dasselbe Verständnis der Gestaltung von Modellierungsaufgaben und -aktivitäten wie bei den genannten Autoren verfolgt wird,<sup>1</sup> so sind wir davon überzeugt, dass auch die hier diskutierte umfangreiche Art des Modellierens gelehrt und gelernt werden kann. Dazu sind die von Blum angesprochenen spezifischen Kompetenzen für das Modellieren bei (angehenden) Lehrkräften umfassend zu fördern. Diese Kompetenzen wurden im Rahmen eines Kompetenzmodells für das Unterrichten von mathematischer Modellierung theoretisch formuliert und bereits um empirische Erfahrungen erweitert (vgl. [Borromeo Ferri et al., 2013, S. 2](#)). Das Kompetenzmodell umfasst vier Dimensionen. Die *theoretische Dimension* beinhaltet das Wissen über verschiedene idealtypische Modellierungskreisläufe und deren Phasen sowie die Kenntnis über die Ziele des Modellierens. Die *aufgabenbezogene Dimension* umfasst das eigenständige Durchführen (komplexer) Modellierung. Ziel ist es, dass Lehrkräfte die Modellierungsprozesse von Lernenden später reflektiert und erfahrungsbasiert begleiten können. Unter diese Dimension fällt zudem die Analyse und (Weiter-)Entwicklung von Modellierungsaufgaben (vgl. [Borromeo Ferri et al., 2013, S. 3](#)). Die *unterrichtsbezogene Dimension* beschreibt die Fähigkeit, Modellierungsunterricht zu planen, durchzuführen und adäquat zu intervenieren. Die *diagnostische Dimension* schließlich bezieht sich auf die Fähigkeit, Phasen im Modellierungsprozess der Lernenden zu identifizieren, Lösungen von Lernenden zu analysieren und mögliche Schwierigkeiten zu diagnostizieren (vgl. [Blum und Borromeo Ferri, 2009, S. 2047](#)).

Die Entwicklung und Erprobung eines ganzheitlichen Aus- und Fortbildungskonzeptes, welches a) die in dieser Arbeit vorgestellte Auffassung und Blickrichtung auf das Modellieren verfolgt, b) Lehrenden authentische Modellierungserfahrungen ermöglicht und c) sich zugleich auf erprobte Kompetenzmodelle beruft, kann und sollte in Zukunft systematisch angegangen werden.

Die Entwicklung eines derartigen Konzeptes mag zwar bei Lehramtsstudierenden oder (im Sinne der Berufsjahre) jungen Lehrkräften aussichtsreich sein, es ist jedoch zu vermuten, dass die Zielgruppe der erfahrenen und lange im Dienst stehenden Lehrkräfte für den Einsatz der Materialien beziehungsweise die Umsetzung der beschriebenen Konzepte schwer über einzelne Fortbildungen zu erreichen ist. Diese Zielgruppe hat vielfach bereits ein langjähriges Portfolio (mehr oder weniger) funktionierender Unterrichtsmaterialien aufgebaut. Von diesen abzuweichen würde einen nicht unbeträchtlichen zeitlichen Aufwand bedeuten und die Bereitschaft erfordern, sich auf (digitale) Innovationen einzulassen. Es bedarf dem Selbstverständnis einer

---

<sup>1</sup>Bei den genannten Autoren liegt der Fokus zumeist stärker auf Aufgaben / Problemen, die weniger zeitintensiv sind und die im Rahmen kürzerer Unterrichtseinheiten im Mathematikunterricht eingesetzt werden können.

---

Lehrkraft stets selbst weiter lernen zu wollen / zu müssen. Für diese Zielgruppe wäre es womöglich sinnvoller in Richtung weniger umfangreicher oder komplexer Modellierungsaktivitäten aus- und weiterzubilden.

Mit Blick auf das notwendige Hintergrundwissen zur Durchführung offener Modellierungsprojekte werden in dieser Arbeit zentrale Überlegungen und Modellentscheidungen beim Bearbeiten von Optimierungsproblemen herausgearbeitet. Es wäre interessant zu untersuchen, inwieweit diese Art der Diskussion rund um das Modellieren (angehenden) Lehrkräften ausreichend Orientierung bietet, um Modellierungsprojekte zu offenen Problemen (mit Fokus auf der Optimierung) zu betreuen.



# Literaturverzeichnis

- Aebli, H. (2006). *Zwölf Grundformen des Lehrens: Eine allgemeine Didaktik auf psychologischer Grundlage. Medien und Inhalte didaktischer Kommunikation, der Lernzyklus*. Klett-Cotta, Stuttgart, 13. Auflage.
- Aggarwal, C. C. (2016). *Recommender Systems: The Textbook*. Springer, Cham, Heidelberg, New York, Dordrecht, London.
- Alt, W. (2004). *Numerische Verfahren der konvexen, nichtglatten Optimierung: Eine anwendungsorientierte Einführung*. Vieweg+Teubner, Wiesbaden.
- Alt, W. (2011). *Nichtlineare Optimierung: Eine Einführung in Theorie, Verfahren und Anwendungen*. Aufbaukurs Mathematik. Vieweg+Teubner, Wiesbaden, 2. Auflage.
- Antoniou, A. und Lu, W.-S. (2007). *Practical Optimization: Algorithms and Engineering Applications*. Springer Science+Business Media, New York.
- Ban, F., Woodruff, D. P., und Zhang, Q. (2019). Regularized Weighted Low Rank Approximation. In *33rd Conference on Neural Information Processing Systems (NeurIPS)*, S. 4059–4069.
- Bandura, A. (1977). Self-efficacy: Toward a Unifying Theory of Behavioral Change. *Psychological Review*, 84(2):191–215.
- Biehler, R. und Fleischer, Y. (2021). Introducing students to machine learning with decision trees using CODAP and Jupyter Notebooks. *Teaching Statistics*, 43:133–142.
- BINE Informationsdienst, FIZ Karlsruhe (2013). Solarthermische Kraftwerke. Konzentriertes Sonnenlicht zur Energieerzeugung nutzen. [www.bine.info/fileadmin/content/Publicationen/Themen-Infos/II\\_2013/themen\\_0213\\_internetx.pdf](http://www.bine.info/fileadmin/content/Publicationen/Themen-Infos/II_2013/themen_0213_internetx.pdf). Zugegriffen: 18. November 2019.
- Blomhøj, M. und Jensen, T. H. (2003). Developing mathematical modelling competence: conceptual clarification and educational planning. *Teaching Mathematics and Its Applications*, 22(3):123–139.
- Bloomfield, P. und Steiger, W. (1980). Least Absolute Deviations Curve-Fitting. *Siam Journal on Scientific and Statistical Computing*, 1(2):290–301.
- Blum, W. (1985). Anwendungsorientierter Mathematikunterricht in der didaktischen Diskussion? In Kahle et al. (Hrsg.), *Mathematische Semesterberichte. Zur Pflege des Zusammenhangs zwischen Schule und Universität*, Band 32, S. 195–232. Vandenhoeck & Ruprecht, Göttingen.

- Blum, W. (2007). Mathematisches Modellieren – zu schwer für Schüler und Lehrer? In *Beiträge zum Mathematikunterricht 2007*, S. 3–12, Hildesheim, Berlin. franzbecker.
- Blum, W. und Borromeo Ferri, R. (2009). Mathematical Modelling: Can it Be Taught and Learnt? *Journal of Mathematical Modelling and Application*, 1(1):45–58.
- Blum, W., Galbraith, P., und Niss, M. (2007). Introduction. In Blum, W., Galbraith, P., Henn, H.-W., und Niss, M. (Hrsg.), *Modelling and Applications in Mathematics Education*, S. 3–32. Springer, New York.
- Blum, W. und Leiß, D. (2005). Modellieren im Unterricht mit der Tanken Aufgabe. *mathematik lehren*, 128:18–21.
- Blum, W. und Niss, M. (1991). Applied mathematical problem solving, modelling, applications, and links to other subjects – state, trends and issues in mathematics instruction. *Educational Studies in Mathematics*, 22:37–68.
- Blum, W. und Wiegand, B. (2000). Offene Aufgaben – wie und wozu? *mathematik lehren*, 100:53–55.
- Bock, W. und Bracke, M. (2015). Applied School Mathematics – Made in Kaiserslautern. In Neuntzer, H. und Prätzel-Wolters, D. (Hrsg.), *Currents in Industrial Mathematics: From Concepts to Research to Education*, S. 403–437. Springer, Berlin, Heidelberg.
- Borromeo Ferri, R., Greefrath, G., und Kaiser, G. (2013). Einführung: Mathematisches Modellieren Lehren und Lernen in Schule und Hochschule. In Borromeo Ferri, R., Greefrath, G., und Kaiser, G. (Hrsg.), *Mathematisches Modellieren für Schule und Hochschule Theoretische und didaktische Hintergründe*, Realitätsbezüge im Mathematikunterricht, S. 1–7. Springer Fachmedien, Wiesbaden.
- Bottou, L., Curtis, F. E., und Nocedal, J. (2018). Optimization Methods for Large-Scale Machine Learning. *SIAM Review*, 60(2):223–311.
- Boyd, S. und Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press, Cambridge.
- Brandt, D., Giersemehl, I., Greulich, D., Herd, E., Jörgens, T., Jürgense-Engl, T., König, A., Reimer, R., Riemer, W., Schmitt-Hartmann, R., Sonntag, R., Spielmanns, M., Stanzel, M., Stühler, A., Zimmermann, P., und Zinser, M. (2014). *Lambacher Schweizer: Mathematik Einführungsphase. Ausgabe Nordrhein-Westfalen*. Ernst Klett Verlag, Stuttgart.
- Bruffee, K. (1995). Sharing Our Toys: Cooperative Learning versus Collaborative Learning. *Change*, 27(1):12–18.
- Bruner, J. (1960). *The Process of Education*. Harvard University Press, Cambridge.
- Burer, S. und Letchford, A. N. (2012). Non-convex mixed-integer nonlinear programming: A survey. *Surveys in Operations Research and Management Science*, 17(2):97–106.

- Caffisch, R. E. (1998). Monte Carlo and quasi-Monte Carlo methods. *Acta Numerica*, 7:1–49.
- Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., und Wirth, R. (2000). CRISP-DM – Step-by-Step Data Mining Guide. [www.kde.cs.uni-kassel.de/wp-content/uploads/lehre/ws2012-13/kdd/files/CRISPWP-0800.pdf](http://www.kde.cs.uni-kassel.de/wp-content/uploads/lehre/ws2012-13/kdd/files/CRISPWP-0800.pdf). Zugegriffen: 04. Januar 2022.
- Computer-Based Math (o. D.). The Computational Thinking Process Poster. [www.computationalthinking.org/helix](http://www.computationalthinking.org/helix). Zugegriffen: 23. Januar 2021.
- Costa, C. J. und Aparicio, J. T. (2020). POST-DS: A Methodology to Boost Data Science. In *15th Iberian Conference on Information Systems and Technologies (CISTI)*, S. 1–6.
- Danckwerts, R. und Vogel, D. (2001). Extremwertaufgaben im Unterricht – Wege der Öffnung. *Der Mathematikunterricht*, 47(4):9–15.
- Das, A. S., Datar, M., Garg, A., und Rajaram, S. (2007). Google News Personalization: Scalable Online Collaborative Filtering. In *Proceedings of the 16th International Conference on World Wide Web*, S. 271–280, New York. ACM.
- Duffie, J. und Beckmann, W. (2013). *Solar Engineering of Thermal Processes*. John Wiley & Sons, Hoboken, 4. Auflage.
- Engel, J. (im Druck). Data Science for informed citizens: On synergies between Digital Literacy and Civic Statistics. In *Proceedings of the 12th CERME conference*.
- Fayyad, U., Piatetsky-Shapiro, G., und Smyth, P. (1996). From Data Mining to Knowledge Discovery in Databases. *AI Magazine*, 17(3):37–54.
- Fest, A. (2007). Vom Modell zum Algorithmus – Kombinatorische Optimierung im Mathematikunterricht. In *Beiträge zum Mathematikunterricht 2007*, S. 344–347, Hildesheim, Berlin. franzbecker.
- Feuerverger, A., He, Y., und Khatri, S. (2012). Statistical Significance of the Netflix Challenge. *Statistical Science*, 27(2):202–231.
- Frank, M., Richter, P., Roeckerath, C., und Schönbrodt, S. (2018). Wie funktioniert eigentlich GPS? Ein Computergestützter Modellierungsworkshop. In Greefrath, G. und Siller, H.-S. (Hrsg.), *Digitale Werkzeuge, Simulationen und mathematisches Modellieren*, S. 137–163. Springer, Wiesbaden.
- Frank, M., Roeckerath, C., und Hattebuhr, M. (2017). Komplexe Modellierung: Solarenergieforschung mit GeoGebra. In Kortenkamp, U. und Kuzle, A. (Hrsg.), *Beiträge zum Mathematikunterricht*, S. 1491–1494, Münster. WTM-Verlag.
- Frank, M., Roeckerath, C., und Schönbrodt, S. (im Druck). Einführung. In Frank, M. und Roeckerath, C. (Hrsg.), *Neue Materialien für einen realitätsbezogenen Mathematikunterricht 9*. Springer Spektrum.
- Freudenthal, H. (1973). *Mathematik als pädagogische Aufgabe*, Band 1. Klett, Stuttgart.

- Frey, K. (2012). *Die Projektmethode: Der Weg zum bildenden Tun*. Pädagogik. Beltz, Weinheim, 12. Auflage.
- Funk, S. (2006). Netflix Update: Try This At Home. <http://sifter.org/simon/journal/20061211.html>. Zugegriffen: 02. Dezember 2021.
- Gallin, P. (2010). Die geometrisch echte Ausgleichsgerade. *Bulletin des VSMP*, S. 23–25. [www.zora.uzh.ch/id/eprint/35404/2/Gallin\\_Die\\_geometrisch\\_echte\\_Ausgleichsgerade.pdf](http://www.zora.uzh.ch/id/eprint/35404/2/Gallin_Die_geometrisch_echte_Ausgleichsgerade.pdf). Zugegriffen: 01. Dezember 2021.
- Gerhard, M., Hattebuhr, M., Schönbrodt, S., und Wohak, K. (im Druck). Aufbau und Einsatzmöglichkeiten des Lehr- und Lernmaterials. In Frank, M. und Roeckerath, C. (Hrsg.), *Neue Materialien für einen realitätsbezogenen Mathematikunterricht 9*. Springer Spektrum.
- Golub, J. (1988). *Focus on Collaborative Learning*. National Council of Teachers of English, ERIC, Urbana, Illinois. <https://files.eric.ed.gov/fulltext/ED297338.pdf>. Zugegriffen: 04. Januar 2022.
- Gould, R. (2021). Toward data-scientific thinking. *Teaching Statistics*, 43:11–22.
- Greefrath, G., Kaiser, G., Blum, W., und Borromeo Ferri, R. (2013). Mathematisches Modellieren – Eine Einführung in theoretische und didaktische Hintergründe. In Borromeo Ferri, R., Greefrath, G., und Kaiser, G. (Hrsg.), *Mathematisches Modellieren für Schule und Hochschule. Realitätsbezüge im Mathematikunterricht*, S. 11–37. Springer Spektrum, Wiesbaden.
- Greefrath, G., Oldenburg, R., Siller, H.-S., Ulm, V., und Weigand, H.-G. (2016). Integralrechnung. In Greefrath, G., Oldenburg, R., Siller, H.-S., Ulm, V., und Weigand, H.-G. (Hrsg.), *Didaktik der Analysis: Aspekte und Grundvorstellungen zentraler Begriffe*, S. 217–286. Springer, Berlin, Heidelberg.
- Greefrath, G. und Siller, H.-S. (2018). Digitale Werkzeuge, Simulationen und mathematisches Modellieren. In Greefrath, G. und Siller, H.-S. (Hrsg.), *Digitale Werkzeuge, Simulationen und mathematisches Modellieren*, S. 3–22. Springer, Wiesbaden.
- Greefrath, G., Siller, H.-S., und Ludwig, M. (2017). Modelling problems in German grammar school leaving examinations (Abitur) – Theory and practice. In *Proceedings of the 10th CERME conference*, S. 932–939.
- Gudjons, H. (2014). *Handlungsorientiert lehren und lernen: Schüleraktivierung – Selbsttätigkeit – Projektarbeit*. Erziehen und Unterrichten in der Schule. Verlag Julius Klinkhardt, Bad Heilbrunn, 8. Auflage.
- Günther, M. (2004). Advanced CSP Teaching Materials. Chapter 6: Linear Fresnel Technology. EnerMENA und DLR. [www.energy-science.org/bibliotheque/cours/1361468614Chapter%2006%20Fresnel.pdf](http://www.energy-science.org/bibliotheque/cours/1361468614Chapter%2006%20Fresnel.pdf). Zugegriffen: 07. Juni 2020.
- Hansen, P. (1979). Methods of nonlinear 0-1 Programming. *Annals of Discrete Mathematics*, 5:53–70.

- Hardt, M. und Recht, B. (2021). Patterns, Predictions, and Actions: A story about machine learning. <https://mlstory.org/pdf/patterns.pdf>. Zugegriffen: 22. Dezember 2021.
- Hastie, T., Mazumder, R., Lee, J. D., und Zadeh, R. (2015). Matrix Completion and Low-Rank SVD via Fast Alternating Least Squares. *Journal of Machine Learning Research*, 16(1):3367–3402.
- Hattebuhr, M. (im Druck). Gibt es den Klimawandel wirklich? – Statistische Analyse von realen Temperaturdaten. In Frank, M. und Roeckerath, C. (Hrsg.), *Neue Materialien für einen realitätsbezogenen Mathematikunterricht 9*. Springer Spektrum.
- Hattebuhr, M., Frank, M., und Roeckerath, C. (2015). Optimierung der Spiegel in einem Solarkraftwerk – Projekttag des EducationLabs CAMMP der RWTH Aachen. In Caluori, F., Linneweber-Lammerskitten, H., und Streit, C. (Hrsg.), *Beiträge zum Mathematikunterricht*, S. 356–359, Münster. WTM-Verlag.
- He, J. und Yu, X. (2001). Conditions for the convergence of evolutionary algorithms. *Journal of Systems Architecture*, 47(7):601–612.
- Heiliö, M. und Pohjolainen, S. (2016). Introduction. In Pohjolainen, S. (Hrsg.), *Mathematical Modelling*, S. 1–5. Springer International Publishing.
- Hemker, T. (2008). *Derivative Free Surrogate Optimization for Mixed-Integer Non-linear Black Box Problems in Engineering*. Dissertation, TU Darmstadt.
- Hinrichs, G. (2008). *Modellierung im Mathematikunterricht*. Springer, Heidelberg.
- Hoeffler, K. (2020). Entwicklung von Unterrichtsmaterial zum Thema Solarenergie im Rahmen eines interdisziplinären mathematischen Modellierungsprojektes. Bachelorarbeit, Karlsruher Institut für Technologie (KIT).
- Hofmann, S. (im Druck). Teaching data science in school: Digital learning material on predictive text systems. In *Proceedings of the 12th CERME conference*.
- Hu, Y., Koren, Y., und Volinsky, C. (2008). Collaborative Filtering for Implicit Feedback Datasets. *2008 Eighth IEEE International Conference on Data Mining*, S. 263–272.
- Humenberger, H. (2010). Das Quadrat als optimales Rechteck – Optimieren als fundamentale Idee erfahren. *mathematik lehren*, 159:44–50.
- Humenberger, H. (2015a). Optimieren – Nicht nur mit Differentialrechnung! *Der Mathematikunterricht*, 61(1):14–19.
- Humenberger, H. (2015b). Zur Einführung. *Der Mathematikunterricht. Schwerpunkt: Optimieren*, 61:2–3.
- Humenberger, H. und Reichel, H.-C. (1995). *Fundamentale Ideen der Angewandten Mathematik und ihre Umsetzung im Unterricht*. BI, Mannheim.
- Hußmann, S. und Lutz-Westphal, B. (2007). *Kombinatorische Optimierung erleben*. Vieweg+Teubner, Wiesbaden.

- Jannach, D., Zanker, M., Felfernig, A., und Friedrich, G. (2010). *Recommender Systems: An Introduction*. Cambridge University Press, Cambridge.
- Janssen, D. (2020). Machine Learning in der Schule – Ein praxisorientierte Einführung in künstliche neuronale Netze, Gesichtserkennung und Co. *Science On Stage Deutschland e. V.*
- Johnson, D. und Johnson, R. (1989). *Cooperation and Competition: Theory and Research*. Interaction Book Company, Edina, Minnesota.
- Johnson, D. und Johnson, R. (2014). Using technology to revolutionize cooperative learning: An opinion. *Frontiers in Psychology*, 5:1–3.
- Jordan, A., Krauss, S., Löwen, K., Blum, W., Neubrand, M., Brunner, M., Kunter, M., und Baumert, J. (2008). Aufgaben im COACTIV-Projekt: Zeugnisse des kognitiven Aktivierungspotentials im deutschen Mathematikunterricht. *Journal für Mathematik-Didaktik*, 29:83–107.
- Kaiser, G. (2007). Modelling and modelling competencies in school. In Haines, C. P., Galbraith, P., Blum, W., und Khan, S. (Hrsg.), *Mathematical Modelling (ICTMA 12) Education, Engineering and Economics*, S. 110–119, Chichester. Horwood Publishing.
- Kaiser, G. und Schwarz, B. (2010). Authentic Modelling Problems in Mathematics Education: Examples and Experiences. *Journal für Mathematik-Didaktik*, 31:51–76.
- Kambezidis, H. und Psiloglou, B. (2008). The Meteorological Radiation Model (MRM): Advancements and Applications. In Badescu, V. (Hrsg.), *Modeling Solar Radiation at the Earth's Surface: Recent Advances*, S. 357–389. Springer, Berlin, Heidelberg.
- Kapp, F. und Schulte, C. (2019). Einsatz von Jupyter Notebooks am Beispiel eines fiktiven Kriminalfalls. In Pasternak, A. (Hrsg.), *Proceedings zur 18. GI-Fachtagung Informatik und Schule "Informatik für Alle"*, Gesellschaft für Informatik, S. 247–256.
- Kincaid, N., Mungas, G., Kramer, N., und Zhu, G. (2019). Sensitivity analysis on optical performance of a novel linear Fresnel concentrating solar power collector. *Solar Energy*, 180:383–390.
- Kistler, B. L. (1986). A User's Manual for DELSOL3: A Computer Code for Calculating the Optical Performance and Optimal System Design for Solar Thermal Central Receiver Plants. *Sandia National Labs*. Technical report. [www.osti.gov/servlets/purl/7228886](http://www.osti.gov/servlets/purl/7228886). Zugegriffen: 04. Januar 2022.
- Klahold, A. (2009). *Empfehlungssysteme: Recommender Systems – Grundlagen, Konzepte und Lösungen*. Vieweg+Teubner, Wiesbaden.
- Kleesiek, J., Murray, J., Strack, C., Kaissis, G., und Braren, R. (2020). Wie funktioniert maschinelles Lernen? *Radiologe*, 60:24–31.

- Koren, Y. (2010). Factor in the Neighbors: Scalable and Accurate Collaborative Filtering. *ACM Trans. Knowl. Discov. Data*, 4(1):1–24.
- Koren, Y. und Bell, R. (2011). Advances in Collaborative Filtering. In Ricci, F., Rokach, L., Shapira, B., und Kantor, P. B. (Hrsg.), *Recommender Systems Handbook*, S. 145–186. Springer, Boston.
- Koren, Y., Bell, R., und Volinsky, C. (2009). Matrix Factorization Techniques for Recommender Systems. *Computer*, 42(8):30–37.
- Korte, B. und Vygen, J. (2018). *Kombinatorische Optimierung*. Springer Spektrum, Berlin, 3. Auflage.
- Krahforst, C. (2016). Didaktisch-methodische Weiterentwicklung des CAMMP day Moduls Spiegelaufstellung in einem Solarkraftwerk für den Einsatz in der Mittelstufe. Schriftliche Hausarbeit im Rahmen der ersten Staatsprüfung, Abschlussarbeit, RWTH Aachen.
- Kreckler, J. (2015). *Standortplanung und Geometrie: Mathematische Modellierung im Regelunterricht*. Springer Spektrum.
- Kultusministerkonferenz (2003). Bildungsstandards im Fach Mathematik für den Mittleren Schulabschluss. [www.kmk.org/fileadmin/Dateien/veroeffentlichungen\\_beschluesse/2003/2003\\_12\\_04-Bildungsstandards-Mathe-Mittleren-SA.pdf](http://www.kmk.org/fileadmin/Dateien/veroeffentlichungen_beschluesse/2003/2003_12_04-Bildungsstandards-Mathe-Mittleren-SA.pdf). Zugegriffen: 03. März 2021.
- Kultusministerkonferenz (2012). Bildungsstandards im Fach Mathematik für die Allgemeine Hochschulreife. [www.kmk.org/fileadmin/Dateien/veroeffentlichungen\\_beschluesse/2012/2012\\_10\\_18-Bildungsstandards-Mathe-Abi.pdf](http://www.kmk.org/fileadmin/Dateien/veroeffentlichungen_beschluesse/2012/2012_10_18-Bildungsstandards-Mathe-Abi.pdf). Zugegriffen: 03. März 2021.
- Kultusministerkonferenz (2017). Bildung in der digitalen Welt – Strategie der Kultusministerkonferenz. [www.kmk.org/fileadmin/Dateien/pdf/PresseUndAktuelles/2018/Digitalstrategie\\_2017\\_mit\\_Weiterbildung.pdf](http://www.kmk.org/fileadmin/Dateien/pdf/PresseUndAktuelles/2018/Digitalstrategie_2017_mit_Weiterbildung.pdf). Zugegriffen: 19. November 2021.
- Kultusministerkonferenz (2020). Bildungsstandards im Fach Physik für die Allgemeine Hochschulreife. [www.kmk.org/fileadmin/Dateien/veroeffentlichungen\\_beschluesse/2020/2020\\_06\\_18-BildungsstandardsAHR\\_Physik.pdf](http://www.kmk.org/fileadmin/Dateien/veroeffentlichungen_beschluesse/2020/2020_06_18-BildungsstandardsAHR_Physik.pdf). Zugegriffen: 16. August 2021.
- Kuntze, S. (2018). Flächeninhalt und Volumen. In Padberg, F. und Büchter, A. (Hrsg.), *Didaktik der Geometrie für die Sekundarstufe I*, S. 149–177. Springer, Berlin, Heidelberg.
- Leary, P. L. und Hankins, J. D. (1979). User’s guide for MIRVAL: A Computer Code for Comparing Designs of Heliostat-Receiver Optics for Central Receiver Solar Power Plants. *Sandia National Labs*. Technical report. [www.osti.gov/servlets/purl/6371450/](http://www.osti.gov/servlets/purl/6371450/), Zugegriffen: 04. Januar 2022.
- Lehrerinnenfortbildung Baden-Württemberg (2016). Das Problem des kürzesten Weges: Dijkstra und die Ameisen – Bildungsplan 2016: IMP Klasse

9. [https://lehrerfortbildung-bw.de/u\\_matnatech/imp/gym/bp2016/fb2/i01\\_duc/1\\_hintergrund/2\\_gang/5\\_dijksra/](https://lehrerfortbildung-bw.de/u_matnatech/imp/gym/bp2016/fb2/i01_duc/1_hintergrund/2_gang/5_dijksra/). Zugegriffen: 25. November 2021.
- Linden, G., Smith, B., und York, J. (2003). Amazon.com recommendations: item-to-item collaborative filtering. *Internet Computing, IEEE*, 7:76–80.
- Lohr, S. (2010). Netflix Cancels Contest After Concerns Are Raised About Privacy. [www.nytimes.com/2010/03/13/technology/13netflix.html](http://www.nytimes.com/2010/03/13/technology/13netflix.html). Zugegriffen: 24. November 2021.
- Lutchman, S. L., Groenwold, A. A., Gauché, P., und Bode, S. (2014). On Using a Gradient-based Method for Heliostat Field Layout Optimization. In *Proceedings of the SolarPACES 2013 International Conference. Energy Procedia*, Band 49, S. 1429–1438.
- Lutz-Westphal, B. (2006). *Kombinatorische Optimierung – Inhalte und Methoden für einen authentischen Mathematikunterricht*. Dissertation, TU Berlin.
- Maaß, K. (2007). *Mathematisches Modellieren – Aufgaben für die Sekundarstufe I*. Cornelsen Scriptor, Berlin.
- Maaß, K. (2010). Classification Scheme for Modelling Tasks. *Journal für Mathematik-Didaktik*, 31(2):285–311.
- Miettinen, K. (1998). *Nonlinear Multiobjective Optimization*. International Series in Operations Research & Management Science. Springer, Bosten.
- Ministerium für Bildung, Wissenschaft, Weiterbildung und Kultur Rheinland-Pfalz (2007). Rahmenlehrplan Mathematik: Klassenstufen 5 bis 9/10. <https://lehrplaene.bildung-rp.de>. Zugegriffen: 25. November 2021.
- Ministerium für Bildung, Wissenschaft, Weiterbildung und Kultur Rheinland-Pfalz (2015). Lehrplan Mathematik: Grund- und Leistungsfach in der gymnasialen Oberstufe (Mainzer Studienstufe). <https://lehrplaene.bildung-rp.de>. Zugegriffen: 25. November 2021.
- Ministerium für Bildung, Wissenschaft, Weiterbildung und Kultur Rheinland-Pfalz (o. D.a). Lehrplan Informatik: Grund- und Leistungsfach Einführungsphase und Qualifikationsphase der gymnasialen Oberstufe (Mainzer Studienstufe). <https://lehrplaene.bildung-rp.de>. Zugegriffen: 25. November 2021.
- Ministerium für Bildung, Wissenschaft, Weiterbildung und Kultur Rheinland-Pfalz (o. D.b). Lehrpläne für die Naturwissenschaftlichen Fächer für die weiterführenden Schulen in Rheinland-Pfalz: Klassenstufen 7 bis 9/10. <https://lehrplaene.bildung-rp.de>. Zugegriffen: 25. November 2021.
- Ministerium für Kultus, Jugend und Sport Baden-Württemberg (2016a). Bildungsplan des Gymnasiums: Ethik. [www.bildungsplaene-bw.de/site/bildungsplan/get/documents/lbw/export-pdf/depot-pdf/ALLG/BP2016BW\\_ALLG\\_GYM\\_ETH.pdf](http://www.bildungsplaene-bw.de/site/bildungsplan/get/documents/lbw/export-pdf/depot-pdf/ALLG/BP2016BW_ALLG_GYM_ETH.pdf). Zugegriffen: 15. Dezember 2021.
- Ministerium für Kultus, Jugend und Sport Baden-Württemberg (2016b). Bildungsplan des Gymnasiums: Informatik. [www.bildungsplaene-bw.de](http://www.bildungsplaene-bw.de)

- [site/bildungsplan/get/documents/lsbw/export-pdf/BP2016BW\\_ALLG\\_GYM/INFWFO/bildungsplan\\_BP2016BW\\_ALLG\\_GYM\\_INFWFO.pdf](http://www.bildungsplaene-bw.de/site/bildungsplan/get/documents/lsbw/export-pdf/BP2016BW_ALLG_GYM/INFWFO/bildungsplan_BP2016BW_ALLG_GYM_INFWFO.pdf). Zugegriffen: 25. November 2021.
- Ministerium für Kultus, Jugend und Sport Baden-Württemberg (2016c). Bildungsplan des Gymnasiums: Mathematik. [www.bildungsplaene-bw.de/site/bildungsplan/get/documents/lsbw/export-pdf/depot-pdf/ALLG/BP2016BW\\_ALLG\\_GYM\\_M.pdf](http://www.bildungsplaene-bw.de/site/bildungsplan/get/documents/lsbw/export-pdf/depot-pdf/ALLG/BP2016BW_ALLG_GYM_M.pdf). Zugegriffen: 25. November 2021.
- Ministerium für Kultus, Jugend und Sport Baden-Württemberg (2016d). Bildungsplan des Gymnasiums: Physik. [www.bildungsplaene-bw.de/Lde/LS/BP2016BW\\_ALLG\\_GYM/PH](http://www.bildungsplaene-bw.de/Lde/LS/BP2016BW_ALLG_GYM/PH). Zugegriffen: 25. November 2021.
- Ministerium für Kultus, Jugend und Sport Baden-Württemberg (2018). Bildungspläne Informatik, Mathematik, Physik (IMP): Profillfach. [www.bildungsplaene-bw.de/site/bildungsplan/get/documents/lsbw/export-pdf/depot-pdf/ALLG/BP2016BW\\_ALLG\\_GYM\\_IMP.pdf](http://www.bildungsplaene-bw.de/site/bildungsplan/get/documents/lsbw/export-pdf/depot-pdf/ALLG/BP2016BW_ALLG_GYM_IMP.pdf). Zugegriffen: 25. November 2021.
- Ministerium für Kultus, Jugend und Sport Baden-Württemberg (2019). Bildungspläne Naturwissenschaft und Technik (NwT): Schulversuch Basisfach in der Kursstufe. [www.bildungsplaene-bw.de/site/bildungsplan/get/documents/lsbw/export-pdf/BP2016BW\\_ALLG\\_GYM/NWTBFO/bildungsplan\\_BP2016BW\\_ALLG\\_GYM\\_NWTBFO.pdf](http://www.bildungsplaene-bw.de/site/bildungsplan/get/documents/lsbw/export-pdf/BP2016BW_ALLG_GYM/NWTBFO/bildungsplan_BP2016BW_ALLG_GYM_NWTBFO.pdf). Zugegriffen: 25. November 2021.
- Ministerium für Schule und Weiterbildung des Landes Nordrhein-Westfalen (2014a). Kernlehrplan für die Sekundarstufe II Gymnasium/Gesamtschule in Nordrhein-Westfalen: Informatik. [www.schulentwicklung.nrw.de/lehrplaene/upload/klp\\_SII/if/KLP\\_GOSt\\_Informatik.pdf](http://www.schulentwicklung.nrw.de/lehrplaene/upload/klp_SII/if/KLP_GOSt_Informatik.pdf). Zugegriffen: 25. November 2021.
- Ministerium für Schule und Weiterbildung des Landes Nordrhein-Westfalen (2014b). Kernlehrplan für die Sekundarstufe II Gymnasium/Gesamtschule in Nordrhein-Westfalen: Mathematik. [www.schulentwicklung.nrw.de/lehrplaene/upload/klp\\_SII/m/KLP\\_GOSt\\_Mathematik.pdf](http://www.schulentwicklung.nrw.de/lehrplaene/upload/klp_SII/m/KLP_GOSt_Mathematik.pdf). Zugegriffen: 25. November 2021.
- Ministerium für Schule und Weiterbildung des Landes Nordrhein-Westfalen (2019a). Kernlehrplan für die Sekundarstufe I Gymnasium in Nordrhein-Westfalen: Mathematik. [www.schulentwicklung.nrw.de/lehrplaene/lehrplan/195/g9\\_m\\_klp\\_3401\\_2019\\_06\\_23.pdf](http://www.schulentwicklung.nrw.de/lehrplaene/lehrplan/195/g9_m_klp_3401_2019_06_23.pdf). Zugegriffen: 25. November 2021.
- Ministerium für Schule und Weiterbildung des Landes Nordrhein-Westfalen (2019b). Kernlehrplan für die Sekundarstufe I Gymnasium in Nordrhein-Westfalen: Physik. [www.schulentwicklung.nrw.de/lehrplaene/lehrplan/208/g9\\_ph\\_klp\\_203411\\_2019\\_06\\_23.pdf](http://www.schulentwicklung.nrw.de/lehrplaene/lehrplan/208/g9_ph_klp_203411_2019_06_23.pdf). Zugegriffen: 25. November 2021.
- Ministerium für Schule und Weiterbildung des Landes Nordrhein-Westfalen (2021). Kernlehrplan für die Sekundarstufe I – Klasse 5 und 6 in Nordrhein-Westfalen: Informatik. [www.schulentwicklung.nrw.de/lehrplaene/lehrplan/260/si\\_kl5u6\\_if\\_klp\\_2021\\_07\\_01.pdf](http://www.schulentwicklung.nrw.de/lehrplaene/lehrplan/260/si_kl5u6_if_klp_2021_07_01.pdf). Zugegriffen: 25. November 2021.
- Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill, New York.

- Mitchell, T. M. (2017). Key Ideas in Machine Learning. [www.cs.cmu.edu/~tom/mlbook/keyIdeas.pdf](http://www.cs.cmu.edu/~tom/mlbook/keyIdeas.pdf). Entwurf eines Kapitels für die 2. Auflage des Lehrbuchs Machine Learning. Zugegriffen: 02. Dezember 2021.
- Mühlehner, P. (2021). Kombination von Informatik und Sport – Theorie und Umsetzung in der Schule. Diplomarbeit, Johannes Kepler Universität Linz. <https://epub.jku.at/obvulihs/download/pdf/5971517?originalFilename=true>. Zugegriffen: 25. November 2021.
- Narayanan, A. und Shmatikov, V. (2008). Robust De-anonymization of Large Datasets (How To Break Anonymity of the Netflix Prize Dataset). *arXiv*. <https://arxiv.org/pdf/cs/0610105.pdf>. Zugegriffen: 26. November 2021.
- Niss, M. (1992). *Applications and modelling in school mathematics – directions for future development*. IMFUFA Roskilde Universitetscenter, Roskilde.
- Niss, M. und Blum, W. (2020). *The Learning and Teaching of Mathematical Modelling*. Routledge, London, New York.
- Nocedal, J. und Wright, S. J. (2006). *Numerical Optimization*. Springer Science+Business Media, New York, 2. Auflage.
- Noone, C. J., Torrillhon, M., und Mitsos, A. (2012). Heliostat Field Optimization: A New Computationally Efficient Model and Biomimetic Layout. *Solar Energy*, 86(2):792–803.
- Oldenburg, R. (2006). Numerische Optimierung – ein schneller Weg zu komplexer Modellbildung. In *ISTRON – Materialien für einen realitätsbezogenen Mathematikunterricht*. franzbecker.
- O’Neil, C. (2016). *Weapons of Math Destruction: How Big Data Increases Inequality And Threatens Democracy*. Crown Books, New York.
- Opel, S., Schlichtig, M., Schulte, C., Biehler, R., Frischemeier, D., Podworny, S., und Wassong, T. (2019). Entwicklung und Reflexion einer Unterrichtssequenz zum Maschinellen Lernen als Aspekt von Data Science in der Sekundarstufe II. In Pasternak, A. (Hrsg.), *Proceedings zur 18. GI-Fachtagung Informatik und Schule “Informatik für Alle”, Gesellschaft für Informatik*, S. 285–294.
- Palm, T. (2002). *The realism of mathematical school tasks: Features and consequences*. Dissertation, Umea University.
- Panitz, T. (1999a). Collaborative versus cooperative learning: A comparison of the two concepts which will help us understand the underlying nature of interactive learning. *ERIC*. <https://files.eric.ed.gov/fulltext/ED448443.pdf>. Zugegriffen: 20. Dezember 2021.
- Panitz, T. (1999b). The Motivational Benefits of Cooperative Learning. *New directions for teaching and learning*, 78:59–67.
- Pimentel, J., Murta, L., Braganholo, und Freire, J. (2021). Understanding and improving the quality and reproducibility of Jupyter notebooks. *Empirical Software Engineering*, 26(65):1–55.

- Pollak, H. (1979). The interaction between mathematics and other school subjects. In UNESCO (Hrsg.), *New Trends in Mathematics Teaching IV*, S. 232–248, Paris.
- Rantzau, L. (2021). Empfehlungssysteme basierend auf Nachbarschaftsmethoden – mathematisch-fachliche Diskussion und Entwicklung digitalen Lernmaterials zur Netflix Challenge für Schüler\*innen der Sekundarstufe II. Bachelorarbeit, Karlsruher Institut für Technologie (KIT).
- Ricci, F., Rokach, L., und Shapira, B. (2011). Introduction to Recommender Systems Handbook. In Ricci, F., Rokach, L., Shapira, B., und Kantor, P. B. (Hrsg.), *Recommender Systems Handbook*, S. 1–35. Springer, Boston.
- Richter, P. (2017). *Simulation and Optimization of Solar Thermal Power Plants*. Dissertation, RWTH Aachen.
- Richter, P., Laukamp, D., Gerdes, L., Frank, M., und Abraham, E. (2016). Helio-stat Field Layout Optimization with Evolutionary Algorithms. In *GCAI 2016*, Band 41, S. 240–252.
- Roberts, T. S. (2004). Preface. In Roberts, T. S. (Hrsg.), *Online Collaborative Learning*, S. vi–xiii. Idea Group Publishing, Hershey, London.
- Roccia, J. P., Piaud, B., Coustet, C., Caliot, C., Guillot, E., Flamant, G., und Delatorre, J. (2012). SOLFAST, a Ray-Tracing Monte-Carlo software for solar concentrating facilities. *Journal of Physics: Conference Series*, 369:1–10.
- Roeckerath, C. (2012). Mathematische Modellierung der Spiegel eines solarthermischen Kraftwerks im Rahmen einer Modellierungswoche und einer Projektwoche in der Sek. II. Schriftliche Hausarbeit im Rahmen der Zweiten Staatsprüfung für das Lehramt an Gymnasien und Gesamtschulen.
- Sarwar, B., Karypis, G., Konstan, J., und Riedl, J. (2001). Item-based Collaborative Filtering Recommendation Algorithms. *Proceedings of the ACM World Wide Web Conference 01*, S. 285–295.
- SAS Institute (2017). Introduction to SEMMA. <https://documentation.sas.com/doc/en/emref/14.3/n061bzurmej4j3n1jnj8bbj1a2.htm>. Zugegriffen: 25. November 2021.
- Schewe, L. und Schmidt, M. (2019). Gemischt-ganzzahlige nichtlineare Optimierungsprobleme (MINLPs). In Schewe, L. und Schmidt, M. (Hrsg.), *Optimierung von Versorgungsnetzen*, Kap. 10, S. 125–144. Springer Spektrum, Berlin, Heidelberg.
- Schmidt, L. (2019). Machine Learning: automatische Bilderkennung mit Mathematik?! – Ein Lehr-Lern-Modul im Rahmen eines mathematischen Modellierungstages für Schülerinnen und Schüler der Sekundarstufe II. Masterarbeit, RWTH Aachen.
- Schmitz, M., Schwarzbözl, P., Buck, R., und Pitz-Paal, R. (2006). Assessment of the potential improvement due to multiple apertures in central receiver systems with secondary concentrators. *Solar Energy*, 80:111–120.

- Schreiber, A. (1983). Bemerkungen zur Rolle universeller Ideen im mathematischen Denken. *math. did.*, 6:65–76.
- Schupp, H. (1992). *Optimieren – Extremwertbestimmung im Mathematikunterricht*. BI, Mannheim.
- Schupp, H. (1997). Optimieren ist fundamental. *mathematik lehren*, 81:4–10.
- Schwarzer, R. und Jerusalem, M. (2002). Das Konzept der Selbstwirksamkeit. In Jerusalem, M. und Hopf, D. (Hrsg.), *Selbstwirksamkeit und Motivationsprozesse in Bildungsinstitutionen*, S. 28–53. Beltz, Weinheim.
- Schönbrodt, S. (2019). *Maschinelle Lernmethoden für Klassifizierungsprobleme – Perspektiven für die mathematische Modellierung mit Schülerinnen und Schülern*. Springer Spektrum.
- Schönbrodt, S. (im Druck). Erneuerbare Energien – Modellierung und Optimierung eines Solarkraftwerks. In Frank, M. und Roeckerath, C. (Hrsg.), *Neue Materialien für einen realitätsbezogenen Mathematikunterricht 9*. Springer Spektrum.
- Schönbrodt, S., Camminady, T., und Frank, M. (2021). Mathematische Grundlagen der Künstlichen Intelligenz im Schulunterricht – Chancen für eine Bereicherung des Unterrichts in linearer Algebra. *Mathematische Semesterberichte*.
- Schönbrodt, S. und Frank, M. (2020). Schüler/innen forschen zu erneuerbaren Energien – Optimierung eines Solarkraftwerks. In Siller, H.-S., Weigel, W., und Wörler, J. F. (Hrsg.), *Beiträge zum Mathematikunterricht*, S. 1534–1534, Münster. WTM-Verlag.
- Schönbrodt, S. und Frank, M. (2021). Digitales Lernmaterial zur Netflix Challenge. In Hein, K., Heil, C., Ruwisch, S., und Prediger, S. (Hrsg.), *Beiträge zum Mathematikunterricht*. WTM-Verlag, Münster.
- Schönbrodt, S., Wohak, K., und Frank, M. (im Druck). Digital Tools to Enable Mathematical Modeling Online. *MSEL*.
- Schüller, K., Koch, H., und Rampelt, F. (2021). Data-Literacy Charta. [www.stifterverband.org/sites/default/files/data-literacy-charta\\_v1\\_2.pdf](http://www.stifterverband.org/sites/default/files/data-literacy-charta_v1_2.pdf). Zugegriffen: 25. November 2021.
- Serrano, L. (2018). How does Netflix recommend movies? Matrix Factorization. [www.youtube.com/watch?v=ZspR5PZemcs](https://www.youtube.com/watch?v=ZspR5PZemcs). Zugegriffen: 02. Dezember 2021.
- Siller, H.-S. und Greefrath, G. (2010). Mathematical Modelling in Class regarding to Technology. In *Proceedings of the 6th CERME conference*, S. 2136–2145.
- Sra, S., Nowozin, S., und Wright, S. J. (2011). Introduction: Optimization for Machine Learning. In Sra, S., Nowozin, S., und Wright, S. J. (Hrsg.), *Optimization for Machine Learning*. The MIT Press, London.
- Srebro, N. und Jaakkola, T. (2003). Weighted Low-Rank Approximations. In Fawcett, T. und Mishra, N. (Hrsg.), *Proceedings of the 20th International Conference on Machine Learning (ICML)*, S. 720–727, New York. ACM.

- Stahl, G., Koschmann, T., und Suthers, D. (2006). Computer-supported collaborative learning: An historical perspective. In Sawyer, R. (Hrsg.), *Cambridge handbook of the learning sciences*, S. 409–426. Cambridge University Press, Cambridge.
- Stein, O. (2021a). *Grundzüge der Globalen Optimierung*. Springer Spektrum, Berlin, 2. Auflage.
- Stein, O. (2021b). *Grundzüge der Nichtlinearen Optimierung*. Springer Spektrum, Berlin, 2. Auflage.
- Stender, P. (2018). Lehrerinterventionen bei der Betreuung von Modellierungsfragestellungen auf Basis von heuristischen Strategien. In Borromeo Ferri, R. und Blum, W. (Hrsg.), *Lehrerkompetenzen zum Unterrichten mathematischer Modellierung: Konzepte und Transfer*, S. 101–122. Springer Fachmedien, Wiesbaden.
- Strang, G. (2009). *Linear Algebra and Learning from Data*. Wellesley-Cambridge Press.
- Sube, M. (2019). *Entwicklung und Evaluation von Unterrichtsmaterial zu Data Science und mathematischer Modellierung mit Schülerinnen und Schülern*. Dissertation, RWTH Aachen.
- Sube, M., Camminady, T., Frank, M., und Roeckerath, C. (2020). Vorschlag für eine Abiturprüfungsaufgabe mit authentischem und relevantem Realitätsbezug. In Greefrath, G. und Maaß, K. (Hrsg.), *Modellierungskompetenzen – Diagnose und Bewertung*, S. 153–187. Springer, Berlin, Heidelberg.
- Sullivan, T. J. (2015). *Introduction to Uncertainty Quantification*. Springer International Publishing.
- Sánchez, M. und Romero, M. (2006). Methodology for generation of heliostat field layout in central receiver systems based on yearly normalized energy surfaces. *Solar Energy*, 80(7):861–874.
- Tietze, U. P., Klika, M., und Wolpers, H. (1997). Historische Entwicklung, Beziehungsnetze und Fundamentale Ideen. In Tietze, U. P., Klika, M., und Wolpers, H. (Hrsg.), *Mathematikunterricht in der Sekundarstufe II: Fachdidaktische Grundfragen – Didaktik der Analysis*, S. 178–212. Vieweg+Teubner, Wiesbaden.
- Vogel, D. (2010a). Maximal, minimal, optimal ... *mathematik lehren*, 159:6–13.
- Vogel, D. (2010b). Optimieren – Leitidee für den Umgang mit Daten. *mathematik lehren*, 159:51–55.
- Von der Bank, M. (2012). Fundamentale Ideen – insbesondere Optimierung. In Filler, A. und Ludwig, M. (Hrsg.), *Wege zur Begriffsbildung für den Geometrieunterricht – Ziele und Visionen 2020. Vorträge auf der 29. Herbsttagung des Arbeitskreises Geometrie in der Gesellschaft für Didaktik der Mathematik vom 14. bis 16. September 2012 in Saarbrücken*. franzbecker.
- Vorhoelter, K. und Freiwald, J. (im Druck). Concept and structure of the Hamburg Modeling Days. *MSEL*.

- Vos, P. (2011). What is 'Authentic' in the Teaching and Learning of Mathematical Modeling? In Kaiser, G., Blum, W., Borromeo Ferri, R., und Stillman, G. (Hrsg.), *Trends in Teaching and Learning of Mathematical Modelling ICTMA 14*, S. 713–722, Dordrecht, Heidelberg, London, New York. Springer.
- Weicker, K. (2015). *Evolutionäre Algorithmen*. Springer, Wiesbaden, 3. Auflage.
- Wendlin, T. (2003). Soltrace: a new optical modeling tool for concentrating solar optics. In *ASME 2003 International Solar*, S. 253–260. ASME.
- Wiegand, B. und Blum, W. (1999). Offene Probleme für den Mathematikunterricht – Kann man Schulbücher dafür nutzen? In *Beiträge zum Mathematikunterricht 1999*, S. 590–593.
- Wild, C. und Pfannkuch, M. (1999). Statistical Thinking in Empirical Enquiry. *International Statistical Review*, 67(3):223–265.
- Winter, H. (1975). Allgemeine Lernziele für den Mathematikunterricht. *ZDM*, 7(3):106–116.
- Winter, H. (1985). Minimumseigenschaft vom Zentralwert und arithmetischem Mittel – Beweise mit Hilfe der SI-Mathematik. *mathematik lehren*, 8:7–15.
- Winter, H. (1995). Mathematikunterricht und Allgemeinbildung. *Mitteilungen der Gesellschaft für Didaktik der Mathematik*, 61:37–46.
- Wohak, K. (2021). *Didaktisch-methodische Entwicklung und Evaluation von computergestützten Lehr- und Lernmaterialien zur Mathematik inverser Probleme*. Dissertation, KIT.
- Wohak, K., Sube, M., Schönbrodt, S., Frank, M., und Roeckerath, C. (2021). Authentische und relevante Modellierung mit Schülerinnen und Schülern an nur einem Tag?! In Bracke, M., Ludwig, M., und Vorhölter, K. (Hrsg.), *Neue Materialien für einen realitätsbezogenen Mathematikunterricht 8. Realitätsbezüge im Mathematikunterricht.*, S. 37–50. Springer Spektrum, Wiesbaden.
- Zech, F. (1998). *Grundkurs Mathematikdidaktik*. Beltz Verlag, Weinheim, Basel, 9. Auflage.
- Zhou, Y., Wilkinson, D., Schreiber, R., und Pan, R. (2008). Large-Scale Parallel Collaborative Filtering for the Netflix Prize. In Fleischer, R. und Xu, J. (Hrsg.), *Algorithmic Aspects in Information and Management. AAIM 2008. Lecture Notes in Computer Science*, Band 5034, S. 337–348, Berlin, Heidelberg. Springer.

# A Ergänzungen zum Basiswissen rund um Solarkraftwerke

## A.1 Modellierung von Verschattungs- und Blockierungseffekten

Hier wird kurz die Grundidee der Modellierung a) der Verschattung durch den Sekundärreflektor und b) der Blockierung reflektierter Sonnenstrahlen durch nebenstehende Spiegel beschrieben. Dieses Kapitel schließt an die Modellierung des Schattenwurfs durch nebenstehende Spiegel in Abschnitt 10.3 an.

### Schattenwurf durch den Sekundärreflektor

Um die Verschattung durch den Sekundärreflektor zu modellieren, beschreiben wir diesen vereinfachend durch ein Rechteck mit Länge  $b$  und Breite  $\frac{b}{2}$ . Je nach Sonnenstand ( $\alpha \leq 90^\circ$  oder  $\alpha > 90^\circ$ ) werden Sonnenstrahlen durch die jeweils relevanten Ecken des Rechtecks betrachtet. Anschließend werden zahlreiche Schnittpunktsberechnungen mit den Strecken, die die Spiegel beschreiben, durchgeführt. Auf detaillierte Ausführungen der geometrischen Überlegungen und Schnittpunktsberechnungen wird hier verzichtet.

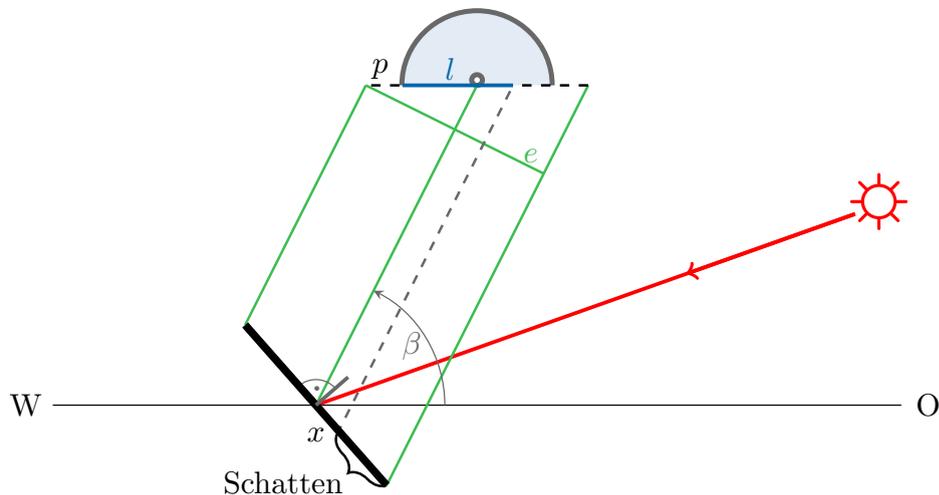
Für jeden Spiegel  $H_{i_0}$  definieren wir die Menge  $I_{\mu_{unshaded}}$ , welche die Werte des Parameters  $\mu$  enthält, die den Teil der Spiegelstrecke beschreiben, der tatsächlich bestrahlt wird. Dabei wurde sowohl die Verschattung durch den Sekundärreflektor als auch durch nebenstehende Spiegel (vgl. Abschn. 10.3) bereits berücksichtigt.

### Blockierung reflektierter Strahlen durch nebenstehende Spiegel

Bei der Modellierung der Blockierung reflektierter Strahlen werden Schnittpunkte zwischen den Spiegelstrecken und den Halbgeraden der reflektierten Strahlen, die durch die Spiegelecken der nebenstehenden Spiegel verlaufen, berechnet. Für den Spiegel  $H_{i_0}$  erhalten wir ein Intervall  $I_{\mu_{reflected}}$ , welches die Werte des Parameters  $\mu$  enthält, die den Teil der Spiegelstrecke beschreiben deren reflektierte Strahlen nicht blockiert werden.

Die Menge der Werte für den Parameter  $\mu$ , die den nicht blockierten **und** nicht verschatteten Bereich eines Spiegels beschreiben, ist gegeben durch die Schnittmenge

$$I_{\mu_{reflected}} \cap I_{\mu_{unshaded}}.$$



**Abb. A.1:** Skizze zur Berechnung der Projektion auf Rohrhöhe  $p$

Diese Schnittmenge ist entweder ein abgeschlossenes Intervall oder, wenn bedingt durch die Sekundärreflektorverschattung nur ein Teil in der Mitte des Spiegels verschattet wird, die disjunkte Vereinigung zweier abgeschlossener Intervalle. Abschließend ist zu bestimmen, welcher Anteil der effektiv reflektierten Strahlen tatsächlich auf den Sekundärreflektor trifft. Dazu werden die Schnittpunkte der reflektierten Strahlen durch die äußeren Punkte des effektiven Spiegelbereichs mit einer Strecke, die die Unterseite des Sekundärreflektors beschreibt, berechnet. Insgesamt müssen entweder zwei (wenn am Rand des Spiegels ein Teil verschattet ist) oder vier (wenn bedingt durch einen Schattenwurf des Sekundärreflektors ein Teil in der Mitte des Spiegels verschattet wird) reflektierte Strahlen betrachtet werden. Dies liefert letztlich die Länge der insgesamt bestrahlten Strecke  $l$  des Sekundärreflektors (vgl. Abb. A.1).

Der Anteil  $\tilde{a}$  des bestrahlten Sekundärreflektorbereichs an der ohne Verschattungs- und Blockierungseffekte bestrahlten Strecke  $p$  (vgl. Abb. A.1) ist gegeben durch

$$\tilde{a} = \min\left\{\frac{l}{p}, 1\right\},$$

wobei sich die Länge der Strecke  $p$  gemäß

$$p = \frac{e}{\cos(90^\circ - \beta)}$$

berechnen lässt. Dieser Anteil kann bei der Berechnung der Leistung in Gleichung (10.7) und damit bei der Beschreibung des Energieumsatzes für eine gegebene Konfiguration an Spiegelpositionen  $\mathbf{x}$  berücksichtigt werden.

# B Lernmaterial zum Thema Solarenergie – Workshop I

## B.1 Ablauf des Workshops

In Abbildung B.1 ist ein möglicher Ablauf des Workshops für die Mittelstufe dargestellt. Die Nummerierung in den Konzepten A und B gibt die Reihenfolge an, in denen die Arbeitsblätter bearbeitet werden können.

Es ist nicht notwendig alle Arbeitsblätter zu bearbeiten, um den Workshop inhaltlich kohärent abzuschließen. Besteht wenig(er) Zeit für die Durchführung, so kann der Workshop bereits nach früheren Arbeitsblättern beendet werden, ohne den Modellierungsprozess für die Lernenden unvollständig erscheinen zu lassen. Geeignete Stellen für den Abschluss des Workshops werden hier kurz aufgeschlüsselt.

### Kurzversionen des Workshops für die Mittelstufe

Folgende Kurzversionen des Workshops für die Mittelstufe sind denkbar:

- **Kurzversion 1:** Der Workshop kann bereits nach dem vierten Arbeitsblatt beendet werden. Die Leitfrage des Workshops ist dann die Modellierung der Leistung, die durch einen Spiegel erzeugt wird (AB 1 – 4).
- **Kurzversion 2:** Der Workshop kann nach dem fünften Arbeitsblatt beendet werden. Die Leitfrage ist in diesem Fall die Modellierung der Leistung mitsamt einer ersten Diskussion der optimalen Absorberrohrhöhe (AB 1 – 5)
- **Kurzversion 3:** Der Workshops kann abschlossen werden, nachdem die Lernenden eine beliebige Modellverbesserung umgesetzt haben (AB 1 – 5 und AB 6 oder AB 7).

## B.2 Arbeitsblätter zum Modell für Leistung und Energie

Die Arbeitsblätter (Jupyter Notebooks) sind mit eingetragenen Lösungen dargestellt. Die Kommentare `#= NaN`; `=#` vor den Lösungen deuten an, dass in der Schülerversion an dieser Stelle der Platzhalter `NaN` steht. Dieser wäre dann durch eine Formel, einen Wert oder ähnliches zu ersetzen. Die auf den Arbeitsblättern verlinkten Tippkarten sind nicht im Anhang enthalten. Diese können jedoch wie im [Vorwort](#) beschrieben über die Workshop-Plattform eingesehen werden.

Mittelstufe: Modellierung der Leistung und Optimierung der Rohrhöhe

	Arbeitsblatt	Inhalt	A <sup>1</sup>	B	Schulbezug	Optimierung	
Baustein 1: Modellierung der Leistung am Rohr	AB 1	Spiegelausrichtung	1	1	Winkelbeziehungen		
	AB 2	Leistung auf Spiegel	2	2	Trigonometrie		
	AB 3	Leistung am Rohr	3	3	Relativer Anteil		
	<b>Diskussion im Plenum</b>						
	AB 4	Modellerweiterung: Verschiebung des Spiegels	4	4	Trigonometrie		
	AB 4 lang						
<b>Diskussion im Plenum</b>							
Baustein 2: Optimierung der Rohrhöhe	AB 5	Optimierung der Rohrhöhe mit Modell von AB 4	5	7	Diskussion Funktionsgraph	Grafisch, Existenz und Eindeutigkeit	
	<b>Diskussion im Plenum</b>						
	AB 6	Modellverbesserung 1: atmosphärische Effekte	6 oder 7	8 oder 9	Pythagoras	Modellierung der Zielfunktion, Existenz & Eindeutigkeit	
	AB 7	Modellverbesserung 2: Fehlerberücksichtigung	6 oder 7	8 oder 9	Trigonometrie, Winkelbeziehungen, relativer Anteil, Interpretation des Funktionsgraphen		
	AB 8	Kombination von Modellverbesserung 1 & 2	8	10			
<b>Diskussion im Plenum</b>							
Zusatzbaustein	Zusatzblatt 1	Mehrere Spiegel	nach AB 4	5			
	Zusatzblatt 2	Energie über einen Tag (s. AB 2 im Workshop für die Oberstufe)	nach AB 4	6	Riemann-Summen		
	Zusatzblatt 3	Modellverbesserung 3 & 4 (Mehrere Spiegel + Energie über Zeitraum)	nach AB 8	11 oder 12	Interpretation des Funktionsgraphen	Modellierung der Zielfunktion, Existenz & Eindeutigkeit	
	Zusatzblatt 4	Modellverbesserung 5: Monte Carlo Simulation	nach AB 8	11 oder 12	Erwartungswert	Optimierung unter Unsicherheiten	

<sup>1</sup> Die Nummerierung in den Konzepten A und B gibt die Reihenfolge an, in denen die ABs bearbeitet werden können.

Abb. B.1: Übersicht über Inhalte und Arbeitsblätter des Workshops für die Mittelstufe

## B.2.1 Arbeitsblatt 1: Ausrichtung eines Spiegels

### Solkraftwerk - AB 1: Die Ausrichtung eines Spiegels unter dem Absorberrohr

Sekundärreflektor  
Absorberrohr  
Reihen flacher Spiegel

**Einleitung**

Das **Fresnelkraftwerk** ist ein bestimmter Typ von Solarkraftwerken. In diesen Kraftwerken reflektieren **flache** Spiegel Sonnenstrahlen auf ein wasserführendes Rohr. Dieses wird auch Absorberrohr genannt. Das Rohr ist von einem gekrümmten Spiegel umgeben. Dieser soll die Strahlen, die knapp neben das Rohr treffen auf das Rohr reflektieren. Dieser Spiegel wird Sekundärreflektor genannt. Die Wärmestrahlung erhitzt das Wasser im Rohr so stark, dass es verdampft. Der Dampf treibt eine Turbine an, die mit einem Generator verbunden ist. Dieser wandelt die Bewegungsenergie in elektrische Energie um.

Wir werden ein mathematisches Modell für ein Fresnelkraftwerk entwickeln. Dazu verwenden wir Daten zum Stand der Sonne und der Stärke der Sonnenstrahlung an dem sonnigen Ort **Daggett in Kalifornien (USA)**. Ziel ist es, anschließend verschiedene Parameter des Kraftwerks zu optimieren!

**Die Sonnenstrahlen optimal auf das Rohr spiegeln**

Wie in der mathematischen Modellierung üblich, starten wir zunächst mit einer (stark) vereinfachten Situation. Wir betrachten also zunächst ein Fresnelkraftwerk mit nur **einem einzigen Spiegel** direkt **unter dem Rohr** Diese Situation ist in der folgenden Abbildung dargestellt. Dabei steht

- $\gamma$  (gamma) für den **Neigungswinkel des Spiegels**. Das ist der Winkel zwischen der Senkrechten auf den Spiegel (im Bild grau dargestellt) und dem Boden.
- $\alpha$  (alpha) für den **Sonneneinfallswinkel**. Das ist der Winkel zwischen dem einfallenden Sonnenstrahl und dem Boden.

Da sich der Sonnenstand im Laufe des Tages ändert, muss auch der Spiegel im Tagesverlauf gedreht werden. Damit der reflektierte Sonnenstrahl genau auf das Absorberrohr trifft, muss der Neigungswinkel des Spiegels an den Sonneneinfallswinkel angepasst werden.

W

**Teil a**

Für den 21. Juni in Daggett (Kalifornien) liegt der Sonneneinfallswinkel  $\alpha$  abhängig von der aktuellen Tageszeit  $t \in [0, 24)$  vor. Diesen werden wir im Workshop als Datengrundlage nutzen. Durch Aufrufen der Funktion `alpha(t)` kannst du dir den Einfallswinkel zu einer beliebigen Tageszeit  $t$  ausgeben lassen.

**!**

- Hierbei wird  $\alpha$  im **Gradmaß**, das heißt mit der Einheit **Grad**, ausgegeben.
- Die Zeit  $t$  wird in **Stunden nach Tagesbeginn** angegeben. Beispielsweise entspricht 11:30 Uhr der Zeit  $t = 11.5$ .

Als erstes verschaffen wir uns einen Eindruck von dem Verlauf der einfallenden und reflektierten Sonnenstrahlung je nach gewählter Neigung des Spiegels.

Ersetze das erste `NaN` durch eine Tageszeit zwischen 0 h und 24 h. Mithilfe der Funktion `alpha(t)` wird dir der Einfallswinkel der Sonnenstrahlen zur gewählten Zeit ausgegeben. Ersetze das zweite `NaN` durch einen Wert für den Neigungswinkel  $\gamma$ . Führe den Code dann aus. Klicke dazu in das Code-Feld und drücke auf den **▶** - Button oder "Shift" + "Enter".

### Teil c

Die Spiegelausrichtung soll automatisch erfolgen. Dazu beschreiben wir den korrekten Neigungswinkel des Spiegels abhängig vom Sonneneinfallswinkel. In der Formel für den Neigungswinkel  $\gamma$  können wir die Funktion `alpha(t)`, mit der der Sonneneinfallswinkel zu einer gewählten Uhrzeit  $t$  bestimmt wird, verwenden.



Stelle eine Formel für den gesuchten Neigungswinkel  $\gamma$  in Abhängigkeit von  $\alpha$  (Eingabe als `alpha(t)`) auf. Nimm deine angefertigte Zeichnung zur Hilfe. Gib deine Formel für das erste `NaN` ein. Wähle dann eine beliebige Tageszeit  $t$  zwischen 0 und 24 und gib sie für das zweite `NaN` ein. Führe den Code aus, indem du auf "Run" klickst. Dir wird für die gewählte Zeit  $t$  und mit deiner Formel für  $\gamma$  der Strahlenverlauf angezeigt. Wenn du alles richtig gemacht hast, fällt der reflektierte Sonnenstrahl genau auf das Absorberrohr - und zwar unabhängig von deiner Wahl für  $t$ .

Falls du Hilfe benötigst, kannst du dir hier einen **Tipp ansehen**.

```
In [ ]:
# Berechnung von Gamma
gamma(t) = NaN; # (alpha(t) + 90) / 2 # Ersetze NaN durch eine Formel, die
# Tageszeit
t = NaN; # Ersetze NaN durch eine beliebige Zeit zwischen 0 und 24
# Hier nichts ändern
checkReflection(t, alpha, gamma)
```

### Fazit

Der Spiegel wird nun automatisch so ausgerichtet, dass der Sonnenstrahl, der von der Mitte des Spiegels reflektiert wird, jederzeit direkt auf das Rohr fällt. Auf dem nächsten Arbeitsblatt werden wir die Leistung berechnen, die durch das Kraftwerk erzeugt wird. Öffne das Arbeitsblatt 2.

Der Verlauf der Strahlen wird zum gewählten Zeitpunkt und basierend auf der von dir gewählten Neigung des Spiegels angezeigt.

**Stelle den Neigungswinkel  $\gamma$  so ein, dass der reflektierte Strahl auf das Rohr trifft.**

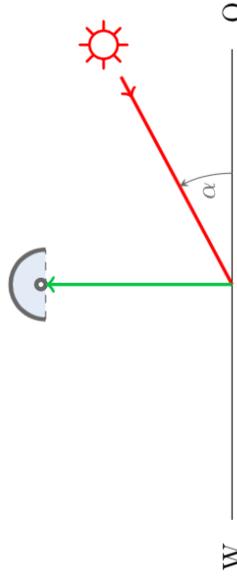
```
In [ ]:
# Hier nichts ändern!
include("../code/setup_AB1.jl");
# Tageszeit
t = NaN; # Ersetze NaN durch eine beliebige Zeit in Stunden nach 7a
# Neigungswinkel
gamma = NaN; # 30; # Ersetze NaN durch einen beliebigen Winkel im Gradmaß
alpha = alpha(t); plotSimpleFresnel(t, gamma, alpha)
```

Vermutlich hast du den Neigungswinkel  $\gamma$  in der Grafik durch Ausprobieren so wählen können, dass der reflektierte Strahl das Rohr trifft. Wählen wir in Teil a jedoch eine andere Tageszeit und damit einen anderen Einfallswinkel  $\alpha$ , so muss auch der Neigungswinkel  $\gamma$  wieder angepasst werden, damit das Licht wie gewünscht reflektiert wird. Ziel ist es, eine Formel zur Berechnung des korrekten Neigungswinkels  $\gamma$  aufzustellen, die vom aktuellen Einfallswinkel der Sonnenstrahlen  $\alpha$  abhängt.

Die Skizze in Teil b soll dir beim Aufstellen der Formel für  $\gamma$  helfen!

### Teil b

Die untere Abbildung zeigt, wie der reflektierte Strahl zum einfallenden Strahl verlaufen muss, damit er genau auf das Absorberrohr trifft. In dieser Abbildung fehlt jedoch der Spiegel.



Übertrage die letzte Zeichnung auf ein Blatt. Ergänze deine Zeichnung um den Spiegel und die Senkrechte auf den Spiegel, sodass eine physikalisch korrekte Reflexion entsteht. Zeichne auch den Neigungswinkel  $\gamma$  ein.

## B.2.2 Arbeitsblatt 2: Leistung am Spiegel

### Solkraftwerk - Aufgabenblatt 2: Die Leistung auf einem Spiegel

#### Einleitung

Zu den wichtigsten Kennzahlen eines Kraftwerkes gehört die Leistung  $P$  (engl. *power*). Sie bezeichnet die in einer bestimmten Zeitspanne umgesetzte Energie  $\Delta E$  bezogen auf diese Zeitspanne ( $P = \frac{\Delta E}{\Delta t}$ ). Die Einheit der Leistung ist **Watt** (kurz: **W**). Mithilfe der Leistung kann ermittelt werden, wie viel Energie durch das Kraftwerk umgesetzt wird.

Ziel der Arbeitsblätter 2 und 3 ist es, schrittweise ein Modell zu entwickeln, mit dem die Leistung am Absorberrohr berechnet werden kann.

- Schritt:** Berechnung der Leistung, die den Spiegel erreicht (Arbeitsblatt 2)
- Schritt:** Berechnung der Leistung, die schließlich das Absorberrohr erreicht (Arbeitsblatt 3)

#### Die Stärke der Sonnenstrahlung

Als Ausgangspunkt für die Berechnung der Leistung am Spiegel werden wir die sogenannte **Bestrahlungsstärke** (engl. *irradiance*) der Sonnenstrahlung  $I_{\text{sun}}$  verwenden. Die **Bestrahlungsstärke**  $I_{\text{sun}}$  ist über den Tag gesehen nicht konstant, sondern hängt genauso wie der **Sonneneinfallswinkel**  $\alpha$  von der Tageszeit ab. Für den Standort unseres Kraftwerks, Daggett, liegt die **Bestrahlungsstärke** der Sonnenstrahlung  $I_{\text{sun}}$  an einem Tag im Juni vor. Für eine beliebige Tageszeit  $t$  können wir uns die **Bestrahlungsstärke** ausgeben lassen, indem wir die Funktion `Isun(t)` aufrufen.

Die **Bestrahlungsstärke** wird für gewöhnlich in **Watt pro Quadratmeter** ( $= \frac{\text{W}}{\text{m}^2}$ ) angegeben. Da wir die Problemstellung jedoch auf den Querschnitt eines Kraftwerks vereinfacht haben (und damit nur die **Breite** und nicht die **Fläche** eines Spiegels betrachten), arbeiten wir mit der Einheit **Watt pro Meter** ( $= \frac{\text{W}}{\text{m}}$ ).

#### Teil a

Zunächst verschaffen wir uns einen Eindruck von der Größenordnung, in der die Stärke der Sonnenstrahlung liegt.



Ersetze im folgenden Code `NaN` durch eine Uhrzeit `t` zwischen 0 und 24. Drücke anschließend auf "Run". Die Leistung der Sonnenstrahlen zu der gewählten Uhrzeit wird dir ausgegeben. Vergleiche die Stärke der Strahlung am Mittag mit der am Morgen kurz nach dem Sonnenaufgang.

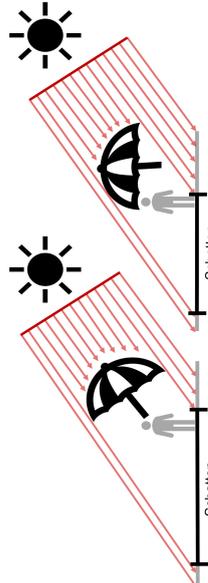
**Erinnerung:** Die Zeit  $t$  muss wieder in **Stunden nach Tagesbeginn** angegeben werden (z. B. entspricht 15:45 Uhr somit  $t = 15.75$ ).

```
In [ ]:
# Hier nichts ändern
include("../code/setup_AB2_MS_j11"); alpha(t) = compute_alpha(beta(t)); Isun(t)
# Tageszeit
t = NaN; # Ersetze NaN durch eine beliebige Uhrzeit t. Du kannst
# Leistung zum gewählten Zeitpunkt t
Isun(t);
# Hier nichts ändern
checkIsun(t) # Ausgabe von Isun zu festgelegter Tageszeit
```

#### Der Kosinus-Effekt

Die **Bestrahlungsstärke der Sonnenstrahlung**  $I_{\text{sun}}$  beschreibt die Leistung, die übertragen wird, wenn die Sonnenstrahlen **senkrecht** auf eine Fläche von einem Quadratmeter treffen. Treffen die Strahlen **nicht** senkrecht auf, so ist die Leistung geringer. Es gilt: Je spitzer bzw. stumpfer der Einfallswinkel, desto geringer die Leistung. Dieser Effekt wird als **Kosinus-Effekt** bezeichnet.

Ein einfaches Gedankenexperiment soll diesen Effekt verdeutlichen: Wird ein Sonnenschirm so gegen die Sonne gehalten, dass dieser senkrecht zu den Sonnenstrahlen steht, wie links in Abbildung dargestellt, so können mehr Strahlen abgefangen werden, als wenn die Sonnenstrahlen schräg auf den Sonnenschirm auftreffen. Dies wird auch durch den Schatten ersichtlich, der im linken Fall (Schirm steht senkrecht zu den Strahlen) länger ist als im rechten Fall.



Ähnlich verhält es sich auch bei der Sonnenstrahlung. Das Strahlenpaket, das den Spiegel erreicht, hat nur die Breite wie die Länge der Strecke, die orthogonal zu den Strahlen steht. In folgender Abbildung ist die Länge dieser senkrechten (Hilfs-)Strecke mit  $c$  bezeichnet. Diese wird durch die zwei äußeren Sonnenstrahlen, die auf den Spiegel treffen, begrenzt. Sie nimmt die gleiche Leistung durch die Sonnenstrahlung auf wie unser Spiegel, steht jedoch senkrecht auf den Sonnenstrahlen – wie es für die Verwendung von  $I_{\text{sun}}$  vorausgesetzt wird. Wir bezeichnen die **Breite des Spiegels** mit  $s$ . Diese setzen wir als fest (mit  $s = 1.1\text{m}$ ) und bekommen voraus. Sie ist ebenfalls in folgendem Bild eingezeichnet.

Sowohl der Sonneneinfallswinkel  $\alpha$ , als auch die Stärke der Sonnenstrahlung  $I_{\text{sun}}$  ändern sich im Laufe des Tages. Die **Leistung am Spiegel**  $P_{\text{mirror}}$  ist somit ebenfalls von der

Stelle eine Formel für die Leistung am Spiegel  $P_{\text{mirror}}$  auf. Gib deine Formel für `NaN` ein. Falls du alles richtig gemacht hast, erscheint eine Abbildung, in der der Verlauf der Leistung am Spiegel über den Tag dargestellt ist. Wie lässt sich der Verlauf der Leistung begründen?

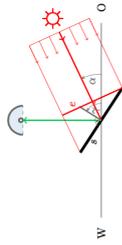
**Hinweis:** Bei deiner Eingabe kannst du die Funktionen `Isun(t)`, `alpha(t)`, `e(t, s)` oder `gamma(t)` nutzen.

```
In [ ]:
# Funktion zur Berechnung der Leistung am Spiegel Pmirror
Pmirror(t,s) = NaN; # Isun(t) * s * cosd(gamma(t) - alpha(t)); # Ersetze
# Hier nichts ändern - Checken
checkPmirror(Pmirror,s);
```

### Fazit

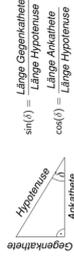
Wir können nun die Leistung am Spiegel zu beliebigen Tageszeiten  $t$  berechnen. Zuletzt müssen wir bestimmen, wie groß die Leistung ist, die dann tatsächlich das Absorberrohr erreicht. Öffne dazu das Arbeitsblatt 3.

Tageszeit abhängig. Um die Leistung  $P_{\text{mirror}}$  berechnen zu können, brauchen wir zunächst die Länge der orthogonalen Hilfsstrecke  $e$ .



### Teil b

Stelle eine Formel für die Breite der Strecke  $e$  auf. Verwende dabei die Spiegelbreite  $s$  und die von Aufgabenblatt 1 bekannten Winkel  $\alpha$  und  $\gamma$ . Nutze die Definitionen von Sinus und Kosinus im rechtwinkligen Dreieck:



$$\sin(\delta) = \frac{\text{Länge Gegenkathete}}{\text{Länge Hypotenuse}}$$

$$\cos(\delta) = \frac{\text{Länge Ankathete}}{\text{Länge Hypotenuse}}$$

**Hinweis:** Sinus und Kosinus werden im Code mit `sind()` und `cosd()` eingegeben. Das `d` steht dabei für die verwendete Einheit Gradmaß (engl. degree).



Ersetze im folgenden Code das `NaN` durch deine Formel, die von  $s$ ,  $\alpha$  und  $\gamma$  abhängt und mit der die Länge der Strecke  $e$  berechnet werden kann.

**Falls du Hilfe benötigst, kannst du dir diesen Tipp ansehen.**

```
In [ ]:
# Breite des Spiegels
s = 1.1;
# Funktion zur Berechnung der Strecke e
e(t,s) = NaN; # s * cosd(gamma(t) - alpha(t)); # Ersetze NaN durch eine
# Hier nichts ändern - Checken
check_eNormal(e,s)
```

### Teil c

Wir können nun ein Modell für die Leistung entwickeln, die den Spiegel (engl. `mirror`) mit einer Breite von  $s = 1.1\text{m}$  erreicht. Wir bestimmen nun eine Funktion, die in Abhängigkeit der Leistung der Sonne  $I_{\text{sun}}$  und des Sonneneinfallswinkels  $\alpha$  die Leistung  $P_{\text{mirror}}$  berechnet, die auf einem Spiegel mit Breite  $s = 1.1\text{m}$  ankommt.



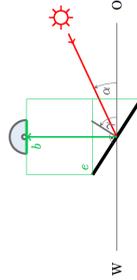
### B.2.3 Arbeitsblatt 3: Leistung am Rohr

#### Solkraftwerk - Aufgabenblatt 3: Die Leistung am Absorberrohr

##### Einleitung

In der folgenden Abbildung erkennt man, dass über dem Absorberrohr ein gekrümmter **Sekundärreflektor** (das ist auch ein Spiegel) angebracht ist. Dieser dient dazu, die Strahlen, die von unten knapp am Rohr vorbei auf ihn fallen, doch noch auf das Rohr zu reflektieren.

Auf dem letzten Blatt haben wir die Leistung am Spiegel berechnet. Diese stimmt im Allgemeinen nicht mit der Leistung am Absorberrohr überein. Ein Grund dafür ist, dass das reflektierte Strahlenpaket breiter als der Sekundärreflektor sein kann, wie in der Abbildung unten dargestellt. Auf diesem Arbeitsblatt werden wir den **Anteil der Leistung am Spiegel**  $P_{\text{mirror}}$  bestimmen, **der am Absorberrohr ankommt** - so erhalten wir die tatsächliche Leistung am Rohr.



Wir nehmen zunächst an, dass die Leistung am Rohr (engl. tube), die wir im Folgenden mit  $P_{\text{tube}}$  bezeichnen, mit der Leistung am Sekundärreflektor übereinstimmt:

$$P_{\text{tube}} \hat{=} \text{Leistung am Absorberrohr} \hat{=} \text{Leistung am Sekundärreflektor.}$$

##### Die Leistung am Absorberrohr

Wir nehmen weiterhin an, dass die Breite des Sekundärreflektors  $b$  (oben in grün eingezeichnet) gegeben ist. Vom zweiten Arbeitsblatt kennen wir bereits den Zusammenhang zwischen der Leistung am Spiegel  $P_{\text{mirror}}$  und der Stärke der Sonnenstrahlung  $I_{\text{sun}}$ :

$$P_{\text{mirror}} = I_{\text{sun}} \cdot e$$

Aufgrund des Reflexionsgesetzes (Einfallswinkel = Ausfallswinkel) entspricht die Breite des Strahlenpakets, das vom Spiegel nach oben reflektiert wird, ebenfalls der Länge der Strecke  $e$ .

Zusammenfassend werden auf diesem Arbeitsblatt folgende Vereinfachungen bzw. Annahmen getroffen:

1. **Alle** Strahlen, die den Sekundärreflektor treffen, werden auf das Absorberrohr reflektiert.

2. Die Breite des Sekundärreflektors  $b$  ist gegeben.
3. Der Strahl, der von der Mitte des Spiegels reflektiert wird, trifft genau auf die Mitte des Absorberrohrs.

##### Teil a

Wir betrachten zunächst zwei Fälle:

1. **Fall:** Das reflektierte Strahlenbündel ist breiter als der Sekundärreflektor, d. h. es gilt  $e > b$ . Nur ein Teil der reflektierten Strahlung trifft den Sekundärreflektor. Folglich kommt nur ein Teil der Leistung, die auf dem Spiegel anliegt, am Rohr an.
2. **Fall:** Das reflektierte Strahlenbündel ist maximal so breit wie der Sekundärreflektor, d. h. es gilt  $e \leq b$ .



Stelle für beide Fälle eine Formel auf, mit der der **relative Anteil der Leistung am Spiegel** berechnet werden kann, der tatsächlich am Rohr ankommt. Ersetze das erste `NaN` durch die Formel für den ersten Fall und das zweite `NaN` durch dir Formel für den zweiten Fall. Du kannst die  $b$  und  $e$  in deinen Formeln verwenden. Drücke anschließend auf "Run".

Falls du Hilfe benötigst, kannst du dir diesen **Tip** ansehen.

```
In [ ]:
# Hier nichts ändern
include("../code/setup_AB3_MS.jl");

# Funktion zur Berechnung des relativen Anteils
ratioCase1(b,e) = NaN; # Ersetze NaN durch eine Formel, die von e
ratioCase2(b,e) = NaN; # Ersetze NaN durch eine Formel mit der der n

# Prüfung der Funktion (ab hier nichts mehr ändern)
checkRatioCases(ratioCase1, ratioCase2)
```

Im folgenden Code-Feld werden die beiden Fälle zur Berechnung des relativen Anteils in der Funktion `ratio(b, e)` kombiniert.

**Hinweis:** Um die kleinere von zwei Zahlen  $x$  und  $y$  zu finden, kann man in Julia die Funktion `min(x,y)` benutzen. Zum Beispiel: `min(5,9) = 5` und `(8,-3) = -3`.



Führe den folgenden Code aus. Die Funktion `ratio(b, e)` wird eingespeichert. Diese kann in den folgenden Aufgaben verwendet werden.

```
In [ ]:
# Hier nichts ändern! - Funktion zur Berechnung des relativen Anteils in beiden
ratio(b,e) = min(ratioCase1(b,e),ratioCase2(b,e));
checkRatio(ratio)
```

##### Teil b

- es auch für horizontal verschobene Spiegel gilt
- nicht nur die durch **einen Spiegel** erzeugte Leistung berechnet werden kann, sondern auch die Leistung, die durch **mehrere Spiegel** erzeugt wird
- nicht nur die Leistung zu (einer bestimmten Uhrzeit), sondern die gesamte erzeugte **Energie über einen längeren Zeitraum** (z.B. ein Jahr oder einen Tag) berechnet wird

Öffne jetzt das nächste Arbeitsblatt.



Stelle eine Formel für die Leistung  $P_{\text{tube}}$  auf, die am Absorberrohr generiert wird. Ersetze NaN durch deine Formel. Bei deiner Eingabe kannst die Funktionen `PhiError(t,s)` und `e(t,s)` von Arbeitsblatt 2 verwenden. Wenn du alles richtig gemacht hast, erscheint der Graph der Funktion  $P_{\text{tube}}$  zu den festgelegten Werten von  $b$  und  $s$ . Diese Funktion beschreibst die Leistung am Absorberrohr in Abhängigkeit der Zeit.

**Hinweis:** Wenn du den relativen Anteil verwendst, so musst du im Code `ratio(b,e(t,s))` schreiben!

```
In [ ]: # --- Festlegung einiger Parameter des Kraftwerks --- #
b = 0.8; # Breite des Sekundärreflektors
s = 1.1; # Breite des Spiegels
# Funktion für die Leistung am Rohr
Prube(t,b,s) = NaN; # e(t,s) * ratio(b,e(t,s)); # Ersetze NaN
# Hier nichts ändern!
checktube(Prube,b,s)
```

### Teil C



Zu welcher Uhrzeit wird dem Kraftwerk die größte Leistung zugeführt? Lies die Antwort am Graphen ab und gib dein Ergebnis für NaN ein, um eine Rückmeldung zu erhalten. Wie kannst du das Ergebnis erklären?

```
In [ ]: timeMax = NaN; # Ersetze NaN durch die Zeit (in Stunden nach Tag)
# Hier nichts ändern!
checktimeMax(timeMax);
```

### Fazit:

Wir haben eine Funktion entwickelt, die die Leistung am Rohr abhängig von Tageszeit  $t$ , Breite des Sekundärreflektors  $b$  und Spiegelbreite  $s$  ausgibt. Jedoch ist dieses Modell nur für einen Spiegel gültig, der **direkt unter dem Rohr** positioniert ist.

### Modellverbesserungen:

Auf den folgenden Arbeitsblättern soll das Modell erweitert und Schritt für Schritt der Realität angenähert werden, sodass

## B.2.4 Arbeitsblatt 4: Horizontal verschobene Spiegel

### Solkraftwerk - Arbeitsblatt 4: Modellerweiterung auf horizontal verschobene Spiegel

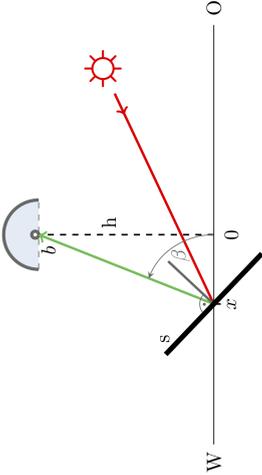
#### Einleitung

Die Annahme, dass sich alle Spiegel direkt senkrecht unter dem Absorberrohr befinden, entspricht nicht der Realität. In Wirklichkeit befinden sich viele Spiegel schräg links und rechts unter dem Absorberrohr, wie die folgende Abbildung zeigt.



Diese Anordnung ermöglicht es, mehr als nur eine Spiegelfläche als Reflexionsfläche zu nutzen und so die umgesetzte Energie zu erhöhen. Um unser Modell besser an die Realität anzupassen, werden wir es auf diesem Arbeitsblatt auf horizontal verschobene Spiegel verallgemeinern. Das heißt, wir gehen auf diesem Arbeitsblatt nicht mehr davon aus, dass sich der Spiegel zwingend senkrecht unter dem Rohr befindet.

#### Teil a - Der Reflexionswinkel $\beta$



Obige Abbildung stellt schematisch die Situation mit einem verschobenen Spiegel dar. Die neu eingeführte Variable  $x$  gibt die **Position des Spieglmittlepunktes** relativ zum Nullpunkt an. Der Nullpunkt befindet sich senkrecht unter dem Rohr. Dementsprechend ergibt eine Verschiebung des Spiegels nach links negative Werte für  $x$  und eine Verschiebung des Spiegels nach rechts positive Werte für  $x$ .

Mithilfe der obigen Abbildung sollst du nun eine Formel für den gewünschten Reflexionswinkel  $\beta$  zwischen Boden und reflektiertem Sonnenstrahl herleiten. Auf den

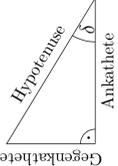
vorherigen Arbeitsblättern gilt stets  $\beta = 90^\circ$ . In der neuen Situation ist  $\beta$  von der Position  $x$  des Spiegels abhängig. Zur Berechnung von  $\beta$  benötigst du zudem die **Höhe des Absorberrohrs**  $h$  über dem Boden (blau im Bild). Wir gehen zunächst davon aus, dass diese Höhe  $h$  fest vorgegeben und bekannt ist.

Bei der Herleitung einer Formel für  $\beta$  werden wir drei Fälle unterscheiden:

- $x < 0$  (dann gilt  $\beta < 90^\circ$ )
- $x = 0$  (dann gilt  $\beta = 90^\circ$ )
- $x > 0$  (dann gilt  $\beta > 90^\circ$ ) (Mache dir dies ggf. an einer eigenen Zeichnung klar)

Der unten stehende Code ist mit dem Gerüst einer *if-else-Anweisung* bereits soweit vorbereitet, dass du die Formeln für die verschiedenen Fälle nur noch an den richtigen Stellen eintragen musst. Dadurch wird die Funktion `beta(x, h)` definiert, die  $\beta$  für gegebenes  $x$  und gegebenes  $h$  berechnet.

**Hinweis:** Für die Berechnung von  $\beta$  wird der Tangens bzw. der Arkustangens benötigt. Die Arkustangensfunktion (in der Schule oft  $\tan^{-1}$  bezeichnet) ist die Umkehrfunktion der (eingeschränkten) Tangensfunktion.



$$\tan(\delta) = \frac{\text{Länge Gegenkathete}}{\text{Länge Ankathete}}$$

$$\delta = \arctan\left(\frac{\text{Länge Gegenkathete}}{\text{Länge Ankathete}}\right)$$

**Hinweise:**

- Die Formeln zur Berechnung von  $\beta$  in allen drei Fällen auf. Du kannst dazu den Arkustangens verwenden.
- Ersetze das erste `NaN` durch eine Formel die von  $x$  und  $h$  abhängt und mit der der Winkel `beta` im Fall  $x < 0$  berechnet werden kann. Ersetze das zweite `NaN` durch einen Ausdruck für `beta` im Fall  $x = 0$  und das dritte `NaN` durch eine Formel für `beta` im Fall  $x > 0$ .

```

In [ ]:
# Hier nichts ändern
include("../code/setup_ABA_MS.jl"); alpha(t) = compute_alpha_beta(t); isun(t)
# Funktion zur Berechnung von beta aufstellen
funktion beta(h, x)
    if (x < 0)
        computeBeta = NaN; # Ersetze NaN durch einen Fo
    elseif (x == 0)
        computeBeta = NaN; # Ersetze NaN durch einen Ausdruck für d
    end
end
    
```

```
# Höhe des Rohrs
h = NaN; # Ersetze NaN durch eine positive Höhe
# Hier nichts ändern!
plotMirrorsB4(x, h, t, beta(h, x), gamma(t, h, x), 1, 1, 1.5)
```

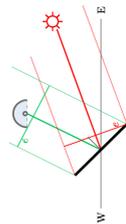
### Teil c - Der Kosinus-Effekt am Rohr

Die Leistung am Spiegel  $P_{\text{minor}}$  kann analog zu Arbeitsblatt 2 berechnet werden. Die Formeln ändern sich nicht. Das heißt, es gilt weiter

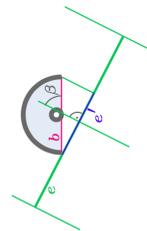
$$P_{\text{minor}} = I_{\text{sun}} \cdot e = I_{\text{sun}} \cdot s \cdot \cos(\gamma - \alpha).$$

mit

- $I_{\text{sun}}$ : Bestrahlungsstärke der Sonnenstrahlung, die **senkrecht** auf eine Fläche von einem Quadratmeter trifft,
- $e$ : Breite der Hilfsfläche, die senkrecht auf dem Sonnenstrahl steht,
- $s$ : Breite des Spiegels.

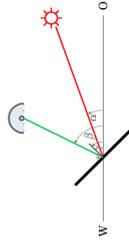


Die Formel für den relativen Anteil (Variable:  $\text{ratio}$ ) der Leistung am Spiegel, die tatsächlich das Rohr erreicht, ist hingegen nur gültig, wenn die reflektierten Strahlen senkrecht auf den Sekundärreflektor treffen. Aufgrund der Position des Spiegels direkt unter dem Rohr, was dies auf Arbeitsblatt 3 der Fall. Bei einem verschobenen Spiegel, treffen die Strahlen jedoch schräg auf das Rohr (siehe Abbildung oben). Wir müssen erneut den **Kosinus-Effekt** berücksichtigen. Dazu konstruieren wir eine Hilfsstrecke, die senkrecht auf den **reflektierten Strahlen** steht und an der die gleiche Leistung umgesetzt wird, wie am Absorberrohr. Diese neue Hilfsstrecke wird mit  $e'$  (=  $eP_{\text{r.ime}}$ ) bezeichnet (siehe Abbildung unten).



```
elseif (x > 0)
    computeBeta = NaN; # Ersetze NaN durch eine F
end
return computeBeta;
# Hier nichts ändern!
checkBeta(beta);
```

### Teil b - Der Neigungswinkel $\gamma$



Wie auf dem ersten Arbeitsblatt, muss die Neigung des Spiegels berechnet werden, die für eine Reflexion der Sonnenstrahlen auf das Absorberrohr notwendig ist. Der Neigungswinkel  $\gamma$  hängt nun nicht mehr allein vom Sonneneinfallswinkel  $\alpha$ , sondern zusätzlich vom erforderlichen Ausfallswinkel  $\beta$  ab.



Stelle mithilfe von der obigen Abbildung eine **Formel zu Berechnung von  $\gamma$**  in Abhängigkeit von  $\alpha$  und  $\beta$  auf. **Gib deine Formel für NaN ein.** Bei der Eingabe kannst du  $\text{alpha}(t)$  und  $\text{beta}(h, x)$  verwenden. Führe den Code aus.

```
In [ ] :
# Funktion zur Berechnung von gamma
gamma(t, h, x) = NaN; # (alpha(t) + beta(h, x)) / 2; # Ersetze NaN durch ei.
# Hier nichts ändern!
checkGamma(gamma, beta);
```

Nun werden wir die definierten Funktionen auf konkrete Werte anwenden und uns das Ergebnis graphisch anzeigen lassen.



Lege im folgenden Code beliebige Werte für die Uhrzeit  $t$ , die Spiegelposition  $x$  und die Rohrhöhe  $h$  fest. Führe den Code dann aus. Für die festgelegten Werte wird die Reflexion der einfallenden Strahlen basierend auf deiner Formel für  $\gamma$  grafisch dargestellt.

**Hinweis:** Teste gerne unterschiedliche Kombinationen von  $x$ ,  $h$  und  $t$ .

```
In [ ] :
# Festlegung von Uhrzeit, Position und Absorberhöhe
t = NaN; # Ersetze NaN durch eine beliebige Uhrzeit
# Position des Spiegels
x = NaN; # Ersetze NaN durch eine beliebige Spiegelposition
```

Stelle eine Formel für  $P_{\text{tube}}$  auf. Du kannst alle Funktionen, die auf diesem Blatt bereits definiert wurden, verwenden. Auch steht dir die Funktion  $I_{\text{sun}}$  zur Verfügung, welche die Stärke der Sonnenstrahlung ausgibt. Gib deine Formel für `NaN` ein und führe den Code aus.

```
In [ ]:
# Funktion zur Berechnung von Ptube
Ptube(t,b,h,s,x) = NaN; # e(t, h, s, x) * ratio(t, b, h, s, x)
# Hier nichts ändern!
checkPtubeB4(Ptube, beta);
```

### Teil f

Wie auf Arbeitsblatt 3 können wir den Verlauf der Leistung am Rohr abhängig von der Variablen  $t$  grafisch betrachten.



Lege Werte für  $b$ ,  $h$ ,  $s$  und  $x$  fest. Die Funktion  $P_{\text{tube}}$  wird in Abhängigkeit von der Zeit  $t$  dargestellt. Vergleiche die maximale Leistung, die durch einen verschobenen Spiegel umgesetzt wird, mit der Leistung, die von einem Spiegel direkt unter dem Rohr umgesetzt wird. Wie kannst du deine Beobachtungen erklären?

```
In [ ]:
# Definition der festen Größen
b = NaN; # 0.8; # Ersetze NaN durch die Breite des Sekundärreflektors i
h = NaN; # 0; # Ersetze NaN durch die Höhe des Absorberrohrs in Meter
s = NaN; # 1.1; # Ersetze NaN durch die Breite des Spiegels in Metern
x = NaN; # 0; # Ersetze NaN durch die Position des Spiegels in Metern
# Plotten - Hier nichts ändern!
f(t) = Ptube(t,b,h,s,x); plotPtubeB4(f,b,h,s,x)
```

### Fazit

Für beliebige Positionen eines Spiegels haben wir ein mathematisches Modell entwickelt, mit dem die Leistung am Rohr im Tagesverlauf berechnet werden kann.

### Optimierung

Bei unserem bisherigen Modell haben wir für die Größen  $b$ ,  $h$  und  $s$  des Kraftwerks stets Werte festgelegt. Bei der Planung eines Kraftwerks können jedoch auch solche Größen variiert werden.

Im weiteren Verlauf des Workshops ist es das Ziel, verschiedene Kraftwerkparameter so zu wählen, dass der Energiegewinn optimiert wird. Frage den Workshopleiter welches Arbeitsblatt du öffnen sollst, um mit der Optimierung zu starten.

### Zusatzmaterial:

Stelle eine Formel für die Hilfsstrecke  $ePrime$  in Abhängigkeit von  $b$  und  $\beta$  (Eingabe:  $beta(h,x)$ ) auf. Gib deine Formel im folgenden Code für `NaN` ein.

```
In [ ]:
# Funktion zur Berechnung von e'
ePrime(b,h,x) = NaN; # b * cosd(abs(90 - beta(h,x))); # Ersetze NaN durch
# Hier nichts ändern!
check_ePrime(ePrime,beta);
```

### Teil d - Der relative Anteil ratio

Wir können nun den Anteil der Leistung am Spiegel  $P_{\text{mirror}}$ , der tatsächlich am Absorberrohr ankommt, mithilfe der Längen der Hilfsstrecken  $e$  und  $e'$  berechnen.



Welcher Anteil der Leistung am Spiegel wird dem Absorberrohr zugeführt? Stelle eine Formel für den relativen Anteil `ratio` auf. Diese sollte von  $e$  und  $e'$  abhängen. Ersetze `NaN` durch deine Formel. Bei der Eingabe kannst du die Funktionen  $e(t, h, s, x)$  und  $ePrime(b, h, x)$  verwenden.

**Hinweis:** Um die kleinere von zwei Zahlen  $x$  und  $y$  zu finden, kann man in Julia die Funktion `min(x,y)` benutzen. Zum Beispiel:

```
min(5,9) = 5 und min(8,-3) = -3.
```

```
In [ ]:
# Hier nichts ändern!
e(t,h,s,x) = s * cosd(gamma(t,h,x) - alpha(t)); # Funktion zur Berechnung der
# Formel zur Berechnung des Anteils
ratio(t,b,h,s,x) = NaN; # min(1, ePrime(b,h,x) / e(t,h,s,x)); # Ersetze
# Hier nichts ändern!
checkRatio2(ratio, e, ePrime, beta);
```

### Teil e - Berechnung der Leistung am Rohr

Zuletzt müssen alle Formeln kombiniert werden, um eine Funktion für die Leistung am Rohr  $P_{\text{tube}}$  zu erhalten.

$P_{\text{tube}}$  sollte dabei nur durch **unabhängige Größen** ausgedrückt werden. In unserem Modell sind die unabhängigen Größen

- Zeit  $t$
- Position des Spiegels  $z$
- Breite des Spiegels  $s$
- Höhe  $h$  des Absorberrohrs
- Breite  $b$  des Sekundärreflektors



Um dein Modell noch weiter zu verbessern, kannst du jetzt oder im späteren Verlauf des Workshops die folgenden beiden Zusatzblätter bearbeiten:

- Zusatzblatt 1: Auf dem Zusatzblatt kann das Modell auf **mehrere** Spiegel erweitert werden.
- Zusatzblatt 2: Auf dem Zusatzblatt kann das Modell auf die Berechnung der **Gesamtenenergie, die über einen längeren Zeitraum** (Tag, Monat, Jahr...) erweitert werden.

## B.2.5 Arbeitsblatt 4 - Kurzversion: Horizontal verschobene Spiegel

### Abkürzung zur Optimierung

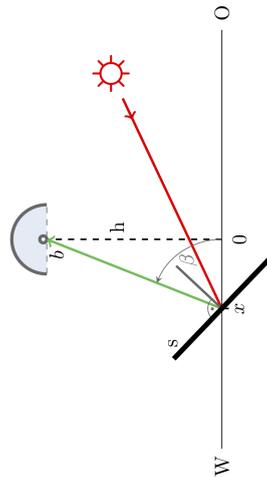


Lies den nächsten Abschnitt dieses Arbeitsblattes durch. Anschließend hast du drei Optionen, wie du weiterarbeiten kannst!

Die Annahme, dass alle Spiegel sich direkt senkrecht unter dem Absorberrohr befinden, entspricht nicht der Realität. In Wirklichkeit befinden sich im Fresnelkraftwerk viele Spiegel schräg links und rechts unter dem Absorberrohr, wie das folgende Bild zeigt. Diese Anordnung ermöglicht es, mehr als nur eine Spiegelreihe zur Reflexion der Sonnenstrahlen auf das Absorberrohr zu nutzen und auf diese Weise die Stromerzeugung zu maximieren.



Das unten stehende Bild stellt schematisch die Situation mit **einem horizontal verschobenen Spiegel** dar. Die neue Variable  $x$  gibt die **Position des Spiegelmittelpunktes** relativ zum Nullpunkt an. Der Nullpunkt befindet sich senkrecht unter dem Rohr und ist eingezeichnet. Dementsprechend ergibt eine Verschiebung des Spiegels nach **links negative Werte** für  $x$  und eine Verschiebung des Spiegels nach **rechts positive Werte** für  $x$ . In der Abbildung ist zudem die **Höhe des Absorberrohrs**  $h$  über dem Boden eingezeichnet. Diese muss bei der Verbesserung des Modells für verschobene Spiegel ebenfalls berücksichtigt werden.



Um unser Modell besser an die Realität anzupassen, wurde es um eine mögliche Verschiebung des Spiegels erweitert.

**Das neue Modell, welches die Leistung am Rohr für beliebige Spiegelpositionen**

berechnet, steht dir zur Bearbeitung des nächsten Teil des Workshops direkt zur Verfügung! Du brauchst dieses nicht selbst zu entwickeln!

Gleich geht es mit dem 2. Teil des Workshops der Optimierung weiter! Zuvor werden wir die Ergebnisse kurz im Plenum diskutieren. In der Zwischenzeit kannst du eines der folgenden Zusatzblätter bearbeiten:

1. Zusatzblatt 1: Erweiterung des Modells für mehrere Spiegel.
2. Zusatzblatt 2: Berechnung der Gesamtenergie über einen längeren Zeitraum wie z. B. einen Tag oder ein Jahr.
3. Zusatz 3: eigenständige Erweiterung des Modells für verschobene Spiegel auf AB 4.

## B.2.6 Zusatzblatt 1: mehrere Spiegel

### Zusatzblatt 1: Leistung am Rohr bei mehreren Spiegeln

Es soll die Leistung bestimmt werden, die von der Gesamtheit aller Spiegel (die auf das betrachtete Rohr reflektieren) am Absorberrohr generiert wird. Dabei gehen wir davon aus, dass keine Effekte wie Verschattung oder Blockierung zwischen den einzelnen Spiegeln auftreten.

Unser Kraftwerk besitzt insgesamt zehn Spiegelreihen. Fünf davon sind schräg links unter dem Rohr auf den Positionen  $x_1 = -1\text{m}$ ,  $x_2 = -2.5\text{m}$ ,  $x_3 = -4\text{m}$ ,  $x_4 = -5.5\text{m}$  und  $x_5 = -7\text{m}$ . Die übrigen fünf Spiegelreihen befinden sich auf den entsprechenden Positionen schräg rechts unter dem Rohr, also  $x_6 = 1\text{m}$ ,  $x_7 = 2.5\text{m}$ ,  $x_8 = 4\text{m}$ ,  $x_9 = 5.5\text{m}$  und  $x_{10} = 7\text{m}$ . Jeder Spiegel hat eine Breite von  $s = 1.1\text{m}$ . Das Absorberrohr befindet sich in einer Höhe von  $h = 7\text{m}$  und der Sekundärreflektor hat eine Breite von  $b = 0.8\text{m}$ . Wir betrachten die Zeit  $t = 11\text{Uhr}$ .



Gib für die ersten vier NaMs die im Text genannten Größen für die Kraftwerkparameter ein. Stelle dann eine Formel für die Leistung auf, die das gesamte oben beschriebene Kraftwerk um 11 Uhr generiert. Du kannst die Funktion  $P_{\text{tube}}$  mit  $P_{\text{tube}}(t, b, h, s, x)$  verwenden. Gib deine Formel für das letzte NaN ein.

```
In [ ]:
# Hier nichts ändern
include("../code/setup_multiMirrors.jl")
# Du kannst die Funktion Ptube(t,b,h,s,x) nutzen.

# --- Festlegung einiger Parameter des Kraftwerks --- #
s = NaN; # 1.1; # Breite des Spiegels
b = NaN; # 0.8; # Breite des Sekundärreflektors
h = NaN; # 6; # Höhe des Absorberrohrs
t = NaN; # 11; # Uhrzeit

# Leistung Kraftwerk
Ptocal = NaN; # Ptube(t,b,h,s,-7) + Ptube(t,b,h,s,-5.5) + Ptube(t,b,h,s,
# Hier nichts ändern!
checkPtocal(Ptocal,Ptube,t,b,h,s);
```

**Tipp:** Du kannst die Leistung auch über eine for-Schleife berechnen. Falls die Verwendung von for-Schleifen neu für dich ist, findest du hier ein Beispiel dazu.

## B.2.7 Zusatzblatt 2: Energie über einen Tag

### Zusatzblatt: Gesamtenergie über einen Zeitraum

#### Einleitung

Wir sind nicht nur daran interessiert, wie viel **Leistung** das Absorberrohr zu einem bestimmten Zeitpunkt  $t$  erreicht, sondern insbesondere daran, wie viel **Energie über einen längeren Zeitraum** insgesamt gewonnen wird. Dieser Zeitraum kann ein Tag, eine Woche oder mehrere Jahre sein.

Das Ziel dieses Arbeitsblattes ist es, die Gesamtenergie über einen Tag zu berechnen. Dazu sollst du die Funktion `Ptube(t, b, h, s, x)` nutzen, welche die **Leistung am Rohr** beschreibt.

#### Zusammenhang zwischen Leistung und umgesetzter Energie

##### Teil a.)

Zunächst werden wir für die Größen  $b$ ,  $h$ ,  $s$  und  $x$  des Kraftwerkes konkrete Werte festlegen.

Führe den folgenden Code aus. Die Werte für die genannten Größen werden in Variablen gespeichert. Die Funktion der Leistung am Rohr `f(t) = Ptube(t, b, h, s, x)`, vom vorherigen Arbeitsblatt, wird graphisch dargestellt.

```
In [ ]:
# Hier nichts ändern!
include("../code/setup_numintegration.jl");

# --- Festlegung einiger Parameter des Kraftwerks --- #
b = 0.8; # Breite des Sekundärreflektors
s = 1.1; # Breite des Spiegels
h = 6; # Höhe des Absorberrohrs
x = -2; # Position des Spiegels

plotPtube(Ptube, b, h, s, x);
```

Ziel ist es, die umgesetzte Energie über einen Tag **näherungsweise** zu berechnen. Dazu ist es sinnvoll, sich zunächst zu überlegen, wie die **Leistung P** und die **Energie E** zusammenhängen. Es gilt:  
Die Leistung  $P$  entspricht der umgesetzten Energie  $\Delta E$  pro Zeiteinheit  $\Delta t$ . Wir können also festhalten:

$$P = \frac{\Delta E}{\Delta t}$$

**Hinweis:** Das Symbol  $\Delta (= \text{Delta})$  steht für die Differenz zwischen zwei Werten. Zum Beispiel steht  $\Delta t$  für die Differenz zwischen zwei Zeitpunkten  $t_1$  und  $t_2$ .



Stelle eine Formel für die Berechnung der umgesetzten Energie über einen Tag auf. Die Energie soll in der Einheit Wattstunden (=  $W \cdot h$  oder kurz **Wh**) berechnet werden! Gib deine Formel im folgenden Code ein. Die Funktion `Ptube(t, b, h, s, x)` vom vorherigen Arbeitsblatt kannst du verwenden.

#### Hinweise:

- Nutze eine for-Schleife. Falls die Verwendung von for-Schleifen neu für dich ist, findest du hier ein Beispiel dazu.
- Falls du Schwierigkeiten beim Lösen dieser Aufgabe hast, findest du hier eine **Hilfekarte**.

```
In [ ]:
# Hinweis: Du kannst die Funktion Ptube(t,b,h,s,x) verwenden.

function Etube(b,h,s,x)
    Energy = 0; # Am Anfang des Tages ist die gesamte Energie noch 0
    DeltaT = NaN; # = NaN; # Wähle einen Wert für die Länge der Zeitabschn
    ## Hier beginnt die for - Schleife
    for t = NaN : NaN : NaN; # 0 : DeltaT + 24 # Berechne mithilfe von d
        Energy = NaN; # Energy + Ptube(t, b, h, s, x) * DeltaT
    end
    return Energy
end

# Hier nichts ändern!
totalEnergy = Etube(b,h,s,x)
checkEnergyOneDay(totalEnergy, b, h, s, x)
```

#### Fazit

Wir haben ein Modell entwickelt, mit dem die **gesamte Energie** erzeugt durch **einen Spiegel** über einen längeren Zeitraum berechnet werden kann. Dabei haben wir für verschiedene Größen des Kraftwerks (d.  $h$ ,  $h$  und  $s$ ) stets konkrete Werte festgelegt. Bei der Planung eines Kraftwerks können auch solche Größen variiert und nach eigenen Vorstellungen bei den Unternehmen, die Fresnelspiegel bzw. Absorberrohre herstellen, geordert werden.

Im weiteren Verlauf des Workshops ist es das Ziel, verschiedene **Kraftwerkparameter zu optimieren**.

#### Diskussion

Ist das Ergebnis für die **Gesamtenergie** über einen Tag viel oder wenig? Um die berechnete Leistung besser einordnen zu können, ist ein Vergleich mit fossilen Kraftwerken und Kernkraftwerken sinnvoll. Außerdem könntest du recherchieren, wie viel Energie - hauptsächlich in Form von Strom - deine Stadt pro Tag benötigt und wie viele Haushalte unser Fresnelkraftwerk in etwa mit Strom versorgen kann.

## B.3 Arbeitsblätter zur Optimierung der Rohrhöhe

### B.3.1 Arbeitsblatt 5: Optimale Rohrhöhe

#### Solkraftwerk - AB 5: Optimierung der Rohrhöhe

mit unserem Modell von AB 4



Notiere deine Ergebnisse während der Optimierung auf diesem Antwortblatt.

#### Einleitung

Die Funktion  $P_{\text{tube}}$  beschreibt die Leistung am Rohr abhängig von den fünf Größen  $t$ ,  $b$ ,  $h$ ,  $s$  und  $x$ . Auf Arbeitsblatt 3 und 4 haben wir die Leistung im Tagesverlauf betrachtet. Die

Variable war die Tageszeit  $t$ . Die **fixen Parameter** waren:

- Breite des Sekundärreflektors  $b$
- Höhe des Absorberrohrs  $h$
- Breite des Spiegels  $s$
- Position des Spiegels  $x$

Für  $b = 0.8\text{m}$ ,  $h = 6\text{m}$ ,  $s = 1.1\text{m}$  und  $x = -2\text{m}$  sieht der Graph der Funktion  $P_{\text{tube}}$  beispielsweise so aus:



Führe den folgenden Code aus, um dir den Graph der Funktion erneut anzeigen zu lassen.

```
In [ ]: # Hier nichts ändern!
include("../code/setup_AB5Hand6_08.jl");
# Plotten
plotPtube(Ptube, 0.8, 6, 1.1, -2)
```

#### Optimierung der Rohrhöhe bei einem Spiegel und fester Uhrzeit

Bevor ein Kraftwerk gebaut wird, müssen in der Regel optimale Werte für die konstanten Parameter bestimmt werden. Exemplarisch werden wir auf diesem Arbeitsblatt die **Rohrhöhe  $h$**  optimieren - sie wird damit zur **Variablen**.

Zunächst treffen wir die Annahme, dass die **Parameter mit festen Werten** wie folgt gegeben sind:



Gib die abgelesene optimale Höhe für das `NaH` ein. Die zugehörige Leistung am Rohr wird berechnet. Erhöhe anschließend den Wert für die Höhe und berechne die Leistung erneut. Notiere deine Beobachtungen auf dem Antwortblatt und begründe, welche Höhe du für die Konstruktion des Kraftwerks wählen würdest.

```
In [ ]:
hOptimal = #= NaN; #= 400;
# Hier nichts ändern!
checkROpt(t, b, hOptimal, s, x)
```

**Teil c**

Das Ablesen der Lösung am Graphen ist nicht unbedingt genau. In Wirtschaft und Forschung kommt bei der Optimierung daher der Computer zum Einsatz. Wir werden in dieser Teilaufgabe mithilfe eines Optimierungstools von Julia die optimale Höhe **numerisch**, das heißt näherungsweise rechnerisch, bestimmen. Dazu legen wir fest, in welchem Bereich wir nach dem Optimum suchen. Dieser Suchbereich ist zunächst auf `Left = 0.0` bis `Right = 100.0` festgesetzt.



Führe den Code aus. Mit dem Optimierungstool von Julia wird das Optimum bestimmt und angezeigt. Notiere den ermittelten Wert für die optimale Höhe und die zugehörige Leistung auf dem Antwortzettel.

Ändere dann die Grenzen des Bereichs, in dem der Algorithmus nach der Lösung sucht (Variable: `Right`) und bestimme die optimale Höhe erneut. Was beobachtest du?

```
In [ ]:
# Linke und rechte Grenze des Suchbereichs
Left = [0.0];
Right = [100.0];
# Optimieren - Hier nichts ändern!
Zielfunktion(h) = -Ptube(t, b, h, s, x); optimizeAndPrint(Zielfunktion, Left, Right)
```



Diskutiere die folgenden Punkte:

- Sind die Ergebnisse realistisch?
- Welche Informationen oder anwendungsrelevanten Phänomene haben wir außer Acht gelassen? Notiere Ideen, wie das Modell für die Leistung am Rohr verbessert werden kann auf dem Antwortzettel.
- Diskutiere deine Überlegungen mit deinem Sitznachbarn.

- die Breite des Sekundärreflektors  $b = 0.8\text{m}$  (vom Hersteller vorgegeben),
- die Breite des Spiegels  $s = 1.1\text{m}$  (vom Hersteller vorgegeben),
- die Position des Spiegels  $x = -2\text{m}$  (exemplarisch festgelegt) und
- die Uhrzeit  $t = 11\text{h}$  (exemplarisch festgelegt).

Die Parameter  $x$  und  $t$  werden wir später variieren. Ziel dieses Arbeitsblattes ist es, die Absorberrohrhöhe zu bestimmen, bei der die **maximale Leistung** umgesetzt wird. Mathematisch gesprochen ist ein **Optimierungsproblem** zu lösen. Die Funktion, deren optimaler Wert gesucht wird, heißt auch **Zielfunktion**. In unserem Fall ist das die Funktion  $P_{\text{tube}}$  mit  $P_{\text{tube}}(t, b, h, s, x)$ .



Öffne dieses **Infoblatt** für mehr Infos rund um Optimierungsprobleme.

**Teil a - Optimum am Graphen ablesen**

Wir bestimmen die optimale Rohrhöhe zunächst graphisch. Für die festen Parameter speichern wir dazu vorab die Werte  $t = 11\text{h}$ ,  $b = 0.8\text{m}$ ,  $s = 1.1\text{m}$  und  $x = -2\text{m}$ .



Führe den Code aus. Die Werte für die Parameter  $t$ ,  $b$ ,  $s$  und  $x$  werden gespeichert. Zudem wird der Graph der Funktion  $P_{\text{tube}}$  in Abhängigkeit von der Höhe  $h$  dargestellt. Bei welcher Höhe wird die Leistung am Rohr maximal? Notiere dein Ergebnis auf dem Antwortblatt.

**Hinweis:** Wir nutzen zunächst alle die gleichen Werte für die Parameter  $t$  und  $x$ , damit wir unsere Ergebnisse leichter untereinander vergleichen können.

```
In [ ]:
# --- Festlegung einiger Parameter des Kraftwerks --- #
t = 11; # Uhrzeit
x = -2; # Position des Spiegels
b = 0.8; # Breite des Sekundärreflektors
s = 1.1; # Breite des Spiegels
# Der Wert von hRight legt die rechte Grenze der ausgegebenen Grafik fest. Die
hRight = 100;
# Plotten - Hier nichts ändern!
g(h) = Ptube(t, b, h, s, x);
plotPtube(g, t, 0, hRight)
```

Auf folgendem **Zusatzblatt** kamst du den Fall, dass der Spiegel direkt unter dem Absorberrohr steht (d. h.  $x = 0$ ) diskutieren.

**Teil b**

Wir überprüfen in dieser Teilaufgabe die Optimalität der abgelesenen Lösung.



## B.3.2 Arbeitsblatt 6: Modellverbesserung 1 – atmosphärische Effekte

### Solkraftwerk – AB 6: Modellverbesserung 1

#### Berücksichtigung von atmosphärischen Effekten

Notiere deine Ergebnisse während der Optimierung auf diesem [Antwortblatt](#).



#### Teil a

Bei unserem bisherigen Modell sind wir davon ausgegangen, dass die Energie eines Sonnenstrahls immer gleich bleibt – egal wie lang die Strecke ist, die der reflektierte Strahl vom Spiegel bis zum Rohr zurücklegt. Dies entspricht nicht der Realität. Tatsächlich **sinkt** die Energie eines Lichtstrahls, wenn die zurückgelegte Strecke wächst. Dies liegt an der sogenannten **atmosphärischen Abschwächung**. Diesen physikalischen Effekt werden wir nun in unser Modell einbauen.

Stelle eine Formel für die Länge der Strecke auf, die ein Lichtstrahl von der Mitte des Spiegels bis zum Rohr zurücklegt. Die gesuchte Länge nennen wir  $d$ . Deine Formel sollte von  $x$  und  $h$  abhängen. Gib deine Formel im nachfolgenden Code für `NaN` ein. Die Berechnung der Streckenlänge wird dann in der Funktion `d(h,x)` gespeichert.

Falls du Hilfe benötigst, kannst du dir diesen [Tipp ansehen](#).

#### Hinweise:

- Um die Quadratwurzel einer Zahl zu berechnen, kannst du den Befehl `sqrt( )` verwenden. Zum Beispiel ergibt: `sqrt(4) = 2`.
- In Julia gibt man Potenzen mit dem Zeichen `^` ein. Zum Beispiel ist `3^2 = 9` oder `10^-2 = 0.01`.

```
In [ ]:
# Hier nichts ändern
include("../code/setup_AB5and6_06.jl");

# --- Festlegung einiger Parameter des Kraftwerks --- #
t = 11; # Uhrzeit
x = -2; # Position des Spiegels
b = 0.8; # Breite des Sekundärreflektors
s = 1.1; # Breite des Spiegels

# Formel für die Strecke von Spiegelmittle bis zum Rohr
d(h,x) = NaN; ## sqrt(x^2 + h^2); # Gib hier deine Formel für die Berechnu
```

```
# Hier nichts ändern! – Checken
checkDistance(d)
```

#### Teil b

Wissenschaftler/innen im Bereich der Solarenergieforschung haben ein bewährtes mathematisches Modell entwickelt, das beschreibt, wie stark das reflektierte Sonnenlicht abgeschwächt wird. Unter Verwendung dieses Modells erhält man die Funktion `atmLoss`. Mit dieser kann berechnet werden, welcher **Anteil** der abgegebenen Leistung tatsächlich am Ziel (in unserem Fall dem Rohr) ankommt. Dieser Anteil hängt von der **Länge  $d$**  ab. Die Funktion `atmLoss` abhängig von der zurückgelegten Streckenlänge  $d$  lautet:

$$\text{atmLoss}(d) = 0.99321 - 1.176 \cdot 10^{-4} \cdot d + 1.97 \cdot 10^{-8} \cdot d^2 \quad \text{für } d \leq 1000 \text{ m}$$

Schauen wir uns dazu ein **Beispiel** an:

Die Leistung des Strahlenpaktes, welches vom Spiegel auf Position  $x = -2\text{m}$  reflektiert wird, beträgt um 11 Uhr ohne Berücksichtigung der atmosphärischen Abschwächung  $560.88\text{W}$  – unabhängig von der zurückgelegten Strecke. Berücksichtigt man jedoch die atmosphärische Abschwächung, so beträgt die Leistung des selben Strahlenpaktes nach  $20\text{m}$  zurückgelegter Strecke nur

$$560.88 \text{ W} \cdot \text{atmLoss}(20) = 560.88 \text{ W} \cdot 0.9915 = 556.59 \text{ W}$$

Gib die Funktion `atmLoss(h,x)`, die die atmosphärische Abschwächung beschreibt, im Code für `NaN` ein. Führe den Code aus. Die Funktion der atmosphärischen Abschwächung (= relativer Anteil der Leistung, die das Rohr erreicht) wird in Abhängigkeit von der Höhe  $h$  gespeichert und grafisch dargestellt.

#### Hinweise:

- In unserem Modell hängt die zurückgelegte Strecke  $d$  von der Position  $x$  und der Höhe  $h$  ab. Deswegen schreiben wir im Code `atmLoss(h,x)` anstelle von `atmLoss(d)`.
- Du kannst deine Funktion `d(h,x)` bei der Eingabe verwenden.

```
In [ ]:
# Funktion für die atmosphärische Abschwächung
atmLoss(h,x) = NaN; ## hähf = 0.99321-1.176*10^(-4)*d(h,x)+1.97*10^(-8)*d(h,x)^2

# Hier nichts ändern! – Checken und plotten
checkAtmLoss(atmLoss, d)
f(h) = atmLoss(h,x); plotAtmLoss(f,0,20)
```

#### Teil c

In das Modell für die Leistung am Rohr muss der Effekt der atmosphärischen Abschwächung aufgenommen werden. Die Funktion, die wir erhalten, bezeichnen wir mit `PabkAtmLoss`.

Stelle die Funktionsgleichung für die Funktion `$P_{abkAtmLoss}`

auf, mit der die Leistung am Rohrer Berücksichtigung atmosphärischer Abschwächung  
 . Du kannst  $P_{\text{tube}}(t, b, h, s, x)$  und  $\text{atmLoss}(h, x)$  verwenden. Der Graph der neuen Funktion  $P_{\text{tube}}(t, b, h, s, x)$  wird in Abhängigkeit der Rohrhöhe dargestellt. Lies die optimale Höhe ab und notiere diese auf dem Antwortzettel.

```
In [ ]: PtubeAtmLoss(t, b, h, s, x) = # NaH; # Ptube(t, b, h, s, x) * atmLoss(h, x); # Gib h
# Hier nichts ändern! - Checken und plotten
checkPtubeAtmLoss(PtubeAtmLoss);
hRight = 500; f(h) = PtubeAtmLoss(t, b, h, s, x); g(h) = Ptube(t, b, h, s, x);
plotPtubeAtmLoss(f, g, t, 0, hRight)
```

**Teil d**

Wir werden die optimale Höhe erneut numerisch mit Hilfe des Optimierungstools von Julia bestimmen.

&

- Führe den Code aus. Die optimale Höhe des Rohres wird basierend auf der neuen Funktion  $P_{\text{tube}}(t, b, h, s, x)$  bestimmt. Notiere das Ergebnis auf dem Antwortzettel.
- Setze die Position des Spiegels anschließend auf  $x = -5$  und führe die Optimierung erneut aus. Was beobachtest du? Notiere die optimale Höhe.
- Ist das Ergebnis der Optimierung praktikabel? Welchen Wert würdest du für die Konstruktion des Rohres wählen? Notiere Beobachtungen und Ideen für mögliche Modellverbesserungen.

```
In [ ]: # Position des Spiegels
x = -2;
# Optimieren - Hier nichts ändern!
Left = 0.0;
Right = 5000.0;
Zielfunktion(h) = -PtubeAtmLoss(t, b, h[1], s, x);
optimizeAndPrint(Zielfunktion, Left, Right, t, b, s, x, 0);
```

**Fazit**

Wir haben atmosphärische Effekte in unser Modell aufgenommen. Je nach Position des Spiegels erhalten wir jedoch immer noch **sehr grobe Werte für die Höhe  $h$** .

**Und so gehts nun weiter...**

1. Hast du die Modellverbesserung zum **Fehlerinfluss noch nicht umgesetzt**? Dann öffne nun [Modellverbesserung 2](#).
2. Hast du die Modellverbesserung zum **Fehlerinfluss schon umgesetzt**? Dann führe die Optimierung der Rohrhöhe auf diesem [Arbeitsblatt](#) mit beiden Modellverbesserungen zusammen durch.

### B.3.3 Arbeitsblatt 7: Modellverbesserung 2 – Fehlerberücksichtigung

#### Solkraftwerk – AB 7: Modellverbesserung 2

#### Berücksichtigung von Einstellungsfehlern



Notiere deine Ergebnisse während der Optimierung auf diesem [Antwortblatt](#).

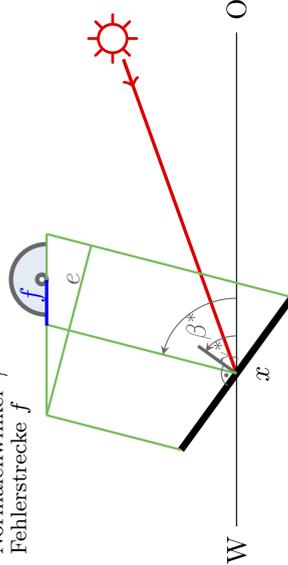
Die Spiegel in einem Fresnel-Kraftwerk werden mithilfe von Motoren im Laufe des Tages gemäß des Sonnenstandes ausgerichtet. Wir haben die notwendige Ausrichtung eines Spiegels mithilfe des Neigungswinkels  $\gamma$  bestimmt. Da es durchaus vorkommt, dass einzelne Motoren ungenau arbeiten, entspricht die tatsächliche Ausrichtung der Spiegel nicht immer exakt der gewünschten Ausrichtung. Wir werden den Einfluss von Ungenauigkeiten bei der Spiegelausrichtung auf die optimale Rohrhöhe untersuchen.

**Hinweis:** Auf diesem Arbeitsblatt werden alle Modellierungsschritte nur für eine Linksverschiebung des Spiegels durchgeführt. Die Position des Spiegels setzen wir zunächst auf  $x = -2$  fest.

#### Teil a - Einfluss von Fehlern auf die Leistung am Rohr

Wird der Spiegel nicht korrekt eingestellt, so führt dies dazu, dass der mittlere reflektierte Strahl nicht direkt auf das Rohr fällt. Stattdessen verfehlt er das Rohr (links oder rechts) um eine Fehlerstrecke  $f$ . Dies ist in der Abbildung unten dargestellt.

Reflexionswinkel  $\beta^*$   
 Normalenwinkel  $\gamma^*$   
 Fehlerstrecke  $f$



Den **fehlerbehafteten Neigungswinkel** bezeichnen wir mit  $\gamma^*$ . Dieser ergibt sich aus dem **gewünschten Neigungswinkel**  $\gamma$ , indem man den Fehler `error` hinzuaddiert:

$$\gamma^* = \gamma + \text{error}$$



Führe den Code aus. Der Fehler auf den Neigungswinkel `error` wird gespeichert. Dieser ist zunächst auf `error = 4` festgelegt. Es wird eine Grafik erzeugt, die den gewünschten reflektierten Strahl und den aufgrund des Fehlers entstandenen reflektierten Strahl darstellt. Variiere den Fehler und beobachte den Einfluss auf den fehlerhaften Ausfallswinkel  $\beta^*$  und die Fehlerstrecke  $f$ .

```
In [ ]:
# Hier nichts ändern!
include("../code/setup_AB7_MS_11");
# --- Festlegung einiger Parameter des Kraftwerks --- #
t = 11; # Uhrzeit
x = -2; # Position des Spiegels
b = 0.8; # Breite des Sekundärreflektors
s = 1.1; # Breite des Spiegels
# Fehler auf den Neigungswinkel
error = 4; # Du kannst diesen Fehler variieren
# Erzeugen der Grafik
plotError(x, 0.8, s, t, error)
```

#### Teil b

Wird der Neigungswinkel  $\gamma$  nicht korrekt eingestellt, so führt dies auch zu einem fehlerhaften Ausfallswinkel  $\beta^*$ .

#### Zur Erinnerung:

Die Funktionen für die gewünschten Winkel, die wir auf Arbeitsblatt 4 entwickelt haben, lauten:

#### Gewünschter Ausfallswinkel

$$\beta = \begin{cases} \arctan\left(\frac{h}{|x|}\right) & \text{für } x < 0 \\ 90^\circ & \text{für } x = 0 \\ 180^\circ - \arctan\left(\frac{h}{x}\right) & \text{für } x > 0 \end{cases}$$

#### Gewünschter Neigungswinkel

$$\gamma = \frac{1}{2} \cdot (\alpha + \beta)$$



Gib für das erste `Math` die oben genannte Formel zur Berechnung von  $\gamma^*$  (`= gammaError`) ein. Stelle dann eine Formel für den fehlerhaften Ausfallswinkel  $\beta^*$  (`= betaError`) auf. Die Formel sollte von der Tageszeit  $t$ , der Höhe  $h$ , der Position  $x$  und dem Fehler `error`

**Hinweis:** Den Betrag einer Zahl kannst du mit dem Befehl `abs()` berechnen. Zum Beispiel ist: `abs(-4) = 4`.  
Den Tangens gibt man als `tand()` ein.

Falls du Hilfe benötigst, kannst du dir diesen [Tipp](#) ansehen.

```
In [ ] :
# Festlegung des Fehlers
error = 1.5;
# Fehlerstrecke f
errorDistanceLeftShift(h,x,error) = # NaN;
# Hier nichts ändern!
checkErrorDistanceLeftShift(errorDistanceLeftShift,t,b,s,x,error);
g(h) = errorDistanceLeftShift(h,x,error);
```

**Teil d**

**Annahme:** Wir gehen im Folgenden davon aus, dass für den Fehler auf den Neigungswinkel gilt:

$$-1.0 \leq \text{error} \leq 1.0$$

Ziel ist es, die Höhe so zu wählen, dass die umgesetzte Energie maximiert, aber zugleich der Einfluss von Fehlern berücksichtigt wird. Zur korrekten Bestimmung der Leistung am Rohr muss die Abweichung um die Fehlerstrecke  $f$  in das mathematische Modell aufgenommen werden. So kann der Anteil der vom Spiegel abgebenen Leistung, der das Rohr tatsächlich erreicht, neu bestimmt werden. Die Länge der **bestrahlten Strecke auf Rohrhöhe** bezeichnen wir wie zuvor mit  $p$  (siehe Abbildung oben).

Zunächst werden wir die Länge der Strecke  $p$  berechnen und anschließend die Länge der Teilstrecke von  $p$  bestimmen, die im Bereich des Sekundärreflektors liegt. Die Länge dieser Teilstrecke nennen wir  $l$  (siehe Abbildung unten)



Stelle eine Formel für die bestrahlte Streckenlänge auf Rohrhöhe  $p$  auf. Gib deine Formel für  $l$  ein. Die Funktion `e(t,b,h,s,x,error)` ist bereits eingespeichert. Diese berechnet die Streckenlänge  $e$  unter Berücksichtigung des Einstellfehlers. Du kannst diese verwenden. Auch kannst du deine Funktionen `betaError(h,x,error)` und `gammaError(t,h,x,error)` verwenden.

**Hinweise:**

- Für die Berechnung von  $p$  benötigst du den **fehlerhaften** Ausfallswinkel  $\beta^*$ .
- Kosinus und Sinus gibt man als `cosd()` bzw. `sind()` ein.

```
In [ ] :
# Bestrahlte Strecke auf Rohrhöhe
computeIllumination(t,b,h,s,x,error) = # NaN;
e(t,b,h,s,x,error)/cosd(abs
```

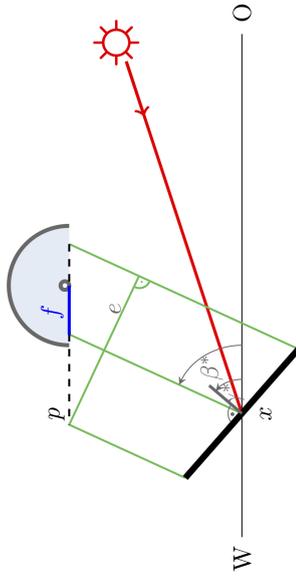
abhängen. Bei der Eingabe der Formeln kannst du `alpha(t)`, `beta(h,x)` und `gamma(t,h,x)` von Arbeitsblatt 4 verwenden.

Falls du Hilfe benötigst, kannst du dir diesen [Tipp](#) ansehen.

```
In [ ] :
# Formeln für den fehlerhaften Neigungswinkel
gammaError(t,h,x,error) = # NaN;
betaError(h,x,error) = # NaN;
# Formeln für und den fehlerhaften Ausfallswinkel
betaError(h,x,error) = # NaN;
# Hier nichts ändern!
checkBetaError(betaError,gammaError,error)
```

**Teil c**

Wenn der tatsächliche Ausfallswinkel vom gewünschten Wert abweicht, hat dies auch Einfluss auf die angestrahlte Strecke auf der Höhe des Rohres, die wir im Folgenden mit  $p$  bezeichnen (siehe Abbildung unten). Diese liegt ggf. nur noch teilweise oder gar nicht mehr auf dem Sekundärreflektor.



Mit der Funktion  $P_{\text{tube}}$  von Arbeitsblatt 5 können wir die tatsächliche Leistung am Rohr nicht mehr exakt beschreiben. Im Folgenden werden wir zunächst die **Fehlerstrecke**  $f$  berechnen und anschließend den Einfluss der Rohrhöhe  $h$  auf die Länge dieser Fehlerstrecke untersuchen. Ziel ist es, die Rohrhöhe so zu wählen, dass auch bei zu erwartenden Ungenauigkeiten bei der Spiegelausrichtung hinreichend viel Leistung am Rohr generiert wird.



Stelle eine Formel für die **Fehlerstrecke**  $f$  im Fall einer Linksverschiebung des Spiegels (d. h.  $x < 0$ ) auf. Deine Formel sollte von  $h$ ,  $x$  und dem Fehler `error` abhängen. Ist deine Formel für beliebige Werte des Einstellfehlers anwendbar?

```
# Hier nichts ändern – Checken
checkillum(computeIllumination, t, b, s, x, error)
```

**Teil e**

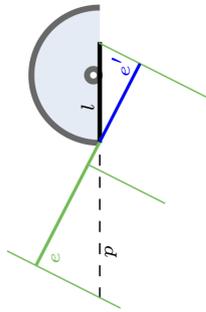
Wie wir von Arbeitsblatt 3 bereits wissen, gilt: Die Leistung, die auf der gesamten Strecke  $p$  anliegt, entspricht der Leistung auf der Hilfsstrecke  $e$ . Diese Leistung kann berechnet werden durch:

$$P_{\text{sun}} \cdot e$$

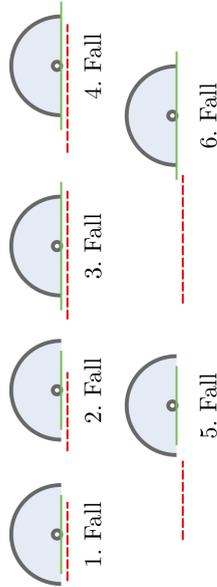
Die Leistung, die am Sekundärreflektor ankommt, entspricht jedoch nur dem Anteil, den die Strecke  $l$  an  $p$  ausmacht. Diesen neuen relativen Anteil bezeichnen wir mit **ratioError**. Mit diesem ergibt sich für die Leistung am Rohr:

$$P_{\text{tube}} = P_{\text{sun}} \cdot e \cdot \text{ratioError}$$

**bestrahlter Teil Sekundärreflektor  $l$**



Abhängig von der Größe des Fehlers, können unterschiedliche Situationen auftreten, die wiederum in sechs Fälle unterteilt werden können. Diese sechs Fälle sind in der folgenden Abbildung dargestellt:



Die untere eingezeichnete Strecke stellt jeweils den Bereich dar, der bei fehlerhafter Einstellung des Neigungswinkels tatsächlich bestrahlt wird (also  $p$  basierend auf  $\gamma$ ). Die obere Strecke stellt den bestrahlten Bereich dar, wenn der mittlere reflektierte Strahl wie gewünscht auf die Mitte des Rohrs trifft (also  $p$  basierend auf  $\gamma$ ).

- Der Fehler ist so klein, dass bei fehlerhafter Einstellung genau so viele Strahlen auf den Sekundärreflektor treffen, wie bei korrekter Einstellung (1. und 3. Fall).
- Ein Teil der Strahlen, die ohne fehlerhafte Einstellung den Sekundärreflektor getroffen hätten, fällt jetzt links oder rechts neben den Sekundärreflektor (2. und 4. Fall).
- Der Fehler ist so groß, dass alle Strahlen den Sekundärreflektor verfehlen (5. und 6. Fall).



Stelle für die Fälle eins bis drei je eine Formel auf, mit der der Anteil der reflektierten Leistung berechnet werden kann, der dem Rohr tatsächlich zugeführt wird (Variable **ratioError**). Du kannst die Sekundärreflektorbreite  $b$ , die Fehlerstrecke  $f$  und die bestrahlte Streckenlänge  $p$  in deinen Formeln verwenden. Gib deine Formeln für die entsprechenden **NaNs** im folgenden Code ein.

Falls du Hilfe benötigst, kannst du dir diesen  **Tipp**  ansehen.

```
In [ ] :
ratioErrorCase1(b, f, p) == NaN; ==# 1; # Gib hier deine Formel für den Anteil
ratioErrorCase2(b, f, p) == NaN; ==# (p - (f - ((b - p)/2))) / p; # Gib hier
ratioErrorCase3(b, f, p) == NaN; ==# b / p; # Gib hier deine Formel für den
checkRatioErrorCases(ratioErrorCase1, ratioErrorCase2, ratioErrorCase3)
```

**Hinweis:** Die Formeln für die anderen drei Fälle brauchst du nicht selbst zu entwickeln.

**Teil f - Die Leistung am Rohr unter Fehlereinfluss**

Ersetzen wir in der Funktion  $P_{\text{tube}}$  den Faktor **ratio** durch **ratioError**, erhalten wir die Funktion  $P_{\text{tubeError}}$ , welche die Leistung am Rohr unter Berücksichtigung möglicher Einstellungsfehler berechnet. Im Folgenden kann  $P_{\text{tubeError}}$  verwendet werden, um für beliebige Positionen (d. h. auch für  $x = 0m$  und  $x > 0m$ ) die Optimierung der Rohrhöhe vorzunehmen.



Führe den folgenden Code aus. Der Graph der Funktion  $P_{\text{tubeError}}$  mit Einstellungsfehler **error = 0.5°** und der Graph der Funktion  $P_{\text{tube}}$  werden dargestellt. Wie lässt sich der Verlauf des Funktionsgraphen von  $P_{\text{tubeError}}$  erklären?

```
In [ ] :
# Fehler auf dein Einstellungsgröße
error = 0.5;
# - Hier nichts ändern! - Plotten
f(h) = PtubeError(t, b, h, s, x, error); g(h) = Ptube(t, b, h, s, x);
plotPtubeError(f, g, t, 0, 50)
```

**Teil g ... zurück zur Optimierung**

Ziel ist es, die Rohrhöhe  $h$ , so zu wählen, dass die Leistung am Rohr maximal ist - und zwar unter Berücksichtigung des Einflusses von Einstellungsfehlern. Auch bei kleineren Einstellungsfehlern des Spiegels sollen also gute oder besser gesagt optimale Werte für die Leistung erreicht werden. Wir lösen damit ein **Optimierungsproblem unter Unsicherheiten**.



- Führe den Code aus. Die optimale Höhe des Rohrs bei dem gegebenen Fehler  $\text{error} = 0.3^\circ$  wird mit einem Optimierungstool bestimmt. Notiere deine Ergebnisse. Ist das Ergebnis für die optimale Höhe realistisch?
- Untersuche zudem, wie stark die optimale Lösung von der Position des Spiegels abhängt. Setze dazu die Position des Spiegels oben auf dem Arbeitsblatt auf  $\alpha = -5\text{m}$  und berechne die optimale Höhe erneut. Welche Höhe würdest du für die Konstruktion des Kraftwerks wählen?
- Hast du Ideen für weitere Modellverbesserungen?

```
In [ ]:
# Position des Spiegels
x = -2;

# Grenzen des Suchbereichs
Left = 0.0;
Right = 200.0;

# - Hier nichts ändern - Optimieren
Zielfunktion(h) = -PtubeError(t, b, h[1], s, x, error); optimizeAndPrint(Ziel);
```

#### Zusatzaufgabe:



Auf diesem Arbeitsblatt haben wir die Einstellung des Spiegels als Fehlerquelle diskutiert. Überlege dir, was weitere Fehlerquellen bei einem Solarkraftwerk sein könnten. Notiere deinen Ideen auf dem Antwortzettel.

#### Diskussion

Wir haben unser Modell so erweitert, dass potentielle Fehler bei der Einstellung eines Spiegels berücksichtigt werden.

Und so gehts nun weiter....

1. Hast du die Modellverbesserung zu den atmosphärischen Effekten **noch nicht** umgesetzt? Dann leg nun damit los, indem du die [Modellverbesserung 1](#) öffnest.
2. Hast du die Modellverbesserung zu den atmosphärischen Effekten bereits umgesetzt? Dann führe die Optimierung der Rohrhöhe auf diesem [Arbeitsblatt](#) mit der kombinierten Funktion `PtubeErrorAtmLoss` durch!

## B.3.4 Arbeitsblatt 8: Kombination von Modellverbesserung 1 & 2

### Solarkraftwerk – AB 8: Kombination von Modellverbesserungen 1 & 2

#### Optimierung der Rohrhöhe



Notiere deine Ergebnisse der Optimierung auf diesem [Antwortblatt](#).

Ziel ist es, die Rohrhöhe  $h$  zu bestimmen, bei der die Leistung am Rohr unter Berücksichtigung **atmosphärischer Effekte** und möglicher **Einstellungsfehler** maximal ist. Die neue Zielfunktion, bei der beide Modellverbesserungen berücksichtigt werden, nennen wir `PtubeErrorAtmLoss`.

#### Teil a



Führe die Optimierung für die Fehlerwerte

1. error = 0
2. error = 0.1
3. error = 0.2
4. error = 0.5

sowie für beliebige andere Fehlerwerte durch. *Vergleiche die Ergebnisse mit den Ergebnissen von Arbeitsblatt 6 und 7. Erkläre deine Beobachtungen.*

**Hinweis:** Der Fehler `error` stellt den Fehler auf den Einstellungswinkel  $\gamma$  dar. Das heißt es gilt:  $\gamma^* = \gamma + \text{error}$

```

In ( ):
# Hier nichts ändern!
include("../code/setup_AB8_MS.jl");
# --- Festlegung einiger Parameter des Kraftwerks --- #
t = 11; # Uhrzeit
x = -2; # Position des Spiegels
b = 0.8; # Breite des Sekundärreflektors
s = 1.1; # Breite des Spiegels
error = NaN; # Gib hier verschiedene Werte für den Fehler ein.
# - Hier nichts ändern - Optimieren
Left = 0.0;
Right = 500.0;
Zielfunktion(h) = -PtubeErrorAtmLoss(t, b, h[1], s, x, error); optimizeAndPri
                    
```

### Teil b

In dieser abschließenden Teilaufgabe wird der Einfluss der Spiegelposition  $x$  und der Tageszeit  $t$  auf das Ergebnis der Optimierung untersucht.



Variere die Position des Spiegels. Setze dazu die Position auf  $x = -5$  m und führe die Optimierung mit der Zielfunktion `PtubeErrorAtmLoss` erneut durch. Was beobachtest du? Notiere Ideen für mögliche Modellverbesserungen. Teste anschließend auch verschiedene Tageszeiten  $t$ .

**Schon fertig mit beiden Modellverbesserungen...?**

Dann kannst du das Modell und damit auch die Planung des Kraftwerks weiter verbessern, indem du auf den nächsten Arbeitsblättern verschiedene Erweiterungen bei der Optimierung der Höhe einbaust:

- Berücksichtigung **mehrerer Spiegel** und die Berechnung der **Energie über einen Tag** (Zusatzblatt 3).
- Der Einfluss stark **variierender Fehler** bei der Einstellung des Spiegel (Zusatzblatt 4).

## B.3.5 Zusatzblatt: Optimierung mit verbessertem Modell

### Optimierung der Rohrhöhe - Modellverbesserung 3 und 4

#### Mehrere Spiegel und Energie über einen Tag

 Notiere deine Ergebnisse der Optimierung auf diesem [Antwortblatt](#).

---

 Damit du dieses Arbeitsblatt bearbeiten kannst, musst du als erstes den folgenden Code ausführen. Klicke dazu in das folgende Code-Feld und drücke auf den "Run"-Button.

```
In [ ] :
# Hier nichts ändern!
include("../code/setup_Ab9_MS.jl");

# --- Festlegung einiger Parameter des Kraftwerks --- #
t = 11; # Uhrzeit
b = 0.8; # Breite des Sekundaerreflektors
s = 1.1; # Breite des Spiegels
```

### Optimierung der Höhe bei mehreren Spiegeln

Variert man die Position des Spiegels und berechnet erneut die optimale Höhe des Kollektors, so wird schnell deutlich, dass die optimale Lösung sich stark ändert. Es ist deswegen naheliegend, die Höhe nicht in Abhängigkeit der durch **alle Spiegel** generierten Leistung zu betrachten.

Wir werden die Optimierung nun unter Berücksichtigung aller Spiegel, die rechts und links von dem Rohr aufgestellt werden durchführen. Die Funktion `TubeError` für die Leistung am Rohr muss dazu zur Berechnung der Leistung erzeugt durch mehrere Spiegel erweitert werden.

Wir betrachten ein Kraftwerk mit 8 Spiegeln, die auf den folgenden Positionen stehen:  
`positions = [-5.5, -4, -2.5, -1, 1, 2.5, 4, 5.5]`

---

 Führe den folgenden Code aus. Der Strahlenverlauf bei mehreren Spiegeln wird dargestellt.

```
In [ ] :
# - Hier nichts ändern! -
positions = [-5.5,-4,-2.5,-1,1,2.5,4,5.5]; # Die Spiegelpositionen werden
plotMultiMirrors(positions,5,2,1.5,5)
```

### Teil a

Die Funktion `TubeErrorAtmLoss(t,b,h,s,x,error)` wurde zur Berechnung der Leistung generiert durch mehrere Spiegel erweitert. Dazu wird die Leistung, die durch die einzelnen Spiegel erzeugt wird, aufsummiert:

$$TubeErrorAtmLoss(t,b,h,s,x_1,error) + TubeErrorAtmLoss(t,b,h,s,x_2,error) + PtubeMultiMirrors$$

Die neue Funktion für die Leistung erzeugt durch alle Spiegel nennen wir `PtubeMultiMirrors`.

---

 Führe den Code aus. Die Funktion `PtubeMultiMirrors` zur Berechnung der durch alle Spiegel generierten Leistung wird gespeichert.

---

**Auf diesem Zusatzblatt kannst du die Erweiterung des Modells zur Berechnung der durch alle Spiegel generierten Leistung selbst implementieren.**

```
In [ ] :
positions = [-5.5, -4, -2.5, -1, 1, 2.5, 4, 5.5];
error = 0.5;

PtubeMultiMirrors(t,b,h,s,positions,error) = PtubeMultiMirrorsSol(t,b,h,s,positions,
printsave(PtubeMultiMirrors))
```

### Teil b

 Führe den nachfolgenden Code aus. Die optimale Rohrhöhe bei dem gegebenen Fehler wird mit dem Optimierungstool von Julia bestimmt. Die neue Funktion der Leistung am Rohr wird grafisch dargestellt. Notiere das Ergebnis auf dem Antwortblatt.

```
In [ ] :
error = 0.5;
xLeft = 0.0;
xRight = 500.0;

# Optimieren
zielFunktion(h) = -PtubeMultiMirrors(t,b,h[1],s,positions,error);
optimizeAndPrint(zielFunktion,xLeft,xRight,t,b,s,positions,0);

# Plotten
f(h) = PtubeMultiMirrors(t,b,h,s,positions,error);
plotTubeMulti(f,t,0,70)
```

---

Beim Einbauen der Modellverbesserung 2 haben wir festgestellt, dass auch die Wahl des Fehlers einen Einfluss auf die optimale Höhe hat. Eine Möglichkeit ist, verschiedene Fehler zuzulassen und in das Modell zu integrieren.

## Optimierung der Energie über einen Tag

Bisher haben wir stets nur **einen festen Zeitpunkt  $t$**  betrachtet. Variiert man die Uhrzeit, so erhält man jedoch deutlich andere Ergebnisse für die optimale Höhe. Zudem ist für den Betreiber eines Kraftwerks nicht nur interessant, wie viel Leistung zu einer speziellen Uhrzeit erzeugt wird, sondern vielmehr, wie viel **Energie über einen Tag**, eine Woche oder ein Jahr eingefangen wird.

### Teil c

Die Funktion `PtubeMultiMirrors` wurde zur Berechnung der Energie über einen Tag erweitert und unter dem Funktionsnamen `EtubeErrorMultiMirrorsOneDay` gespeichert.

**Auf diesem Zusatzblatt kannst du die Erweiterung des Modell zu Berechnung der erzeugten Energie über einen längeren Zeitraum (z. B. einen Tag oder ein Jahr) selbst entwickeln.**



*Führe den nachfolgenden Code aus. Die optimale Rohrhöhe mit der neuen Zielfunktion wird mit dem Optimierungstool von Julia bestimmt. Auch wird die neue Zielfunktion der Energie am Rohr über einen Tag grafisch dargestellt. Notiere das Ergebnis für die optimale Höhe auf dem Antwortblatt.*

```
In [ ] :
error = 0.5;
left = 0.0;
right = 50.0;

# Optimieren
Zielfunktion(h) = -EtubeErrorMultiMirrorsOneDay(b, h[1], s, positions, error, 0.05)
optimizeAndPrintEtube(Zielfunktion, left, right, t, b, s, positions, error);

# Plotten
f(h) = EtubeErrorMultiMirrorsOneDay(b, h, s, positions, error, 0.05); plotEtubeDay
```

### Fazit

Im Laufe dieses **2. Teil des Workshops** hast du dein Modell zur Berechnung der Leistung am Rohr nach und nach verbessert. Damit hast du auch die Zielfunktion, die die Leistung am Rohr beschreibt, Schritt für Schritt verändert. Dies hat zu unterschiedlichen optimalen Werten der Höhe  $h$  geführt.

### B.3.6 Infoblatt: Optimierung unter variablem Fehler

#### Infoblatt: Optimierung der Rohrhöhe unter variablem Einstellungsfehler



Damit du dieses Arbeitsblatt bearbeiten kannst, musst du als erstes den folgenden Code ausführen. Klicke dazu in das Code-Feld und drücke auf den "Run"-Button.

```
In [ ]:
# Hier nichts ändern
include("../code/setup_ab9_ms_jl");
# --- Festlegung einiger Parameter des Kraftwerks --- #
t = 11; # Uhrzeit
b = 0.8; # Breite des Sekundärreflektors
s = 1.1; # Breite des Spiegels
# Positionen
positions = [-4, 2, 0, 2, 4] # [-5.5, -4, -2.5, -1, 1, 2.5, 4, 5.5];
```

Ein und den selben Fehler für alle Spiegel festzulegen, entspricht nicht der Realität. Womöglich arbeiten einige Spiegelmotoren äußerst korrekt, andere hingegen mit großem Fehler. Deswegen werden wir nicht **einen festen Fehler für alle Spiegel** festlegen, sondern für jeden Spiegel einzeln einen zufälligen, individuellen Fehler bestimmen. Die Leistung am Rohr wird dann pro Spiegel mit dessen individuellen Fehlern berechnet.

Wir gehen aber noch einen Schritt weiter: Wir bestimmen die Leistung am Rohr für alle Spiegel und ihre individuellen Fehler nicht nur einmal, sondern wir wiederholen das Zufallsexperiment! Für jeden Spiegel bestimmen wir wieder einen zufälligen Fehler und berechnen die Leistung, die insgesamt von unseren Spiegel am Rohr ankommt erneut! Dies wird viele Male wiederholt und die Ergebnisse für die Leistung aufsummiert. Anschließend teilen wir durch die Anzahl der Wiederholungen. So erhalten wir die **gemittelte oder zu erwartende Leistung**.

Leistung Zufallsexperiment 1 + Leistung Zufallsexperiment 2 + ... + Leistung Zufallsexperiment M

Ziel ist es, die optimale Höhe der Funktion, die die **gemittelte Leistung** beschreibt, zu bestimmen. So möchte man sozusagen auf viele zu erwartende Fehlerszenarien gefasst sein.



Gib für **NAN** die Anzahl der Fehler ein, die pro Spiegel erzeugt werden sollen. Starte mit kleinen Werten, da die Optimierung für viele Fehler **seeeehr** lange dauern kann. Führe den Code aus. Die neue Zielfunktion mit variablem Fehler wird grafisch dargestellt.

```
In [ ]:
nErrorPerMirror = NaN; # 50; # Lege hier die Anzahl der Fehler fest, die
loops = size(positions, 1) * nErrorPerMirror;
```

```
Left = 0.0;
Right = 50.0;
# Plotten
f(h) = PtubeErrorExpValueMultiMirrors(t, b, h, s, positions, loops, 0.7)
plotPtubeMulti(f, t, 0.1, 50)
```



Variere den Wert der Variable **nErrorPerMirror** im oberen Codefeld. Beobachte, welchen Einfluss die Änderung auf den Graph der Funktion hat. Welchen Wert für **nErrorPerMirror** würdest du wählen, um an dem erhaltenen Graphen die optimale Rohrhöhe abzulesen? Begründe deine Antwort.



Führe den folgenden Code aus. Die optimale Rohrhöhe wird bestimmt. Vergleiche das erhaltene Ergebnis mit dem Ergebnis von Arbeitsblatt 7.

```
In [ ]:
# Optimierung unter Fehlereinfluss zu festem Zeitpunkt t
Zielfunktion(h) = -PtubeErrorExpValueMultiMirrors(t, b, h[1], s, positions, 1)
optimize(Zielfunktion, 0.2, 0)
```

## B.4 Antwort- und Dokumentationsblätter

### B.4.1 Dokumentationsblatt zu Arbeitsblättern 1 – 4



**Dokumentationsblatt: Modellierung eines Solarkraftwerks**

Mithilfe dieses Arbeitsblatts sollst du die Formeln und Gleichung, die du am Computer eingibst, übersichtlich dokumentieren. Dies soll dir helfen, den Überblick zu behalten.  
 In der Spalte **Modellannahmen** kannst du alle **Annahmen und Vereinfachungen** notieren, die wir treffen. Die Spalte **Modellverbesserungsideen** kannst du nutzen, um im Laufe des Workshops Ideen zu notieren, die dir zur Verbesserung des Modells einfallen.

		Berechnete Größe	Formel / Funktion	Modellannahmen	Modellverbesserungsidee
AB 1	Teil c	Neigungswinkel $\gamma$			
AB 2	Teil b	Länge der Hilfsstrecke $e$			
	Teil c	Leistung am Spiegel $P_{\text{Mirror}}$			
AB 3	Teil b	Anteil der Leistung die das Rohr erreicht $\text{ratio}$			
	Teil c	Leistung am Rohr $P_{\text{Tube}}$			
AB 4	Teil a	Ausfallwinkel $\beta$	$\beta = \begin{cases} \text{für } x < 0 \\ \text{für } x = 0 \\ \text{für } x > 0 \end{cases}$		
	Teil b	Neigungswinkel $\gamma$			

		Länge der Hilfsstrecke $e$ <small>(wie im Modell oben!)</small>			
		Leistung am Spiegel $P_{\text{Mirror}}$ <small>(wie im Modell oben!)</small>			
	Teil d	Hilfslänge $e'$			
	Teil e	Anteil der Leistung die das Rohr erreicht			
	Teil f	Leistung am Rohr $P_{\text{Tube}}$			
Zusatzblatt 1		Leistung am Rohr $P_{\text{Tube}}$ erzeugt durch mehrere Spiegel			
Zusatzblatt 2		Gesamtenergie über einen Tag			

### B.4.2 Dokumentationsblatt zu Arbeitsblättern 6 – 7



**Dokumentationsblatt: Optimierung der Höhe des Absorberrohrs**

Mithilfe dieses Arbeitsblatts sollst du die Formeln und Gleichung, die du am Computer eingibst, übersichtlich dokumentieren. Dies soll dir helfen, den Überblick zu behalten.  
 In der Spalte **Modellannahmen** kannst du alle **Annahmen und Vereinfachungen** notieren, die wir treffen. Die Spalte **Modellverbesserungsideen** kannst du nutzen, um im Laufe des Workshops Ideen zu notieren, die dir zur Verbesserung des Modells einfallen.

		Berechnete Größe	Formel	Modellannahmen	Modellverbesserungsideen
<b>AB 6</b>	<b>Teil a</b>	Abstand Spiegelmitte zu Rohrmitte <i>distance</i>			
		Atmosphärische Abschwächung <i>atmLoss</i>			
	<b>Teil b</b>	Leistung am Rohr unter atm. Abschwächung			
<b>AB 7</b>	<b>Teil b</b>	Fehlerhafter Neigungswinkel $\gamma'$			
		Fehlerhafter Ausfallswinkel $\beta'$			
	<b>Teil c</b>	Fehlerstrecke $f$			
	<b>Teil d</b>	Länge der bestrahlten Strecke auf Rohrhöhe $p$			
	<b>Teil e</b>	Leistung am Rohr unter Fehlerinfluss			

### B.4.3 Antwortblätter zu Arbeitsblättern 5 – 8



**Antwortblatt 1: Optimierung der Höhe des Absorberrohrs**

- Spiegelbreite  $s = 1.1$  m
- Breite des Sekundärreflektors  $b = 0.8$  m

	Modell (Name der Zielfunktion)	Position $x$ [m]	Zeit $t$	error	$h_{opt}$ [m]	$P_{tube}$ [Watt]	Beobachtung / Erkenntnis	Verbesserungsidee
<b>AB 5</b>	<b>Ptube</b>	-2	11	■	grafisch (Teil a)			
					Optimierungstool (Teil c)			
<b>AB 6</b> Modellverbesserung 1	<b>Ptube</b> <b>AtmLoss</b>	-2	11	■	grafisch (Teil c)			
					Optimierungstool (Teil d)			
		-5	11	■	Optimierungstool (Teil d)			
<b>AB 7</b> Modellverbesserung 2	<b>PtubeError</b>	-2	11	0.5	Optimierungstool (Teil g)			
		-5	11	0.5	Optimierungstool (Teil g)			

Platz für Skizzen:



Antwortblatt 2: Optimierung der Höhe des Absorberrohrs

	Modell (Name der Zielfunktion)	Position x [m]	Zeit t	error	$h_{opt}$ [m]	$P_{tube}$ [Watt]	Beobachtung / Erkenntnis	Verbesserungsidee	
<b>AB 8</b> Kombination: Modell- verbesserung 1 & 2	PtubeError AtmLoss	-2	11	0					
		-2	11	0.1					
		-2	11	0.2					
		-2	11	0.5					
		-5	11	0.5					
		-2	13	0.5					
		-2	16	0.5					
		(weitere Experimente)							



Antwortblatt 3: Optimierung der Höhe des Absorberrohrs

	Modell (Name der Zielfunktion)	Position x [m]	Zeit t	error	$h_{opt}$ [m]	$P_{tube}$ [Watt]	Beobachtung / Erkenntnis	Verbesserungsidee
<b>Zusatzblatt 3</b> Modell verbesserung 3 & 4	Ptube MultiMirrors	-5.5, -4, -2.5, -1, 1, 2.5, 4, 5.5	11	0.5				
		-5.5, -4, -2.5, -1, 1, 2.5, 4, 5.5 <i>(mögliche weitere Experimente)</i>						
	EtubeErrorMulti MirrorsOneDay	-5.5, -4, -2.5, -1, 1, 2.5, 4, 5.5	■	0.5				
		-5.5, -4, -2.5, -1, 1, 2.5, 4, 5.5 <i>(mögliche weitere Experimente)</i>	■					
<b>Zusatz</b> (Platz für weitere mögliche Ergebnisse)								

## B.5 Präsentationsfolien

### B.5.1 Folien für den Einstiegsvortrag



Quelle: ebl

**Wie funktioniert ein Solarkraftwerk...  
und was hat das mit Mathe zu tun?**

Einführung in die Problemstellung



Warum ist die Forschung zur Verbesserung erneuerbarer Energien so wichtig?




Workshop | Solarenergie | 3/14

Wie kann man durch Sonnenstrahlung Energie gewinnen?




Workshop | Solarenergie | 5/14

#### Intensive Forschung zu erneuerbaren Energien

##### In der Wirtschaft

- Vattenfall
- ebl
- Flagsol
- ... u.v.m ...

##### In der Wissenschaft

- Karlsruher Institut für Technologie
  - Deutsches Zentrum für Luft- und Raumfahrt
  - RWTH Aachen
  - Universidad de Sevilla
  - ... u.v.m ...
- 
- Workshop | Solarenergie | 2/14

#### Warum ist die Forschung zur Verbesserung erneuerbarer Energien so wichtig?

- saubere Alternative zu herkömmlichen Energiequellen
- hoher CO<sub>2</sub>-Ausstoß in Kohlekraftwerken
- Klimawandel
- Nachhaltigkeit
- Kosten erneuerbarer Energiequellen senken
- Erneuerbare Energiequellen effizienter gestalten
- ...

**Forschung auf dem Gebiet der erneuerbaren Energie ist wichtig für heutige und zukünftige Generationen!**



Workshop | Solarenergie | 4/14

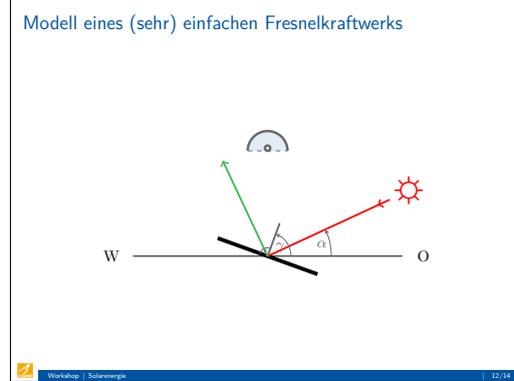
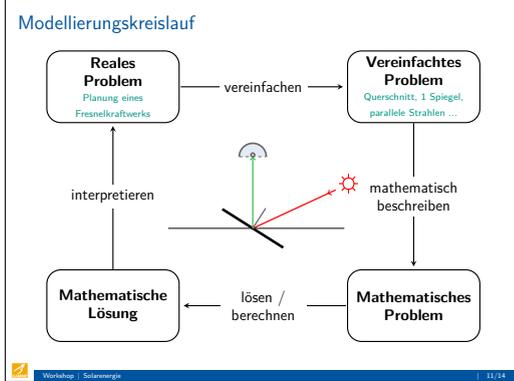
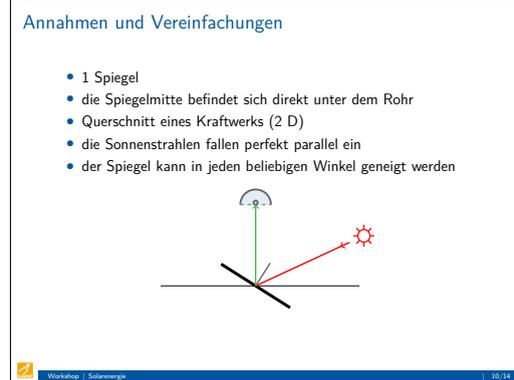
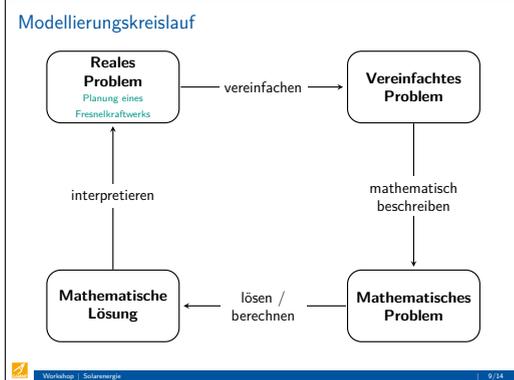
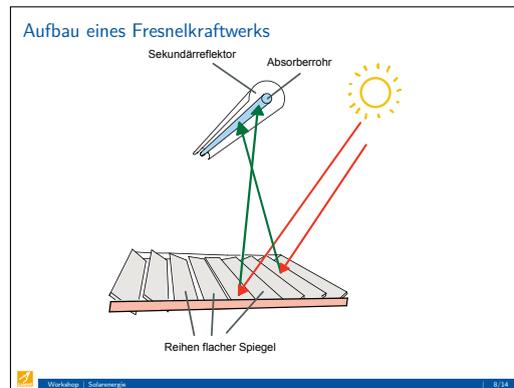
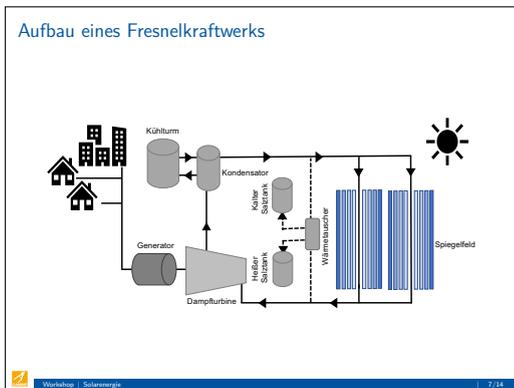


[www.youtube.com/watch?v=7P7p081aE](http://www.youtube.com/watch?v=7P7p081aE)

[www.youtube.com/watch?v=2t5Xj806e4&list=PL108](http://www.youtube.com/watch?v=2t5Xj806e4&list=PL108)



Workshop | Solarenergie | 6/14



### Fragen

- Wie können die Sonnenstrahlen optimal aufs Rohr reflektiert werden?
- Wie viel Leistung erreicht den Spiegel?
- Wie viel Leistung erreicht das Rohr?
- Wie viel Energie wird über einen Tag umgesetzt?
- Wie sollten verschiedene Kraftwerkparameter gewählt werden, um den Energieumsatz zu maximieren?

### Wie findet ihr Antworten auf alle diese Fragen?

**DON'T PANIC!**

**ORGANIZE!**

- Bearbeit die Arbeitsblätter!
- Aufgabenstellung sorgfältig lesen!
- Teamwork!
- Hilfekarten!
- Nutzt das Internet!
- Fragt die Betreuer/innen!
- Helft euch gegenseitig!

## B.5.2 Folien für die Zwischen- und Abschlussdiskussionen

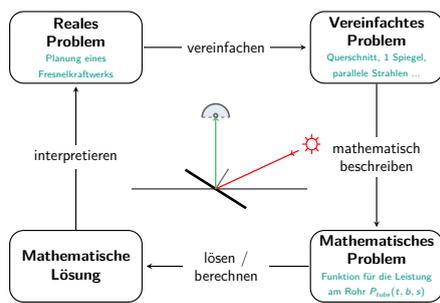


Wie funktioniert ein Solarkraftwerk... und was hat das mit Mathe zu tun?

Diskussion

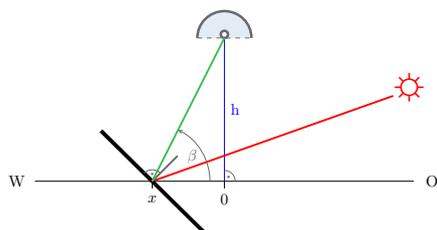


### Modellierungskreislauf



Workshop | Solarenergie | 3/15

### Horizontal verschobener Spiegel



Workshop | Solarenergie | 5/15

### Optimierung

Bei der Planung eines Kraftwerks sollten verschiedene Parameter noch vor der Konstruktion optimal gewählt werden!

- Höhe des Absorberrohrs
- Position des Spiegels
- Form des Sekundärreflektors
- Breite des Spiegels
- ...

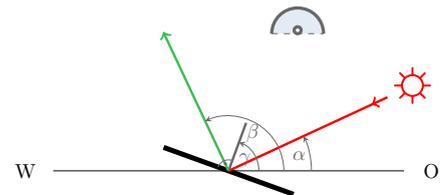
Workshop | Solarenergie | 7/15

### Zusammenfassung

## Diskussion zu AB 1 - 3

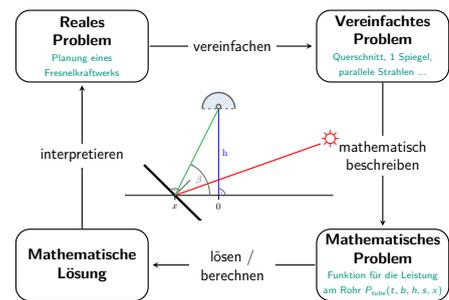
Workshop | Solarenergie | 9/15

### Horizontal verschobener Spiegel



Workshop | Solarenergie | 4/15

### Modellierungskreislauf



Workshop | Solarenergie | 6/15

### Ihr seid dran!

- Öffne Arbeitsblatt **AB 4 short!**
- Lies den ersten Abschnitt des Arbeitsblatts und starte mit der Optimierung auf AB 5.
- Notiere deine Ergebnisse auf dem **Antwortblatt**.

Antwortblatt 1: Optimierung der Höhe des Absorberrohrs

• Optimierung  $x = 1.1$   
 • Zielwert des Gütefunktionswertes  $= 0.91$

Aktivität 1	Modell	Position	Zeit	error	kurz-PM		lang-PM		Skalierung	Iterationsanzahl	Optimierungsergebnis
					Startzeit	Zielf.	Startzeit	Zielf.			
Modell verfeinern 1	Fläche	-2	11	*							
	Fläche Absorber	-2	11	*							
Modell verfeinern 2	Fläche	-6	11	*							
	Fläche Absorber	-6	11	0.8							
Modell verfeinern 3	Fläche	-2	11	0.8							
	Fläche Absorber	-6	11	0.8							

Workshop | Solarenergie | 8/15

Wie findet ihr Antworten auf alle diese Fragen?



- Bearbeite die Arbeitsblätter!
- Arbeite im Team mit deinem/r Partner/in!
- Nutze die Tippkarten!
- Frag' die Betreuer/innen!
- Nutze das Internet!
- Helft euch gegenseitig!

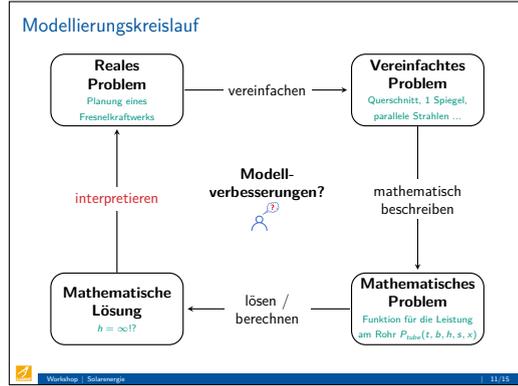
Workshop | Solarenergie | 9/15

Diskussion nach AB 5 | Die optimale Rohrhöhe



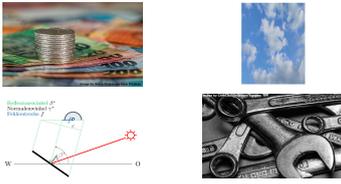
- Wie lautet die optimale Rohrhöhe  $h$ ?
- Wie lässt sich das Ergebnis anhand des Funktionsgraphen erklären?

Workshop | Solarenergie | 10/15



Diskussion - Modellverbesserungen

- Materialkosten
- atmosphärische Effekte
- Unsicherheiten / Fehler
- Wartungsaufwand / Reinigung
- ...



Workshop | Solarenergie | 12/15

Ihr seid dran!

- Für Modellverbesserung **atmosphärische Effekte** öffne **AB 6**
- Für Modellverbesserung **Unsicherheiten** öffne **AB 7**



- Bearbeite die Arbeitsblätter!
- Arbeite im Team mit deinem/r Partner/in!
- Nutze die Tippkarten!
- Frag' die Betreuer/innen!
- Nutze das Internet!
- Helft euch gegenseitig!

Workshop | Solarenergie | 13/15

Diskussion nach AB 6 - 8 | Die optimale Rohrhöhe

- Welche Erkenntnisse habt ihr während der Optimierung der Rohrhöhe gemacht?
- Wie hat sich der Graph der Funktion  $P_{tube}$ , die die Leistung beschreibt, geändert?
- Welchen Einfluss hat die Variation der Uhrzeit oder der Spiegelposition auf die optimale Rohrhöhe?
- Wie hoch würdet ihr das Rohr nun bauen? Warum?
- Habt ihr Ideen für weitere Modellverbesserungen?

Workshop | Solarenergie | 14/15

Ein Blick in die Forschung



- Weitere Forschungsfragen ...
  - Bau von Solarturmkraftwerken
  - optimale Verkabelung der Spiegel
  - optimale Reinigungsrouten
  - optimale Form der Spiegel (flache Spiegel oder Parabolspiegel?)
  - Wettermodelle für die Simulation von Leistung und Energieumsatz

Workshop | Solarenergie | 15/15

# C Lernmaterial zum Thema Solarenergie – Workshop II

Die Arbeitsblätter (Jupyter Notebooks) sind mit eingetragenen Lösungen dargestellt. Die Kommentare `#= NaN; =#` vor den Lösungen deuten an, dass in der Schülerversion an dieser Stelle der Platzhalter `NaN` steht. Dieser wäre dann durch eine Formel, einen Wert oder ähnliches zu ersetzen.

## C.1 Ablauf des Workshops

In Abbildung C.1 ist ein möglicher Ablauf des Workshops für die Oberstufe dargestellt. Die Nummerierung in den Konzepten A und B gibt die Reihenfolge an, in denen die Arbeitsblätter bearbeitet werden können.

Es ist nicht notwendig alle Arbeitsblätter zu bearbeiten, um den Workshop inhaltlich kohärent abzuschließen. Besteht wenig(er) Zeit für die Durchführung, so kann der Workshop auch bereits nach früheren Arbeitsblättern beendet werden, ohne den Modellierungsprozess für die Lernenden unvollständig erscheinen zu lassen:

### Kurzversionen des Workshops für die Oberstufe

Folgende Kurzversionen des Workshops für die Oberstufe sind denkbar:

- **Kurzversion 1:** Der Workshop kann abgeschlossen werden, nachdem die Lernenden **einen beliebigen** Algorithmus von Arbeitsblatt 4 bearbeitet haben. Die Lernenden haben bei dieser Kurzversion ein Verfahren zur Optimierung kennengelernt. Das Projekt kann mit einer Diskussion zu Ideen für die Erweiterung bzw. Verbesserung des Modells (z. B. der Schattenwurf zwischen Spiegeln) abgeschlossen werden. Diese Verbesserungen werden dann jedoch nicht mehr umgesetzt (AB 1 – 3 und **ein** Algorithmus von AB 4).
- **Kurzversion 2:** Analog zur Kurversion 1 kann der Workshop nach **zwei** bis **vier** Algorithmen von Arbeitsblatt 4 abgeschlossen werden.

Oberstufe: Spiegelausrichtung und Optimierung der Spiegelpositionen

	Arbeitsblatt	Inhalt	A	B	Schulbezug	Optimierung	
Baustein 1: Modellierung der Energie am Rohr	AB 1	Ausrichtung eines Spiegels direkt unter dem Rohr	1	1	Winkelbeziehungen		
	AB 2	Energie über Zeitraum	2	4	Riemann-Summen		
	Zusatzblatt 1	Modellierung der Leistung am Rohr (s. AB 2-3 im Workshop der Mittelstufe)		2	Trigonometrie		
	Zusatzblatt 2	Verschiebung des Spiegels (s. AB 4 Workshop Mittelstufe)		3	Trigonometrie		
<b>Diskussion im Plenum (Präsentationsfolien)</b>							
Baustein 2: Optimierung der Spiegelpositionen	AB 3	Optimierung: 1 Spiegelposition (grafisch)	3	5	Diskussion von Funktionsgraphen (erst $\mathbb{R}^2$ dann $\mathbb{R}^3$ )	Grafisch	
		Modellerweiterung: Energie mehrere Spiegel					
		Optimierung: 2 Spiegelpositionen (grafisch)					
		<b>Diskussion im Plenum</b>					
		Nebenbedingung für Mindestabstand und Optimierung von 2 Spiegelpositionen durch systematische Variation			Ungleichung	Systematische Variation	
		Optimierung automatisieren: kreative Entwicklung eines Verfahrens mit Stift und Papier			Algorithmisch Denken	Beliebiges Optimierungsverfahren	
	<b>Diskussion im Plenum</b>						
	Alg 1	Brute-Force / kombinatorisch	4	6	Kombinatorik, Algorithmisch Denken, Ungleichungen	(a) Existenz & Eindeutigkeit von Lösungen, (b) Modellieren von Nebenbed. bzw. der zulässigen Menge (c) Rechenzeit vs. Güte der Lösung	
	Alg 2	Greedy			Algorithmisch Denken		
	Alg 3	Musterbasiert (Spiegelabstand)					
Alg 4	Freie Variablen Optimierung	Ungleichung, lokale und globale Extrema					
Infoblatt	Formelbasierte Optimierung						
<b>Diskussion im Plenum (Präsentationsfolien)</b>							
AB 5 <sup>2</sup>	Effekte zwischen Spiegeln: Modellierung von Verschattung	5	7	Geraden in Parameterform, Schnittpunkte von Geraden, Diskussion von Funktionsgraphen (im $\mathbb{R}^3$ )	Grafisch		
AB 6	Optimierung mit neuem Modell von AB 5 durch Anwendung von Alg 1 - 4	6	8		s. o. (a) – (c)		
<b>Diskussion im Plenum (Präsentationsfolien)</b>							

<sup>2</sup> Alternativ kann die Modellierung ab hier unangeleitet durch die Lernenden durchgeführt werden.

Abb. C.1: Übersicht über Inhalte und Arbeitsblätter des Workshops für die Oberstufe



**Teil c**

Die Spiegelausrichtung soll automatisch erfolgen. Dazu beschreiben wir den korrekten Neigungswinkel des Spiegels abhängig vom Sonneneinfallswinkel. In der Formel für den Neigungswinkel  $\gamma$  können wir die Funktion `alpha(t)`, mit der der Sonneneinfallswinkel zu einer gewählten Uhrzeit  $t$  bestimmt wird, verwenden.



Stelle eine Formel für den gesuchten Neigungswinkel  $\gamma$  in Abhängigkeit von  $\alpha$  (Eingabe als `alpha(t)`) auf. Nimm deine angefertigte Zeichnung zur Hilfe. Gib deine Formel für das erste `NaN` ein. Wähle dann eine beliebige Tageszeit  $t$  zwischen 0 und 24 und gib sie für das zweite `NaN` ein. Führe den Code aus, indem du auf "Run" klickst. Dir wird für die gewählte Zeit  $t$  und mit deiner Formel für  $\gamma$  der Strahlenverlauf angezeigt. Wenn du alles richtig gemacht hast, fällt der reflektierte Sonnenstrahl genau auf das Absorberrohr - und zwar unabhängig von deiner Wahl für  $\alpha$ .

Falls du Hilfe benötigst, kannst du dir hier einen **Tipp** ansehen.

```
In [ ]:
# Berechnung von Gamma
gamma(t) = NaN; # (alpha(t) + 90) / 2 # Ersetze NaN durch eine Formel, die
# Tageszeit
t = NaN; # Ersetze NaN durch eine beliebige Zeit zwischen 0 und 24
# Hier nichts ändern
checkReflection(t, alpha, gamma)
```

**Fazit**

Der Spiegel wird nun automatisch so ausgerichtet, dass der Sonnenstrahl, der von der Mitte des Spiegels reflektiert wird, jederzeit direkt auf das Rohr fällt. Die Annahme, dass alle Spiegel sich direkt senkrecht unter dem Absorberrohr befinden, entspricht jedoch nicht der Realität. In Wirklichkeit befinden sich viele Spiegel schräg links und rechts unter dem Absorberrohr. Diese Anordnung ermöglicht es, mehr als nur eine Spiegereihe zur Reflexion der Sonnenstrahlen zu nutzen und auf diese Weise die Stromerzeugung zu maximieren.

**Modellerweiterung: horizontal verschobene Spiegel**

Das unten stehende Bild stellt schematisch die Situation mit **einem horizontal verschobenen Spiegel** dar. Die neue Variable `x` gibt die **Position des Spiegelmittelpunktes** relativ zum Nullpunkt an. Der Nullpunkt befindet sich senkrecht unter dem Rohr und ist eingezeichnet. Dementsprechend ergibt eine Verschiebung des Spiegels nach **links negative Werte** für `x` und eine Verschiebung des Spiegels nach **rechts positive Werte** für `x`. In der Abbildung ist zudem die **Höhe des Absorberrohrs** `h` über dem Boden eingezeichnet. Diese muss bei der Verbesserung des Modells für verschobene Spiegel ebenfalls berücksichtigt werden.

Der Verlauf der Strahlen wird zum gewählten Zeitpunkt und basierend auf der von dir gewählten Neigung des Spiegels angezeigt.  
**Stelle den Neigungswinkel  $\gamma$  so ein, dass der reflektierte Strahl auf das Rohr trifft.**

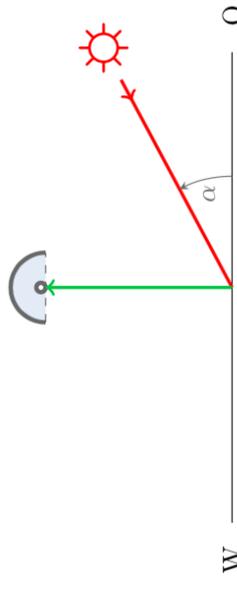
```
In [ ]:
# Hier nichts ändern!
include("../code/setup_AB1.jl");
# Tageszeit
t = NaN; # Ersetze NaN durch eine beliebige Zeit in Stunden nach Tag
# Neigungswinkel
gamma = NaN; # Ersetze NaN durch einen beliebigen Winkel im Gradmaß
# Hier nichts ändern!
alpha = alpha(t); plotSimpleFresnel(t, gamma, alpha)
```

Vermutlich hast du den Neigungswinkel  $\gamma$  in der Grafik durch Ausprobieren so wählen können, dass der reflektierte Strahl das Rohr trifft. Wählen wir in Teil a jedoch eine andere Tageszeit und damit einen anderen Einfallswinkel  $\alpha$ , so muss auch der Neigungswinkel  $\gamma$  wieder angepasst werden, damit das Licht wie gewünscht reflektiert wird. Ziel ist es, eine **Formel** zur Berechnung des korrekten Neigungswinkels  $\gamma$  aufzustellen, **die vom aktuellen Einfallswinkel der Sonnenstrahlen  $\alpha$  abhängt**.

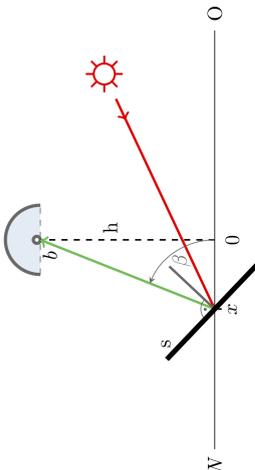
Die Skizze in **Teil b** soll dir beim Aufstellen der Formel für  $\gamma$  helfen!

**Teil b**

Die untere Abbildung zeigt, wie der reflektierte Strahl zum einfallenden Strahl verlaufen muss, damit er genau auf das Absorberrohr trifft. In dieser Abbildung fehlt jedoch der Spiegel.



Übertrage die letzte Zeichnung auf ein Blatt. Ergänze deine Zeichnung um den **Spiegel** und die **Senkrechte auf den Spiegel**, sodass eine **physikalisch korrekte Reflexion** entsteht. Zeichne auch den Neigungswinkel  $\gamma$  ein.



Um unser Modell besser an die Realität anzupassen, wurde es um eine mögliche Verschiebung des Spiegels erweitert. Diese Erweiterung braucht ihr nicht eigenständig vornehmen.

---

Öffne nun das nächste [Arbeitsblatt](#).

## C.3 Arbeitsblatt 2: Energie über einen Tag

### Solkraftwerk – AB 2: Die Energie des Kraftwerks

#### Die Leistung des Kraftwerks

Zu den wichtigsten Kennzahlen von Kraftwerken gehört die **Leistung  $P$**  (engl. *power*). Sie bezeichnet die in einer bestimmten Zeitspanne umgesetzte **Energie  $\Delta E$**  bezogen auf die **Zeitspanne**:

$$P = \frac{\Delta E}{\Delta t}$$

Die Einheit der Leistung ist **Watt**. Das Symbol  $\Delta$  (= Delta) steht für die Differenz zwischen zwei Werten. In unserem Fall steht  $\Delta t$  für die Differenz zwischen zwei Zeitpunkten  $t_1$  und  $t_2$ . Zum Beispiel:  $\Delta t = t_2 - t_1 = 10.50 - 10.00 = 0.5$ .

Basierend auf unseren Modellen für die Berechnung des Neigungswinkels  $\gamma$  und unter Verwendung von realen Daten zur Stärke der Sonnenstrahlung in Daggett wurde die Funktion  $P_{\text{tube}}$  mit  $P_{\text{tube}}(t, b, h, s, x)$  entwickelt. Diese gibt an, wie viel Leistung dem Absorberrohr (engl. *tube*) durch einen Spiegel zugeführt wird. Die Leistung hängt von der Tageszeit  $t$ , der Reflektorbreite  $b$ , der Absorberrohrhöhe  $h$ , der Spiegelbreite  $s$  und der Spiegelposition  $x$  ab.

#### Teil a



Führe den folgenden Code aus. Der Graph der Funktion  $P_{\text{tube}}$  wird im Tagesverlauf dargestellt. Für alle anderen Parameter wurden konstante Werte festgelegt ( $b = 0.8\text{m}$ ,  $h = 6\text{m}$ ,  $s = 1.1\text{m}$  und  $x = -2\text{m}$ ). Da wir das Kraftwerk lediglich im Querschnitt betrachten, vernachlässigen wir die Tiefe der Spiegel und des Rohres. Die Funktion  $P_{\text{tube}}$  dient als Ausgangspunkt für die anstehende Optimierung.

```
In [ ] :
# Hier nichts ändern!
include("../code/setup_numintegration.jl");

# --- Festlegung einiger Parameter des Kraftwerks --- #
b = 0.8; # Breite des Sekundaerreflektors
s = 1.1; # Breite des Spiegels
h = 6;   # Höhe des Absorberrohrs
x = -2; # Position des Spiegels

plotPtube(ptube, b, h, s, x)
```

#### Berechnung der umgesetzten Energie

Mithilfe der (momentanen) Leistung lässt sich berechnen, wie viel Energie ein Kraftwerk über einen bestimmten Zeitraum (bspw. einen Tag, einen Monat oder ein Jahr) umsetzt. Solche

Werte sind von großem Interesse. Ziel dieses Arbeitsblattes ist es, die **über einen Tag umgesetzte Energie** näherungsweise zu berechnen.



Berechne die über einen Tag umgesetzte Energie. Die Energie soll in der Einheit Wattstunden (=  $\text{Wh}$  oder **kurz Wh**) angegeben werden. Gib deine Formel im Code ein. Du kennst die Funktion  $P_{\text{tube}}(t, b, h, s, x)$  verwenden. Diese beschreibt die Leistung am Rohr.



- Nutze bei deiner Eingabe im Code eine **for-Schleife**. Falls die Verwendung von **for-Schleifen** neu für dich ist, findest du **hier** ein Beispiel.
- Falls du Schwierigkeiten beim Lösen dieser Aufgabe hast, findest du hier einen **Tipp**.

```
In [ ] :
## Hinweis: Du kannst die Funktion Ptube(t,b,h,s,x) verwenden.
function Etube(b,h,s,x)
    Energy = 0; # Am Anfang des Tages ist die gesamte Energie noch 0
    Deltat = NaN; # 0.1; # Wähle einen Wert für die Länge der Zeitabschn
    ### Hier beginnt die for - Schleife
    for t = NaN : NaN : NaN; # 0 : Deltat : 24 # Berechne mit diesem for
        Energy = Energy + Ptube(t, b, h, s, x) * Deltat # Gib hi
    end
    return Energy
end
# Hier nichts ändern!
totalEnergy = Etube(b,h,s,x)
checkEnergyOneDay(totalEnergy, b, h, s, x)
```

#### Teil b



Ordnung die über einen Tag umgesetzte Energie unseres Kraftwerks (bestehend aus einem Spiegel) mit Blick auf die Energieversorgung ein. Ist es viel oder wenig Energie? Recherchiere im Internet, welchen täglichen Bedarf an elektrischer Energie ein Privathaushalt mit vier Personen in Deutschland durchschnittlich hat, um einen Anhaltspunkt zu erhalten.

Berücksichtige bei deiner Bewertung, dass nicht die gesamte Strahlungsenergie in elektrische Energie umgesetzt wird. Stattdessen muss der sogenannte Wirkungsgrad berücksichtigt werden. Dieser beschreibt die Effizienz einer technischen Anlage und gibt in unserem Fall an, wieviel Prozent der empfangenen Strahlungsenergie tatsächlich in elektrische Energie überführt wird. Der Wirkungsgrad liegt bei Fresnelkraftwerken je nach Standort und abhängig von den technischen Bauteilen bei circa 10-20%.

---

### Fazit

Wir haben die Energie berechnet, die innerhalb eines Tages von einem Spiegel umgesetzt wird. Dabei haben wir für verschiedene Kraftwerkparameter ( $t$ ,  $h$ ,  $s$  und  $x$ ) konkrete Werte festgelegt. Bei der Planung eines Kraftwerks sollten solche Parameter variiert und bestmöglich gewählt werden. Ziel dieses Workshops ist es, die Positionen mehrerer nebeneinander angeordneter Spiegel zu optimieren.

---



### Diskussion

Bevor es mit der **Optimierung** weitergeht, werden wir die bisherigen Ergebnisse im Plenum diskutieren.

Bis zur Diskussion kamst du

- die Ausrichtung von horizontalen **verschobene Spiegel** auf diesem [Zusatzblatt](#) modellieren oder
  - die Modellierung der Leistung am Absorberrohr auf diesem [Zusatzblatt](#) eigenständig durchführen.
- 

Um mit der Optimierung zu beginnen, öffne dieses [Arbeitsblatt](#).

## C.4 Arbeitsblatt 3: Optimierung der Spiegelpositionen

### Solkraftwerk – AB 3: Optimierung der Spiegelpositionen



Notiere deine Ergebnisse während der Optimierung auf diesem [Antwortblatt](#).

Eine wesentliche Frage bei der Planung eines Kraftwerks ist die **Positionierung der Spiegel**. Diese sollen so aufgestellt werden, dass das Kraftwerk besonders wirtschaftlich arbeitet, also die Energieausbeute bei möglichst geringen Kosten maximiert wird. In diesem Workshop werden wir die **Energiemaximierung** diskutieren. Wirtschaftlichkeitsfaktoren werden wir nicht berücksichtigen.

Ziel ist es, die Spiegel so aufzustellen, dass **maximal** viel Energie am Rohr ankommt. Mathematisch gesprochen, ist ein **Optimierungsproblem** zu lösen. Die Funktion, deren optimaler Wert gesucht wird, heißt **Zielfunktion**.



Öffne dieses [Infoblatt](#) für mehr Informationen rund um Optimierungsprobleme.

#### Teil a - Optimale Position von einem Spiegel – graphisch

Wie in der Modellierung üblich starten wir mit einer vereinfachten Situation: Wir betrachten nur **einen Spiegel** und werden die Position dieses Spiegels optimieren. Als Zielfunktion nutzen wir die Funktion  $E_{\text{tube}}$ , welche die über einen Tag am Rohr umgesetzte Energie beschreibt. Anschließend werden wir das Modell auf **mehrere Spiegel** erweitern.



Führe den Code aus. Für die Parameter  $b$ ,  $h$  und  $s$  werden konstante Werte gespeichert. Die über einen Tag umgesetzte Energie wird in Abhängigkeit von der Spiegelposition  $x$  graphisch dargestellt. Lies die optimale Position  $x$  am Graphen ab. Ist das Ergebnis plausibel? Erkläre die optimale Lösung im Hinblick auf die reale Situation.

**Hinweis:** Falls der Graph nicht direkt erscheint, führe den Code ein zweites Mal aus.

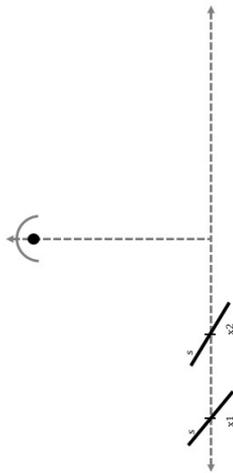
```
In [ ] :
# Hier nichts ändern!
include("../code/setup_graphoptimization.jl");

# --- Festlegung einiger Parameter des Kraftwerks --- #
b = 0.8; # Breite des Sekundaerreflektors
s = 1.1; # Breite des Spiegels
h = 6; # Höhe des Absorberrohrs
xleft, xright = [-10.0, 10.0]; # Linke und rechte Grenze des Kraftwerkareals
```

```
# Hier nichts ändern! Plotten
f(x) = Etube(b,h,s,x); plotPos(Etube(f,-10,10))
```

#### Teil b - Modellerweiterung

Bisher wurde die umgesetzte Energie lediglich für **einen** Spiegel auf Position  $x$  berechnet. Das Modell soll auf **mehrere** Spiegel erweitert werden. Wir betrachten zunächst zwei Spiegel und definieren deren Lage über die Variablen  $x_1$  bzw.  $x_2$ , welche die Positionen der Spiegelmittelpunkte beschreiben (siehe Skizze).



Mit  $E_{\text{tube}}(b, h, s, x)$  können wir die Energie berechnen, die von einem Spiegel auf Position  $x$  über den Tag umgesetzt wird. Wie viel Energie wird von **zwei Spiegeln** auf den Positionen  $x_1$  und  $x_2$  umgesetzt? Stelle eine Formel für die Berechnung dieser Energie in Abhängigkeit von  $x_1$  und  $x_2$  auf. Welche Vereinfachungen oder Annahmen triffst du? Gib deine Formel für `NaI` ein. Du kannst die Funktion  $E_{\text{tube}}(b, h, s, x)$  bei der Eingabe verwenden.

```
In [ ] :
# Funktion zur Berechnung der Energie von 2 Spiegeln
Etube2Mirrors(b, h, s, x1, x2) = # NaI; ==# Etube(b,h,s,x1) + Etube(b, h, s, x2); #
# Hier nichts ändern! Checken
checkKtotal(Etube2Mirrors, Etube, b, h, s)
```

Das Modell für die Energie über einen Tag haben wir für 2 Spiegel erweitert. Genauso kann das Modell für 3 oder mehr Spiegel erweitert werden. Dies brauchst du nicht selbst implementieren. Stattdessen haben wir die Funktion  $E_{\text{tube}}$  bereits so gespeichert, dass damit für beliebig viele Spiegel die Energie berechnet werden kann.

Im Code schreiben wir am Beispiel von 5 Spiegeln: `Etube(b, h, s, [x1, x2, x3, x4, x5])`.

### Teil c - Optimale Positionen von 2 Spiegeln - graphisch

Auch im Falle von zwei Spiegeln kann die Optimierung durch Ablesen am nun dreidimensionalen Funktionsgraphen erfolgen.



- Führe den folgenden Code aus. Der Graph der Funktion `Etube` wird in Abhängigkeit von den Spiegelpositionen `x1` und `x2` dargestellt.
- Lies die optimalen Spiegelpositionen ab.
- Diskutiere das erhaltene Ergebnis auch im Hinblick auf die Konstruktion eines realen Kraftwerks. Notiere Ideen, wie das Modell besser an die reale Situation angepasst werden kann.

```
In [ ]: # Hier nichts ändern! Plotten
f(x1,x2) = Etube(b, h, s, [x1, x2]); plotPosEtube(bew(f, -2, 2))
```



Gib für die beiden NAs die Positionen `x1` und `x2` ein, die gemäß des Graphen aus Teil c die maximale Energie liefern. Dir wird der Strahlenverlauf mit der eingegebenen Lösung angezeigt.

```
In [ ]: # Darstellung des Spiegels und der Sonnenstrahlen
x1_opt = NaN; # 0; # Ersetze NaN durch die optimale Position für den ers
x2_opt = NaN; # 0; # Ersetze NaN durch die optimale Position für den zwe
# Hier nichts ändern! Checken und Plotten
checkOptPosAB1(x1_opt, x2_opt, s); plotMultiMirrors([x1_opt, x2_opt], h, b, s, 11)
```

! Bitte erst Teil c beantworten, bevor du zum nächsten Teil runterscrollst!

### Fazit

Wir haben festgestellt, dass nach unserem bisherigen Modell die meiste Energie umgesetzt wird, wenn beide Spiegel direkt unter dem Rohr stehen. Dies ist nicht umsetzbar. Damit die Spiegel sich nicht gegenseitig überdecken und ihre volle Beweglichkeit erhalten bleibt, fordern wir einen **Mindestabstand** zwischen benachbarten Spiegeln.



In der Sprache der mathematischen Optimierung:

Für die Variablen `x1` und `x2` werden wir eine **Nebenbedingung** aufstellen. Nur die Werte der Variablen `x1` und `x2` sind mögliche Lösungen des Optimierungsproblems, die diese **Nebenbedingung** erfüllen. Im Folgenden werden wir diese Nebenbedingung in Form einer Ungleichung formulieren, in der die Variablen `x1` und `x2` auftauchen.

### Mindestabstand zwischen den Spiegeln

Teil d



- Stelle eine Formel für den Abstand zwischen den beiden Spiegelmitteln  $x_1$  und  $x_2$  auf und gib sie für das erste `NaN` ein.
- Wie groß muss dieser Abstand **mindestens** sein, damit die Spiegel sich nicht überlagern? Gib den Mindestabstand für das zweite `NaN` ein.
- Nutze den Mindestabstand (Variable: `minDistance`), um eine Bedingung zu formulieren, die von den Positionen  $x_1$  und  $x_2$  erfüllt werden muss. Diese Bedingung sollte eine **Ungleichung** sein. Gib die Ungleichung für das dritte `NaN` ein.

**Hinweise:**

- Den Betrag einer Zahl gibt man in Julia mit `abs()` ein. Zum Beispiel ist `abs(-3-1) = abs(-4) = 4`.
- Ein Beispiel dazu, wie du eine Ungleichung eingibst: `abs(a - 20) > 9 * b`.
- Erinnerung: Die Breite der Spiegel ist in der Variablen `s` gespeichert und beträgt `s = 1.1m`.

Falls du Probleme beim Lösen dieser Aufgabe hast, öffne diesen [Tipp](#).

```
In [ ]:
# Formel für den Abstand zwischen den Spiegelmitteln
distance(x1,x2) = NaN; #= NaN; #= abs(x1 - x2); # Gib hier die Formel für den Abst.

# Mindestabstand
minDistance = NaN; #= s; # Gib hier den Mindestabstand ein.

# Nebenbedingung für den Mindestabstand
constraint1(x1,x2,minDistance) = NaN; #= NaN; #= abs(x1 - x2) > minDistance # GI

# Hier nichts ändern! Checken
checkDistance(distance), checkMinDistance(minDistance), checkConstraint1(conse
```

**Teil e - Optimierung mit Nebenbedingung Mindestabstand**



Finde die **bestmögliche** Kombination der beiden Positionen, bei der der Mindestabstand eingehalten wird. Variiere dazu die Spiegelpositionen  $x_1$  und  $x_2$  und lasse dir für jede Kombination die umgesetzte Energie ausgeben. Nutze den Antwortzettel, um deine Ergebnisse zu dokumentieren.

```
In [ ]:
x1 = -4; # Variiere die Position des 1. Spiegels
x2 = 2; # ... und des 2. Spiegels

# Hier nichts ändern!
### Deine Nebenbedingung für den Mindestabstand wird gecheckt und die Energie
checkPositionsMinDist(x1,x2,minDistance); plotMultiMirrors(x1,x2,h,b,s,l1)
```

**! Bitte erst Teil e bearbeiten, bevor du zum nächsten Teil runterscrollst!**

Wir haben eine Bedingung formuliert, die sicherstellt, dass die Spiegel den erforderlichen Mindestabstand einhalten. Anschließend haben wir die optimale Lösung durch Einsetzen unterschiedlicher Kombinationen von  $x_1$  und  $x_2$  in die Zielfunktion gesucht. Dieses Ausprobieren per Hand ist schon für zwei Spiegel aufwendig und für eine größere Zahl an Spiegeln überhaupt nicht mehr praktikabel. Die Optimierung soll deswegen **automatisiert** und dem **Computer** überlassen werden.

**Teil f - Entwicklung eines Optimierungsverfahrens**

... mit Stift und Papier



## C.6 Infoblatt zur Optimierung

### Infoblatt: Mathematische Optimierung

- Optimierungsprobleme bestehen im Allgemeinen aus einer Funktion, deren Funktionswert maximiert oder minimiert werden soll. Diese Funktion wird auch **Zielfunktion** genannt. In unserem Fall ist das die Funktion  $g(\mathbf{x}) = E_{\text{rohr}}(h, s, \mathbf{x})$ , mit der die Leistung am Rohr in Abhängigkeit von der Position des Spiegels  $\mathbf{x}$  berechnet wird.
- Gesucht werden in der Optimierung die **Variablen**, die beim Einsetzen in die Zielfunktion einen maximalen oder minimalen Wert liefern. In unserem Fall soll die Variable Position  $\mathbf{x}$  so gewählt werden, dass die Zielfunktion  $g(\mathbf{x})$  (d. h. die Energie am Rohr) einen maximalen Wert annimmt.
- Meist ist man nur an den Werten der Variablen interessiert, die gewisse **Nebenbedingungen** erfüllen.

# C.7 Arbeitsblatt zur Brute-Force-Methode

## Arbeitsblatt zu Algorithmus 1: Brute-Force Optimierung automatisieren



Damit du dieses Arbeitsblatt bearbeiten kannst, musst du als erstes den folgenden Code ausführen. Klicke dazu in das nächste Code-Feld und drücke auf "Run".

```

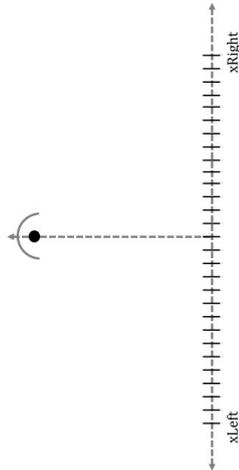
In [ ]:
# Hier nichts ändern!
include("../code/setup_discreteAlgorithm_1.jl");

# --- Festlegung einiger Parameter des Kraftwerks --- #
b 0.8; # Breite des Sekundärreflektors
s 1.1; # Breite des Spiegels
h 6; # Höhe des Absorbers
xLeft, xRight = [-10.0, 10.0]; # Linke und rechte Grenze des Kraftwerks
mindistance s; # Mindestabstand zwischen den Spiegeln
    
```

### Brute-Force Algorithmus

Bei der Optimierung von zwei Spiegelpositionen mit dem **Brute-Force-Ansatz** werden nacheinander die folgenden Schritte durchgeführt:

1. In dem Bereich, in dem die Spiegel stehen dürfen, legen wir mögliche Positionen fest, wobei benachbarte Positionen jeweils den **gleichen Abstand** zueinander haben. Eine Skizze dazu findest du in der folgenden Abbildung.
2. Für alle möglichen Kombinationen der Positionen von  $x_1$  und  $x_2$ , bei denen der Mindestabstand erfüllt ist, wird die Energie berechnet.
3. Die Kombination, die den größten Energiewert liefert, wählen wir als Lösung.



## Teil a

Zunächst unterteilen wir den Kraftwerksbereich von  $xLeft$  bis  $xRight$ , in dem Spiegel platziert werden dürfen, in gleich große Abschnitte. Anschließend wenden wir den beschriebenen Algorithmus an. Es werden die Kombinationen ausgegeben, bei denen eine maximale Energieumsetzung erzielt wird. Dabei werden nur Kombinationen berücksichtigt, bei denen der **Mindestabstand eingehalten** wird.



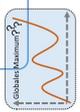
Lege fest, in wie viele Abschnitte (Variable:  $nPieces$ ) du den Kraftwerksbereich von  $xLeft = -10$  bis  $xRight = 10$  unterteilen möchtest. Gib den Wert für  $nAll$  ein. Die vordefinierten Positionen (das sind gerade die Positionen, die als Spiegelpositionen in Frage kommen) sind dann genau die Grenzen der einzelnen Abschnitte. Diese sind in der Skizze oben als senkrechte Striche dargestellt. Wende den Algorithmus an, indem du den Code ausführst. Variiere die Anzahl der Abschnitte und beobachte die Auswirkungen.

```

In [ ]:
nPieces = NaN; # Anzahl Stücke, in die der Spiegelbereich unterteilt wird
# Hier nichts ändern! Algorithmus wird ausgeführt.
checkNPieces(xLeft, xRight, nPieces);
combinatoricAlg(ktube, xLeft, xRight, nPieces, b, h, s, mindistance, constraint1)
    
```

### Vier wichtige Fragestellungen, die man sich beim Formulieren und Lösen von Optimierungsproblemen stellt:

**Existiert überhaupt eine optimale Lösung?** (z. B. es gibt eine Kombination der Positionen  $x_1$  und  $x_2$ , die optimal ist?)



**Einzigkeit der Lösung**

ist die Lösung **einzigartig**? In unserem Fall: gibt es höchstens eine Kombination der Werte für  $x_1$  und  $x_2$ , die zum gleichen Ergebnis der Energie führt?



**Rechenzeit eines Optimierungsverfahrens**

Wie hoch ist die **Rechenzeit**, die das Verfahren braucht, um eine Lösung zu finden?



**Güte der gefundenen Lösung**

Wie **nah** ist die Lösung, die ein Verfahren (z. B. das kombinatorische Verfahren) findet, an der **optimalen Lösung** (falls diese überhaupt existiert)?



## Teil b - Bewertung des Verfahrens



Diskutiere die folgenden Punkte und notiere deine Überlegungen auf dem Antwortzettel

- Welche Vor- oder Nachteile hat die Brute-Force Methode?
- Inwiefern sind die **vier Fragestellungen der Optimierung** erfüllt?
- Welchen Einfluss hat die Anzahl der Abschnitte `nPieces` auf die Rechenzeit und die Güte der Lösung?
- Notiere Ideen für die Verbesserung des Algorithmus.



### Diskussion

Wir werden eure Überlegungen nachher diskutieren!

**! Bitte erst Teil b beantworten, bevor du zum nächsten Teil runterscrollst! !**

### Teil c

In dieser Aufgabe beschäftigen wir uns mit der **Eindeutigkeit der Lösung**.

Wir haben in Teil b festgestellt, dass die Brute-Force Methode **zwei Kombinationen für die besten Positionen x1 und x2 mit dem gleichen Wert für die Energie liefert**. Um mehrere Lösungen mit identischem Spiegelfeldlayout zu vermeiden, werden wir eine Nebenbedingung für die Positionen `x1` und `x2` formulieren, die nicht nur den Mindestabstand berücksichtigt, sondern zusätzlich die genannten gleichwertigen Lösungen verhindert.



Stelle eine Nebenbedingung für die Positionen `x1` und `x2` auf, sodass nur noch eine der beiden Kombinationen, die das gleiche Spiegelfeldlayout beschreiben, als Lösung erlaubt ist. Die Nebenbedingung sollte eine **Ungleichung** sein. Gib diese Ungleichung für `NaN` ein.

Falls du Hilfe benötigst, findest du hier einen **Tipp**.

```
In [ ]:
# Neue Nebenbedingung
constraint2(x1,x2,minDistance) = constraint1(x1,x2,minDistance)
## #= NaN; == x1 < x2 ; # Gib hier deine Ne
# Hier nichts ändern! Checken
checkConstraint(constraint2, minDistance)
```



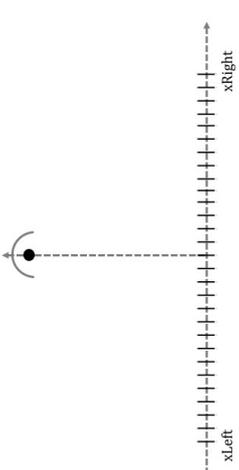
Führe den folgenden Code aus. Der Brute-Force Algorithmus wird mit der neuen Nebenbedingung erneut durchgeführt. Notiere, wie die vier Fragen der Optimierung nun zu beantworten sind.

```
In [ ]:
nPieces = #= NaN; == 250; # Anzahl Stücke, in die du den Spiegelsbereich unte
# Hier nichts ändern! Algorithmus wird ausgeführt.
combinatorial(Brute, xLeft, xRight, nPieces, b, h, s, minDistance, constraint2);
```

### Fazit

# C.8 Arbeitsblatt zum Greedy-Algorithmus

6. Dieses Vorgehen wird wiederholt bis alle Spiegel gesetzt wurden oder keine weiteren Spiegel mehr im Bereich von  $x_{Left}$  bis  $x_{Right}$  platziert werden können.



**Hinweis:** Auch bei diesem Verfahren wird der Mindestabstand stets berücksichtigt! D.h. in jedem Schritt wird geprüft, dass der neu zu setzende Spiegel nicht zu nah an den bereits positionierten Spiegeln steht.

**Teil a**

Zunächst werden wir den Kraftwerkbereich, in dem wir nach der optimalen Lösung suchen, in **gleichgroße Abschnitte** unterteilen und dann den beschriebenen Algorithmus anwenden.



Lege fest, in wie viele Abschnitte (Variable:  $nPieces$ ) du den Kraftwerkbereich von  $xLeft = -10$  bis  $xRight = 10$  unterteilen möchtest. Gib den Wert für  $nAW$  ein. Die vordefinierten Positionen (das sind gerade die Positionen, die als Spiegelpositionen in Frage kommen) sind dann genau die Grenzen der einzelnen Abschnitte. Diese sind in der Skizze oben als senkrechte Striche dargestellt. Führe die Optimierung erneut durch und beobachte, was sich ändert.

```
In [ ]:
nPieces = NaN; # Anzahl Stücke, in die der Spiegelbereich unterte
mirrors = 9; # Anzahl der Spiegel, die positioniert werden sollen
# Hier nichts ändern!
checkPieces(xLeft, xRight, nPieces); optPosl = optimizationAlgorithmGreedy(f
```

**Vier Fragestellung, die man sich beim Formulieren und Lösen von Optimierungsproblemen stellt:**

- 
- 
- 

## Arbeitsblatt zu Algorithmus 2: Greedy

### Optimierung automatisieren



Damit du dieses Arbeitsblatt bearbeiten kannst, musst du als erstes den folgenden Code ausführen. Klicke dazu in das nächste Code-Feld und drücke auf den "Run"-Button.

```
In [ ]:
### -- Hier nichts ändern! -- ###
include("../code/setup_discreteAlgorithm.jl");
# --- Festlegung einiger Parameter des Kraftwerks --- #
b = 0.8; # Breite des Sekundärreflektors
s = 1.1; # Breite des Spiegels
h = 6; # Höhe des Absorberrohrs
xLeft, xRight = [-10, 0, 10, 0]; # Linke und rechte Grenze des Kraftwerkareals
minDistance = s; # Mindestabstand zwischen den Spiegeln
```

### Greedy-Algorithmus

Bei diesem Verfahren wird das Kraftwerkareal in **vordefinierte Positionen** unterteilt. Die Spiegel werden dann nacheinander auf diese Positionen verteilt. Ziel ist dabei, in jedem Schritt eine möglichst hohe Energieumsetzung zu erzielen. Der Algorithmus sucht also in jedem Schritt **gierig** (engl. greedy) nach den Positionen, die zu maximaler (Energie-)Ausbeute führen. Dieses Verfahren wird daher **Greedy-Algorithmus** genannt.

Etwas detaillierter werden die folgenden Schritte durchgeführt:

1. Im Kraftwerkbereich von  $x_{Left}$  bis  $x_{Right}$  legen wir Positionen  $x_i$  fest, die als mögliche Spiegelpositionen in Frage kommen. Benachbarte Positionen haben dabei jeweils **den gleichen Abstand** (siehe Abbildung unten).
2. Für jede der vordefinierten Position  $x_i$  wird die zugehörige umgesetzte Energie durch Einsetzen in die Funktion  $E_{tube}(b, h, s, [x_i, x_i])$  berechnet. Der 1. Spiegel wird auf die Position mit maximaler Energieumsetzung gesetzt. Diese nennen wir  $x_1^*$ .
3. Aus der Liste der vordefinierten Positionen werden die Positionen gelöscht, die den **Mindestabstand** zum ersten Spiegel  $x_1^*$  nicht einhalten.
4. Für alle übrigen Positionen wird die jeweilige Energieumsetzung zusammen mit Spiegel 1 durch die Funktion  $E_{tube}(b, h, s, [x_1^*, x_i])$  berechnet. Der 2. Spiegel wird auf die Position gesetzt, die in Kombination mit  $x_1^*$  die maximale Energie umsetzt. Diese nennen wir  $x_2^*$ .
5. Aus der Liste der vordefinierten Positionen werden die Positionen gelöscht, die den Mindestabstand zum 2. Spiegel nicht einhalten.

**Teil b - Bewertung des Verfahrens**

Wir haben einen Algorithmus entwickelt, der aus einer vorher festgelegten Menge an möglichen Spiegelpositionen eine Lösung für die Positionen von **fünf Spiegeln** bestimmt.

*Diskutiere die folgenden Punkte und notiere deine Überlegungen auf dem Antwortzettel*

- Welche Vor- oder Nachteile hat der Greedy Algorithmus?
- Inwieweit sind die vier **Fragestellungen der Optimierung** erfüllt?
- Welchen Einfluss hat die Anzahl der Abschnitte **nPieces** auf die Rechenzeit und die Güte der Lösung?
- Notiere Ideen für die Verbesserung des Algorithmus.

**Diskussion**

**Wir werden eure Überlegungen nachher diskutieren!**

---



---

## C.9 Arbeitsblatt zur musterbasierten Optimierung

## Arbeitsblatt zu Algorithmus 3: Musterbasierte Optimierung

## Optimierung automatisieren



Damit du dieses Arbeitsblatt bearbeiten kannst, musst du als erstes den folgenden Code ausführen. Klicke dazu in das Code-Feld und drücke auf den "Run"-Button oder "Shift" + "Enter".

```
In [ ]:
### -- Hier nichts ändern! -- ###
include("../code/setup_continousAlgorithms.jl");

# --- Festlegung einiger Parameter des Kraftwerks --- #
b = 0.8; # Breite des Sekundärreflektors
s = 1.1; # Breite des Spiegels
h = 6; # Höhe des Absorbers
xLeft, xRight = [-10.0, 10.0]; # Linke und rechte Grenze des Kraftwerks als
minDistance = s; # Mindestabstand zwischen den Spiegeln
nMirrors = 4; # Anzahl der zu setzenden Spiegel
```

## Musterbasierte Optimierung

Bei diesem Verfahren werden nicht direkt die Position eines jeden Spiegels  $x_1, x_2, x_3, \dots$  optimiert. Stattdessen wird lediglich der Abstand zwischen den Spiegeln als Variable betrachtet und optimiert. Im einfachsten Fall können wir fordern, dass der **Abstand zwischen zwei benachbarten Spiegeln immer gleich groß** sein soll. Diesen Abstand bezeichnen wir mit  $\Delta x$  (Variable: `deltaX`).

## Teil a

Die Funktion `ETube(b, h, s, [x1, x2, x3, ...])` erhält bisher die Positionen  $x_1, x_2, x_3, \dots$  als Eingabe. Wir schreiben diese so um, dass eine Funktion entsteht, die direkt vom Spiegelabstand  $\Delta x$  abhängt. Dazu setzen wir die Position des ersten Spiegels (das ist der Spiegel, der sich in unserer Anordnung ganz links befindet) auf einen Startwert  $x_1 = x_{\text{Start}}$  fest. Wir nutzen dann die Variablen Spiegelabstand  $\Delta x$  und Startposition  $x_{\text{Start}}$ , um die Positionen  $x_1, x_2, x_3, \dots$  zu beschreiben.



Stelle jeweils eine Formel auf, mit der die Position  $x_2$  bzw.  $x_3$  in Abhängigkeit von  $x_{\text{Start}}$  und `deltaX` berechnet werden kann. Stelle dann eine allgemeine Formel für die Position des  $N$ -ten Spiegels (Position  $x_N$ ) auf. Gib die drei Formeln für die `NaN` ein und führe den Code aus.

```
In [ ]:
x1(xStart) = xStart;
x2(xStart, deltaX) = NaN; # Gib hier eine Formel für d
x3(xStart, deltaX) = NaN; # Gib hier eine Formel f
xN(xStart, deltaX, N) = NaN; # Gib hier eine
checkPositionsDelta(x1, x2, x3, xN)
```

Nun können wir unsere Funktion für die Energie `ETube(b, h, s, [x1, x2, x3, ...])` so speichern, dass nur noch der Spiegelabstand  $\Delta x$  und die Position des 1. Spiegels  $x_1$  als Eingabe übergeben werden.



Führe den folgenden Code aus. Die Funktion für die Energie am Rohr über einen Tag wird in Abhängigkeit von der Startposition  $x_{\text{Start}}$  und dem Spiegelabstand `deltaX` gespeichert. Dazu wird keine Funktion `xN(xStart, deltaX, N)` aus Teil a verwendet.

```
In [ ]:
# Hier nichts ändern!
funktion ETubeDelta(b, h, s, xStart, deltaX)
positions = [xN(xStart, deltaX, i) for i in 1 : nMirrors];
ETube(b, h, s, positions);
end
saveETubeDelta(xN)
```

## Teil b - Optimierung des Spiegelabstands

Julia verfügt über verschiedene Methoden, mit denen Optimierungsprobleme mit einer oder mehreren Variablen gelöst werden können. Wir werden eine solche Methode anwenden, um für einen gegebenen Startwert  $x_{\text{Start}}$  den optimalen Spiegelabstand  $\Delta x$  zu finden. Hierbei wird die Nebenbedingung  $\Delta x \geq s$  (d. h. die Einhaltung des Mindestabstands) berücksichtigt.



Führe den Code aus, um mit dem Optimierungstool von Julia den Spiegelabstand zu bestimmen, der zur maximalen umgesetzten Energie führt. Das Ergebnis der Optimierung und die Rechendauer werden angezeigt. Variiere den Startwert  $x_{\text{Start}}$  und führe den Algorithmus erneut aus. Notiere deine Beobachtungen und Ideen für Verbesserung.

**Hinweis:** Die Anzahl der zu setzenden Spiegel ist zunächst auf 4 Spiegel festgelegt. Du kannst diese Zahl im Verlauf des Arbeitsblattes variieren.

```
In [ ]:
xStart = -10; # Du kannst den Startwert ändern!
# Hier nichts ändern!
zielfunktion(deltaX) = -ETubeDelta(b, h, s, xStart, deltaX); optimierdelta(ziel
f1(deltaX) = ETubeDelta(b, h, s, xStart, deltaX); plotPosDeltaETube(f1, xStart, min)
```

## Teil c - Optimierung von Startwert und Spiegelabstand

- Beantworte die vier Fragestellungen der Optimierung.
- Notiere mögliche Ideen zur Verbesserung des Verfahrens.



**Diskussion**

Wir werden eure Überlegungen nachher diskutieren!

Bisher haben wir die Optimierung des Spiegelabstands  $\Delta x$  mit einem festen Startwert für den ersten Spiegel durchgeführt. Da die **Lösung stark von der Wahl dieses Startwertes abhängt**, ist es sinnvoll, diesen ebenfalls als Variable bei der Optimierung zu berücksichtigen. Das Optimierungsproblem hat damit nun zwei Variablen, deren **optimale Kombination** gesucht wird: der Startwert  $x_{\text{start}}$  und der Spiegelabstand  $\Delta x$ .

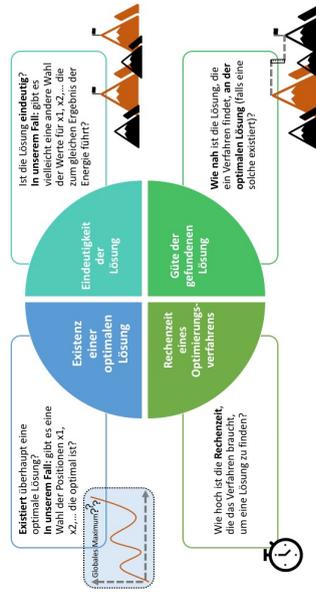


Führe den Code aus, um mit der Optimierungsmethode von Julia die Kombination aus Startwert  $x_{\text{start}}$  und Spiegelabstand  $\Delta x$  zu finden, die zur maximalen Energieumsetzung führt. Das Ergebnis der Optimierung und die Rechendauer werden angezeigt. Ändere auch mal die Anzahl der zu setzenden Spiegel `nMirrors`. Vergleiche das Ergebnis mit dem Ergebnis aus Teil b. Wie sollten die Spiegel nun am besten positioniert werden? Bewerte das Ergebnis auch im Hinblick auf die reale Situation...

```
In [ ]: nMirrors = 2;
# Hier nichts ändern!
Zielfunktion(xStart, deltaX) = @bubbeDelta(b,h,s,xStart,deltaX); optimizerDel
```

**Teil d - Bewertung des Verfahrens**

Vier wichtige Fragestellung, die man sich beim Formulieren und Lösen von Optimierungsproblemen stellt:



Diskutiere die folgenden Punkte und notiere deine Überlegungen auf dem Antwortzettel

- Was sind Vor- oder Nachteile der Musterbasierten Optimierung?

# C.10 Arbeitsblatt zur freien Variablen Optimierung

## Arbeitsblatt zu Algorithmus 4: Freie Optimierung der Positionsvariablen

### Optimierung automatisieren



Damit du dieses Arbeitsblatt bearbeiten kannst, musst du als erstes den folgenden Code ausführen. Klicke dazu in das nächste Code-Feld und drücke auf den "Run"-Button.

```

In [ ] :
### -- Hier nichts ändern! -- ###
include("../code/setup_continuousAlgorithms.jl");

# --- Festlegung einiger Parameter des Kraftwerks --- #
b = 0.8; # Breite des Sekundärreflektors
s = 1.1; # Breite des Spiegels
h = 6; # Höhe des Absorberrohrs
xLeft, xRight = (-10.0, 10.0); # Linke und rechte Grenze des Kraftwerkareals
mindistance = s; # Mindestabstand zwischen den Spiegeln
            
```

### Freie Optimierung der Positionsvariablen

Bei diesem Verfahren dürfen die Variablen der Positionen  $x_1, x_2, x_3, \dots$  nicht nur diskrete, vorher festgelegte Werte annehmen, sondern sie können **jeden reellen Wert** innerhalb des Kraftwerkbereichs annehmen.

#### Teil a

Die Funktion  $E_{\text{tube}}$  mit  $E_{\text{tube}}(b, h, s, [x_1, x_2, x_3, \dots])$  ist die **Zielfunktion** der Optimierung. Die Positionen  $x_1, x_2, x_3, \dots$  sind die **Variablen**, für die optimale Werte gesucht werden. Dabei sollen die folgenden beiden Nebenbedingungen berücksichtigt werden:

- Der Mindestabstand zwischen benachbarten Spiegeln muss eingehalten werden.
- Die Variablen müssen im Bereich des Kraftwerkareals von  $x_{\text{left}}$  bis  $x_{\text{right}}$  liegen.

Julia verfügt über verschiedene Tools, mit denen Optimierungsprobleme mit einer oder mehreren Variablen gelöst werden können. Wir werden eine dieser Methoden anwenden und dabei die genannten Nebenbedingungen berücksichtigen. Die Optimierung wird zunächst für zwei Spiegel durchgeführt.

---

**Hinweis:** Der Optimierer startet die Suche nach der Lösung von einem Startwert aus und liefert **eine Lösung**. Das heißt aber **nicht**, dass es **die einzige Lösung** ist. Manchmal findet der Optimierer auch nur eine Lösung, die **lokales Maximum oder Minimum** ist. Mehr Infos und ein Beispiel dazu findest du auf diesem [Infoblatt](#).

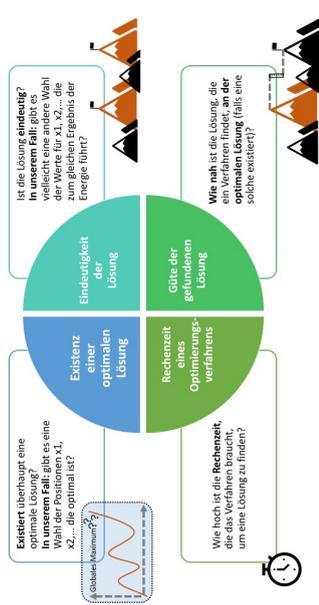


1. Ersetze `NaN!` durch einen Startwert für die Position des ersten Spiegels. Dieser sollte zwischen `xLeft = -10.0` und `xRight = 10.0`.
2. Führe den Code aus. Das Ergebnis der Optimierung und die Rechenzeit werden angezeigt.
3. Ändere den Startwert für die Position des ersten Spiegels und führe die Optimierung erneut aus. Gib sowohl negative, als auch positive Startwerte ein. Was beobachtest du?

```

In [ ] :
nMirrors = 2; # Anzahl der Spiegel, die positioniert werden sollen
startValueX1 = NaN; # NaN; # Startwert für die Position des ersten Spiege
# Hier nichts ändern!
# Hier nichts ändern!
Zielfunktion(positions...) = -Etube(b, h, s, positions);
PosOpt = optimizerFreeVariables3(startValueX1, Zielfunktion, nMirrors, mindistance
            
```

### Vier wichtige Fragestellungen, die man sich beim Formulieren und Lösen von Optimierungsproblemen stellt:



**Existenz einer optimalen Lösung:** Existiert überhaupt eine optimale Lösung? In unserem Fall: Gibt es eine Wahl der Positionen  $x_1, x_2, \dots$ , die optimal ist?

**Eindeutigkeit der Lösung:** Ist die Lösung eindeutig? Gibt es vielleicht eine andere Wahl der Werte für  $x_1, x_2, \dots$ , die zum gleichen Ergebnis der Energie führt?

**Qualität der gefundenen Lösung:** Wie nah ist die Lösung, die wir gefunden haben, an der optimalen Lösung (falls eine solche existiert)?

**Rechenzeit eines Optimierungsverfahrens:** Wie hoch ist die Rechenzeit, die das Verfahren braucht, um eine Lösung zu finden?

### Teil b - Bewertung des Verfahrens



Diskutiere die folgenden Punkte und notiere deine Überlegungen:

- Welche Vor- oder Nachteile hat diese Optimierung?
- Beantworte die vier Fragestellungen zur Optimierung.
- Notiere mögliche Ideen zur Verbesserung der Optimierung.

**! Bitte erst Teil b beantworten, bevor du zum nächsten Teil**



Stelle eine Nebenbedingung für die Positionen  $x_1$  und  $x_2$  auf, sodass nur noch eine der beiden Kombinationen mit gleichem Spiegelfeldlayout als Lösung erlaubt ist. Die Nebenbedingung sollte eine Ungleichung sein. Gib die neue Nebenbedingung, die von den Positionen  $x_1$  und  $x_2$  erfüllt werden muss, für `NaN` ein.

Falls du Schwierigkeiten beim Lösen der Aufgabe hast, kannst du diesen **Tip** öffnen.

```
In [ ]:
# Neue Nebenbedingung
constraint2(x1, x2, mindistance) == constraint1(x1, x2, mindistance)
&& #!= NaN; ==# x1 < x2 ; # Gib hier deine Nebenbedingung ein
# Hier nichts ändern! Checken
checkConstraint2(constraint2, mindistance)
```



Führe die Optimierung mit der **neuen Nebenbedingung** erneut durch, indem du den folgenden Code ausführst. Gib für `NaN` zunächst wieder einen Startwert für die Position des ersten Spiegels ein. Variiere den Startwert für die Position  $x_1$ . Was beobachtest du?

```
In [ ]:
nMirrors = 2; # Anzahl der Spiegel, die positioniert werden sollen
startValueX1 = #!= NaN; ==# -1; # Startwert für die Position des ersten Spiegels
# Hier nichts ändern!
Zielfunktion(positions...) == -Etube(b, h, s, positions);
PosOpt = optimizerFreeVariablesZ(startValueX1, Zielfunktion, nMirrors, mindistance)
```

### Zusatzaufgabe

Führe die Optimierung für mehr als zwei Spiegel durch, indem du den Wert der Variablen `nMirrors` erhöhst. **Hinweis:** deine Nebenbedingung für den Mindestabstand und die Nebenbedingung für die auf- bzw. absteigende Sortierung der Spiegel wurde im Hintergrund erweitert. Sie wird bei der Optimierung stets für je zwei benachbarte Spiegel überprüft.



### Diskussion

Wir werden eure Überlegungen nachher diskutieren!

runterscrollst!

### Teil c

In dieser Aufgabe beschäftigen wir uns mit der **Eindeutigkeit der Lösung**. Wir haben in Teil b festgestellt, dass abhängig vom gewählten Startwert zwei optimale Kombinationen für die Spiegelpositionen gefunden werden. Beide Kombinationen liefern den **gleichen Wert der Zielfunktion** und führen zum **selben Spiegelfeldlayout**. Um solche gleichwertigen Lösungen zu vermeiden, werden wir eine weitere Nebenbedingung an die Positionen  $x_1$  und  $x_2$  formulieren.

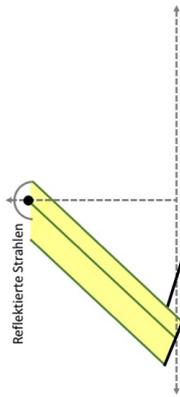
# C.11 Arbeitsblatt 5: Verschattungseffekte

## Solkraftwerk - Arbeitsblatt 5: Modellierung von Verschattungseffekten

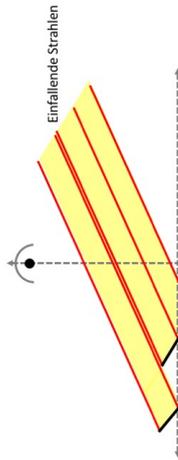
Bisher sind wir davon ausgegangen, dass die Spiegel sich nicht gegenseitig beeinflussen. Tatsächlich treten jedoch verschiedene Effekte auf.

Betrachte die folgenden drei Abbildungen. Welche Effekte sind in den einzelnen Skizzen dargestellt?

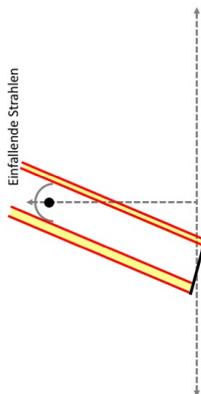
**Effekt 1:**



**Effekt 2:**



**Effekt 3:**



Einfallende Strahlen können durch nebenstehende Spiegel oder durch den Sekundärreflektor abgefangen werden. Auch ist es möglich, dass reflektierte Strahlen von nebenstehenden Spiegeln blockiert werden und nicht auf das Absorberrohr treffen. Tatsächlich ist umgesetzte Energie am Rohr also **kleiner als wir bisher angenommen haben**.

Im Folgenden werden wir das Modell für die umgesetzte Energie um die genannten Verschattungseffekte erweitern und den **Einfluss auf die optimalen Spiegelpositionen** untersuchen.



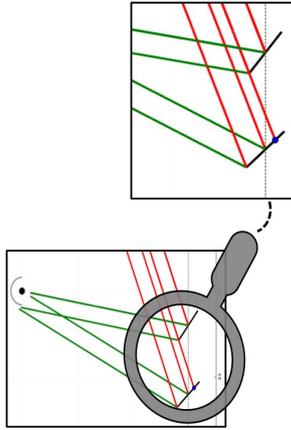
Damit du dieses Arbeitsblatt bearbeiten kannst, musst du als erstes den folgenden Code ausführen. Klicke dazu in das nächste Code-Feld und drücke "Run".

```
In [ ]:
# Hier nichts ändern
include("../code/setup_mirrorEffects.jl");
# --- Festlegung einiger Parameter des Kraftwerks --- #
t = 8.5; # Uhrzeit
b = 0.6; # Breite des Sekundärreflektors
s = 1.1; # Breite des Spiegels
h = 6; # Höhe des Absorberrohrs
x1 = -3; # Position Spiegel 1
x2 = -1; # Position Spiegel 2
```

Abhängig von Tageszeit und Spiegelpositionen ist es möglich, dass Spiegel teilweise im Schatten von anderen Spiegeln stehen. Wir werden auf diesem Arbeitsblatt die Verschattung von Spiegeln modellieren und in unserem Modell berücksichtigen.

Wir betrachten exemplarisch zwei Spiegel auf den Positionen  $x_1 = -3$  und  $x_2 = -1$  um 8:30 ( $t = 8.5$ ) Uhr. Der Strahlenverlauf um diese Uhrzeit ist in der folgenden Abbildung dargestellt. Ein Teil der einfallenden Strahlen wird vom rechten Spiegel abgefangen und trifft nicht auf den linken Spiegel. Man erkennt, dass vom linken Spiegel nur die Fläche effektiv bestrahlt wird, die **von der linken Ecke des Spiegels bis zu dem in der Abbildung blau markierten Punkt reicht**. Dieser blaue Punkt ist der Schnittpunkt zwischen

- der Gerade, die den einfallenden Strahl durch die linke Ecke des rechten Spiegels beschreibt und
- der Gerade, die den linken Spiegel enthält.



Ziel ist es, Spiegel und Sonnenstrahlen mathematisch zu beschreiben und den markierten Schnittpunkt zu bestimmen. Dazu legen wir zunächst ein zweidimensionales kartesisches Koordinatensystem in unser Kraftwerkareal, dessen Ursprung direkt unter dem Rohr liegt. Die x-Achse läuft durch die Spiegelmitten.

**Achtung:** Wir werden auf diesem Arbeitsblatt die Parameterform von Geraden verwenden. Wenn du diese noch nicht kennst, findest du [hier](#) ein Infoblatt dazu. Dieses liefert dir alle Infos, die du für die weitere Bearbeitung des Arbeitsblattes brauchst!

### Teil a - Mathematische Beschreibung der einfallenden Sonnenstrahlen

Wir stellen die einfallenden Sonnenstrahlen als Geraden mit Stützvektor  $\vec{p}$  und Richtungsvektor  $\vec{r}$  dar. Einen einfallenden Strahl beschreiben wir dann als

$$\vec{p} + \lambda \cdot \vec{r} = \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} + \lambda \cdot \begin{pmatrix} r_1 \\ r_2 \end{pmatrix}$$

**Zusatzüberlegung:** Wir nutzen zur Darstellung der Geraden die Parameterform. Alternativ könnten die Sonnenstrahlen durch Graphen linearer Funktionen der Form  $f(x) = m \cdot x + b$  mit Steigung  $m$  und y-Achsenabschnitt  $b$  modelliert werden. Welche Vor- oder Nachteile hat diese Darstellungsform?



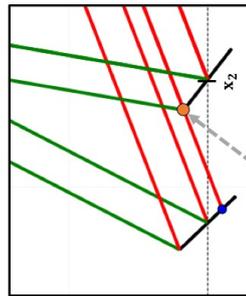
Stelle Formeln für die Berechnung der Einträge  $r_1$  und  $r_2$  des Richtungsvektors  $\vec{r}$  der einfallenden Strahlen auf. Diese müssen vom Einfallswinkel der Sonnenstrahlen  $\alpha_1$  (=  $\alpha[\text{alpha}_1]$ ) abhängen. Gib deine Formeln für die beiden `Maths` ein.

**Hinweise:**

- Eine Erinnerung dazu, welche Winkel `beta`, `gamma` und `alpha` beschreiben, findest du [hier](#).
- Falls du unsicher bist, was ein Richtungsvektor ist, findest du [hier](#) eine kurze Erinnerung
- Falls du Schwierigkeiten beim Aufstellen des Richtungsvektors hast, kamst du dir diesen [Tipp](#) ansehen.
- Sinus und Cosinus gibt man bei Julia mit `sind()` und `cosd()` ein. Eine kurze Erinnerung zur Verwendung von Sinus und Cosinus findest du [hier](#).

```
In [ ]:
alpha1 = alpha(t); # Den Einfallswinkel zum Zeitpunkt t speichern wir in der
r1 = #= Math; # cosd(alpha1); # Ersetze NaN durch den 1. Eintrag des Richtung
r2 = #= Math; # sind(alpha1); # Ersetze NaN durch den 2. Eintrag des Richtung
# Hier nichts ändern!
checkray(r1,r2,t)
```

Um den bestrahlten Bereich des linken Spiegels zu bestimmen, ist gerade der einfallende Strahl relevant, der durch die linke Ecke des rechten Spiegels verläuft. Als Stützvektor  $\vec{p}$  dieses Strahls können wir daher die Koordinaten ebendieser linken Ecke verwenden.



Gesucht: linke Ecke des Spiegels auf Position  $x_2$

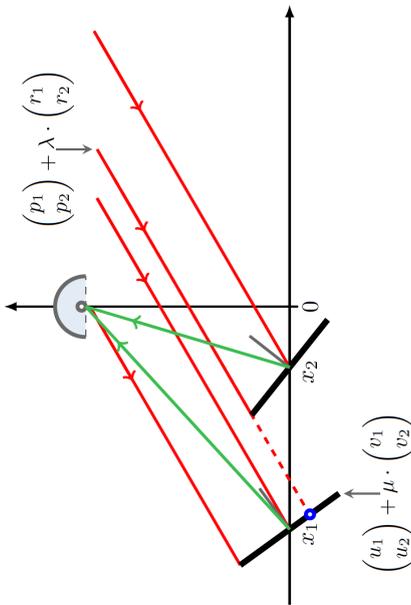


Stelle Formeln für die Koordinaten der linken Ecke des Spiegels auf Position  $x_2$  auf (siehe Skizze). Du kannst den Neigungswinkel  $\gamma_2$  (= `gamma2`) des Spiegels verwenden. Wir betrachten den Fall, dass der Neigungswinkel kleiner als  $90^\circ$  ist. Gib deine Formeln zur Berechnung der beiden Einträge der linken Ecke für die beiden `Maths` ein.

### Teil c - Bestimmung des nicht verschatteten Spiegelbereichs

Wie lang ist der Bereich des linken Spiegels, der nicht verschattet ist? Um dies zu bestimmen, berechnen wir den **ersten verschatteten Punkt** auf dem linken Spiegel. Das ist gerade der Schnittpunkt zwischen

- der Gerade, die den linken Spiegel enthält, und
- der Gerade des **einfallenden Strahls** durch die **linke Ecke** des rechten Spiegels.



Wie kann der Schnittpunkt zwischen den beiden Geraden bestimmt werden? Überlege dir eine konkrete Vorgehensweise und diskutiere diese falls möglich mit einer Mitschülerin oder einem Mitschüler. Halte deine Lösungsstrategie schriftlich fest. Führe erst danach das folgende Codefeld aus. Daraufhin wird dir die Lösung zu dieser Aufgabe, d. h. die Vorgehensweise bei der Bestimmung des Schnittpunktes, verraten.

```
In [ ]:
# Ausgabe der Lösung
print(solutionsR())
```



Stelle mithilfe von den gespeicherten Parameterwerten `muSP` oder `LambdaSP` Formeln zur Berechnung der Koordinaten des ersten verschatteten Punktes auf dem Spiegel auf. In diesen Formeln kannst du folgende Variablen verwenden:

Hinweis: Falls du Schwierigkeiten hast, findest du hier einen **Tipp**.

```
In [ ]:
# Hier nichts ändern!
gamma = gamma(t,h,x2)
x2dge = #= NaN; #= s / 2 * cosd(90 - gamma2); # Ersetze NaN durch den 1.
y2dge = #= NaN; #= s / 2 * sind(90 - gamma2); # Ersetze NaN durch den 2. Ein
## -- Hier nichts ändern! -- ##
p1,p2 = x2dge, y2dge # Speichern der Koordinaten der linken Ecke als Einträge
checkLeftEdge(p1,p2,t,h,x2)
```

### Teil b - Mathematische Beschreibung des Spiegels

Um den Schnittpunkt und anschließend den nicht verschatteten Bereich berechnen zu können, werden wir den Spiegel durch eine Gerade der Form

$$\vec{u} + \mu \cdot \vec{v} = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + \mu \cdot \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$$

mit Stützvektor  $\vec{u}$  und Richtungsvektor  $\vec{v}$  beschreiben.



1. Stelle Formeln für die Einträge  $v_1$  und  $v_2$  des Richtungsvektors des Spiegels auf Position  $x_1$  auf. Du kannst den Neigungswinkel  $\gamma_1$  (= `gamma1`) des Spiegels verwenden.
2. Welchen Vektor kannst du als Stützvektor verwenden?
3. Welche Werte darf der Parameter  $\mu$  annehmen, damit tatsächlich nur der Spiegel beschrieben wird und nicht eine Gerade, die über die Spiegelfläche hinausragt.

Gib den Richtungsvektor, den Stützvektor und das Intervall für  $\mu$  (= `mu`) für die **NaNs** ein.

#### Hinweise:

- Wir betrachten erneut den Fall, dass der Neigungswinkel des Spiegels `gamma1` kleiner als 90 Grad ist.
- Falls du eine **Tipp** für das Aufstellen der Formeln des Richtungsvektors brauchst, öffne diesen **Tipp**.

```
In [ ]:
# Hier nichts ändern!
gamma1 = compute_gammaBeg(t,h,x1); # Den Neigungswinkel zur Zeit t speichern
gamma1
### -- Beschreibung des Spiegels durch Stützvektor und Richtungsvektor --- #
u1 = #= NaN; #= x1; # Ersetze NaN durch den 1. Eintrag eines Stützvektors für
u2 = #= NaN; #= 0; # Ersetze NaN durch den 2. Eintrag eines Stützvektors für
v1 = #= NaN; #= s * cosd(90 - gamma1); # Ersetze NaN durch den 1. Eintrag des
v2 = #= NaN; #= -s * sind(90 - gamma1); # Ersetze NaN durch den 2. Eintrag des
##### -- Intervall für mu -- #####
muLeft = #= NaN; #= -1/2; # Gib hier die linke Intervallgrenze für mu ein.
muRight = #= NaN; #= 1/2; # Gib hier die rechte Intervallgrenze für mu ein.
# Hier nichts ändern!
checkMirror(u1,u2,v1,v2,muLeft,muRight,t,h,s,x1)
```

```
unshadedLength = #= NaN; =# sqrt((xEdge - xSP)^2 + (yEdge - ySP)^2); # Gib hier
# Hier nichts ändern!
checkKillIlluminatedLength(unshadedLength, xEdge, yEdge, xSP, ySP)
```

### Fazit

Mit unserem Modell können wir den Bereich eines Spiegels ermitteln, der durch Spiegel zu seiner Rechten verschattet wird. Dieses Vorgehen kann unter Berücksichtigung verschiedener Fallunterscheidungen auf eine Verschattung durch linksseitige Spiegel (am Nachmittag) verallgemeinert werden.

Abschließend muss noch berechnet werden, welcher **Anteil der nicht verschatteten Strahlen auf den Sekundärreflektor trifft** (und diesen nicht verfehlt). Dies brauchst du auf diesem Arbeitsblatt nicht selbst machen. Stattdessen kannst die Funktion `EtubeEffects` verwenden, in der **alle drei Effekte** (Verschattung durch Spiegel, Verschattung durch Sekundärreflektor, Blockierung) eingebaut sind.



In der Forschung findet ein etwas anderes Modell Anwendung, um die Effekte zwischen den Spiegel zu modellieren. Ein sogenannter **RayTracer**. Mehr Infos dazu findest du auf diesem [Zusatzblatt](#).

### Teil e - Modell mit Verschattungs- und Blockierungseffekten

Wir werden das um Verschattungs- und Blockierungseffekte erweiterte Modell abschließend mit dem alten Modell ohne Berücksichtigung dieser Effekte vergleichen.

Im folgenden Code kannst du entscheiden, welche Effekte (**Verschattung, Blockierung, Verschattung durch Sekundärreflektor**) du berücksichtigen möchtest. Du kannst **keinen, einen, zwei oder alle** Effekte berücksichtigen.



**Entscheide, welche Effekte (Verschattung durch Spiegel, Blockierung reflektierter Strahlen, Verschattung durch Sekundärreflektor) du berücksichtigen möchtest. Du kannst keinen, einen, zwei oder alle Effekte berücksichtigen. Ersetze im folgenden Code die drei NaNs durch eine 1 oder eine 0.**

- **1** bedeutet: der Effekt wird in der Zielfunktion berücksichtigt.
  - **0** bedeutet: der Effekt wird **nicht** in der Zielfunktion berücksichtigt.
- Führe den Code dann aus. Die Energieumsätze gemäß des alten Modells `Etube` und gemäß des neuen Modells `EtubeEffects` werden für Spiegel auf den exemplarisch

- Die Einträge der Stütz- und Richtungsvektoren `p1`, `p2` und `r1`, `r2` (Sommenstrahl) bzw. `u1`, `u2` und `v1`, `v2` (Spiegel).
- Die Parameter `muSP` und `lambdaSP`.

Gib deine Formel zur Berechnung der x-Koordinate des ersten verschatteten Punktes für das erste `NaN` ein und die Formel für die y-Koordinate für das zweite `NaN`.

```
In [ ] :
# Hier nichts ändern!
muSP, lambdaSP = computeSP(p1,p2,r1,r2,u1,u2,v1,v2);
# Berechnung des ersten verschatteten Punktes auf dem linken Spiegel:
xSP = #= NaN; =# u1 + muSP * v1; # Gib hier deine Formel zur Berechnung der x
ySP = #= NaN; =# u2 + muSP * v2; # Gib hier deine Formel zur Berechnung der y
# Hier nichts ändern!
checkSP(muSP, lambdaSP, xSP, ySP, u1, u2, v1, v2); plotWallWithMirrorsEffectsSP1(x1, x2
```

### Zusatzaufgabe

Führe die Rechenschritte zur Bestimmung der Werte der Parameter  $\mu_{SP}$  und  $\lambda_{SP}$  im Schnittpunkt selbstständig durch.

### Teil d - Anteil des nicht verschatteten Spiegelbereichs

Nun muss noch der Anteil berechnet werden, den der **nicht verschattete Bereich an der Gesamtlänge des Spiegels** ausmacht. Dies entspricht dem Anteil der Energie, die tatsächlich am Spiegel ankommt und in Richtung Absorberrohr reflektiert wird. Dazu berechnen wir zunächst die **Länge des nicht verschatteten Spiegelbereichs**. Dieser reicht von der linken Ecke des Spiegels bis zum berechneten Schnittpunkt.



Julia nutzt deine Eingabe aus Teil b, um die Koordinaten der linken Ecke des Spiegels auf Position `x1` zu berechnen. Diese werden unter `xEdge`, `yEdge` gespeichert. Stelle eine Formel für die Länge `l` des nicht verschatteten Spiegelbereichs auf. Gib deine Formel für `NaN` ein. Du kannst die Koordinaten der linken Ecke `xEdge`, `yEdge` und die des Schnittpunkts `xSP`, `ySP` in deiner Formel verwenden.

#### Hinweis:

- Die Quadratwurzel einer Zahl berechnet man im Code mit `sqrt()`. Zum Beispiel: `sqrt(4) = 2`
- Das Quadrat einer Zahl berechnet man im Code mit `^2`. Zum Beispiel: `5^2 = 25`
- Bei Schwierigkeiten kannst du diesen [Tipp](#) öffnen.

```
In [ ] :
# Hier nichts ändern!
xEdge, yEdge = computeLeftEdgeSol(t, h, x1);
# Nicht verschatteten Spiegelbereich berechnen
```

gewählten Positionen  $x_1 = -2.5$  m,  $x_2 = 0$  m und  $x_3 = 1.5$  m ausgegeben. In dem neuen Modell werden genau die Effekte berücksichtigt, die du zuvor ausgewählt hast.

```
In [ ]:
# Blockierungseffekt
blockiongOn = NaN; # 1; # Blockierungseffekt
shadingOn = NaN; # 1; # Verschattungseffekt
collectorShadingOn = NaN; # 1; # Verschattung durch Sekundärreflektor

# Positionen
positions = [-2.5, 0, 1.5]; # Du kannst die Spiegelpositionen ändern oder erw
### -- Hier nichts ändern! -- ###
EnergyOld = Etube(b,h,s,positions); EnergyNew = EtubeEffects(b,h,s,positions)
```



Führe den folgenden Code aus. Die Graphen der ursprünglichen Funktion (ohne Effekte) und der weiterentwickelten Funktionen (mit Effekten) werden für die umgesetzte Energie in Abhängigkeit von der Position  $x$  eines Spiegels angezeigt. Hat sich die optimale Position eines Spiegels geändert? Erkläre den Verlauf der beiden Graphen im Hinblick auf die reale Situation.

```
In [ ]:
# Hier nichts ändern!
f(x) = Etube(b,h,s,[x]); g(x) = EtubeEffects(b,h,s,[x]); plotPos1Etube2Func(g
```



Führe den folgenden Code aus. Der Graph der weiterentwickelten Funktionen (mit Effekten) wird für die umgesetzte Energie in Abhängigkeit von den Positionen  $x_1$  und  $x_2$  von zwei Spiegeln angezeigt. Hat sich die optimale Position eines Spiegels geändert? Erkläre den Verlauf des Graphen im Hinblick auf die reale Situation.

**Hinweis:** Da bei unserem neuen Modell viele einzelne Schnittpunktsberechnungen durchgeführt werden, dauert es länger bis die Grafik erscheint!

```
In [ ]:
# Hier nichts ändern! Plotten
f(x1,x2) = EtubeEffects(b,h,s,[x1,x2]); plotPos2EtubeNew(f,-3,3)
```

Öffne nun das nächste Arbeitsblatt zur Optimierung. Auf diesem kannst du die verschiedenen Optimierungsverfahren auf die neue Zielfunktion `EtubeEffects` anwenden.

## C.12 Arbeitsblatt 6: Optimierung unter Verschattungseffekten

### Solarkraftwerk - Optimierung der Spiegelpositionen

#### Modell mit Verschattungs- und Blockierungseffekten

Auf diesem Arbeitsblatt kannst du den Einfluss des verbesserten Modells, bei dem alle drei Verschattungs- und Blockierungseffekte berücksichtigt werden, auf die Optimierung der Spiegelpositionen untersuchen.

Du kannst dazu die verschiedenen Optimierungsverfahren der vorherigen Arbeitsblätter anwenden:

- Brute-Force Algorithmus (nur für 2 Spiegel)
- Greedy-Algorithmus
- Optimierung des Spiegelabstands und der Startposition
- Freie Variablen Optimierung



Damit du dieses Arbeitsblatt bearbeiten kannst, musst du als erstes den folgenden Code ausführen. Klicke dazu in das Code-Feld und drücke auf den "Run"-Button oder "Shift" + "Enter".

```
In [ ]:
# Hier nichts ändern!
include("../code/setup_AllAlgorithms.jl");

# --- Festlegung einiger Parameter des Kraftwerks --- #
b = 0.8; # Breite des Sekundärreflektors
s = 1.1; # Breite des Spiegels
h = 6; # Höhe des Absorberrohrs
xLeft,xRight = [-10,10.0]; # Linke und rechte Grenze des Kraftwerkareals
minDistance = 8; # Mindestabstand zwischen den Spiegeln
```



Wende einen der Algorithmen zur Optimierung der Positionen auf das verbesserte Modell an. Führe dazu eines der folgenden Code-Felder aus. Vergleiche die Ergebnisse mit den Ergebnissen, die du unter Verwendung des alten Modells ohne Verschattungs- und Blockierungseffekte erhalten hast.



Da mit unserem neuen Modell viele einzelne (Schnittpunkts-)Berechnungen durchgeführt werden, um die Verschattung und Blockierung zu berechnen, ist die Rechenzeit womöglich hoch.

#### Algorithmus 1: Kombinatorischer Algorithmus

**Hinweis:** Beide Nebenbedingungen (Mindestabstand und aufsteigende Sortierung der Spiegel) werden berücksichtigt.

```
In [ ]:
nPieces = 100; # Anzahl der Abschnitte, in die der Kraftwerkbereich unterteilt
# Hier nichts ändern!
combinatorialAlg(Etubeeffects,xLeft,xRight,nPieces,b,h,s,minDistance,constrain
```

#### Algorithmus 2: Greedy Algorithmus

```
In [ ]:
nMirrors = 4; # Anzahl der Spiegel, die gesetzt werden sollen
nPieces = 100; # Du kannst die Anzahl der Abschnitte, in die der Kraftwerkber
# Hier nichts ändern!
optimizationAlgorithmgreedy(fEnergyEffects,xLeft,xRight,nPieces,minD
```

#### Algorithmus 3: Optimierung des Spiegelabstands und der ersten Spiegelposition

**Hinweis:**

Die Optimierung dauert sehr lange. Du kannst die Zeit, die der Optimierer maximal suchen soll, einschränken, indem du eine Zeit für die Variable `maxTime` vorgibst. Ersetze dazu das `NaN` durch eine Zeit in Sekunden.

```
In [ ]:
nMirrors = 4; # Anzahl der Spiegel, die gesetzt werden sollen
maxTime = NaN; # 60; # maximale Zeit in Sekunden, die der Optimierer s
# Hier nichts ändern!
optimizeDeltaEffects(fEnergyEffectsDelta,xLeft,xRight,b,h,s,minDistance,nMirr
```

#### Algorithmus 4: Freie Optimierung der Positionsvariablen

**Hinweis:**

Die Optimierung dauert sehr lange. Du kannst die Zeit, die der Optimierer maximal suchen soll, einschränken, indem du eine Zeit für die Variable `maxTime` vorgibst. Ersetze dazu das `NaN` durch eine Zeit in Sekunden.

```
In [ ]:
nMirrors = 2; # Anzahl der Spiegel, die gesetzt werden sollen
maxTime = NaN; # 50; # maximale Zeit in Sekunden, die der Optimierer suc
# Hier nichts ändern!
optimizeAndPrintPos(fEnergyEffectsPenalty2,xLeft,xRight,b,h,s,nMirrors,maxTim
```

## C.13 Antwortblätter

### C.13.1 Antwortblatt zu Arbeitsblatt 3



**Antwortblatt zur Optimierung der Spiegelposition**

**AB 3 – Teil e**  
Tabelle zur Dokumentation deiner Ergebnisse:

Position x1	Position x2	Energie [Wh]

**AB 3 – Teil f**  
Notiere hier die Schritte deines Verfahrens zur Optimierung von 2 Spiegelpositionen.

Lässt sich dein Verfahren auch für 8 Spiegel erweitern? Begründe deine Antwort.

Welche Vor- oder Nachteile hat dein Verfahren?

### C.13.2 Antwortblatt zu Arbeitsblatt 4

<p> <b>Antwortblatt zu Algorithmus 1:</b> _____ (Name des Algorithmus)</p> <p>1. Welche Vor- oder Nachteile hat das Verfahren?</p> <p>2. Inwieweit sind die vier Fragestellungen der Optimierung erfüllt?</p> <p>a. <u>Existenz und Eindeutigkeit:</u></p> <p>b. <u>Rechenzeit und Güte der gefundenen Lösung:</u></p> <ul style="list-style-type: none"><li>• Welchen Einfluss hat die Anzahl der Abschnitte <math>n</math> Pieces auf Rechenzeit und Güte der gefundenen der Lösung?</li></ul> <p>• Lässt sich der Algorithmus problemlos auf viele Spiegel erweitern?</p> <p><u>Verbesserungsvorschläge:</u></p>	<p> <b>Antwortblatt zu Algorithmus 2:</b> _____ (Name des Algorithmus)</p> <p>1. Welche Vor- oder Nachteile hat das Verfahren?</p> <p>2. Inwieweit sind die vier Fragestellungen der Optimierung erfüllt?</p> <p>a. <u>Existenz und Eindeutigkeit:</u></p> <p>b. <u>Rechenzeit und Güte der gefundenen Lösung:</u></p> <ul style="list-style-type: none"><li>• Welchen Einfluss hat die Anzahl der Abschnitte <math>n</math> Pieces auf Rechenzeit und Güte der gefundenen der Lösung?</li></ul> <p>• Lässt sich der Algorithmus problemlos auf viele Spiegel erweitern?</p> <p><u>Verbesserungsvorschläge:</u></p>
--	--

<p> <b>Antwortblatt zu Algorithmus 3:</b> _____ (Name des Algorithmus)</p> <p>1. Welche Vor- oder Nachteile hat diese Optimierung?</p> <p>2. Inwieweit sind die vier Fragestellungen der Optimierung erfüllt?</p> <p>a. <u>Existenz und Eindeutigkeit:</u></p> <p>b. <u>Rechenzeit und Güte der gefundenen Lösung:</u></p> <p><u>Verbesserungsvorschläge:</u></p>	<p> <b>Antwortblatt zu Algorithmus 4:</b> _____ (Name des Algorithmus)</p> <p>1. Welche Vor- oder Nachteile hat diese Optimierung?</p> <p>2. Inwieweit sind die vier Fragestellungen der Optimierung erfüllt?</p> <p>a. <u>Existenz und Eindeutigkeit:</u></p> <p>b. <u>Rechenzeit und Güte der gefundenen Lösung:</u></p> <p><u>Verbesserungsvorschläge:</u></p>
--	--

## C.14 Präsentationsfolien

Die Folien für den Einstiegsvortrag entsprechen denen, die im Mittelstufenworkshop genutzt werden (vgl. Anh. B.5.1).

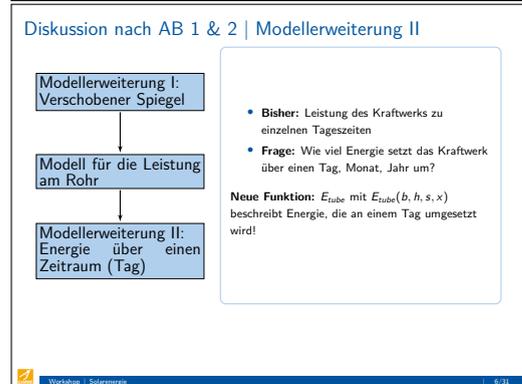
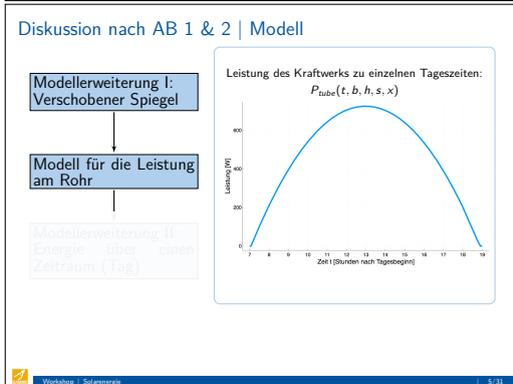
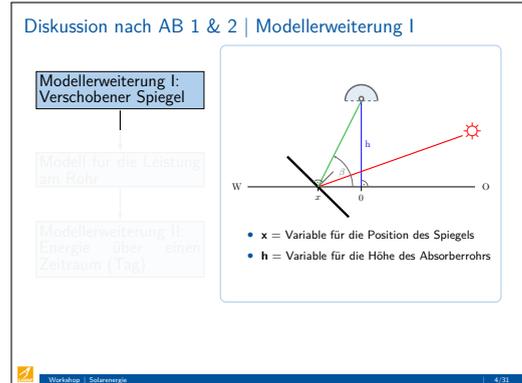
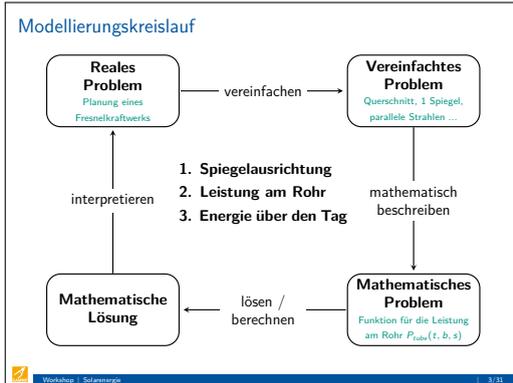
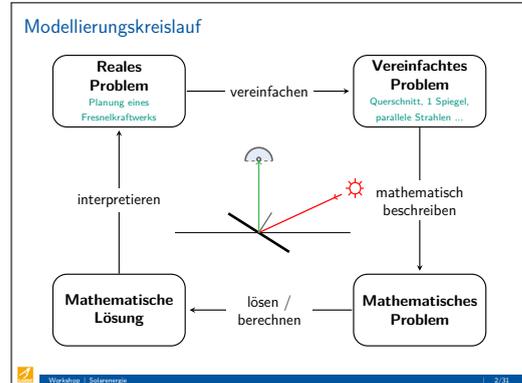
### C.14.1 Folien für die Zwischen- und Abschlussdiskussionen

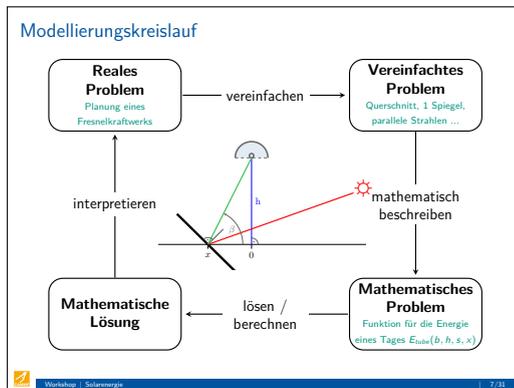


Quelle: abt

**Wie funktioniert ein Solarkraftwerk... und was hat das mit Mathe zu tun?**

Diskussion



### Planung und Bau eines Kraftwerks

**Optimierung** verschiedener Kraftwerkparameter wäre möglich:

- die Höhe des Absorberrohrs
- **Position der Spiegel**
- Form des Sekundärreflektors
- Breite des Spiegels
- ...

### Diskussion nach AB 3 | Optimierung der Spiegelpositionen

Optimale Position von 1 Spiegel?

Modellerweiterung: Energie erzeugt durch mehrere Spiegel

Optimale Positionen von 2 Spiegeln

**Zielfunktion:**  $E_{tube}(b, h, s, x)$   
**Optimale Lösung** für die Variable  $x$ ?

### Optimierung der Spiegelpositionen

Optimale Position von 1 Spiegel?

Modellerweiterung: Energie erzeugt durch mehrere Spiegel

Optimale Positionen von 2 Spiegeln

**Zielfunktion:**  $E_{tube}(b, h, s, [x_1, x_2])$   
**Optimale Lösung** für die Variablen  $x_1$  und  $x_2$ ?

### Planung und Bau eines Kraftwerks

Unser Kraftwerk soll so geplant werden, dass der Energiegewinn maximiert wird.

**Frage:** Was sind mögliche Parameter des Kraftwerks, die wir **optimieren** könnten?

### Ihr seid dran!

**DON'T PANIC!**

- Bearbeite die Arbeitsblätter!
- Arbeite im Team mit deinem/r Partner/in!
- Nutze die Tippkarten!
- Frag' die Betreuer/innen!
- Nutze das Internet!
- Helft euch gegenseitig!

**ORGANIZE!**

Öffnet **AB 3** zur Optimierung der Spiegelpositionen!

### Optimierung der Spiegelpositionen (Modellerweiterung)

Optimale Position von 1 Spiegel?

Modellerweiterung: Energie erzeugt durch mehrere Spiegel

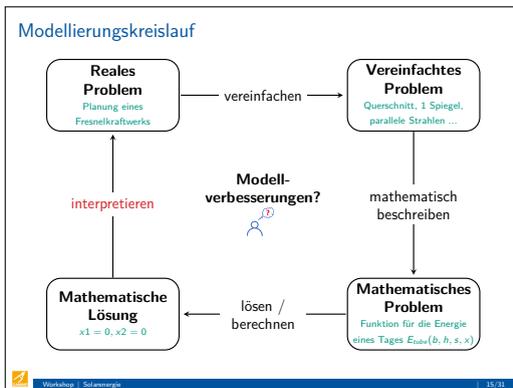
Optimale Positionen von 2 Spiegeln

- Welche Modellannahmen oder Vereinfachungen habt ihr getroffen?

### Optimierung der Spiegelpositionen

Das Optimierungsproblem etwas formaler:

Maximiere die Zielfunktion  $E_{tube}(b, h, s, [x_1, x_2])$   
 Variablen  $x_1$  und  $x_2$   
**Lösung?**



### Optimierung der Spiegelpositionen | Mindestabstand

**Optimierungsproblem mit Nebenbedingung:**

Maximiere  $E_{tube}(b, h, s, [x_1, x_2])$   
 Variablen  $x_1$  und  $x_2$   
 Nebenbedingung  $|x_1 - x_2| \geq \text{minDist}$   
**Lösung?**

### Optimierung der Spiegelpositionen | Mindestabstand

**Optimierungsproblem mit Nebenbedingung:**

Maximiere  $E_{tube}(b, h, s, [x_1, x_2])$   
 Variablen  $x_1$  und  $x_2$   
 Nebenbedingung  $|x_1 - x_2| > \text{minDist}$   
**Lösung**  
 $x_1 = -\frac{\text{minDist}}{2}, x_2 = +\frac{\text{minDist}}{2}$   
 $x_1 = +\frac{\text{minDist}}{2}, x_2 = -\frac{\text{minDist}}{2}$   
 Existenz?  
 Eindeutigkeit?

### Optimierung der Spiegelpositionen

**Wichtige Fragen beim Lösen von Optimierungsproblemen:**

- Existenz einer optimalen Lösung:** Existiert überhaupt eine optimale Lösung? In unserem Fall: gibt es eine Kombination der Positionen  $x_1$  und  $x_2$ , die optimal ist?
- Eindeutigkeit der Lösung:** Ist die Lösung eindeutig? In unserem Fall: gibt es vielleicht eine andere Wahl der Werte für  $x_1$  und  $x_2$ , die zum gleichen Ergebnis der Energie führt?
- Güte der gefundenen Lösung:** Wie nah ist die Lösung, die ein Verfahren (z. B. das kombinatorische Verfahren) findet, an der optimalen Lösung (falls eine solche existiert)?
- Rechenzeit eines Optimierungsverfahrens:** Wie hoch ist die Rechenzeit, die das Verfahren braucht, um eine Lösung zu finden?

### Optimierung automatisieren

Was sind eure Ideen für Optimierungsverfahren?

### Ihr seid dran!

**DON'T PANIC!**

- Bearbeite die Arbeitsblätter!
- Arbeite im Team mit deinem/r Partner/in!
- Nutze die Tippkarten!
- Nutze das Internet!
- Helft euch gegenseitig!

**ORGANIZE!**

Öffnet AB 4 und startet mit einem der 4 Algorithmen.

### Diskussion nach AB 4 | Brute-Force-Algorithmus

- Funktionsweise des Verfahrens?
- Vor- und Nachteile des Verfahrens?
- Eure Einschätzung zu folgenden Punkten:
  - Existenz und Eindeutigkeit einer Lösung?
  - Rechenzeit und Güte der gefundenen Lösung?
- Verbesserungsideen?

### Diskussion nach AB 4 | Greedy-Algorithmus

- Funktionsweise des Verfahrens?
- Vor- und Nachteile des Verfahrens?
- Eure Einschätzung zu folgenden Punkten:
  - Existenz und Eindeutigkeit einer Lösung?
  - Rechenzeit und Güte der gefundenen Lösung?
- Verbesserungsideen?

Diskussion nach AB 4 | Musterbasiertes Verfahren

- Funktionsweise des Verfahrens?
- Vor- und Nachteile des Verfahrens?
- Eure Einschätzung zu folgenden Punkten:
  - Existenz und Eindeutigkeit einer Lösung?
  - Rechenzeit und Güte der gefundenen Lösung?
- Verbesserungsideen?



Workshop | Solarenergie | 23/31

Diskussion nach AB 4 | Freie Variablen Verfahren

- Funktionsweise des Verfahrens?
- Vor- und Nachteile des Verfahrens?
- Eure Einschätzung zu folgenden Punkten:
  - Existenz und Eindeutigkeit einer Lösung?
  - Rechenzeit und Güte der gefundenen Lösung?
- Verbesserungsideen?



Workshop | Solarenergie | 24/31

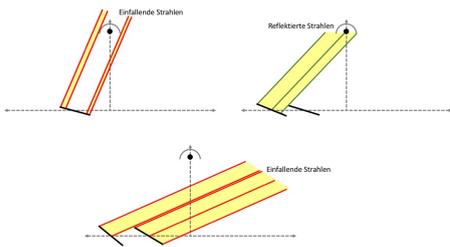
Modellverbesserungen

Weitere Ideen zur Verbesserung unseres Modells?



Workshop | Solarenergie | 25/31

Modellverbesserungen



Workshop | Solarenergie | 26/31

Modellierung von Spiegeleffekten



**DON'T PANIC!**

- Bearbeite die Arbeitsblätter!
- Arbeite im Team mit deinem/r Partner/in!
- Nutze die Tippkarten!
- Nutze das Internet!
- Helft euch gegenseitig!

**ORGANIZE!**

Öffnet Arbeitsblatt **AB 5** zu den Verschattungseffekten!

Workshop | Solarenergie | 27/31

Diskussion nach AB 5 | Schattenwurf

- Welche Vor- oder Nachteile haben die beiden Ansätze
  - Graphen linearer Funktionen
  - Parameterdarstellung von Geraden
 zur Modellierung von Spiegeln und Sonnenstrahlen?
- Inwieweit unterscheiden sich die optimalen Positionen von
  - einem Spiegel
  - zwei Spiegel
 basierend auf dem alten und neuen Modell? Wie lassen sich die Ergebnisse erklären?

Workshop | Solarenergie | 28/31

Diskussion nach AB 6 | Optimierung der Spiegelpositionen

- Welche Beobachtungen habt ihr im Hinblick auf die **Rechenzeit** gesammelt?
- Inwieweit unterscheiden sich die Ergebnisse der Optimierung basierend auf dem alten und neuen Modell?
- Für welchen Optimierungsansatz würdet ihre euch entscheiden? Warum?

Workshop | Solarenergie | 29/31

---

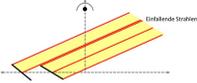
Offene Modellierungsprojekte

**Problem 1:** Eigene Optimierungsverfahren zur Positionsoptimierung implementieren, Verfahren mit Stift und Papier

**Problem 2:** Schattenwurf modellieren, Mathem. Modell mit Stift und Papier, Implementieren nur optional

**Problem 3:** Optimierung beliebiger Kraftwerkparameter, Implementieren, Verfahren mit Stift und Papier

**Problem 4:** Optimierung der Rohrhöhe, Angeleitet durch Arbeitsaufträge auf ABs



Workshop | Solarenergie | 31/31

Ein Blick in die Forschung



- Forschung begann historisch mit **diskreten** und **musterbasierten Ansätzen**
- Viele weitere Forschungsfragen ...
  - Bau von Solarturmkraftwerken
  - optimale Verkabelung der Spiegel
  - optimale Reinigungsrouten
  - optimale Form der Spiegel (flache Spiegel oder Parabolspiegel?)
  - verschiedene Wettermodelle für die Simulation der Energie

Quelle: DLR

Workshop | Solarenergie | 30/31



# D Arbeitsblätter zu offenen Modellierungsproblemen – Solarenergie

## D.1 Übersichtsblatt

### Übersichtsblatt: Offene Modellierungsaufgaben

In diesem Teil des Solarprojektes kannst du dich gemeinsam mit deiner Gruppe unterschiedlichen Fragestellungen widmen! Beispielsweise könnt ihr euer eigenes entwickeltes Optimierungsverfahren oder andere Verbesserungen des Modells theoretisch modellieren oder sogar implementieren. Ihr entscheidet dabei, welche eurer Ideen ihr umsetzt.

Mögliche Problemstellungen

- **Problem 1: Implementieren eines Verfahrens für die Optimierung der Spiegelpositionen.**  
Hier gelangt ihr zu dem Arbeitsblatt.
- **Problem 2: Den Schattenwurf von einem Spiegel auf einen anderen modellieren.**  
Hier gelangt ihr zu dem Arbeitsblatt.
- **Problem 3: Die Optimierung auf andere Kraftwerkparameter (Rohrhöhe  $h$ , Spiegelbreite  $s$ , ...) erweitern.**  
Hier gelangt ihr zu dem Arbeitsblatt.

Für Probleme 1 - 3 liegt ein Jupyter Notebook bereit, in dem die Probleme nochmal beschrieben werden. Diese Notebooks könnt ihr auch für euren eigenen Programmcode nutzen. **Hinweis:** Eure Ergebnisse werden nicht mehr überprüft. Stattdessen werden wir eure Modelansätze und Ergebnisse gemeinsam diskutieren.

Falls ihr lieber wie bisher mit vorstrukturierten Aufgaben weiterarbeiten wollt, so könnt ihr das folgende Problem bearbeiten:

- **Problem 4: Optimierung der Rohrhöhe und Berücksichtigung von Einstellungsfehlern mit Arbeitsaufträgen.**  
Hier gelangt ihr zu dem Arbeitsblatt.

Vorab findet ihr auf diesem Arbeitsblatt Beispiele, um mit Julia Grafiken zu erstellen. Falls ihr in eurem eigenen Code etwas visualisieren möchtet, könnt ihr euch an diesen Beispielen orientieren.

### Plot einer Funktion

Führe den folgenden Code aus. Die Funktion  $f$  mit  $f(x) = x^2$  wird grafisch dargestellt.

```
In [ ]: # Das Paket zum Plotten wird geladen
using Plots

# Definiere eine Funktion:
f(x) = x^2;

# Definiere wie kleinschrittig die Funktion ausgewertet werden soll:
xi = Array(-10:0.1:10); # Werte von -10, -9.9, ..., 9.9, 10
y = f.(xi)
```

```
# Plotten
plot(x1,y,linewidth = 3, title = "Titelbeschriftung", size = [700,600], xlabel
```

### Plotten einer Funktion mit zwei Variablen

Führe den folgenden Code aus. Die Funktion  $g$  mit  $g(x,y) = x + y$  wird grafisch dargestellt.

```
In [ ]: # Definiere eine Funktion:
g(x,y) = x + y;

# Definieren wir kleinschrittig die Funktion ausgewertet werden soll:
x1 = Array(-10:0.1:10) # Werte, für die die x-Koordinate ausgewertet werden so
x2 = Array(-10:0.1:12) # Werte, für die die y-Koordinate ausgewertet werden so

# Plotten
plotly()
Plots.plot(x1,x2,g,st=surface,linewidth = 3, title = "Titelbeschriftung", sl:
xlabel = "x-Achse beschriften", ylabel = "y-Achse beschriften", xlabel =
```

### Plotten von einzelnen Datenpunkten

Führe den folgenden Code aus.

```
In [ ]: x = [3, -4, 4, 2, 1, 5]
y = [1, 4, 4, -2, 1, 0, 1]
plot(x,y,seriestype=:scatter, title="My Scatter Plot")
```

# D.2 Problem 1: Optimierungsverfahren implementieren

## Problem 1:

### Implementieren eines Verfahrens für die Optimierung der Spiegelpositionen

Dir stehen alle bisher entwickelten Funktionen zur Verfügung. Diese werden in dem ersten Code-Feld geladen und können von dir eingesetzt werden.



#### Teamwork:

Nutzt das **Whiteboard-Tool in Mattermost**, um eure Ideen gemeinsam zu diskutieren und um Skizzen zu machen.



Damit du dieses Arbeitsblatt bearbeiten kannst, **musst du als erstes den folgenden Code ausführen.**

```
In [ ]:
#### -- Hier nichts ändern! -- ####
include("../code/setup_Abl_OpenModelling.jl");
alpha(t) = compute_alpha(t); # Someneinfallswinkel
tsun(t) = compute_tsun(t); # Stärke der Sonnenstrahlung
gamma(t,h,x) = compute_gamma(t,h,x); # Neigungswinkel der Spiegel
beta(h,x) = compute_beta(t,h,x); # Ausfallswinkel der reflektierten Strahlen

# --- Festlegung einiger Parameter des Kraftwerks --- #
b = 0.8; # Breite des Sekundärreflektors
s = 1.1; # Breite des Spiegels
h = 6; # Höhe des Absorberrohrs
xLeft,xRight = [-10,0,10,0]; # Linke und rechte Grenze des Kraftwerkareals

# Funktion für die Energie am Rohr über einen Tag von bisherigen Arbeitsblättern
Etube(b,h,s,positions) = EtubeMultiMirrorsOneDay(b,h,s,positions);
```

Auf dem Arbeitsblatt zur Optimierung der Spiegelpositionen hast du mit Stift und Papier ein eigenes Optimierungsverfahren entwickelt. Dieses könntest du im folgenden Code-Feld implementieren. Dein Verfahren soll die optimalen Positionen einer beliebigen Anzahl von Spiegeln ermitteln. Die Spiegel sollten nur in dem Bereich des Kraftwerkareals von `xLeft = -10` bis `xRight = 10` positioniert werden.

Für die Optimierung kannst du die Funktion `Etube(b,h,s,positions)` verwenden. Diese wurde bereits geladen. Die Variablen, die optimiert werden sollen, sind die Einträge des Arrays `positions = [x1,x2,x3,x4,x5,x6,x7,...]`.



Implementiere dein Verfahren im nächsten Code-Feld. Nutze dazu die folgenden Hinweise. Natürlich kannst du auch im Internet nach nützlichen Befehlen suchen oder das CAMMP Team fragen.

**Tipp:** Führe die Optimierung zunächst für wenige (z. B. 2) Spiegel durch. Erweitere das Verfahren anschließend für viele Spiegel.

#### Hinweise:

- Falls du in deinem Verfahren eine **for-Schleife** benötigst, so findest du [hier](#) ein Beispiel dazu. For-Schleifen sind nützlich für Berechnungen, die mehrfach wiederholt werden sollen.
- Falls du in deinem Verfahren eine **if-Abfragen** benötigst, so findest du [hier](#) nochmal eine Erklärung dazu. If-Abfragen sind nützlich für Berechnungen, die nur durchgeführt werden sollen, wenn eine Bedingung erfüllt ist.
- Einträge in einem Vektor kannst du folgendermaßen speichern: `optimalPositions[2] = -0.4`. Als Position für den 2. Spiegel wird in diesem Vektor der Wert `-0.4` gespeichert
- So kannst du direkt alle Positionen gleichzeitig abspeichern: `optimalPositions = [-2.2,1]`. Als Position für den 1. Spiegel wird der Wert `-2.2` und als Wert für den 2. Spiegel der Wert `1.0` gespeichert.

```
In [ ]:
# Anzahl Spiegel, die positioniert werden sollen
nMirrors = 2;

function myOptimizationAlgorithm(xLeft,xRight,nMirrors)
# Hier nichts ändern!
optimalPositions = zeros(nMirrors) .* NaN; # Ein Vektor (eine Art Liste)

##### -- Hier kannst du deinen Algorithmus implementieren. -- #####

# In dem Vektor optimalPositions sollen die besten Positionen
# als Einträge gespeichert werden.
optimalPositions = #= NaN; =#

##### -- Ab hier nichts mehr ändern! -- #####

return optimalPositions # Der Vektor mit den optimalen Positionen wird au
end;

# Hier nichts ändern! - Algorithmus anwenden
myOptimizationAlgorithm(xLeft,xRight, nMirrors)
```

## D.3 Problem 2: Verschattung modellieren

### Problem 2:

#### Den Schattenwurf von einem Spiegel auf einen anderen modellieren

Dir stehen alle bisher entwickelten Funktionen zur Verfügung. Diese werden in dem ersten Code-Feld geladen und können von dir eingesetzt werden.



#### Teamwork:

Nutzt das Whiteboard-Tool in Mattermost, um eure Ideen gemeinsam zu diskutieren und um Skizzen zu machen.



Damit du dieses Arbeitsblatt bearbeiten kannst, musst du als erstes den folgenden Code ausführen.

```
In [ ]:
### -- Hier nichts ändern! -- ###
include("../code/setup_Abl_OpenModeling.jl");
alpha(t) = compute_alphaDeg(t); # Sonneneinfallswinkel
Isun(t) = compute_Isun(t); # Stärke der Sonnenstrahlung
gamma(t,h,x) = compute_gammaDeg(t,h,x); # Neigungswinkel der Spiegel
beta(h,x) = compute_betaDeg(h,x); # Ausfallswinkel der reflektierten Strahlen

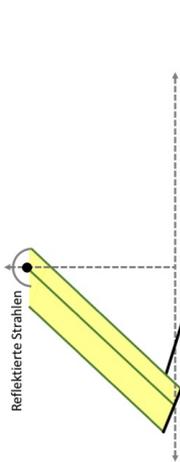
# --- Festlegung einiger Parameter des Kraftwerks --- #
b = 0.8; # Breite des Sekundärreflektors
s = 1.1; # Breite des Spiegels
h = 6; # Höhe des Absorberrohrs
positions = [-2, 3] # Hier wurden exemplarisch 2 Spiegel auf die Positionen -
xLeft,xRight = [-10,0,10]; # Linke und rechte Grenze des Kraftwerkareals
# Funktion für die Energie am Rohr über einen Tag von bisherigem Arbeitsblatt
Etube(b,h,s,positions) = EtubeMult.MirrorOneDay(b,h,s,positions);
```

Bisher sind wir davon ausgegangen, dass die Spiegel sich nicht gegenseitig beeinflussen. Dies ist jedoch nicht korrekt! Zwischen den Spiegeln können verschiedene Effekte auftreten. Die umgesetzte Energie ist also kleiner als wir bisher modelliert haben! Ändert dies womöglich auch die optimalen Positionen für die aufzustellenden Spiegel?

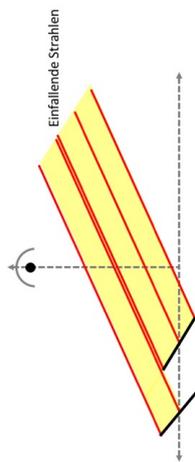
Du kannst dir zunächst einen Überblick über diese Effekte verschaffen und anschließend einen der Effekte selbst mathematisch modellieren!

Betrachte die folgenden drei Abbildungen. Welche Effekte beobachtest du in den einzelnen Skizzen der einfallenden oder reflektierten Strahlen?

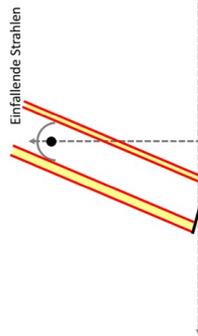
#### Effekt 1:



Effekt 2:



Effekt 3:



#### Modellierung von Verschattung



Lasse dir im nachfolgenden Code das Ergebnis für die Energie am Rohr für 4 Spiegel mit dem alten und dem neuen Modell ausgeben.

```
In [ ]:
# Festlegung der Positionen
positions = [-4,-2,3,6]

# Altes Modell
EtotalOld = Etube(b,h,s,positions) # hier wird die Energie mit dem Modell ohne
println("Das Ergebnis mit dem alten Modell lautet: $EtotalOld J")

# Neues Modell
EtotalNew = EtubeNew(b,h,s,positions)
println("Das Ergebnis mit dem neuen Modell lautet: $EtotalNew J")
```



- Starte mit der Modellierung des Schattenwurfs zu einer festen Tageszeit (z. B.  $t = 11$ ) und für zwei Spiegel auf von dir festgelegten Positionen (z. B.  $x1 = -2$  und  $x1 = -0.5$ ). Berechne welcher Bereich der Spiegel verschattet ist.
- Anschließend kannst du dein Modell auch auf andere Tageszeiten verallgemeinern (z. B. für den Nachmittag).

**Tip:** Entwickle dein Modell zunächst mit Stift und Papier (bzw. mit dem Whiteboard-Tool). Wenn noch Zeit ist, kannst du dein Modell auch implementieren. Dazu kannst du das folgende Code-Feld nutzen und die bisherige Funktion `Etube` beliebig verändern.

**Hinweise:**

- Eine Erinnerung dazu, welche Winkel `beta`, `gamma` und `alpha` beschreiben, findest du [hier](#).
- Falls du in deinem Verfahren eine **for-Schleife** benötigst, so findest du [hier](#) ein Beispiel dazu. For-Schleifen sind nützlich für Berechnungen, die mehrfach wiederholt werden sollen.
- Falls du in deinem Verfahren eine **if-Abfrage** benötigst, so findest du [hier](#) nochmal eine Erklärung dazu. If-Abfragen sind nützlich für Berechnungen, die nur durchgeführt werden sollen, wenn eine Bedingung erfüllt ist.

```
In [ ]:
### --- Nutze dieses Code-Feld für die Modellierung von Effekten --- ###
# Bisherige Funktion zur Berechnung der Energie am Rohr. Du kannst beliebige
function EtubeNew(b,h,s,positions)
    sumEnergy = 0;
    nMirrors = size(positions,1);
    nDays = 1;
    DeltaT = 0.5;

    for d = 1 : nDays
        for t = 7 : DeltaT + 22
            for i = 1 : nMirrors
                ratio = min(1, b / s + cosd(abs(beta(h,positions[i]) - 90)) / cosd(beta(
                Ptube = Isun(t) * s + cosd((beta(h,positions[1]) - alpha(t)/2) * Ta
                sumEnergy = sumEnergy + Ptube * DeltaT;
            end
        end
    end
    return sumEnergy
end;
```

Vergleich des alten Modells (ohne Schattenwurf) und des neuen Modells (mit Schattenwurf)

## D.4 Problem 3: Beliebige Kraftwerksparameter optimieren

### Problem 3:

#### Die Optimierung auf andere Kraftwerksparameter (Rohrhöhe $h$ , Spiegelbreite $s$ , ...) erweitern

Du stehst alle bisher entwickelten Funktionen zur Verfügung. Diese werden in dem ersten Code-Feld geladen und können von dir eingesetzt werden.



#### Teamwork:

Nutzt das **Whiteboard-Tool** in **Mattermost**, um eure Ideen gemeinsam zu diskutieren und um Skizzen zu machen.



Damit du dieses Arbeitsblatt bearbeiten kannst, **musst du als erstes den folgenden Code ausführen**.

```
In [ ]:
### -- Hier nichts ändern! -- ###
include("../code/setup_Abl_OpenModeling.jl");
alpha(t) = compute_alphaDeg(t); # Sonneneinfallswinkel
Isun(t) = compute_Isun(t); # Stärke der Sonnenstrahlung
gamma(t,h,x) = compute_gammaDeg(t,h,x); # Neigungswinkel der Spiegel
beta(h,x) = compute_betaDeg(h,x); # Ausfallswinkel der reflektierte Strahlen

# --- Festlegung einiger Parameter des Kraftwerks --- #
b = 0.8; # Breite des Sekundärreflektors
s = 1.1; # Breite des Spiegels
h = 6; # Höhe des Absorberrohrs
positions = [-2, 3] # Hier wurden exemplarisch 2 Spiegel auf die Positionen -
xLeft,xRight = [-10,0,10,0]; # Linke und rechte Grenze des Kraftwerksareals
# Funktion für die Energie am Rohr über einen Tag von bisheriger Arbeitsblatt
Etube(b,h,s,positions) = EtubeMultiMirrorsOneDay(b,h,s,positions);
```

#### Optimales Design des Kraftwerks

Ziel ist es, für alle Kraftwerksgrößen, die man auch in der Realität variieren und unterschiedlich konstruieren kann, optimale Werte zu finden. Überlege dir, wie du die Optimierung auf weitere Kraftwerksparameter erweitern kannst. Nutze das folgende Code-Feld, um dein Optimierungsverfahren zu implementieren.

Bei der Optimierung kannst du die Funktion `Etube(b,h,s,positions)` verwenden. Diese wurde im Code oben geladen.



Implementiere dein Verfahren im nächsten Code-Feld. Nutze dazu die folgenden Hinweise. Natürlich kannst du auch im Internet nach nützlichen Beispielen suchen oder das CAMMP Team fragen.

#### Tipps:

- Suche dir zunächst einen Parameter raus, dessen optimalen Wert du bestimmen willst. Zum Beispiel die Breite des Sekundärreflektors  $b$ .
- Überlege dir anschließend, wie du nicht nur die beste Reflektorbreite  $b$  findest, sondern gleichzeitig auch die besten Werte für andere Parameter (z. B. die Spiegelbreite  $s$  oder die Rohrhöhe  $h$ ).
- Oft kann es helfen die Zielfunktion der Optimierung grafisch darstellen zu lassen.

#### Hinweise:

- Falls du in deinem Verfahren eine **for-Schleife** benötigst, so findest du [hier](#) ein Beispiel dazu. For-Schleifen sind nützlich für Berechnungen, die mehrfach wiederholt werden sollen.
- Falls du in deinem Verfahren eine **if-Abfrage** benötigst, so findest du [hier](#) nochmal eine Erklärung dazu. If-Abfragen sind nützlich für Berechnungen, die nur durchgeführt werden sollen, wenn eine Bedingung erfüllt ist.

```
In [ ]: # Ausgabe der Energie für bisher gewählte Parameter
Etube(b,h,s,positions)
```

# E Ergänzungen zum Basiswissen rund um Empfehlungssysteme

## E.1 Das stochastische Gradientenverfahren

Das nichtlineare Optimierungsproblem (16.21) kann alternativ mit einem auf dem Gradientenverfahren basierenden Algorithmus, dem stochastischen Gradientenverfahren (SGD), angegangen werden. Dieses eignet sich insbesondere für Optimierungsprobleme, bei denen die Zielfunktion in Summenform dargestellt werden kann, zum Beispiel:

$$f(\mathbf{x}) = \sum_{i=1}^N g_i(\mathbf{x}).$$

Anstatt wie bei dem in Abschnitt 2.2.1 beschriebenen *gewöhnlichen* Gradientenverfahren in jeder Iteration einen Schritt in Richtung des negativen Gradienten ( $-\nabla f$ ) bezüglich **aller** Variablen voranzuschreiten, wird die Berechnung beim SGD-Verfahren in den einzelnen Iterationen entkoppelt. Es wird in jedem Iterationsschritt ein zufälliger Parameter  $i \in \{1, \dots, N\}$  gewählt und das Update

$$\mathbf{x} \leftarrow \mathbf{x} - \alpha \cdot \nabla g_i(\mathbf{x})$$

ausgeführt. Dies lässt sich auf unser Empfehlungssystem übertragen und wurde im Rahmen des Netflix Preises insbesondere durch Funk (2006) bekannt.

Für einen beliebigen Trainingsdatenpunkt  $r_{ij}$  mit  $(i, j) \in \kappa$  definieren wir zunächst den Vorhersagefehler als

$$e_{ij} = r_{ij} - \mathbf{u}_{i,\bullet} \mathbf{m}_{\bullet,j} = r_{ij} - \sum_{k=1}^{n_f} u_{ik} m_{kj}.$$

Die Zielfunktion des Optimierungsproblem (16.21) kann geschrieben werden als (vgl. Aggarwal, 2016, S. 101)

$$f(U, M) = \frac{1}{2} \cdot \left( \sum_{(i,j) \in \kappa} (e_{ij})^2 + \lambda \sum_{i=1}^{n_1} \sum_{k=1}^{n_f} u_{ik}^2 + \lambda \sum_{j=1}^{n_2} \sum_{k=1}^{n_f} m_{kj}^2 \right).$$

Die partiellen Ableitungen der Zielfunktion  $f$  nach einer beliebigen Variable  $u_{iq}$  bzw.  $m_{qj}$  sind gegeben durch

$$\begin{aligned} \frac{\partial f}{\partial u_{iq}} &= \sum_{j:(i,j) \in \kappa} (e_{ij}) (-m_{qj}) + \lambda u_{iq} & \forall i = 1, \dots, n_1, q = 1, \dots, n_f \\ \frac{\partial f}{\partial m_{qj}} &= \sum_{i:(i,j) \in \kappa} (e_{ij}) (-u_{iq}) + \lambda m_{qj} & \forall j = 1, \dots, n_2, q = 1, \dots, n_f. \end{aligned}$$

Anschließend werden für alle  $(i, j) \in \kappa$  in zufälliger Reihenfolge die Updates

$$\begin{aligned} u_{iq} &\leftarrow u_{iq} + \alpha (e_{ij} m_{qj} - \lambda u_{iq}) \\ m_{qj} &\leftarrow m_{qj} + \alpha (e_{ij} u_{iq} - \lambda m_{qj}) \end{aligned}$$

durchgeführt. Hierbei ist  $\alpha > 0$  die Schrittweite, die entweder fest gewählt oder während der Iterationen adaptiert werden kann (vgl. Aggarwal, 2016, S. 102 f.). Die Updates werden vielfach auch vektoriell durchgeführt:

$$\begin{aligned} \mathbf{u}_{i,\bullet} &\leftarrow \mathbf{u}_{i,\bullet} + \alpha (e_{ij} \mathbf{m}_{\bullet,j} - \lambda \mathbf{u}_{i,\bullet}) \\ \mathbf{m}_{\bullet,j} &\leftarrow \mathbf{m}_{\bullet,j} + \alpha (e_{ij} \mathbf{u}_{i,\bullet} - \lambda \mathbf{m}_{\bullet,j}). \end{aligned}$$

In Abbildung E.1 sind die Ergebnisse des SGD-Verfahrens bei Anwendung auf die Bewertungsmatrix aus Beispiel 16.2.2 für verschiedene Regularisierungsparameter dargestellt. Es wird ersichtlich, dass das SGD-Verfahren mehr Iterationen benötigt bis das Verfahren konvergiert, als es mit dem ALS-Verfahren der Fall ist.

Das SGD-Verfahren hat den Vorteil, dass es bei dünnbesetzten Matrizen effizienter ist und weniger aufwendiger Berechnungen Bedarf. Hingegen bietet sich das ALS-Verfahren an, wenn die zu faktorisierende Matrix sehr dicht besetzt ist, da das SGD-Verfahren über jeden einzelnen Eintrag der Trainingsdaten iterieren müsste (vgl. Koren et al., 2009, S. 33).

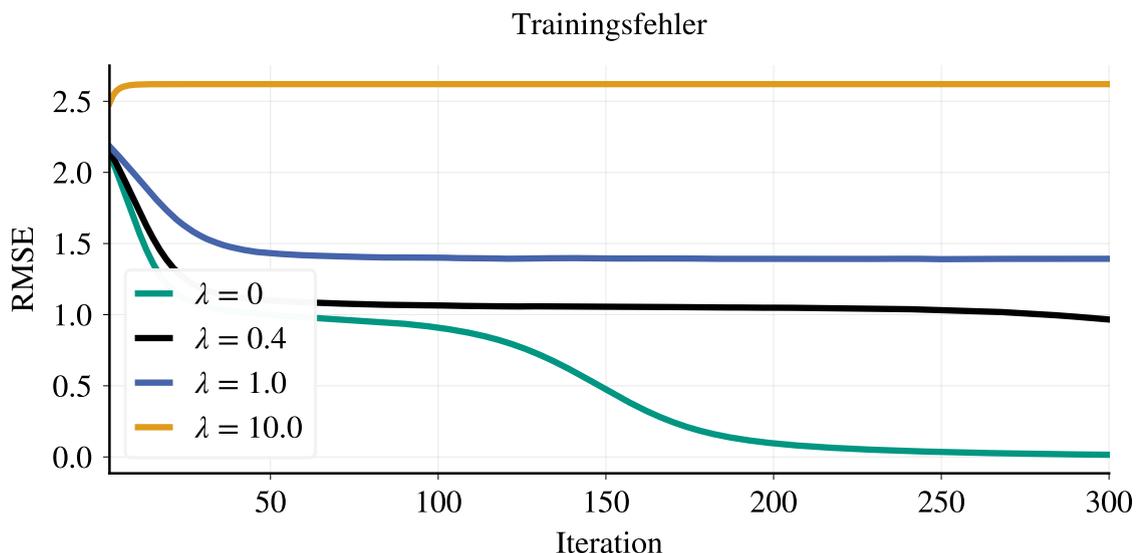


Abb. E.1: Fehler bezüglich der Trainingsdaten für unterschiedliche Werte von  $\lambda$ . In allen Fällen beträgt die Anzahl der Eigenschaften  $n_f = 2$  und die Lernrate  $\alpha = 0.01$

## E.2 ALS- und SGD-Verfahren für die Optimierung mit Bias-Termen

In diesem Abschnitt wird zunächst kurz beschrieben, wie das ALS- und das SGD-Verfahren zum Lösen des Optimierungsproblems (16.30) mit Bias-Termen erweitert

werden kann. Für das ALS-Verfahren wird dies im Folgenden an dem Schritt beschrieben, in dem die Movievektoren optimiert werden. Dazu definieren wir zunächst für  $i = 1, \dots, n_1, j = 1, \dots, n_2$

$$\begin{aligned}\bar{r}_{ij} &= r_{ij} - o_i - \mu \\ \bar{\mathbf{u}}_{i,\bullet} &= (\mathbf{u}_{i,\bullet}, 1) \\ \bar{\mathbf{m}}_{\bullet,j} &= \begin{pmatrix} \mathbf{m}_{\bullet,j} \\ p_j \end{pmatrix}.\end{aligned}$$

Damit lässt sich unser Optimierungsproblem schreiben als<sup>1</sup>

$$\min_{\bar{U}, \bar{M}} \sum_{(i,j) \in \kappa} (\bar{r}_{ij} - \bar{\mathbf{u}}_{i,\bullet} \bar{\mathbf{m}}_{\bullet,j})^2 + \lambda \left( \sum_{i=1}^{n_1} \|\bar{\mathbf{u}}_{i,\bullet}\|^2 + \sum_{j=1}^{n_2} \|\bar{\mathbf{m}}_{\bullet,j}\|^2 \right). \quad (\text{E.1})$$

Zum Update der Movievektoren im ALS-Verfahren, lösen wir das folgende regularisierte lineare Ausgleichsproblem:

$$\min_{\bar{\mathbf{m}}_{\bullet,j}} \|W_j (\bar{U} \bar{\mathbf{m}}_{\bullet,j} - \bar{\mathbf{r}}_{\bullet,j})\|^2 + \lambda \sum_{j=1}^{n_2} \|\bar{\mathbf{m}}_{\bullet,j}\|^2. \quad (\text{E.2})$$

Dieses hat  $(n_f + 1)$  Variablen. Die Lösung für einen beliebigen Movievektor  $\bar{\mathbf{m}}_{\bullet,j}$  ist gegeben durch

$$\bar{\mathbf{m}}_{\bullet,j} = \left( \tilde{U}_j^T \tilde{U}_j + \lambda \mathbb{I} \right)^{-1} \tilde{U}_j^T \bar{\mathbf{r}}_{\bullet,j}, \quad (\text{E.3})$$

wobei  $\mathbb{I} \in \mathbb{R}^{(n_f+1) \times (n_f+1)}$ . Analog ließe sich das Update der Uservektoren durchführen.

Auch das Optimierungsproblem in der gewichteten regularisierten Form (16.24) kann mit Bias-Termen formuliert werden. In (E.3) wäre dann  $\lambda$  durch  $\lambda \cdot n_j^{\text{movie}}$  zu ersetzen.

Das in Abschnitt E.1 beschriebene SGD-Verfahren lässt sich ebenfalls auf die neu hinzugefügten Variablen erweitern. Erneut wird in zufälliger Reihenfolge über alle  $(i, j) \in \kappa$  iteriert und folgende Updates der Variablen vorgenommen (vgl. Koren und Bell, 2011, S. 152)

$$o_i \leftarrow o_i + \alpha (e_{ij} - \lambda o_i) \quad (\text{E.4})$$

$$p_j \leftarrow p_j + \alpha (e_{ij} - \lambda p_j) \quad (\text{E.5})$$

$$\mathbf{u}_{i,\bullet} \leftarrow \mathbf{u}_{i,\bullet} + \alpha (e_{ij} \mathbf{m}_{\bullet,j} - \lambda \mathbf{u}_{i,\bullet}) \quad (\text{E.6})$$

$$\mathbf{m}_{\bullet,j} \leftarrow \mathbf{m}_{\bullet,j} + \alpha (e_{ij} \mathbf{u}_{i,\bullet} - \lambda \mathbf{m}_{\bullet,j}), \quad (\text{E.7})$$

wobei  $e_{ij} = r_{ij} - \mathbf{u}_{i,\bullet} \mathbf{m}_{\bullet,j} - \mu - o_i - p_j$ . Die Lernrate  $\alpha$  und der Regularisierungsparameter  $\lambda$  können per Kreuzvalidierung bestimmt werden. Um die Genauigkeit des System weiter zu erhöhen, wäre es möglich für jede der Variablentypen (Item-Bias, User-Bias, latente Faktoren) jeweils eine individuelle Lernrate und einen individuellen Regularisierungsparameter zu wählen (vgl. Aggarwal, 2016, S. 108).

<sup>1</sup>Die Zielfunktion (E.1) ist nicht direkt äquivalent zu (16.30), da  $\sum_{i=1}^{n_1} \|\bar{\mathbf{u}}_{i,\bullet}\|^2 = \sum_{i=1}^{n_1} \|\mathbf{u}_{i,\bullet}\|^2 + o_i^2 + 1^2$ . Die Addition der Konstante 1 hat jedoch keinen Einfluss auf die Minimalpunkte der Fehlerfunktion. Zudem wird ersichtlich, dass der Term in den einzelnen Iterationen des ALS-Verfahrens verschwindet.

## E.3 Numerische Experimente

Das ALS-Verfahren wurde eingesetzt, um für die in Abschnitt 16.2 beschriebenen Formulierungen des Optimierungsproblems eine Lösung zu bestimmen. Konkret wurden die Ergebnisse für die folgenden Modelle berechnet:

- Modell mit Regularisierung (16.21) (R-ALS)
- Modell mit gewichteter Regularisierung (16.24) (WR-ALS)
- Modell mit Regularisierung und Bias-Termen (16.30) (R-ALS-Bias)
- Modell mit gewichteter Regularisierung und Bias-Termen (WR-ALS-Bias)

Exemplarisch wurde zudem das SGD-Verfahren eingesetzt, um für das ungewichtete Problem mit und ohne Bias-Terme eine Zerlegung zu bestimmen.

Für die numerischen Experimente, die in diesem Kapitel beschrieben werden, wurde (wie im Lernmaterial) lediglich ein kleinerer Teil des Netflixdatensatzes verwendet. Grund für die Arbeit mit einem kleineren Datensatz war, dass die Berechnungen zum einen auf beschränkten Ressourcen laufen sollten (auf der Workshop-Plattform, vgl. Kap. 5) und die Lernenden nach annehmbarer Rechenzeit auf Ergebnisse zugreifen sollten. Als weitere Vereinfachung wurde zudem ein Teildatensatz ausgewählt, der weniger spärlich ist, als der originale Datensatz. Dies sollte es den Lernenden, erlauben die Ergebnisse auch anschaulich vergleichen zu können.

Im Folgenden wird kurz beschrieben, wie der Teildatensatz erzeugt wurde und welche wesentlichen Charakteristika dieser aufweist. Anschließend werden die Ergebnisse auf dem Teildatensatz beschrieben und diskutiert. Bei der Berechnung der Ergebnisse lag der Fokus weniger auf dem Verfeinern und Optimieren der Hyperparameter (insbesondere  $n_f$  und  $\lambda$ ), als viel mehr auf der Darstellung wesentlicher Einflüsse eben dieser Parameter und der unterschiedlichen Modelle.

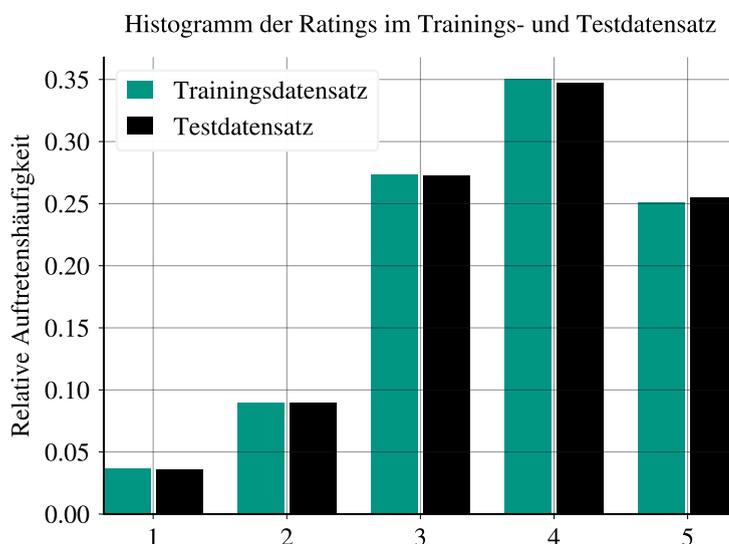


Abb. E.2: Histogramm der Bewertungen im Trainings- und Testdatensatz

### E.3.1 Auswahl eines Teildatensatzes

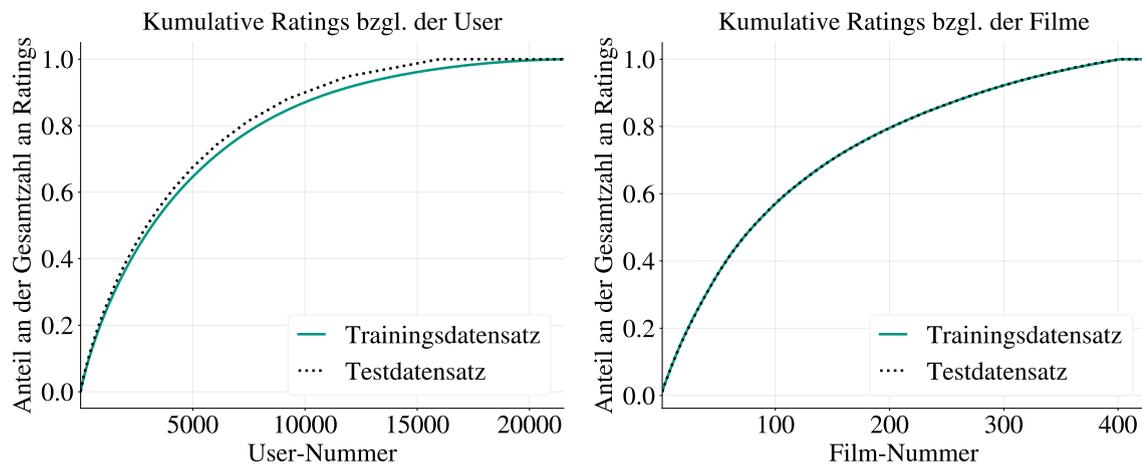
Aus dem Trainingsdatensatz des offiziellen Netflix Preises wurde in folgenden Schritten ein Teildatensatz erzeugt:

1. Zunächst wurden alle Filme herausgefiltert, die von weniger als 10000 Nutzerinnen und Nutzern bewertet worden waren.
2. Aus dem verbliebenen Datensatz wurden zufällig 800 Filme ausgewählt mit samt aller Nutzerinnen und Nutzer, die diese Filme bewertet haben.
3. Aus dem verbliebenen Datensatz wurden zufällig 21617 Nutzerinnen und Nutzer ausgewählt und lediglich die Filme beibehalten, die sie bewertet hatten.
4. Der Datensatz wurde in einen Test- und einen Trainingsdatensatz unterteilt, indem zufällig 10 % der Daten als Testdaten ausgewählt wurden.
5. Abschließend wurden alle Nutzerinnen und Nutzer und alle Filme entfernt, die keine einzige Bewertung im Trainingsdatensatz aufwiesen. Dies entspricht dem Entfernen von leeren Zeilen bzw. Spalten aus der Trainingsmatrix und folglich aus der Testmatrix. Das Kaltstartproblem sollte vereinfachend nicht auftreten.

Der erhaltene, für alle Berechnungen fixe, Teildatensatz besteht aus 21529 Nutzerinnen und Nutzern, 424 Filmen, 704097 Bewertungen im Trainingsdatensatz und 78138 Bewertungen im Testdatensatz. Knapp 7.71 % der Bewertungen im Trainingsdatensatz und 0.856 % der Bewertungen im Testdatensatz sind bekannt. Damit ist der ausgewählte Teildatensatz deutlich dichter, als der von Netflix veröffentlichte Datensatz (vgl. Abschn. 16.1). Das mittlere Rating des Trainingsdatensatzes beträgt 3.689 und das mittlere Rating des Testdatensatzes 3.696. Abbildung E.2 verdeutlicht, wie häufig die Bewertungen von 1 bis 5 im Trainings- bzw. Testdatensatz auftauchen. Aus Abbildung E.3(a) wird ersichtlich, dass die Bewertungen nicht gleichermaßen von allen Nutzenden abgegeben wurden, sondern dass einige Nutzende mit sehr vielen Bewertungen vorliegen (z.T. haben diese Personen mehr als die Hälfte der Filme im Trainingsdatensatz bewertet), wohingegen von der Mehrheit nur wenige Filmbewertungen bekannt sind. Abbildung E.3(b) verdeutlicht einen ähnlichen Effekt bei den Filmen. Die 100 bzw. 200 am meisten bewerteten Filme machen bereits über 50 bzw. 80 % der Bewertungen im Datensatz aus. Die Mehrheit der Filme wurde hingegen nur von wenigen Personen bewertet.

### E.3.2 Ergebnisse basierend auf simplen Vorhersageregeln

Da vereinfachend mit einem kleineren Datensatz gearbeitet wurde, der zudem weniger dünnbesetzt ist als der originale Datensatz des Netflix Preises, lassen sich die Ergebnisse nicht direkt mit den Ergebnissen der Teilnehmenden des Wettbewerbes vergleichen. Um die Ergebnisse dennoch bewerten und einordnen zu können, wurde die Vorhersage  $\hat{r}_{ij}$  für alle  $i = 1, \dots, n_1, j = 1, \dots, n_2$  basierend auf den in Tabelle E.1 dargestellten vier simplen Modellen vorgenommen und jeweils der RMSE auf den Trainings- und den Testdaten berechnet.



- (a) Kumulativer Anteil der Bewertungen bzgl. der Nutzenden. Die Nutzenden wurden aufsteigend nach der Anzahl abgegebener Bewertungen sortiert
- (b) Kumulativer Anteil der Bewertungen bzgl. der Filme. Die Filme wurden aufsteigend nach der Anzahl erhaltener Bewertungen sortiert

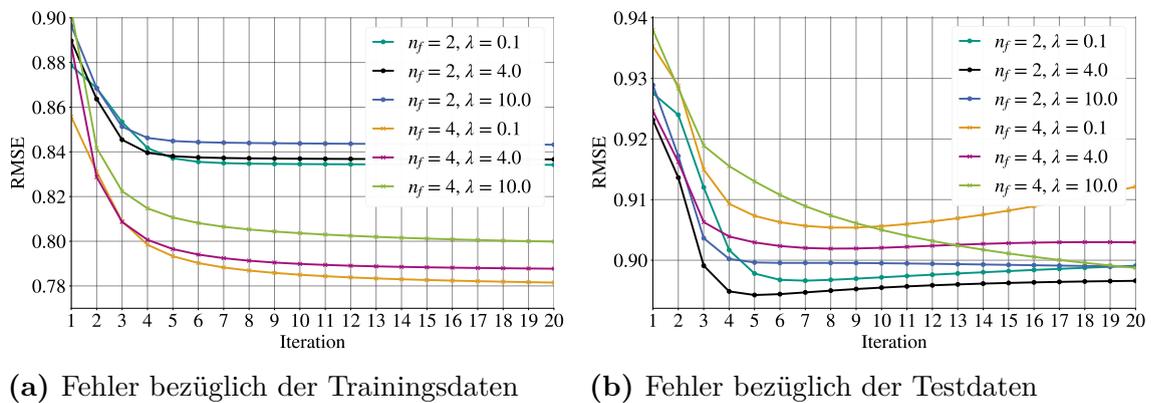
**Tab. E.1:** RMSE bezüglich der Trainings- und Testdaten für verschiedene simple Vorhersageregeln

Vorhersageregeln	Formel	RMSE Training	RMSE Test
Mittleres Rating der Trainingsdaten	$\hat{r}_{ij} = \mu$	1.055	1.057
Mittleres Rating einer jeden nutzen-den Person	$\hat{r}_{ij} = \frac{\sum_{j:(i,j) \in \kappa} r_{ij}}{ \{j (i,j) \in \kappa\} }$	0.962	0.997
Mittleres Rating eines jeden Filmes	$\hat{r}_{ij} = \frac{\sum_{i:(i,j) \in \kappa} r_{ij}}{ \{j (i,j) \in \kappa\} }$	0.985	0.986
Als Lösung des Optimierungspro-blems (16.29) (via SGD mit $\lambda = 0.1$ , Lernrate $\alpha = 0.005$ , 200 Iterationen)	$\hat{r}_{ij} = \mu + o_i + p_j$	0.891	0.920

### E.3.3 Ergebnisse mit dem ALS-Verfahren

Das ALS-Verfahren wurde eingesetzt, um eine Zerlegung der Trainingsmatrix basierend auf dem regularisierten, ungewichteten Problem (16.21) sowie dem regularisierten, gewichteten Problem (16.24) zu bestimmen.

In Abbildung E.4 sind die Ergebnisse der Optimierung im Verlauf der Iterationen für verschiedene Wahlen der Parameter  $\lambda$  und  $n_f$  basierend auf der ungewichteten Formulierung des Problem (R-ALS) dargestellt. Die Ergebnisse werden ab der ersten Iteration (d. h. nach dem ersten Update von  $U$  und  $M$ ) geplottet. Es wird ersichtlich, dass es bei sehr kleinen Werten für den Regularisierungsparameter zu einer Überanpassung an die Trainingsdaten kommt. Der Fehler bezüglich der Testdaten ist in diesem Fall hoch. Zudem verdeutlicht Abbildung E.4, dass die Überanpassung an die Trainingsdaten verstärkt wird, wenn die Anzahl der Eigenschaften  $n_f$  bei gleich-



**Abb. E.4:** Ergebnisse basierend auf dem ungewichteten, regularisierten Problem (16.21) (R-ALS) für verschiedene Wahlen der Parameter  $n_f$  und  $\lambda$ . Die Ergebnisse werden ab der ersten Iteration (d. h. nach dem ersten Update von  $U$  und  $M$ ) dargestellt.

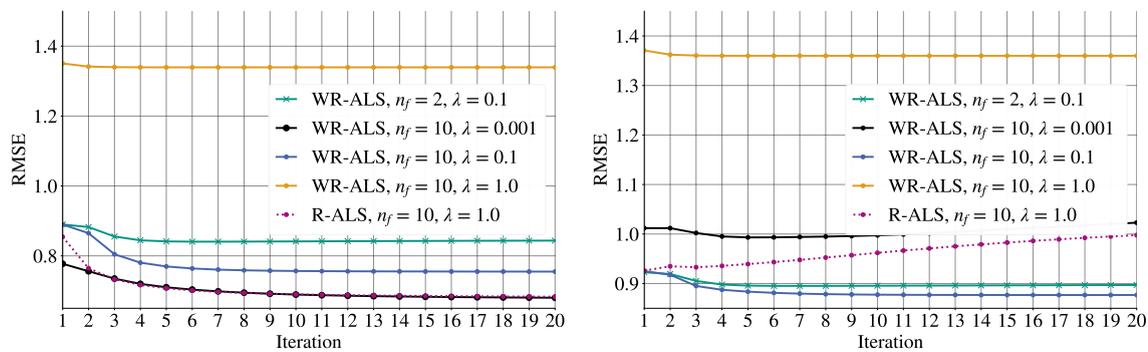
bleibendem Regularisierungsparameter  $\lambda$  erhöht wird (vgl. die Graphen für  $n_f = 4$  und  $n_f = 2$  mit  $\lambda = 0.1$ ).

Der Effekt einer Überanpassung an die Trainingsdaten bei Erhöhung der Anzahl an Eigenschaften wird bei der gewichteten Formulierung des Problems (WR-ALS) gemäß (16.24) abgeschwächt, was aus Abbildung E.5 ersichtlich wird. Eine mögliche Erklärung ist, dass bei Anwendung des Optimierungsverfahrens auf die ungewichtete Formulierung (d. h. bei gleicher Regularisierung für alle latenten Vektoren) insbesondere, die Spalten und Zeilen in der Ratingmatrix besonders gut dargestellt werden, in denen viele Einträge bekannt sind. Grund ist, dass eben diese Spalten und Zeilen mit deutlich mehr Termen in die Summe der Fehlerquadrate eingehen, als es bei Zeilen bzw. Spalten mit wenigen Einträgen der Fall ist. Die Ergebnisse unterstreichen, dass insbesondere die User- bzw. Movievektoren, die in die Modellierung der Vorhersage von Nutzerinnen und Nutzern bzw. Filmen mit vielen Bewertungen eingehen, stärker reguliert werden sollten, um eine Überanpassung an die Trainingsdaten zu vermeiden. Durchaus wäre es interessant weitere Strategien für die Wahl der Regularisierungsparameter und deren Einfluss auf die Ergebnisse zu untersuchen.

Um geeignete Werte für den Regularisierungsparameter  $\lambda$  und die Anzahl an betrachteten Eigenschaften  $n_f$  zu finden, wurde eine grobe Parameterstudie durchgeführt. Ein Teil der Ergebnisse ist in Tabelle E.2 und E.3 dargestellt. Im Falle des ungewichteten Problems (16.21) wurde das beste Ergebnis auf den Testdaten mit den Parametern  $\lambda = 2.0$  und  $n_f = 2$  mit einem RMSE von 0.8948 erzielt. Für das gewichtete Problem (16.24) wurde das beste Ergebnis auf den Testdaten mit den Parametern  $\lambda = 0.09$  und  $n_f = 10$  mit einem RMSE von 0.8774 erzielt.

### E.3.4 Ergebnisse unter Berücksichtigung von Bias-Termen

Das Optimierungsproblem (16.30) mit Bias-Termen wurde mit dem ALS-Verfahren gelöst, wobei die Updates der Movievektoren gemäß (E.3) (und analog für die Uservektoren) durchgeführt wurden. Exemplarische Ergebnisse sind in Abbildung E.6



(a) Fehler bezüglich der Trainingsdaten

(b) Fehler bezüglich der Testdaten

**Abb. E.5:** Ergebnisse basierend auf dem gewichteten regularisierten Problem (16.24) (WR-ALS) für verschiedene Wahlen der Parameter  $n_f$  und  $\lambda$ . Zum Vergleich wird das Ergebnis des ungewichteten regularisierten Problems (R-ALS) für ein Szenario gezeigt.

$n_f$	$\lambda$	RMSE Training	RMSE Test
2	0.2	0.8347	0.8964
2	0.5	0.8349	0.8955
2	1.0	0.8351	0.8950
2	2.0	0.8356	0.8948
2	3.0	0.8362	0.8951
2	4.0	0.8369	0.8955
2	5.0	0.8378	0.8959
10	10.0	0.7302	0.9470
10	20.0	0.7659	0.9283
10	30.0	0.8018	0.9341
20	10.0	0.6395	0.9969
20	20.0	0.6951	0.9468
20	30.0	0.7500	0.9391
20	40.0	0.8030	0.9488

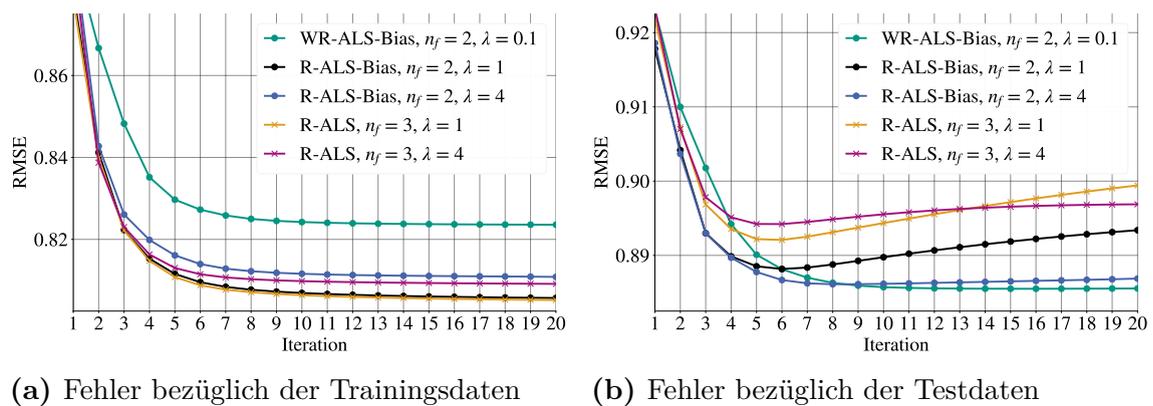
**Tab. E.2:** RMSE nach 10 Iterationen mit dem ALS-Verfahren für das regularisierte ungewichtete Problem (16.21) (R-ALS)

$n_f$	$\lambda$	RMSE Training	RMSE Test
2	0.06	0.8362	0.8964
2	0.09	0.8398	0.8953
2	0.1	0.8413	0.8955
2	0.2	0.8701	0.9134
2	0.5	1.0202	1.0474
2	1.0	1.3394	1.3600
2	2.0	2.1899	2.2027
10	0.08	0.7356	0.8786
10	0.09	0.7459	0.8774
10	0.1	0.7567	0.8775
10	0.2	0.8685	0.9140
20	0.09	0.7076	0.8796
20	0.1	0.7267	0.8792
20	0.2	0.8704	0.9167

**Tab. E.3:** RMSE nach 10 Iterationen mit dem ALS-Verfahren für das regularisierte gewichtete Problem (16.24) (WR-ALS)

dargestellt. Vergleichen wir die Ergebnisse der Modelle mit und ohne Bias-Terme bei gleicher Anzahl an freien Variablen (bspw. R-ALS mit  $n_f = 3$  und R-ALS-

Bias mit  $n_f = 2$ ) so wird ersichtlich, dass das Modell mit Bias-Termen zu besseren Ergebnissen bezüglich der Testdaten führt, als das Modell ohne Bias-Terme. Eine mögliche Erklärung ist, dass die Bias-Terme dem Vorhersagemodell letztlich eine Eigenschaft hinzufügen, die entweder bezüglich der Nutzenden oder der Filme global ist. Insbesondere bei Nutzenden oder Filmen mit sehr wenigen Bewertungen, können diese globalen Komponenten gegebenenfalls zu sinnvolleren Vorhersagen führen, als es durch die ausschließliche Modellierung über das Skalarprodukt aus User- und Movievektor möglich ist (vgl. Aggarwal, 2016, S. 108). Das beste Ergebnis auf den Testdaten wurde basierend auf dem Modell mit gewichteter Regularisierung und Bias-Termen (WR-ALS-Bias) mit  $\lambda = 0.09$  und  $n_f = 10$  und einem RMSE von 0.8762 erzielt.



**Abb. E.6:** Ergebnisse basierend auf dem regularisierten Problem ohne Bias-Terme (16.24) (R-ALS) und dem Problem mit Bias-Termen (16.30) (R-ALS-Bias) für verschiedene Wahlen der Parameter  $n_f$  und  $\lambda$ . Zum Vergleich wird das Ergebnis des gewichteten regularisierten Problems mit Bias-Termen (WR-ALS-Bias) für ein Szenario gezeigt.

Zum Vergleich wurde das Problem (16.30) mit Bias-Termen mit dem SGD-Verfahren gelöst und die Iterationen gemäß (E.4) durchgeführt. Exemplarische Ergebnisse sind in Abbildung E.7 dargestellt. Es wird ersichtlich, dass im Vergleich zum ALS-Verfahren mehr Iterationen notwendig sind, bis das Verfahren konvergiert. Eine mögliche Erklärung ist, dass beim ALS-Verfahren in jeder Iteration die optimale Lösung des linearen Ausgleichsproblems gefunden wird, wohingegen beim SGD-Verfahren in jeder Iteration lediglich kleine Schritte in Richtung etwas besserer Funktionswerte voranschritten werden.

### E.3.5 Abschließende Kommentare

In diesem Abschnitt werden die Ergebnisse kurz zusammengefasst und im Hinblick auf die simplen Vorhersageregeln aus Abschnitt E.3.2 eingeordnet.

- Mit allen auf einer Matrix-Faktorisierung beruhenden Modellen konnten bessere Ergebnisse für die Testdaten erzielt werden, als es mit den simplen Vorhersagemodellen aus Abschnitt E.3.2 möglich war.

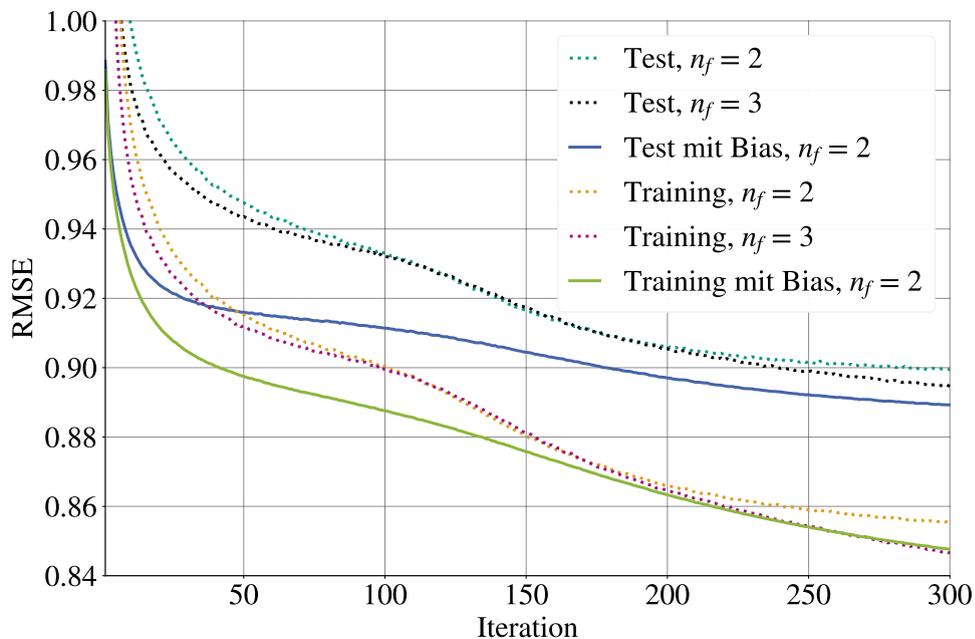


Abb. E.7: Fehler bezüglich der Test- und Trainingsdaten für  $\alpha = 0.001$  und  $\lambda = 0.1$  mit dem SGD-Verfahren

- Eine gute Wahl des Regularisierungsparameters und der Anzahl an betrachteten Eigenschaften ist essentiell, um die Überanpassung an die Trainingsdaten zu vermeiden und gute Ergebnisse bezüglich der Testdaten zu erzielen.
- Der Einbau von gewichteten Regularisierungstermen führte zu besseren Ergebnissen bezüglich der Testdaten als ohne die Berücksichtigung der Gewichte.
- Das Modell mit Bias-Termen führte zu besseren Ergebnissen bezüglich der Testdaten als das Modell ohne Bias-Terme. Ein wesentlicher Teil der Daten kann durch die globalen Tendenzen der Nutzenden bzw. Filme beschrieben werden. Werden diese globalen Effekte berücksichtigt, ist eine bessere Generalisierbarkeit des Modells auf die Testdaten zu beobachten.

# F Lernmaterial zum Thema Empfehlungssysteme

Die Arbeitsblätter (Jupyter Notebooks) sind mit eingetragenen Lösungen dargestellt. Die Kommentare `#= NaN; =#` vor den Lösungen deuten an, dass in der Schülerversion an dieser Stelle der Platzhalter `NaN` steht. Dieser wäre dann durch eine Formel, einen Wert oder ähnliches zu ersetzen. Die auf den Arbeitsblättern verlinkten Tippkarten sind nicht im Anhang enthalten. Diese können jedoch wie im [Vorwort](#) beschrieben über die Workshop-Plattform eingesehen werden.

## F.1 Ablauf des Workshops

In Abbildung [F.1](#) ist ein möglicher Ablauf des Workshops dargestellt.

Arbeitsblatt / Thema	Inhalt	Schulbezug	Weitere mathematische Inhalte	Dauer (Min.)
<b>Einstiegspräsentation (Folien) Technik-Einführung</b>				15 10
AB 1 Datenanalyse	Ein- und Überblick über Netflix Datensatz	Verständnis von Daten		30 – 40
<b>Diskussion im Plenum (Folien)</b>				
AB 2 Das Modell	Zusammenhänge in Bewertungsdaten erkennen, Idee der Faktorisierung erarbeiten	Verständnis von Daten, Skalarprodukt	Matrix-Multiplikation	30
<b>Diskussion im Plenum (Folien)</b>				<b>10</b>
AB3 Die Fehlerfunktion	Bestimmung einer Zerlegung per Hand, Bewertung der Güte einer gefundenen Zerlegung (Faktorisierung)	Gleichungssysteme, Fehlermaß	Summe der Fehlerquadrate	30 – 40
<b>Diskussion im Plenum (Folien)</b>				<b>10</b>
AB 4 Die Optimierung	<ul style="list-style-type: none"> <li>Entwicklung eines Optimierungsverfahrens zur Bestimmung einer Faktorisierung (Methode alternierender Fehlerquadrate)</li> <li>Variierende Anzahl Eigenschaften</li> <li>Einbau Regularisierungsterms</li> </ul>	(Partielle) Ableitung, Minimierungsproblem, lineares Gleichungssystem	Funktion in zwei Variablen, Gradient	50 – 60
AB 4 short	<ul style="list-style-type: none"> <li>lediglich Anwendung des Optimierungsverfahrens von AB 4</li> </ul>			15
<b>Diskussion im Plenum (Folien)</b>				<b>10</b>
AB 5 Anwendung	<ul style="list-style-type: none"> <li>Verständnis von Trainings- und Testdaten</li> <li>Anwendung des Optimierungsverfahrens auf den Netflixdatensatz</li> <li>Erkundung des Einflusses verschiedener Parameter (Anzahl Eigenschaften, Regularisierungsterm...)</li> <li>Kritische Reflexion von Empfehlungssystemen</li> </ul>	Interpretation und kritische Reflexion der Ergebnisse	Prinzip des Maschinellen Lernens	20 – 25
<b>Abschlussdiskussion (Folien) + Evaluation</b>				<b>20</b>
Zusatzblatt: Skalarprodukt <i>(verlinkt auf AB 2)</i>	Mathematische Begründung für größeren bzw. kleinere vorhergesagte Bewertungen	Skalarprodukt, Cosinus, Vektoren, Winkel		
Zusatzblatt: Fehlermaße <i>(verlinkt auf AB 4)</i>	Vergleich verschiedener Fehlerfunktionen	Funktionen / Funktionsgraphen	Optimierung: Eigenschaften der Zielfunktion	
Zusatzblatt: Regularisierung <i>(verlinkt auf AB 4)</i>	Regularisierung – Vermeidung von Überanpassung an die Trainingsdaten	Funktionsgraphen	Optimierung: Existenz und Eindeutigkeit einer Lösung	
Zusatzblatt: eigene Empfehlungen <i>(verlinkt auf AB 5)</i>	Eigene Bewertungen abgeben und individuelle Empfehlungen berechnen			

Abb. F.1: Übersicht über Inhalte und Arbeitsblätter des Workshops zu Empfehlungssystemen

## F.2 Arbeitsblatt 1: Datenanalyse

### Netflix | Arbeitsblatt 1: Datenanalyse

Auf diesem Arbeitsblatt kannst du dir einen Eindruck von dem Datensatz der Netflix Challenge verschaffen.

Wenn du die Codefelder von **Aufgabe 1** ausführst, werden dir jeweils eine Tabelle oder eine Grafik angezeigt.  
 Schau dir die Tabellen und Grafiken **erst alle in Ruhe an**. Notiere dann die Antworten zu folgenden Fragen auf dem **Antwortblatt**.



1. Was ist der beliebteste Film in dem Datensatz?
2. Aus welchem Jahr stammen die ältesten Filme in dem Datensatz? Aus welchem Jahr die jüngsten?
3. Wie hat sich die Verteilung der Bewertungen im Laufe der Erscheinungsjahre geändert?
4. Welcher Film wurde am häufigsten bewertet?
5. Bei welchem Film weichen die Bewertungen der einzelnen Nutzer am wenigsten vom durchschnittlichen Rating ab?

⚠ Antwortblatt runterladen und auf deinem PC bearbeiten. Sonst werden deine Notizen nicht gespeichert.

### Aufgabe 1 | Darstellung des Netflix-Datensatzes

#### Teil a | Grafische Darstellung: Erscheinungsjahr und durchschnittliches Rating



Führe den folgenden Code aus. Klicke dazu in das Codefeld und drücke auf den ▶-Button:

```
In [ ]: # Hier nichts ändern!  
import sys; sys.path.append("../code"); from setup_AB1 import * ;  
interactive_plot(create_rls_av_std_plot)
```

Dir wird eine Grafik angezeigt, in der das Erscheinungsjahr auf der x-Achse und das durchschnittliche Rating auf der y-Achse aufgetragen sind. Du kannst entscheiden, ab welchem Erscheinungsjahr die Filme angezeigt werden, indem du den Slider bewegst.

⚠ Hinweis zur Grafik:

- Wenn du mit der Maus über die Grafik fährst, werden dir weitere Infos zu den einzelnen Filmen angezeigt.
- Die Standardabweichung gibt an, wie stark die Werte eines Merkmals (hier Bewertungen) rund um dessen Mittelwert streuen. Vereinfacht gesagt, gibt die

Standardabweichung die durchschnittliche Entfernung aller Bewertungen eines Filmes vom durchschnittlichen Rating des Filmes an.

```
In [ ]: # Hier nichts ändern!  
import sys; sys.path.append("../code"); from setup_AB1 import * ;  
interactive_plot(create_rls_av_std_plot)
```

#### Teil b | Grafische Darstellung: Erscheinungsjahr und Anzahl Ratings



Führe den folgenden Code aus. Dir wird eine Grafik angezeigt, in der das Erscheinungsjahr auf der x-Achse und die Anzahl der Ratings eines jeden Filmes auf der y-Achse aufgetragen sind.

```
In [ ]: # Hier nichts ändern!  
interactive_plot(create_rls_count_av_plot)
```

#### Teil c | Tabellarische Übersicht



Führe den folgenden Code aus.

Dir wird eine Tabelle mit Infos zu den Filmen des Netflixdatensatzes angezeigt. Du kannst die Tabelle nach einer beliebigen Spalte sortieren. Lege fest, nach welcher Spalte (z. B. Anzahl Ratings, Erscheinungsjahr...) sortiert werden soll. Trage dazu den Namen der Spalte für die Variable `sort1` ein. Du kannst außerdem festlegen, ob die Tabelle aufsteigend (`aufsteigend = True`) oder absteigend (`aufsteigend = False`) sortiert werden soll.

⚠ Hinweis: Die Gänsefüßchen " " müssen um die Bezeichnung der Spalte stehen bleiben.

```
In [ ]: # Sortierung nach Spalte "sort1"  
sort1 = "durchschnittl. Rating"  
# True = aufsteigende Sortierung, False = absteigende Sortierung  
aufsteigend = True  
# Ab hier nichts ändern!  
df_table_sort_values(by=[sort1], ascending = aufsteigend)
```

⚠ Hinweis zur Tabelle: Jeder Film hat eine eindeutige Kennzeichnung bzw. Nummer: seine ID.

### Aufgabe 2 | Die Rating-Tabelle



Welche Zusammenhänge zwischen verschiedenen Usern oder Filmen kannst du in der Rating-Tabelle finden? Beschreibe, was dies für die jeweiligen Nutzer bzw. Film bedeutet.

Hinweis: Es sind mindestens 3 interessante Zusammenhänge zu erkennen.



Falls du Hilfe benötigst, kannst du dir hier einen [Tipp](#) ansehen.

### Teil c | Vorhersage einzelner Bewertungen

In dem Beispiel aus Teil b hat

- Nutzer 2 den Film 6 nicht gesehen und
- Nutzer 4 den Film 1 nicht gesehen.



Diskutiere mit deinem Gruppenmitglied: Wie würdet ihr die beiden fehlenden Bewertungen vorhersagen? Und welchen Film solltet wir dem Nutzer dann vorschlagen? Begründet eure Antwort und notiert diese auf dem Antwortblatt.

### Fazit

Nachdem wir uns mit dem Netflixdatensatz vertraut gemacht haben, werden wir auf den nächsten Arbeitsblättern an kleinen Datensätzen ein Empfehlungssystem entwickeln.

### Zusatzaufgabe für Schnellere:



Notiere auf dem Antwortblatt: Wie viel Prozent der Bewertungen (Ratings) des originale Netflixdatensatzes sind bekannt? Wie viel Prozent der Bewertungen sind nicht bekannt?

Erinnerung: Der Netflixdatensatz besteht aus 17770 Filmen, 480189 Nutzern, 100480507 Bewertungen

### Diskussion im Plenum

Wir werden eure bisherigen Ergebnisse gemeinsam diskutieren!

Öffne nun das nächste Arbeitsblatt.

Wir werden nun alle Bewertungen, die die Nutzer für die Filme abgegeben haben, in einer Art Tabelle bzw. in einem rechteckigen Zahlenschema notieren. Jede Zeile steht für einen User und jede Spalte für einen Film. Die Einträge der Tabelle geben an, welche Bewertung ein Nutzer für einen Film abgegeben hat. Jedoch sind nicht alle Bewertungen bekannt.

Ein Auszug aus der Tabelle könnte wie folgt aussehen:

	3	?	1	?	1	4	...
	?	2	4	1	3	1	...
	3	1	?	3	?	?	...
	4	3	?	?	4	?	...
	4	?	?	?	4	?	...
...	...	...	...	...	...	...	...

Der Netflixdatensatz besteht aus:

- 17770 Filmen
- 480189 Nutzern
- 100480507 Bewertungen (engl. ratings) von 1 - 5

### Teil a | Brainstorming



Sammle Ideen, wie man die **bekanntesten Bewertungen aus der Bewertungstabelle** (auch **Rating-Tabelle** genannt) nutzen kann, um einem User Filme zu empfehlen, die er noch nicht gesehen hat - und zwar so, dass die Filme dem User vermutlich auch gefallen. Notiere deine Ideen auf dem Antwortblatt.

### Teil b | Zusammenhänge in den Bewertungsdaten

Schau dir die folgende beispielhafte **Rating-Tabelle** an. Wir haben dabei ein Beispiel gewählt, bei dem wir den Großteil der Bewertungen bereits kennen.

	3	1	1	3	1	4
	1	2	4	1	3	?
	3	1	1	3	1	4
	?	3	5	4	4	5
	2	3	1	2	2	1

## F.3 Arbeitsblatt 2: Entwicklung eines Modells

### Netflix | Arbeitsblatt 2: Modell zur Vorhersage von Bewertungen

Nutze bei der Bearbeitung des Arbeitsblattes dieses [Antwortblatt](#).

Wir werden zunächst mit kleinen Beispieldatensätzen arbeiten und versuchen interessante Eigenschaften und Zusammenhänge in den Daten auszunutzen, um ein Empfehlungssystem zu entwickeln. Am Ende des Workshops werden wir das Empfehlungssystem auf dem Netflixdatensatz testen!

#### Aufgabe 1 | Die Zerlegung der Rating-Tabelle

##### Teil a

Wir starten mit einem kleinen Beispiel, bei dem die folgenden sechs Filme von vier Nutzern bewertet wurden:

	Name des Films		Name des Films
1.	Batman Returns	4	Der Herr der Ringe: Die Rückkehr des Königs
2.	Spongebob Schwammkopf: Home Sweet Pineapple	5	Miss Undercover (Miss Congeniality)
3.	Die Simpson - Staffel 5	6	The Fast and the Furious 1

Wir starten mit einem Beispiel, bei dem alle Bewertungen bekannt sind. Die Rating-Tabelle lautet:

$$R = \begin{matrix} \begin{matrix} \text{User 1} \\ \text{User 2} \\ \text{User 3} \\ \text{User 4} \end{matrix} & \begin{matrix} \text{F1} \\ \text{F2} \\ \text{F3} \\ \text{F4} \end{matrix} & \begin{matrix} \text{F5} \\ \text{F6} \\ \text{F7} \\ \text{F8} \end{matrix} & \begin{matrix} \text{F9} \\ \text{F10} \\ \text{F11} \\ \text{F12} \end{matrix} & \begin{matrix} \text{F13} \\ \text{F14} \\ \text{F15} \\ \text{F16} \end{matrix} & \begin{matrix} \text{F17} \\ \text{F18} \\ \text{F19} \\ \text{F20} \end{matrix} \\ \begin{matrix} \text{User 1} \\ \text{User 2} \\ \text{User 3} \\ \text{User 4} \end{matrix} & \begin{pmatrix} 3 & 1 & 1 & 3 & 1 & 4 \\ 1 & 2 & 4 & 1 & 3 & 1 \\ 3 & 1 & 1 & 3 & 1 & 4 \\ 4 & 3 & 5 & 4 & 4 & 5 \end{pmatrix} \end{matrix}$$

Wir unterscheiden die sechs Filme bezogen auf zwei Genres: **Action (A)** und **Comedy (C)**. Dazu haben wir ungefähr festgelegt, wie groß der Action- und Comedy-Anteil (von 1 = gering bis 5 = hoch) bei den sechs Filmen ist. Dies haben wir in der folgenden Movie-Tabelle *M* zusammengefasst:

$$M = \begin{matrix} \begin{matrix} \text{F1} \\ \text{F2} \\ \text{F3} \\ \text{F4} \\ \text{F5} \\ \text{F6} \end{matrix} & \begin{matrix} \text{A} \\ \text{C} \end{matrix} & \begin{pmatrix} 3 & 1 \\ 1 & 2 \\ 1 & 1 \\ 1 & 3 \\ 1 & 4 \\ 1 & 1 \end{pmatrix} \end{matrix}$$

**So kannst du die Tabelle lesen:**

An der 6. Spalte kann man ablesen, dass der Film 6 (The Fast and the Furious 1) einen hohen Anteil an Action und einen geringen Anteil an Comedy hat.

*Beurteile, ob die User 1 bis 4 Action bzw. Comedy mögen oder nicht. Schau dir dazu die Movie-Tabelle M und die Rating-Tabelle R genauer an. Ersetze die NaN im folgenden Code*

- durch eine 1, wenn der Nutzer das Genre mag
- durch eine 0, wenn er das Genre nicht mag.

Führe den Code aus, um dein Ergebnis überprüfen zu lassen. Klicke dazu in das Codefeld und drücke auf den ▶ - Button.

```
In [ ]:
# User 1:
A1 = #NaN; # 1 # Action
C1 = #NaN; # 0 # Comedy

# User 2:
A2 = #NaN; # 0 # Action
C2 = #NaN; # 1 # Comedy

# User 3:
A3 = #NaN; # 1 # Action
C3 = #NaN; # 0 # Comedy

# User 4:
A4 = #NaN; # 1 # Action
C4 = #NaN; # 1 # Comedy

# Hier nichts ändern!
include("../code/setup_AB2_j1"); checkPreferences(A1, C1, A2, C2, A3, C3, A4, C4)
```

**Annahme:** Wir gehen davon aus, dass die Nutzer die Genres "Comedy" bzw. "Action" stets komplett (=1) oder gar nicht (=0) mögen. Natürlich wäre es auch denkbar, dass ein Nutzer eine Kategorie nur "teilweise" mag. Dies werden wir im Verlauf des Workshops berücksichtigen.

##### Teil b

Wenn wir die User-Tabelle und die Movie-Tabelle kennen, so können wir diese nutzen, um fehlende Einträge in der Rating-Tabelle vorherzusagen. Das schauen wir uns an einem Beispiel mit der Rating-Tabelle *R2* und einer passenden User-Tabelle *U2* und Movie-Tabelle *M2* an:

### Netflix | Arbeitsblatt 2: Modell zur Vorhersage von Bewertungen

Nutze bei der Bearbeitung des Arbeitsblattes dieses [Antwortblatt](#).

Wir werden zunächst mit kleinen Beispieldatensätzen arbeiten und versuchen interessante Eigenschaften und Zusammenhänge in den Daten auszunutzen, um ein Empfehlungssystem zu entwickeln. Am Ende des Workshops werden wir das Empfehlungssystem auf dem Netflixdatensatz testen!

#### Aufgabe 1 | Die Zerlegung der Rating-Tabelle

##### Teil a

Wir starten mit einem kleinen Beispiel, bei dem die folgenden sechs Filme von vier Nutzern bewertet wurden:

	Name des Films		Name des Films
1.	Batman Returns	4	Der Herr der Ringe: Die Rückkehr des Königs
2.	Spongebob Schwammkopf: Home Sweet Pineapple	5	Miss Undercover (Miss Congeniality)
3.	Die Simpson - Staffel 5	6	The Fast and the Furious 1

Wir starten mit einem Beispiel, bei dem alle Bewertungen bekannt sind. Die Rating-Tabelle lautet:

$$R = \begin{matrix} \begin{matrix} \text{User 1} \\ \text{User 2} \\ \text{User 3} \\ \text{User 4} \end{matrix} & \begin{matrix} \text{F1} \\ \text{F2} \\ \text{F3} \\ \text{F4} \end{matrix} & \begin{matrix} \text{F5} \\ \text{F6} \\ \text{F7} \\ \text{F8} \end{matrix} & \begin{matrix} \text{F9} \\ \text{F10} \\ \text{F11} \\ \text{F12} \end{matrix} & \begin{matrix} \text{F13} \\ \text{F14} \\ \text{F15} \\ \text{F16} \end{matrix} & \begin{matrix} \text{F17} \\ \text{F18} \\ \text{F19} \\ \text{F20} \end{matrix} \\ \begin{matrix} \text{User 1} \\ \text{User 2} \\ \text{User 3} \\ \text{User 4} \end{matrix} & \begin{pmatrix} 3 & 1 & 1 & 3 & 1 & 4 \\ 1 & 2 & 4 & 1 & 3 & 1 \\ 3 & 1 & 1 & 3 & 1 & 4 \\ 4 & 3 & 5 & 4 & 4 & 5 \end{pmatrix} \end{matrix}$$

Wir unterscheiden die sechs Filme bezogen auf zwei Genres: **Action (A)** und **Comedy (C)**. Dazu haben wir ungefähr festgelegt, wie groß der Action- und Comedy-Anteil (von 1 = gering bis 5 = hoch) bei den sechs Filmen ist. Dies haben wir in der folgenden Movie-Tabelle *M* zusammengefasst:

$$M = \begin{matrix} \begin{matrix} \text{F1} \\ \text{F2} \\ \text{F3} \\ \text{F4} \\ \text{F5} \\ \text{F6} \end{matrix} & \begin{matrix} \text{A} \\ \text{C} \end{matrix} & \begin{pmatrix} 3 & 1 \\ 1 & 2 \\ 1 & 1 \\ 1 & 3 \\ 1 & 4 \\ 1 & 1 \end{pmatrix} \end{matrix}$$

**So kannst du die Tabelle lesen:**

An der 6. Spalte kann man ablesen, dass der Film 6 (The Fast and the Furious 1) einen hohen Anteil an Action und einen geringen Anteil an Comedy hat.

*Beurteile, ob die User 1 bis 4 Action bzw. Comedy mögen oder nicht. Schau dir dazu die Movie-Tabelle M und die Rating-Tabelle R genauer an. Ersetze die NaN im folgenden Code*

- durch eine 1, wenn der Nutzer das Genre mag
- durch eine 0, wenn er das Genre nicht mag.

Führe den Code aus, um dein Ergebnis überprüfen zu lassen. Klicke dazu in das Codefeld und drücke auf den ▶ - Button.

```
In [ ]:
# User 1:
A1 = #NaN; # 1 # Action
C1 = #NaN; # 0 # Comedy

# User 2:
A2 = #NaN; # 0 # Action
C2 = #NaN; # 1 # Comedy

# User 3:
A3 = #NaN; # 1 # Action
C3 = #NaN; # 0 # Comedy

# User 4:
A4 = #NaN; # 1 # Action
C4 = #NaN; # 1 # Comedy

# Hier nichts ändern!
include("../code/setup_AB2_j1"); checkPreferences(A1, C1, A2, C2, A3, C3, A4, C4)
```

**Annahme:** Wir gehen davon aus, dass die Nutzer die Genres "Comedy" bzw. "Action" stets komplett (=1) oder gar nicht (=0) mögen. Natürlich wäre es auch denkbar, dass ein Nutzer eine Kategorie nur "teilweise" mag. Dies werden wir im Verlauf des Workshops berücksichtigen.

##### Teil b

Wenn wir die User-Tabelle und die Movie-Tabelle kennen, so können wir diese nutzen, um fehlende Einträge in der Rating-Tabelle vorherzusagen. Das schauen wir uns an einem Beispiel mit der Rating-Tabelle *R2* und einer passenden User-Tabelle *U2* und Movie-Tabelle *M2* an:

$$R3 = \begin{matrix} \text{User 1} & \text{Movie 1} & \text{Movie 2} & \text{Movie 3} & \text{Movie 4} \\ \begin{matrix} \text{User 1} \\ \text{User 2} \\ \text{User 3} \\ \text{User 4} \end{matrix} & \begin{pmatrix} ? & ? & ? & ? \\ 5 & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \end{pmatrix} & \begin{matrix} \text{Movie 1} \\ \text{Movie 2} \\ \text{Movie 3} \\ \text{Movie 4} \end{matrix} \end{matrix}$$

$$U3 = \begin{matrix} A & C \\ \begin{matrix} \text{User 1} \\ \text{User 2} \\ \text{User 3} \\ \text{User 4} \end{matrix} & \begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 1 & 0 \\ 1 & 0.5 \end{pmatrix} \end{matrix} \quad M3 = \begin{matrix} A & C \\ \begin{matrix} \text{Movie 1} \\ \text{Movie 2} \\ \text{Movie 3} \\ \text{Movie 4} \end{matrix} & \begin{pmatrix} 3 & 2 & 1 & 1 \\ 2 & 2 & 4 & 4 \end{pmatrix} \end{matrix}$$



Berechne aus der gegebenen User-Tabelle  $U3$  und der Movie-Tabelle  $M3$  alle fehlenden Bewertungen und trage sie im folgenden Codefeld für die `NaN` ein. Führe den Code aus, um dein Ergebnis überprüfen zu lassen. Die berechneten Bewertungen sind eine Vorhersage dafür, wie ein User einen noch nicht gesehenen Film wohl bewerten würde.

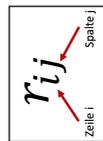
Falls du Hilfe benötigst, kannst du dir hier einen [Tipp](#) ansehen.

```
In [ ]:
R3 = #[NaN NaN 4 NaN; ## [2 2 4 4;
      ## 5 NaN NaN NaN; ## 5 4 5 5;
      ## NaN NaN NaN NaN; ## 3 2 1 1;
      ## NaN NaN NaN 3] ## 4 3 3 3]
checkRatingmatrix(R3)
```

### Aufgabe 2 | Formeln für die Berechnung eines Ratings

#### Teil a

Ziel ist es, die Vorhersage von fehlenden Bewertungen zu verallgemeinern. Wir stellen dazu eine Formel auf, mit der eine beliebige Bewertung in der Rating-Tabelle  $R$  aus der User-Tabelle  $U$  und der Movie-Tabelle  $M$  vorhergesagt werden kann. Dazu beschreiben wir die Einträge von  $R$ ,  $U$  und  $M$  durch Variablen:



$$R2 = \begin{matrix} \text{User 1} & \text{Movie 1} & \text{Movie 2} & \text{Movie 3} & \text{Movie 4} & \text{Movie 5} & \text{Movie 6} \\ \begin{matrix} \text{User 1} \\ \text{User 2} \\ \text{User 3} \\ \text{User 4} \\ \text{User 5} \end{matrix} & \begin{pmatrix} 3 & 2 & 1 & 3 & 2 & 3 \\ 4 & 4 & 5 & 4 & 3 & 5 \\ 3 & 2 & 1 & 3 & 2 & 3 \\ 4 & 4 & 5 & 4 & 3 & 5 \\ 1 & ? & ? & 1 & 1 & 2 \end{pmatrix} & \begin{matrix} \text{Movie 1} \\ \text{Movie 2} \\ \text{Movie 3} \\ \text{Movie 4} \\ \text{Movie 5} \\ \text{Movie 6} \end{matrix} \end{matrix}$$

$$U2 = \begin{matrix} A & C \\ \begin{matrix} \text{User 1} \\ \text{User 2} \\ \text{User 3} \\ \text{User 4} \\ \text{User 5} \end{matrix} & \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{pmatrix} \end{matrix} \quad M2 = \begin{matrix} A & C \\ \begin{matrix} \text{Movie 1} \\ \text{Movie 2} \\ \text{Movie 3} \\ \text{Movie 4} \\ \text{Movie 5} \\ \text{Movie 6} \end{matrix} & \begin{pmatrix} 3 & 2 & 1 & 3 & 2 & 3 \\ 1 & 2 & 4 & 1 & 1 & 2 \end{pmatrix} \end{matrix}$$

In diesem Beispiel hat Nutzer 5 die Filme 2 und 3 noch nicht gesehen.



Überlege zunächst, wie die bekannten Einträge in der Rating-Tabelle  $R2$  aus den Zeilen der User-Tabelle  $U2$  und den Spalten der Movie-Tabelle  $M2$  berechnet werden können. Bestimme auf gleiche Weise die fehlenden Ratings von Nutzer 5 und gib sie im Code für die `NaN` ein. Führe den Code aus.

Falls du Hilfe benötigst, kannst du dir hier einen [Tipp](#) ansehen.

```
In [ ]:
RatingUser5Movie2 = #[NaN; ## 2;
RatingUser5Movie3 = #[NaN; ## 4;
# Hier nichts ändern!
checkPrefUser5(RatingUser5Movie2, RatingUser5Movie3)
```

#### Teil c

Wir schauen uns noch ein zweites Beispiel an, bei dem viele Bewertungen in der Rating-Tabelle  $R3$  fehlen. Die User-Tabelle  $U3$  und die Movie-Tabelle  $M3$  kennen wir:



Stelle eine Formel auf, mit der der Eintrag  $r_{34}$  der Rating-Tabelle  $R$  aus den Einträgen von  $U$  und  $M$  berechnet werden kann.  
Gib deine Formel für das `MaN` ein. Notiere deine Formel auch auf dem Antwortblatt.

**Hinweis:** Du kannst die Einträge von  $U$  (`u11`, `u12`, ...) und  $M$  (`m11`, `m12`, ...) bei der Eingabe deiner Formel verwenden.

**Falls du Hilfe benötigst, kannst du dir hier einen Tipp ansehen.**

```
In [ ]:
# Formel für den Eintrag r34
compute_r34 = #NaN; =# u31 * m14 + u32 * m24;
# Hier nichts ändern!
checkCompute_r34(compute_r34)
```

**Aufgabe 3 | Multiplikation von Matrizen**

**Hinweis:** Du kannst diesen Workshop ohne Vorwissen zu Matrizen problemlos bearbeiten!

Bisher haben wir  $R$ ,  $U$  und  $M$  stets als Tabelle bzw. als Zahlenschema aufgefasst. Dieses Zahlenschema wird in der Mathematik auch **Matrix** genannt. Die Rechnung aus Aufgabe 2 können wir vereinfachen bzw. **kompakter schreiben**, denn hinter deinen Formeln verbergen sich genau die Rechenregeln, die bei der Multiplikation von zwei Matrizen Anwendung finden.





Stelle eine Formel auf, mit der der Eintrag  $r_{34}$  aus den Einträgen von  $U$  und  $M$  berechnet werden kann. Gib deine Formel für das `MaN` ein.

**Hinweis:** Du kannst die Einträge von  $U$  (`u1A`, `u1C`, ...) und  $M$  (`mA1`, `mA2`, ...) bei der Eingabe deiner Formel im Code verwenden.

**Falls du Hilfe benötigst, kannst du dir hier einen Tipp ansehen.**

```
In [ ]:
# Formel für den Eintrag r34
compute_r34_AC = #NaN; =# u3A * mA4 + u3C * mC4;
# Hier nichts ändern!
checkCompute_r34_AC(compute_r34_AC)
```

**Teil b | Beliebige Eigenschaften**

Wir notieren die Berechnung der Bewertungen aus  $U$  und  $M$  nun noch allgemeiner, indem wir nicht mehr ausgerechnet die Eigenschaften **Action** und **Comedy** betrachten. Neben diesen beiden Genres könnten ja durchaus auch andere Genres (Romantik, Kids etc.) oder ganz andere Eigenschaften wie ein/e berühmte/r Regisseur/in relevant sein.

Wir sprechen deswegen ab jetzt allgemein von **Eigenschaft 1** und **Eigenschaft 2**.

$$R = \begin{matrix} & \begin{matrix} \text{u1} & \text{u2} & \text{u3} & \text{u4} & \text{u5} & \text{u6} \end{matrix} \\ \begin{matrix} \text{r1} \\ \text{r2} \\ \text{r3} \\ \text{r4} \\ \text{r5} \end{matrix} & \begin{pmatrix} r_{11} & r_{12} & r_{13} & r_{14} & r_{15} & r_{16} \\ r_{21} & r_{22} & r_{23} & r_{24} & r_{25} & r_{26} \\ r_{31} & r_{32} & r_{33} & r_{34} & r_{35} & r_{36} \\ r_{41} & r_{42} & r_{43} & r_{44} & r_{45} & r_{46} \\ r_{51} & r_{52} & r_{53} & r_{54} & r_{55} & r_{56} \end{pmatrix} \end{matrix}$$

$$U = \begin{matrix} \text{u1} \\ \text{u2} \\ \text{u3} \\ \text{u4} \\ \text{u5} \end{matrix} = \begin{matrix} \text{A} & \text{C} \\ \begin{pmatrix} u_{1A} & u_{1C} \\ u_{2A} & u_{2C} \\ u_{3A} & u_{3C} \\ u_{4A} & u_{4C} \\ u_{5A} & u_{5C} \end{pmatrix} \end{matrix}$$

$$M = \begin{matrix} \text{mA1} & \text{mA2} & \text{mA3} & \text{mA4} & \text{mA5} & \text{mA6} \\ \text{mC1} & \text{mC2} & \text{mC3} & \text{mC4} & \text{mC5} & \text{mC6} \end{matrix}$$

So kannst du die Tabellen lesen:

- Die Variable  $r_{34}$  entspricht dem Eintrag von  $R$  in Zeile 3 und Spalte 4. Die Variable  $r_{34}$  steht also für die Bewertung, die User 3 für Film 4 abgegeben hat (oder für ein Fragezeichen, falls die Bewertung unbekannt ist).
- Die Variable  $u_{3C}$  entspricht dem Eintrag von  $U$  in Zeile 3 und Spalte 2. Die Variable gibt also an, wie sehr User 3 die Eigenschaft  $C$  (=Comedy) mag.

Überzeuge dich, dass das Produkt der Matrizen tatsächlich das gleiche Ergebnis liefert, wie die Formeln aus **Aufgabe 2**. Berechne dazu mithilfe des Computers das Produkt aus User-Tabelle (= User-Matrix)  $U3$  und Movie-Tabelle (=Movie-Matrix)  $M3$ . Gib dazu im Code  $U3 * M3$  ein. Überprüfe, ob deine Formeln aus Teil d das gleiche Ergebnis liefern.

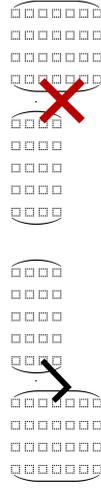
$$R3 = \begin{pmatrix} \text{User}_1 & ? & ? & 4 & ? \\ \text{User}_2 & 5 & ? & ? & ? \\ \text{User}_3 & ? & ? & ? & ? \\ \text{User}_4 & ? & ? & ? & 3 \end{pmatrix}$$

$$U3 = \begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 1 & 0 \\ 1 & 0.5 \end{pmatrix} \quad M3 = \begin{pmatrix} 3 & 2 & 1 & 1 \\ 2 & 2 & 2 & 4 \\ 1 & 0 & 4 & 4 \\ 1 & 0.5 & 4 & 4 \end{pmatrix}$$

```
In [ ]:
# Hier nichts ändern!
# Speichern der User-Matrix
U3 = [0 1;
      1 1;
      1 0;
      1 0.5]
# Speichern der Movie-Matrix
M3 = [3 2 1 1;
      2 2 4 4]
# Berechne das Produkt der beiden Matrizen U3 und M3:
#=# NaN =# U3 * M3
```

**Hinweis:** Die Reihenfolge ist bei der Multiplikation von Matrizen wichtig! Damit zwei Matrizen multipliziert werden können, muss gelten:

**Anzahl Spalten der linken Matrix = Anzahl Zeilen der rechten Matrix.**



**Fazit**

Wir haben die Rating-Tabelle  $R$  (= Rating-Matrix) in eine User-Tabelle  $U$  und eine Movie-Tabelle  $M$  zerlegt. Diese beiden kleineren Tabellen (bzw. Matrizen) liefern uns Infos darüber,

- wie sehr die einzelnen User gewisse Eigenschaften mögen und

- wie sehr die einzelnen Filme diese Eigenschaften aufweisen.

**Info:** Unser bisheriges Modell wird auch **Matrix Faktorisierung** genannt. Wir zerlegen die Rating-Matrix in ein Produkt aus zwei kleineren Matrizen. Die kleineren Matrizen können wir nutzen, um Bewertungen zu berechnen und vorherzusagen!

**Diskussion im Plenum**

Wir werden eure bisherigen Ergebnisse gemeinsam diskutieren! Anschließend geht es weiter mit Arbeitsblatt 3.

**Schon fertig?**

Bearbeite die folgende Zusatzaufgabe oder schau dir das Infoblatt an.

**Zusatzaufgabe**



Welches mathematische Konzept verbirgt sich hinter der Berechnung eines Ratings aus den Zeilen der User-Matrix und den Spalten der Movie-Matrix? Wie lässt sich anschaulich erklären, wann diese Berechnung ein größeres und wann ein kleineres Rating liefert? Deine Ideen kannst du auf diesem Zusatzblatt überprüfen.

**Zusatzinfo:**

Wenn du mehr zu Matrizen und deren Rechenregeln erfahren möchtest, öffne dieses Infoblatt.

# F.4 Arbeitsblatt 3: Die Fehlerfunktion

## Netflix | Arbeitsblatt 3: Berechnung einer Zerlegung in U und M

Nutze bei der Bearbeitung des Arbeitsblattes dieses [Antwortblatt](#).

Der Nettfixdatensatz beinhaltet nur die Rating-Tabelle  $R$ . Wie können wir die User-Tabelle  $U$  und die Movie-Tabelle  $M$  ausgehend von  $R$  berechnen? Bevor wir uns mit dieser Frage beschäftigen, treffen wir zwei Annahmen:

**Annahmen:** 1. Die Einträge in  $U$  und  $M$  dürfen ab jetzt **\*\*beliebige Zahlenwerte\*\*** annehmen. Die Einträge sind dann zwar nicht mehr so schön interpretierbar, dafür können wir die Rating-Tabelle womöglich leichter in  $U$  und  $M$  zerlegen. 2. Wir berücksichtigen weiterhin nur zwei Eigenschaften: **\*\*Eigenschaft 1 (E1)\*\*** und **\*\*Eigenschaft 2 (E2)\*\***. Wir legen nicht fest, welche Eigenschaften wir genau betrachten. Neben dem Genre könnten durchaus auch andere Eigenschaften relevant sein. Zum Beispiel eine besonders berühmte Schauspielerei.

### Aufgabe 1 | Zerlegung per Hand berechnen

Wir suchen für die Rating-Matrix  $R$

$$R = \begin{matrix} & \begin{matrix} \text{E1} & \text{E2} \end{matrix} \\ \begin{matrix} \text{U1} \\ \text{U2} \end{matrix} & \begin{pmatrix} 5 & 3 \\ 3 & 1 \end{pmatrix} \end{matrix}$$

eine Zerlegung in eine User-Matrix  $U_1$  und eine Movie-Matrix  $M_1$ .

$$U_1 = \begin{matrix} \text{U1} & \text{E2} \\ \text{U2} & \text{E2} \end{matrix} \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{pmatrix} \quad \text{und} \quad M_1 = \begin{matrix} \text{E1} & \text{E2} \\ \text{E1} & \text{E2} \end{matrix} \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix}$$



Bestimme Werte für die Einträge von  $U_1$  und  $M_1$ , sodass deine Formeln von Aufgabenblatt 2 für alle Bewertungen in  $R$  erfüllt sind. Gib die gefundenen Werte im Code für die `NaN` ein und führe den Code aus. **Vergleiche deine User-Matrix und deine Movie-Matrix mit denen deines Gruppenmitglieds. Findest du noch eine zweite passende Zerlegung?**

Falls du Hilfe benötigst, kannst du dir hier einen [Tipp](#) ansehen.

**A Hinweis:** Dezimalzahlen werden mit einem Punkt eingegeben. Zum Beispiel: 1.5

```
In [ ]:
# Gib hier die einzelnen Einträge der User-Matrix U ein.
U1 = #=[NaN NaN; #= [2 1;
      #=[NaN NaN] #= [ 2 0]

# Gib hier die einzelnen Einträge der Movie-Matrix M ein.
M1 = #=[NaN NaN; #= [1.5 0.5;
      #=[NaN NaN] #= [ 2.0 2.0]

# Hier nichts ändern!
include("../code/setup_AB3_31"); checkFactorization(R1,U1,M1); U1=M1
```

### Zusatzaufgabe für Interessierte

Du möchtest dich an einer größeren Rating-Matrix versuchen und für diese eine passende Zerlegung berechnen? Dann bearbeite diese [Zusatzaufgabe](#).

### Zwischenfazit

Bei kleinen Rating-Matrizen ist es möglich *per Hand* eine Zerlegung in  $U$  und  $M$  zu finden. Bei großen Rating-Matrizen wird das schwierig... und die Matrix ist beim Nettfixdatensatz mit 480189 Zeilen (=) und 17770 Spalten (=) ziemlich groß. Wir werden die Suche nach einer passenden Zerlegung dem Computer überlassen und dazu auf dem nächsten Arbeitsblatt ein schrittweises Verfahren entwickeln.

Vorab benötigen wir noch ein Fehlermaß, mit dem wir (und der Computer) bewerten können, ob **eine gefundene Zerlegung schon ausreichend gut** ist. Wir definieren dazu in dieser Aufgabe ein Fehlermaß, mit dem wir folgende Fragen beantworten können: *Wann ist eine Zerlegung gut? Wann ist eine Zerlegung besser als eine andere?*

### Aufgabe 2 | Bewertung einer gefundenen Zerlegung

#### Teil a | Vergleich verschiedener Vorhersage-Matrizen

Wir betrachten die folgende Rating-Matrix

$$R = \begin{matrix} & \begin{matrix} \text{E1} & \text{E2} & \text{E3} & \text{E4} \end{matrix} \\ \begin{matrix} \text{U1} \\ \text{U2} \end{matrix} & \begin{pmatrix} 3 & 2 & 1 & 4 \\ 1 & 2 & 3 & 4 \end{pmatrix} \end{matrix}$$

Gegeben sind außerdem drei Zerlegungen, die die Ratings in  $R$  alle drei nicht exakt widerspiegeln.

**Zerlegung 1:**

$$U_4 = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1.5 \end{pmatrix} \quad \text{und} \quad M_4 = \begin{pmatrix} 2 & 1 & 0 & 2 \\ 0 & 1 & 2 & 1 \end{pmatrix}$$

**Zerlegung 2:**

dem Antwortblatt. Die Berechnung kannst du entweder mit Stift und Papier oder mit einem Taschenrechner durchführen.

**Falls du keine Idee hast, wie du den Fehler berechnen kannst, findest du hier einen Tipp.**

### Teil b | Allgemeine Formel für den Fehler

Damit wir den Fehler für eine beliebige Rating-Matrix und eine gefundene Zerlegung in  $U$  und  $M$  berechnen können, stellen wir in dieser Teilaufgabe eine allgemeine Formel für den Fehler auf.



Stelle eine Formel auf, mit der der Fehler zwischen der folgenden **Rating-Matrix R** und der **Vorhersage-Matrix P**

$$R = \begin{matrix} & \begin{matrix} \text{u}_1 & \text{u}_2 \end{matrix} \\ \begin{matrix} \text{r}_{11} & \text{r}_{12} \\ \text{r}_{21} & \text{r}_{22} \end{matrix} & \begin{pmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{pmatrix} \end{matrix}$$

berechnet werden kann. In deiner Formel kannst du alle Variablen verwenden, die in der Rating-Matrix  $R$  ( $r_{11}, r_{12}, \dots, r_{12}, \dots$ ) und der Vorhersage-Matrix  $P$  ( $p_{11}, p_{12}, \dots$ ) vorkommen. Ersetze das NaN durch deine Formel und führe den Code aus.

**Bei der Eingabe deiner Formel können diese Hinweise hilfreich sein:**

- Addieren & subtrahieren:  $2 + 3$  oder  $2 - 4$
- Multiplizieren & dividieren:  $2 * 6$  oder  $6 / 3$
- Quadrieren:  $4^2$
- Den Betrag einer Zahl berechnen:  $\text{abs}(-4)$

**Falls du keine Idee hast, wie du die Formel aufstellen sollst, dann findest du hier einen Tipp.**

```
In [ ] :
# Formel für den Fehler
error_P(r11,r12,r21,r22,p11,p12,p21,p22) = #NaN; =# (r11-p11)^2 + (r12-p12)^2
# Hier nichts ändern!
checkError_P(error_P,r,U4,M4,U5,M5,U6,M6);
```

**Nun wird es noch etwas allgemeiner ...**

Wir definieren die Formel für den Fehler in Abhängigkeit der Einträge von  $U$  und  $M$ . Denn dies sind gerade die Einträge, die wir (bzw. der PC mithilfe eines Optimierungsverfahrens) so wählen wollen, dass der Fehler zwischen Rating-Matrix und Vorhersage-Matrix minimal wird.

$$U_5 = \begin{pmatrix} 2 & 0 \\ 1 & 1.5 \end{pmatrix} \text{ und } M_5 = \begin{pmatrix} 1.5 & 1 & 1 & 2 \\ 0 & 2 & 1 & 2 \end{pmatrix}$$

Zerlegung 3:

$$U_6 = \begin{pmatrix} 1.8 & 0 \\ 1 & 1.5 \end{pmatrix} \text{ und } M_6 = \begin{pmatrix} 1.5 & 1 & 1 & 1.8 \\ 0 & 2 & 1.2 & 2 \end{pmatrix}$$



Führe die drei folgenden Codefelder aus. Für alle drei Zerlegungen wird das Produkt aus der User- und der Movie-Matrix berechnet. Das Ergebnis ist eine Matrix, die alle Vorhersagen der Bewertungen (engl. predictions) enthält. Wir nennen diese Matrix deswegen auch **Vorhersage-Matrix P**. Zum leichteren Vergleich wird jedes Mal auch die Rating-Matrix  $R$  erneut angezeigt.

```
In [ ] :
# Zerlegung 1
U4 = [1 0.5;
      0.5 1.5];
M4 = [2 1 0 2;
      0 1 2 1];
# Berechnung der Vorhersage-Matrix
P4 = U4 * M4; printRP(R,P4)
```

```
In [ ] :
# Zerlegung 2
U5 = [2 0;
      1 1.5];
M5 = [1.5 1 1 2;
      0 2 1 2];
# Berechnung der Vorhersage-Matrix
P5 = U5 * M5; printRP(R,P5)
```

```
In [ ] :
# Zerlegung 3
U6 = [1.8 0;
      1 1.5];
M6 = [1.5 1 1 1;
      0 2 1.2 2];
# Berechnung der Vorhersage-Matrix
P6 = U6 * M6; printRP(R,P6)
```



**Welche der drei Zerlegungen ist am besten, welche am schlechtesten?**

Diskutiere mit deinem Gruppenmitglied, wie die Abweichung bzw. der Fehler zwischen der jeweiligen Vorhersage-Matrix und der tatsächlichen Rating-Matrix  $R$  berechnet werden kann. Berechne dann für jede der drei Zerlegungen den Fehler. Notiere deine Ergebnisse auf

## Diskussion im Plenum

Wir werden eure bisherigen Ergebnisse gemeinsam diskutieren!

### Zusatzaufgabe zur Matrix-Schreibweise

Mit verschiedenen Rechenregeln für Matrizen kann man die Formel für den Fehler enorm abkürzen. Diejenigen, die bereits mit Matrizen gearbeitet haben, können den Fehler in Matrizen-Schreibweise in dieser [Zusatzaufgabe](#) herleiten.

**Frage deine Betreuer/innen, mit welchem der folgenden beiden Arbeitsblätter du weiterarbeiten sollst.**

1. Die eigenständige Entwicklung eines Optimierungsverfahrens auf [Arbeitsblatt 4 - Langversion](#) oder
2. Die Anwendung und Untersuchung eines Optimierungsverfahrens auf [Arbeitsblatt 4 - Kurzversion](#)



Stelle eine Formel auf mit der der Fehler zwischen der folgenden **Rating-Matrix R** und der **Vorhersage-Matrix P**

$$R = \begin{matrix} & \begin{matrix} \text{u}_1 & \text{u}_2 \end{matrix} \\ \begin{matrix} \text{m}_1 \\ \text{m}_2 \end{matrix} & \begin{pmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{pmatrix} \end{matrix}$$

$$P = \begin{matrix} & \begin{matrix} \text{u}_1 & \text{u}_2 \end{matrix} \\ \begin{matrix} \text{m}_1 \\ \text{m}_2 \end{matrix} & \begin{pmatrix} u_{11} \cdot m_{11} + u_{12} \cdot m_{21} & u_{11} \cdot m_{12} + u_{12} \cdot m_{22} \\ u_{21} \cdot m_{11} + u_{22} \cdot m_{21} & u_{21} \cdot m_{12} + u_{22} \cdot m_{22} \end{pmatrix} \end{matrix}$$

berechnet werden kann. Ersetze das **NaN** durch eine Formel für den Fehler, die nur noch von den Einträgen von  $R$  ( $r_{11}, r_{12}, \dots$ ), von  $U$  ( $u_{11}, u_{12}, \dots$ ) und von  $M$  ( $m_{11}, m_{12}, \dots$ ) abhängt. Führe den Code aus. Notiere deine Formel auch auf dem Antwortblatt.

**Tipp:** Falls du keine Idee hast, wie du die Formel aufstellen sollst, findest du [hier](#) einen

In [ ] :

```
# Formel für den Fehler
error(r11,r12,r21,r22,u11,u12,u21,u22,m11,m12,m21,m22) = #NaN; # (r11-(u11*
# Hier nichts ändern!
checkError(error,R,u4,M4,u5,M5,u6,M6);
```

### Fehlende Bewertungen

Tatsächlich sind in der Rating-Matrix nicht alle Einträge bekannt. Bei der Berechnung des Fehlers sollen fehlende Einträge nicht berücksichtigt werden. Dazu wurde deine Formel für den Fehler um einen Programmierbefehl von Julia ergänzt. Dieser fängt alle Terme ab, bei denen die Bewertung nicht bekannt ist.

### Teil c | Vergleichbarkeit des Fehlers



Angenommen wir haben je eine **Vorhersage-Matrix**

- für eine Rating-Matrix mit **1000 Usern und 15000 Filmen** und
- für eine andere Rating-Matrix mit **10 Usern und 15 Filmen**.

berechnet. Kann das Fehlermaß aus **Teil b** verwendet werden, um vergleichen zu können, welche **Vorhersage-Matrix** (und damit auch welche **Zerlegung**) besser ist? Begründe deine Antwort.

# F.5 Arbeitsblatt 4: Das Optimierungsverfahren

## F.5.1 Kurzversion

### Netflix | Arbeitsblatt 4: Das Optimierungsverfahren

Ziel ist es, den Fehler zwischen der Vorhersage und den tatsächlichen Bewertungen zu **minimieren**. Dazu haben wir auf Arbeitsblatt 3 geeignete Einträge von User-Matrix  $U$  und Movie-Matrix  $M$  per Hand bestimmt. Mit anderen Worten haben wir ein Optimierungsproblem gelöst!

Um uns die Berechnung per Hand zu ersparen, werden wir auf diesem Arbeitsblatt ein **Verfahren** einsetzen, um mithilfe des Computers Einträge von  $U$  und  $M$  zu suchen, bei denen der Fehler minimal wird.

Aber vorab zu der Frage: **Was ist eigentlich ein Optimierungsproblem?**

#### Optimierungsprobleme aus der Schule (Funktionen in Abhängigkeit von einer Variablen)

Aus dem Matheunterricht kennt ihr bereits Optimierungsprobleme! Als Beispiel: Ihr wollt das Minimum der Funktion  $f$  mit  $f(x) = 0.1 \cdot x^3 + 2 \cdot x^2$  bestimmen. Ihr löst dann das folgende Optimierungsproblem:

**Finde  $x$ , sodass  $f(x) = 0.1 \cdot x^3 + 2 \cdot x^2$  minimal wird.**

- Möglichkeit: Wir könnten das Minimum in diesem Fall am Funktionsgraphen ablesen.
- Möglichkeit: Wir könnten (wie in der Oberstufe üblich) die Ableitung von  $f$  berechnen und diese gleich 0 setzen.
- Möglichkeit: Wir gehen **schrittweise** vor. Dazu starten wir erstmal mit irgendeinem beliebigen Wert für  $x$  und nähern uns mithilfe einer cleveren Strategie schrittweise dem

Minimum. Solche schrittweise Strategien nennt man auch Optimierungsverfahren oder Optimierungsalgorithmen.

### Das Optimierungsproblem im Netflix-Beispiel (Funktionen in Abhängigkeit von mehreren Variablen)

Die Funktion, deren Minimum wir suchen, ist die Fehlerfunktion von Arbeitsblatt 3! Diese Funktion hängt nicht nur von 1 oder 2 Variablen ab, sondern von ganz schön vielen. Nämlich von  $u_{11}, u_{12}, \dots$  und von  $m_{11}, m_{12}, \dots$ . Mit anderen Worten lösen wir das folgende Optimierungsproblem:

**Finde  $u_{11}, m_{11}, u_{12}, m_{12}, \dots$ , sodass  $error(u_{11}, m_{11}, u_{12}, m_{12}, \dots) = (r_{11} - (u_{11} \cdot m_{11} + u_{12} \cdot m_{21}))^2 + (r_{12} - (u_{11} \cdot m_{12} + u_{12} \cdot m_{22}))^2$  minimal wird.**

Wir werden zum Lösen dieses Optimierungsproblems ein **schrittweises Optimierungsverfahren** nutzen. Schritt für Schritt werden die Einträge von  $U$  und  $M$  in zahlreichen Wiederholungen so verbessert, dass der Fehler immer kleiner wird.

Das Verfahren, welches wir anwenden, heißt **alternierendes Verfahren der kleinsten Fehlerquadrate**. Falls das Verfahren im Plenum noch nicht erklärt wurde, kannst du dir im folgenden Video anschauen, wie das Optimierungsverfahren funktioniert.

**Falls du Probleme mit dem Ton des Videos hast, findest du [hier](#) eine Beschreibung des Verfahrens.**

**Alternierendes Optimierungsverfahren**

Ziel: Finde Einträge von  $U$  und  $M$ , sodass der Fehler minimal wird.

$$R = \begin{pmatrix} 2 & 7 & 4 & 5 \\ 1 & 2 & 4 & 1 \\ 2 & 7 & 4 & 1 \end{pmatrix} \quad R = \begin{pmatrix} ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \end{pmatrix}$$

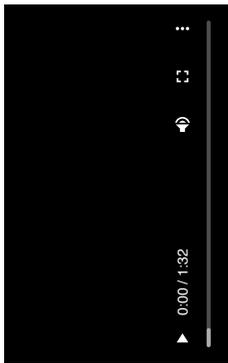
$$U = \begin{pmatrix} ? & ? \\ ? & ? \end{pmatrix} \quad M = \begin{pmatrix} ? & ? & ? \\ ? & ? & ? \end{pmatrix}$$

Gen: 0.00 / 6:28 quadrat

**Fazit:** Mithilfe des **Optimierungsverfahrens** berechnet der Computer schrittweise geeignete Einträge für  $U$  ( $u_{11}, u_{12}, \dots$ ) und für  $M$  ( $m_{11}, m_{12}, \dots$ ). Und zwar so, sodass der Fehler zwischen der Vorhersage und den tatsächlichen Bewertungen **minimiert** wird.

### Aufgabe 1 | Anwendung des Verfahrens auf Rating-Matrizen

Du kannst das Optimierungsverfahren nun auf eine kleine Rating-Matrix anwenden und eine Zerlegung in  $U$  und  $M$  berechnen.



### Teil b | Regularisierung

Wir addieren einen zusätzlichen Term an den Funktionsterm der bisherigen Fehlerfunktion und lösen ein geändertes Optimierungsproblem:

Finde  $u_{11}, m_{11}, u_{12}, m_{12}, \dots$ , sodass

$$\text{error}_{\text{Reg}}(u_{11}, m_{11}, u_{12}, m_{12}, \dots) = (r_{11} - (u_{11} \cdot m_{11} + u_{12} \cdot m_{21}))^2 + (r_{12} - (u_{11} \cdot m_{12} + u_{12} \cdot m_{22}))^2 + \dots + \lambda \cdot (u_{11}^2 + u_{12}^2 + m_{11}^2 + m_{12}^2 + m_{21}^2 + m_{22}^2 + \dots)$$

minimal wird.

**Hinweis:** Der hinzu addierte Term ist *rot* markiert. Durch den Parameter  $\lambda$  kann der Einfluss der neuen Terms von uns gesteuert (=reguliert) werden. Wir können für  $\lambda$  **beliebige Werte**  $\geq 0$  festlegen.

### Der Einfluss des Parameters $\lambda$



Setze  $\lambda$  (Variable: **Lambda**) auf einen beliebigen Wert  $\geq 0$ . Führe den Code aus. Wie ändern sich die Werte in der Vorhersage-Matrix, wenn du  $\lambda$  größer oder kleiner wählst? Wie Antworten auf dem Antwortblatt.

**Hinweis:** Der Fehler auf den bekannten Bewertungen wird für jeden einzelnen Optimierungsschritt auch grafisch dargestellt.

```
In [ ]:
# Anzahl der Optimierungswiederholungen:
iterations = 10;
# Regularisierungsparameter
lambda = NaN; ## 0.6;
# Anzahl der betrachteten Eigenschaften
features = 2;
```

### Teil a



Führe den folgenden Code aus, um die Rating-Matrix  $R_1$  einzuspeichern.

**Hinweis:** eine 0 in der Rating-Matrix entspricht einer fehlenden / unbekanntem Bewertung.

```
In [ ]:
# Hier nichts ändern!
include("../code/setup_AB4_short.jl");
R1 = [3 1 0 2 1 0 ;
      1 0 4 1 3 1 ;
      0 1 1 0 1 4 ;
      0 0 5 0 4 0];
printsave()
```



Führe den Code aus, um mithilfe des Optimierungsverfahrens eine Zerlegung der Rating-Matrix  $R_1$  zu berechnen.

- Beurteile, wie gut die bekannten Bewertungen vorhergesagt werden.
- Ist die Zerlegung auch für die unbekanntem Bewertungen (die Nullen in  $R_1$ ) gut geeignet?

```
In [ ]:
# Anzahl der Optimierungswiederholungen:
iterations = 10;
# Anzahl der betrachteten Eigenschaften
features = 2;
# Hier nichts ändern! Die Matrizen M und U werden iterativ verbessert
train_error, P, U, M = computeALS(iterations, features, R1, [], 0, false, false, false);
println("train_error: ", train_error);
println("P: ", P);
println("U: ", U);
println("M: ", M);
```

### Zwischenfazit

Schau dir dieses kurze Video an, bevor du mit Teil b weiter arbeitest.

```
# Hier nichts ändern! Die Matrizen M und U werden iterativ verbessert
train_error, P, U, M = computeALS(iterations, features, R1, [], lambda, false, false,
printPretty(R1, P, U, M);
```

► Klicke hier, um eine Zusatzaufgabe auszuklappen.

### Teil c | Anzahl der Eigenschaften variieren

Bisher haben wir nur zwei Eigenschaften (z. B. Genres wie Action und Comedy) betrachtet. Du kannst die Anzahl der Eigenschaften nun variieren. Zum Beispiel würde eine Zerlegung in eine User-Matrix und eine Movie-Matrix bei 3 Eigenschaften, 4 Nutzern und 5 Filmen diese Form haben:

$$R = \begin{matrix} & \begin{matrix} E_1 & E_2 & E_3 & E_4 & E_5 \end{matrix} \\ \begin{matrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{matrix} & \begin{pmatrix} r_{11} & r_{12} & r_{13} & r_{14} & r_{15} \\ r_{21} & r_{22} & r_{23} & r_{24} & r_{25} \\ r_{31} & r_{32} & r_{33} & r_{34} & r_{35} \\ r_{41} & r_{42} & r_{43} & r_{44} & r_{45} \end{pmatrix} \end{matrix}$$

$$M = \begin{matrix} \begin{matrix} E_1 & E_2 & E_3 \end{matrix} & \begin{pmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{pmatrix} \\ \begin{matrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{matrix} & \begin{pmatrix} u_{11} & u_{12} & u_{13} \\ u_{21} & u_{22} & u_{23} \\ u_{31} & u_{32} & u_{33} \\ u_{41} & u_{42} & u_{43} \end{pmatrix} \end{matrix}$$



Gib im Code von Teil b an, wie viele Eigenschaften (Variable: `features`) berücksichtigt werden sollen. Führe den Code erneut aus. Für die gewählte Anzahl an Eigenschaften wird die Zerlegung berechnet.

### Diskussion im Plenum

Wir werden die Ergebnisse von diesem Arbeitsblatt gemeinsam diskutieren! Anschließend geht es weiter mit [Arbeitsblatt 5](#). Dort werden wir uns mit der Anwendung des Verfahrens auf unbekannte Daten beschäftigen und das Verfahren auf einen Teil des originalen Netflixdatensatzes anwenden.

## F.5.2 Langversion

### Netflix | Arbeitsblatt 4 - Langversion

#### Entwicklung eines Optimierungsverfahrens

Wir werden ein Verfahren entwickeln, mit dem eine User-Matrix  $U$  und eine Movie-Matrix  $M$  gefunden werden, sodass der Fehler zwischen der Vorhersage und den tatsächlichen Bewertungen **minimiert** wird. Wir lösen dazu das folgende Optimierungsproblem:

**Finde die Einträge von  $U$  und  $M$ , sodass**

$$(r_{11} - (u_{11} \cdot m_{11} + u_{12} \cdot m_{21}))^2 + (r_{12} - (u_{11} \cdot m_{12} + u_{13} \cdot m_{22}))^2 + \dots$$

**minimal wird.**

Falls das Optimierungsverfahren im Plenum noch nicht erklärt wurde, schau dir das folgende Video an. In diesem wird kurz beschrieben, wie das Verfahren funktioniert, dass wir auf diesem Arbeitsblatt entwickeln werden.

**Falls du Probleme mit dem Ton des Videos hast, findest du hier eine Beschreibung des Verfahrens.**

Alternierendes Optimierungsverfahren

The slide shows the alternating optimization process. It includes the equation  $r_{ij} = u_{i1} \cdot m_{1j} + u_{i2} \cdot m_{2j}$  and a video player interface with a play button and volume icon.

**Merke:** Bei dem Optimierungsverfahren **Alternierende Methode kleinster Fehlerquadrate** werden immer **abwechselnd** alle Einträge der User-Matrix oder alle Einträge der Movie-Matrix optimiert!

#### Aufgabe 1 | Schrittweise Entwicklung des Optimierungsverfahrens

Wir erarbeiten das beschriebene Vorgehen Schritt für Schritt an einem kleinen Beispiel für **einen Nutzer und zwei Filme**. Die Rating-Matrix  $R$  lautet:

Gesucht sind die Einträge der User-Matrix  $U$  und der Movie-Matrix  $M$ :

$$R = \begin{matrix} & \begin{matrix} \text{E1} & \text{E2} \end{matrix} \\ \begin{matrix} \text{U1} \\ \text{U2} \end{matrix} & \begin{pmatrix} 2 & 3 \\ 1 & 1 \end{pmatrix} \end{matrix}$$

$$U = \begin{matrix} \text{E1} & \text{E2} \\ \text{U1} & \text{U2} \end{matrix} \quad \text{und} \quad M = \begin{matrix} \text{E1} & \text{E2} \\ \text{M1} & \text{M2} \\ \text{M1} & \text{M2} \end{matrix}$$

### 1. Schritt | Movie-Matrix festlegen

Die Einträge der Movie-Matrix werden auf beliebige Werte festgelegt. Zum Beispiel auf:

$$M = \begin{matrix} \text{E1} & \text{E2} \\ \text{M1} & \text{M2} \\ \text{E2} & \text{M1} \end{matrix} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

### 2. Schritt | Einträge der User-Matrix optimieren

Die Einträge der User-Matrix  $u_{11}$  und  $u_{12}$  werden variiert. Diese beiden Einträge sollen so gewählt werden, dass der Fehler minimal wird. Dazu nutzen wir die Fehlerfunktion vom letzten Arbeitsblatt und setzen in diese alle festen Werte von  $R$  und  $M$  ein. Die Fehlerfunktion lautet dann:

$$\text{error}(u_{11}, u_{12}) = (2 - (u_{11} \cdot 1 + u_{12} \cdot 1))^2 + (3 - (u_{11} \cdot 1 + u_{12} \cdot 1))^2$$

Diese Funktion hängt nur noch von den Variablen  $u_{11}$  und  $u_{12}$  ab.

**In [ ] :**

```
# Fehlerfunktion in Abhängigkeit von u11 und u12
error(u11,u12) = (2 - (u11 + u12))^2 + (3 - (u11 + u12))^2;
# Hier nichts ändern!
include("../code/setup_AB4_31"); checkError(errorU)
```

Die Einträge der User-Matrix sollen nun so gewählt werden, dass die Fehlerfunktion **errorU** einen minimalen Wert annimmt. Wir lösen ein Minimierungsproblem. Aus der Schule kennst du bereits Minimierungsprobleme dieser Art:

**Minimierungsproblem mit einer Variablen:**



Hinweis: eine 0 in der Rating-Matrix entspricht einer fehlenden / unbekanntem Bewertung.

```
In [ ] :
# Hier nichts ändern!
R1 = [3 1 0 2 1 0 ;
      1 0 4 1 3 1 ;
      0 1 1 0 1 4 ;
      0 0 5 0 4 0];
printsave()
```



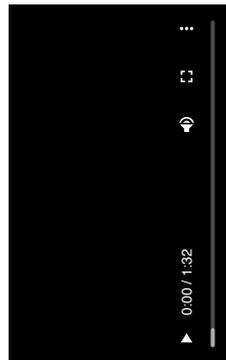
Führe den Code aus, um mithilfe des Optimierungsverfahren eine Zerlegung der Rating-Matrix  $R_1$  zu berechnen.

- Beurteile, wie gut die bekannten Bewertungen vorhergesagt werden.
- Ist die Zerlegung auch für die unbekanntem Bewertungen (die Nullen in  $R_1$ ) gut geeignet?

```
In [ ] :
# Anzahl der Wiederholungen:
iterations = 40;
# Hier nichts ändern! Die Matrizen M und U werden iterativ verbessert
train_error, P, U, M = computeALS(iterations, 2, R1, [], 0, false, false, false); p;
```

### Zwischenfazit

Schau dir dieses kurze Video an bevor du mit Teil b weiterarbeitest.



### Teil b | Regularisierung

Wir addieren einen zusätzlichen Term an den Funktionsterm der bisherigen Fehlerfunktion. Dieser sogenannte Regularisierungsterm lautet:

$$\lambda \cdot (u_{11}^2 + u_{21}^2 + u_{31}^2 + u_{22}^2 + u_{32}^2 + u_{23}^2 + \dots + m_{11}^2 + m_{21}^2 + m_{31}^2 + \dots)$$

```
In [ ] :
# Speichern der Fehlerfunktion (nichts ändern)
errorM(m1,m2,m22) = (2-(u11_min*m1+u12_min*m21))^2+(3-(u11_min*m12+u12_min*m22)
# Berechnung der Ableitungen bzgl. m11, m12, m21 und m22
ErrorGradientM1(m1,m2,m22) = 2*(2-u11_min*m11-u12_min*m21)*(-(u11_min);
ErrorGradientM12(m1,m2,m22) = 2*(3-u11_min*m12-u12_min*m22)*(-(u11_min);
ErrorGradientM21(m1,m2,m22) = #NaN; # 2*(2-u11_min*m11-u12_min*m21)*(-
ErrorGradientM22(m1,m2,m22) = #NaN; # 2*(3-u11_min*m12-u12_min*m22)*(-
# Hier nichts ändern!
checkGradient(ErrorGradientM1,ErrorGradientM12,ErrorGradientM21,ErrorGradi
```

Wie im Falle eines Minimierungsproblems mit nur einer Variablen bestimmen wir die Nullstelle des Gradienten der Fehlerfunktion. Dazu müssen wir erneut alle Teilableitungen gleich null setzen. Dies liefert ein lineares Gleichungssystem.



Stelle das gesuchte lineare Gleichungssystem auf und löse es. Gib dein Ergebnis und damit die Lösungen für  $m_{11}$ ,  $m_{12}$ ,  $m_{21}$  und  $m_{22}$  für die vier  $\lambda$ W ein.

```
In [ ] :
# Nullstelle des Gradienten
m11_min = #NaN; ## 1
m12_min = #NaN; ## 0
m21_min = #NaN; ## 0
m22_min = #NaN; ## 6
# Hier nichts ändern!
checkMinM(m11_min,m12_min,m21_min,m22_min,errorM,ErrorGradientM1,ErrorGradi
```

### Zwischenfazit

In diesem kleinen Beispiel war es ausreichend zwei Wiederholungen des Optimierungsverfahrens durchzuführen, um eine gute bzw. sogar exakte Zerlegung der Rating-Matrix  $R$  zu finden. In der nächsten Aufgabe kannst du dieses Vorgehen auf größere Rating-Matrix anwenden.

### Aufgabe 2 | Anwendung des Verfahrens auf größere Rating-Matrizen

Das beschriebene Optimierungsverfahren wurde in der Funktion `computeALS` gespeichert. Diese kannst du verwenden, um das Verfahren auf eine Rating-Matrizen anzuwenden und eine Zerlegung in  $U$  und  $M$  zu berechnen.

### Teil a



Führe den folgenden Code aus, um die Rating-Matrix  $R_1$  einzuspeichern.

Am Beispiel von Aufgabe 1 mit Rating-Matrix  $R$  und gesuchten Matrizen  $U$  und  $M$

$$R = \begin{matrix} & \text{E1} & \text{E2} \\ \text{U1} & 2 & 3 \end{matrix}$$

$$U = \begin{matrix} \text{U1} & \text{E2} \\ \text{U1} & u_{11} & u_{12} \end{matrix} \quad \text{und} \quad M = \begin{matrix} \text{E1} & \text{E2} \\ m_{11} & m_{12} \\ m_{21} & m_{22} \end{matrix}$$

Lösen wir nun das folgende Minimierungsproblem:

Finde die Einträge von  $U$  und  $M$ , sodass:

$$\text{error}_{\text{log}}(u_{11}, m_{11}, u_{12}, m_{12}, \dots) = (r_{11} - (u_{11} \cdot m_{11} + u_{12} \cdot m_{21}))^2 + (r_{12} - (u_{11} \cdot m_{12} + u_{12} \cdot m_{22}))^2 + \dots + \lambda \cdot (u_{11}^2 + u_{12}^2 + m_{11}^2 + m_{12}^2 + m_{21}^2 + m_{22}^2 + \dots)$$

minimal wird.

Hinweis: Der Einfluss des addierten Terms kann durch den Parameter  $\lambda$  reguliert werden. Für  $\lambda$  können wir beliebige Werte größer gleich 0 festlegen.

### Der Einfluss des Parameters $\lambda$



Beantworte folgende Fragen und notiere deine Antworten auf dem Antwortblatt

- Für welchen Wert von  $\lambda$  entspricht die Funktion mit Regularisierungsterm der Funktion ohne Regularisierungsterm?
- Angenommen wir wählen einen großen Wert für  $\lambda$ , zum Beispiel  $\lambda = 10$ . Wie müssen die Einträge von  $U$  und  $M$  gewählt werden, damit die Fehlerfunktion mit Regularisierungsterm dennoch einen kleinen Wert liefert?



Lege fest, wie groß der Einfluss des Regularisierungsterms sein soll. Setze  $\lambda$  (Variable: Lambda) dazu auf einen beliebigen Wert  $\geq 0$ . Führe den Code aus. Wie ändern sich die Werte in der Vorhersage-Matrix, wenn du  $\lambda$  größer oder kleiner wählst? Wie ändern sich die Vorhersagen der unbekannt Bewertungen (Nullen in  $R_{11}$ ).

Hinweis: Der Fehler auf den bekannten Bewertungen wird für jeden einzelnen Optimierungsschritt auch grafisch dargestellt.

```
In [ ]:
# Anzahl der Wiederholungen:
iterations = 40;
# Regularisierungsparameter
lambda = #NaN;  =# 0.5;
# Anzahl der betrachteten Eigenschaften
features = 2;
# Hier nichts ändern! Die Matrizen M und U werden iterativ verbessert
train_error, P, U, M = computeALS(iterations, features, R1, [], lambda, false, false, false)
printPretty(R1,P,U,M); plotError(train_error)
```

### Teil c | Anzahl der Eigenschaften variieren

Bisher haben wir nur zwei Eigenschaften (z. B. Genres wie Action und Comedy) betrachtet. Du kannst die Anzahl der Eigenschaften nun variieren. Zum Beispiel würde eine Zerlegung in eine User-Matrix und eine Movie-Matrix bei 3 Eigenschaften, 4 Nutzern und 5 Filmen diese Form haben:

$$R = \begin{matrix} & \text{E1} & \text{E2} & \text{E3} & \text{E4} & \text{E5} \\ \text{U1} & r_{11} & r_{12} & r_{13} & r_{14} & r_{15} \\ \text{U2} & r_{21} & r_{22} & r_{23} & r_{24} & r_{25} \\ \text{U3} & r_{31} & r_{32} & r_{33} & r_{34} & r_{35} \\ \text{U4} & r_{41} & r_{42} & r_{43} & r_{44} & r_{45} \end{matrix}$$

$$U = \begin{matrix} \text{U1} & \text{E2} & \text{E3} \\ u_{11} & u_{12} & u_{13} \\ u_{21} & u_{22} & u_{23} \\ u_{31} & u_{32} & u_{33} \\ u_{41} & u_{42} & u_{43} \end{matrix}$$

$$M = \begin{matrix} \text{E1} & \text{E2} & \text{E3} \\ m_{11} & m_{12} & m_{13} & m_{14} & m_{15} \\ m_{21} & m_{22} & m_{23} & m_{24} & m_{25} \\ m_{31} & m_{32} & m_{33} & m_{34} & m_{35} \end{matrix}$$



Gib im Code von Teil b an, wie viele Eigenschaften (Variable: features) berücksichtigt werden sollen. Führe den Code erneut aus. Für die gewählte Anzahl an Eigenschaften wird die Zerlegung berechnet.

### Diskussion im Plenum

Wir werden die Ergebnisse von diesem Arbeitsblatt gemeinsam diskutieren!

Anschließend geht es weiter mit Arbeitsblatt 5. Dort werden wir uns mit der Anwendung des Verfahrens auf unbekannt Daten beschäftigen und das Verfahren anschließend auf einen Teil des originalen Netflixdatensatzes anwenden.

### Schon fertig?

Bearbeite eines der folgenden beiden Zusatzblätter!

#### Zusatzblatt 1: Die Wahl des Fehlermaßes

Auf AB 3 haben wir uns für das Fehlermaß die "Summe der Fehlerquadrate" entschieden. Eine andere Möglichkeit wäre, als Fehlermaß die "Summe der absoluten Fehler" zu verwenden. Die Wahl des Fehlermaßes hat Auswirkungen auf die Lösung des Minimierungsproblems. Auf diesem [Zusatzblatt](#) kannst du Vor- und Nachteile der Fehlermaße diskutieren.

#### Zusatzblatt 2: Einfluss der Regularisierung

Der Einfluss des Regularisierungsterms auf das Optimierungsverfahren lässt sich auch grafisch analysieren. Auf diesem [Zusatzblatt](#) kannst du diesen Einfluss untersuchen.

## F.6 Arbeitsblatt 5: Anwendung auf den Netflixdatensatz

### Netflix | Arbeitsblatt 5: Anwendung auf den Netflixdatensatz

Nutze bei der Bearbeitung des Arbeitsblattes dieses [Antwortblatt](#).

Ziel ist es, die gefundene Zerlegung auch auf unbekannte Daten (Filme, die ein Nutzer noch nicht gesehen hat) anzuwenden und passende Empfehlungen auszusprechen. Um zu bewerten, wie gut sich unser Vorgehen für unbekannte Daten eignet, nutzen wir die folgende Strategie aus dem Bereich des **überwachten Maschinellen Lernens** (engl. *supervised learning*):

$$\text{Rating-Matrix } R = \begin{pmatrix} 2 & 1 & ? & 1 & 3 \\ 3 & 7 & 2 & 7 & 5 \\ 7 & 5 & 7 & 7 & 2 \\ ? & 2 & 1 & 4 & ? \\ 1 & 7 & 7 & 7 & 3 \end{pmatrix}$$

**!** Bei Bedarf findest du auf diesem [Infoblatt](#) eine kurze Beschreibung der Animation.

Auf diesem Arbeitsblatt wenden wir das Optimierungsverfahren auf einen Teil des originalen Netflixdatensatzes und damit auf einen größeren Datensatz an. Dabei werden wir die einzelnen Schritte aus der Animation nacheinander ausführen.

### Aufgabe 1

Beim Maschinellen Lernen arbeitet man normalerweise mit großen Datenmengen (Stichwort **Big Data**). Das Verfahren hat dann mehr Daten, aus denen es wesentliche Eigenschaften der Nutzer und der Filme "lernen" und auf unbekannte Filme übertragen kann.

**Wir arbeiten nun mit größeren Datenmengen. Rechnungen werden deswegen deutlich länger dauern.**

**Schritt 1 | Daten laden & in Test- und Trainingsdaten unterteilen**

Führe den folgenden Code aus. Die Daten des Netflixdatensatzes werden geladen.

```
In [ ]:
# Hier nichts ändern!
include("../code/setup_AB5_j1");
movieIndex, userIndex, ratings = load_netflix_data(); printDatabaseInfo(movieI
```

#### Auswahl eines Teildatensatzes

Um nicht direkt auf dem kompletten, sehr großen Datensatz zu arbeiten, wählen wir einen Teil der Daten aus: Wir betrachten von allen Filmen in dem Datensatz nur die Filme, die mindestens von 120000 Nutzern bewertet wurden. Da der Datensatz immer noch sehr groß ist und Rechnungen sehr lange dauern würden, wählen wir von den verbleibenden Nutzern zufällig `n_user` Nutzer aus.

Führe den folgenden Code aus.

```
In [ ]:
n_ratings = 120000
n_user = 6500
# Laden der Filme mit mindestens n_ratings Bewertungen
movie_subset, user_subset, ratings_subset = subset_top_movies(movieIndex, use
# Zufällige Auswahl von n_user Nutzern
movie_subset, user_subset, ratings_subset = subset_random_users(movie_subset,
```

Unterteilung in Test- und Trainingsdaten

Führe den folgenden Code aus. Die Daten werden in Trainings- und Testdaten unterteilt. Ein Auszug aus der Trainings- und der Testmatrix werden ausgegeben. Üblicherweise werden mehr Daten für das Training (z.B. 80-90 % der bekannten Daten) als für das Testen genutzt. Die Auswahl der Werte, die in die Testmatrix übernommen werden, erfolgt zufällig.

```
In [ ]:
# Festlegung der Größe des Testdatensatzes in Prozent
percentage = 10;
# Hier nicht ändern!
R_train, R_test, ids = split_data(movie_subset, user_subset, ratings_subset, pe
print(trainTest(R_train, R_test, 20, 10));
```

**Schritt 2 | Berechnung einer Zerlegung auf den Trainingsdaten**

**Info:** Vor der Netflix Challenge erzielte das Empfehlungssystem von Netflix einen mittleren quadratischen Fehler von 0.9073! Diesen galt es auf einem unbekanntem Datensatz um rund 10 % zu verbessern, um den Hauptgewinn von 1 Million Dollar zu gewinnen.

## Aufgabe 2 | Parameterstudie



Welchen Einfluss haben die beiden Parameter

- Anzahl der Eigenschaften `features`
- Regularisierungsparameter `lambda`

auf den Trainings- bzw. Testfehler? Variiere dazu im folgenden Codefeld nacheinander die beiden Parameter, wie in der Tabelle auf dem [Antwortblatt](#) vorgegeben. Lasse das Verfahren jeweils erneut durchlaufen und notiere deine Ergebnisse. Mit welchen Werten erhältst du den kleinsten Fehler auf den Testdaten?

```
In [ ]: # Anzahl der Optimierungswiederholungen
iterations = 20;
# Regularisierungsparameter
lambda = 1.0;
# Eigenschaften
features = 8;
# Hier nichts ändern!
test_loss, train_loss, U, M, P = compute_factorization_als_bias(iterations, f
```

## Aufgabe 3 | Ausgabe der Film-Empfehlung für einen beliebigen User

In dieser Aufgabe kamst du dir die Filmpfehlung für einen beliebigen User ausgeben lassen. Dazu wird in der Zeile der Vorhersage-Matrix des ausgewählten Nutzers der Film herausgesucht, bei dem das vorhergesagte Rating den höchsten Wert hatte.



Gib für `NaN` die Zeilennummer eines beliebigen Nutzers ein. Führe den Code dann aus.

```
In [ ]: # Eingabe der Zeilennummer eines beliebigen Nutzers
user = #NaN; ## 2
# Hier nichts ändern!
compute_recommendation(P, Rtrain, ids, user);
```

Schon fertig?

Führe den folgenden Code aus. Mit dem Optimierungsverfahren vom vorherigen Arbeitsblatt wird eine Zerlegung der Trainingsmatrix berechnet. Dir wird zudem der Fehler auf den Trainingsdaten (kurz: **Trainingsfehler**) angezeigt.

```
In [ ]: # Anzahl der Wiederholungen:
iterations = 20;
# Anzahl der betrachteten Eigenschaften
features = 10;
# Regularisierungsparameter
lambda = 0.5;
# Hier nichts ändern!
test_loss, train_loss, U, M, P = compute_factorization_als_bias(iterations, f
print(train_error());
```

## Schritt 3 | Ausgabe der Vorhersagematrix



Führe den folgenden Code aus. Das Produkt  $U \cdot M$  (d. h. unsere Vorhersagematrix) wird berechnet. Ein Auszug der Vorhersagematrix wird ausgegeben. Zum Vergleich werden erneut die Trainingsmatrix und die Testmatrix angezeigt.

```
In [ ]: # Ausgabe der Vorhersagematrix
print(PrettyTrainTest(Rtrain, Rtest, P, 20, 9)
```

## Schritt 4 | Berechnung des Fehlers auf den Testdaten



Führe den folgenden Code aus. Der quadratisch gemittelte Fehler auf den Testdaten (kurz: **Testfehler**) wird berechnet.

```
In [ ]: # Berechnung des Fehlers auf den Testdaten
compute_mse(Rtest, P)
```



Führe den folgenden Code aus. Der Trainings- und Testfehler in den einzelnen Optimierungsschritten wird grafisch dargestellt.

```
In [ ]: # Grafische Darstellung des Fehlers in den einzelnen Iterationen der Optimierung
visualize_training(train_loss, test_loss)
```



## Diskussion im Plenum

*Wir werden die Ergebnisse von diesem Arbeitsblatt gemeinsam diskutieren!*

Bearbeite eine beliebige Zusatzaufgabe.

### Zusatzaufgabe 1: Cold Start Problem

Unter dem Cold Start Problem versteht man, dass User, die sich neu bei Netflix anmelden noch gar keine Bewertungen abgegeben haben.

- *Wie können wir diesen Usern dennoch Filme vorschlagen? Notiert erste Ideen auf dem Antwortblatt.*

### Zusatzaufgabe 2: Andere Modellierungsansätze

Neben der Matrix Faktorisierung gibt es noch andere Ansätze für Empfehlungssysteme. Beispielsweise sogenannte Nachbarschaftsmethoden. Bei diesen wird modelliert, wie ähnlich sich verschiedene Nutzer sind. Diese Ähnlichkeiten werden dann ausgenutzt, um einem User neue Filme zu empfehlen.

- *Überlegt, wie man Ähnlichkeiten zwischen verschiedenen Usern ausgehend von der Ratingmatrix messen kann! Notiert erste Ideen auf dem Antwortblatt.*

### Zusatzaufgabe 3: Empfehlungen für dich berechnen lassen

Auf diesem **Zusatzaufgabe 3** kannst du selbst Bewertungen für verschiedene Filme abgeben und dir persönliche Empfehlungen ausgeben lassen.



### Zusatzaufgabe 4: Datenspeicher

Bearbeite dieses **Zusatzaufgabe 4** zu den Vorteilen, die die Methode der **Matrix-Faktorisierung** für das Speichern von Daten mit sich bringt.

### Infovideo: Video des Gewinnerteams

Schaue dir [hier](#) ein Video an, in dem das Gewinnerteams der Netflix Challenge von seinen Erfahrungen berichtet.

# F.7 Zusatzmaterial

## F.7.1 Zusatzblatt: Skalarprodukt

### Zusatzaufgabe | Große oder kleine Ratings!?

#### Teil a | Geometrische Interpretation

Wir betrachten die folgende Rating-Matrix  $R$  und eine Zerlegung in User-Matrix  $U$  und Movie-Matrix  $M$ :

$$R = \begin{matrix} \text{U}_1 & \text{U}_2 & \text{U}_3 \\ \begin{pmatrix} 4 & 1 & 2 \\ 1 & 4 & 2 \\ 4 & 5 & 4 \end{pmatrix} \end{matrix}$$

$$U = \begin{matrix} \text{U}_1 & \text{U}_2 & \text{U}_3 \\ \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix} \end{matrix} \quad M = \begin{matrix} \text{E}_1 & \text{E}_2 & \text{E}_3 \\ \begin{pmatrix} 4 & 1 & 2 \\ 1 & 4 & 2 \\ 1 & 4 & 2 \end{pmatrix} \end{matrix}$$

Die Zeilen der User-Matrix und die Spalten der Movie-Matrix können wir auch als Vektoren betrachten:

$$U = \begin{matrix} \text{U}_1 \\ \text{U}_2 \\ \text{U}_3 \end{matrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix} \quad M = \begin{matrix} \text{E}_1 \\ \text{E}_2 \\ \text{E}_3 \end{matrix} \begin{pmatrix} 4 & 1 & 2 \\ 1 & 4 & 2 \\ 1 & 4 & 2 \end{pmatrix}$$



Führe den folgenden Code aus. Die Zeilenvektoren der User-Matrix und die Spaltenvektoren der Movie-Matrix werden in ein Koordinatensystem eingezeichnet.

```
In [ ]:
# Hier nichts ändern!
include("../code/setup_SKP.jl"); plotUserMovieVectors(U,M)
```

#### Teil b | Mathematisches Grundkonzept



Welches mathematische Konzept steckt hinter der Berechnung eines Ratings aus einem Zeilenvektor der User-Matrix und einem Spaltenvektor der Movie-Matrix? Vervollständige den folgenden Lückentext und gib deine ausgeschriebene Lösung für  $\text{NaN}$  ein. Deine Lösung wird automatisch überprüft.

Hinter der Berechnung der gesuchten Ratings verbirgt sich das folgende mathematische Konzept: "NaN".

Hinweis: Es ist wichtig, dass die Anführungszeichen "" um das Wort stehen bleiben, sonst wird es nicht überprüft.

 Falls du gar keine Idee hast, kannst du dir hier die Lösung ansehen.

```
In [ ] :
# Lückentext
word1 = "# "NaN"; ==# "Skalarprodukt"; # Ersetze NaN durch das Wort, das den
# Hier nichts ändern!
checkWords(word1)
```

Teil c



Begründe, inwieweit sich im zweidimensionalen Fall geometrisch begründen lässt, wenn das Ergebnis ein größeres bzw. kleineres Rating liefert. Nutze dazu die geometrische Darstellung und die Definition des mathematischen Konzepts aus Teil b. Bei Bedarf kannst du im Internet nach der Definition suchen.

**Zusatzinfo:** Diese Interpretation lässt sich genauso auch auf Vektoren mit mehr Einträgen übertragen. Beispielsweise, wenn man nicht nur die 2 Eigenschaften (Action & Comedy), sondern noch weitere Eigenschaften berücksichtigen würde.

## F.7.2 Zusatzblatt: Die Wahl eines Fehlermaßes

### Zusatzblatt: Die Wahl der Fehlerfunktion



Nutze bei der Bearbeitung des Zusatzblattes dieses Antwortblatt.

Wir werden auf diesem Arbeitsblatt die folgenden drei Fehlermaße und deren Vor- und Nachteile diskutieren:

1. Summe der Fehlerquadrate (dieses Fehlermaß nutzen wir auf AB 2 und 3)
 
$$(r_{11} - (u_{11} \cdot m_{11} + u_{12} \cdot m_{21}))^2 + (r_{12} - (u_{11} \cdot m_{12} + u_{12} \cdot m_{22}))^2 + \dots$$
2. Summe der absoluten Abweichungen
 
$$|r_{11} - (u_{11} \cdot m_{11} + u_{12} \cdot m_{21})| + |r_{12} - (u_{11} \cdot m_{12} + u_{12} \cdot m_{22})| + \dots$$
3. Summe der Abweichungen
 
$$r_{11} - (u_{11} \cdot m_{11} + u_{12} \cdot m_{21}) + r_{12} - (u_{11} \cdot m_{12} + u_{12} \cdot m_{22}) + \dots$$

### Vor- und Nachteile verschiedener Fehlermaße bei der Optimierung

Wir bleiben für die Diskussion der Fehlermaße bei dem Beispiel von AB 3 mit der Ratingmatrix

$$R = \begin{matrix} & \begin{matrix} \text{B}_1 & \text{B}_2 \end{matrix} \\ \begin{matrix} \text{A}_1 \\ \text{A}_2 \end{matrix} & \begin{pmatrix} 2 & 3 \\ 1 & 1 \end{pmatrix} \end{matrix}$$

Wir werden den ersten Schritt des Optimierungsverfahrens mit fester Moviematrix  $M$

$$M = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

mit verschiedenen Fehlermaßen diskutieren. Gesucht sind die Einträge der Usermatrix  $U$

$$U = \begin{pmatrix} u_{11} & u_{12} \end{pmatrix}$$

### Teil a | Speichern der einzelnen Fehlerfunktionen

In allen drei Fällen können wir die bekannten, festen Einträge von  $R$  und  $M$  in die Fehlerfunktionen einsetzen. Dies liefert:

1. Summe der Fehlerquadrate:
 
$$(2 - (u_{11} \cdot 1 + u_{12} \cdot 1))^2 + (3 - (u_{11} \cdot 1 + u_{12} \cdot 1))^2$$

### Teil c | Optimierung mit dem Fehlermaß: Summe der Abweichungen



- Führe den folgenden Code aus. Die Fehlerfunktion wird grafisch dargestellt.
- Diskutiere, inwieweit die Summe der Fehler ein geeignetes Fehlermaß darstellt.
- Berechne für die Fehlerfunktion "Summe der Abweichungen" die Teilableitung und berechne das Minimum. Notiere deine Beobachtungen auf dem Antwortblatt.

Es kann etwas dauern bis die grafische Darstellung erscheint.

```
In [ ] : plotError(errorSum,-1,3)
```

### Zum Vergleich: Summe der Fehlerquadrate

Zum Vergleich kannst du dir hier auch nochmal den Graphen der Fehlerfunktion "Summe der Fehlerquadrate" ausgeben lassen. Führe dazu den folgenden Code aus.

```
In [ ] : plotError(errorSquared,-1,3)
```

### 2. Summe der absoluten Abweichungen:

$$|2 - (u_{11} \cdot 1 + u_{12} \cdot 1)| + |3 - (u_{11} \cdot 1 + u_{12} \cdot 1)|$$

### 3. Summe der Abweichungen

$$(2 - (u_{11} \cdot 1 + u_{12} \cdot 1)) + (3 - (u_{11} \cdot 1 + u_{12} \cdot 1))$$



Führe den folgenden Code aus, um die drei Funktionen einzuspeichern.

```
In [ ] : # Summe der Fehlerquadrate
errorSquared(u1,u2) = (2 - (u11 + u12))^2 + (3 - (u11 + u12))^2;

# Summe der Beträge
errorAbsolut(u1,u2) = abs(2 - (u11 + u12)) + abs(3 - (u11 + u12));

# Summe der Fehler
errorSum(u1,u2) = (2 - (u11 + u12)) + (3 - (u11 + u12));

# Hier nichts ändern!
include("../code/setup_AB4_j1");
```

### Teil b | Optimierung mit dem Fehlermaß: Summe der absoluten Abweichungen

Auf Arbeitsblatt 3 haben wir das Minimum der Fehlerfunktion berechnet, indem wir zunächst die partiellen Ableitungen (Teilableitungen) der Funktion berechnet haben und diese gleich Null gesetzt haben.



- Führe den folgenden Code aus. Die Fehlerfunktion wird grafisch dargestellt. Es kann etwas dauern bis die grafische Darstellung erscheint.
- Lies das Minimum der Funktion ab und notiere es auf dem Antwortblatt.
- Betrachte den Funktionsgraphen. Ist unser Vorgehen (partielle Ableitungen berechnen und gleich 0 setzen) auch bei dieser Funktion ohne weiteres möglich? Begründe deine Antwort und notiere diese auf dem Antwortblatt.
- Diskutiere, inwieweit die Fehlerfunktion "Summe der Fehlerquadrate" besser für unser Optimierungsverfahren geeignet ist.

### Für Interessierte:

Berechne für die Fehlerfunktion "Summe der absoluten Abweichungen" den Gradienten (beide partiellen Ableitungen).

Recherchiere die Ableitung der Betragsfunktion bei Bedarf im Internet.

```
In [ ] : plotError(errorAbsolut,-2,4)
```

## F.7.3 Zusatzblatt: Regularisierung

### Zusatzblatt: Regularisierung



Nutze bei der Bearbeitung dieses Zusatzblattes dieses [Antwortblatt](#).

Wir werden den Einfluss der Regularisierung auf die Lösung des Optimierungsproblems untersuchen. Dazu bleiben wir zunächst bei dem kleinen Beispiel von ABS:

$$R = \begin{pmatrix} 2 & 3 \\ 1 & 1 \end{pmatrix}$$

Gesucht sind die Einträge der Usermatrix  $U$  und der Moviematrix  $M$ :

$$U = \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{pmatrix} \quad \text{und} \quad M = \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix}$$

### Der Einfluss des Parameters $\lambda$ in den einzelnen Optimierungsschritten

Wir untersuchen den Einfluss des Regularisierungsterms in den einzelnen Schritten des Optimierungsverfahrens. Dazu betrachten wir die Optimierung der Einträge von  $U$  und vergleichen die Fehlerfunktionen mit und ohne Regularisierungsterm.

Die Moviematrix haben wir im ersten Optimierungsschritt festgelegt auf:

$$M = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

Der Funktionsterm ohne Regularisierung lautet:

$$(2 - (u_{11} \cdot 1 + u_{12} \cdot 1))^2 + (3 - (u_{11} \cdot 1 + u_{12} \cdot 1))^2$$

Der Funktionsterm mit Regularisierung lautet:

$$(2 - (u_{11} \cdot 1 + u_{12} \cdot 1))^2 + (3 - (u_{11} \cdot 1 + u_{12} \cdot 1))^2 + \lambda \cdot (u_{11}^2 + u_{12}^2 + 1 + 1 + 1 + 1)$$



Führe den folgenden Code aus. Die Fehlerfunktion mit Regularisierungsterm wird gespeichert und grafisch dargestellt. Variiere den Wert des Regularisierungsparameters  $\lambda$  (Variable: `lambda`). Beantworte folgende Fragen und notiere deine Antworten:

- Wie ändert sich der Graph für große Werte von  $\lambda$ ?
- Wie ändert sich der Graph für Werte nahe 0?
- Wann hat der Graph ein eindeutiges globales Minimum?

```
In [ ]:
# Regularisierungsparameter lambda
lambda = 2.0;
# Hier nichts ändern!
errorSquaredRegular(u11, u12, lambda) = (2 - (u11 + u12))^2 + (3 - (u11 + u12))^2 + lambda *
include("../code/setup_regularization.jl"); plotErrorReg(errorSquaredRegul
```

## F.7.4 Zusatzblatt: Eigene Empfehlungen

### Netflix | Zusatzblatt: Eigene Empfehlungen berechnen lassen

Auf diesem Zusatzblatt hast du die Möglichkeit der Bewertungsmatrix eine weitere Zeile mit deinen eigenen Bewertungsdaten hinzuzufügen und dir Empfehlungen anzeigen zu lassen.

#### Teil a | Laden der Daten



Führe den folgenden Code aus. Um einen Teil des Netflixdatensatzes in dieses Arbeitsblatt zu laden.

```
In [ ] :
# Hier nichts ändern!
cd("../"); include("../code/setup_myRecommendation.jl");
Rtrain, Rtest, ids = load_and_split_data(); cd("../Zusatzmaterial")
```

#### Teil b | Eigene Bewertungen abgeben

Für folgende 15 Filme kannst du Bewertungen abgeben. Deine Bewertungen werden in den Trainingsdatensatz aufgenommen. Dazu wurde die Ratingmatrix um eine Zeile erweitert. In dieser neuen Zeile stehen deine Bewertungen.



Gib für mindestens 5 der folgenden Filme eine Bewertung zwischen 1 und 5 für das `NaN` ein. Wenn du keine 5 der 15 Filme kennst, kannst du 5 beliebige Filme googlen und ungefähr abschätzen, ob sie dir gut gefallen würden oder nicht.

Ersetze bei den übrigen Filmen das `NaN` durch eine 0. Führe den folgenden Code aus.

```
In [ ] :
# Miss Congeniality (in DB: Miss Undercover)
col[1] = 57
rating[1] = NaN; ==# 4

# Pretty Women
col[2] = 72
rating[2] = NaN; ==# 5

# Pearl Harbor
col[3] = 8
rating[3] = NaN; ==# 3

# Spider-Man 2
col[4] = 87
rating[4] = NaN; ==# 1

# Lord of the Rings: The Two Towers (in DB: Der Herr der Ringe)
col[5] = 18
rating[5] = NaN; ==# 0
```

```
# American Beauty
col[6] = 23
rating[6] = NaN; ==# 2

# Monsters, Inc = Monster AG
col[7] = 71
rating[7] = NaN; ==# 0

# Titanic
col[8] = 21
rating[8] = NaN; ==# 5

# How to Lose a Guy in 10 Day (in DB: Wie werde ich ihn los in 10 Tagen)
col[9] = 26
rating[9] = NaN; ==# 2

# The Day After Tomorrow
col[10] = 75
rating[10] = NaN; ==# 0

# 50 First Dates (in DB: 50 erste Dates)
col[11] = 45
rating[11] = NaN; ==# 5

# Independence Day
col[12] = 36
rating[12] = NaN; ==# 1

# The Green Mile
col[13] = 35
rating[13] = NaN; ==# 2

# Men in Black
col[14] = 32
rating[14] = NaN; ==# 0

# Fight Club
col[15] = 62
rating[15] = NaN; ==# 1

# Hier nicht ändern!
Rtrain_neu, Rtest_neu = create_extended_dataset(Rtrain, Rtest, col, rating);
```

#### Teil c | Berechnung einer Zerlegung

Für die neue Trainingsmatrix wird nun eine Zerlegung berechnet.



Führe den folgenden Code aus.

```
# Anzahl der Optimierungswiederholungen
iterations = 10;

# Regularisierungsparameter
lambda = 0.2;

# Eigenschaften
features = 5;
```

```
In [ ] :
```

```
# Hier nichts ändern!
test_loss, train_loss, U, M, P = compute_factorization_as_biae(iterations, f
```

**Teil d | Deine persönliche Empfehlung**

Aus der letzten Zeile der berechnete Empfehlungsmatrix `P` wird nun der Eintrag gesucht, der den größten Wert hat. Der zugehörige Wert wird als deine persönliche Empfehlung ausgegeben.



Führe den folgenden Code aus.

```
# Hier nichts ändern!
cd("../"); compute_recommendation(P, Rtrain_neu, ids, size(Rtrain_neu, 1)); cd
```

```
In [ ] :
```

## F.8 Antwortblätter

**Wichtig: Antwortblatt downloaden und auf deinem PC ausfüllen!**  
**Antwortblatt zum Arbeitsblatt 1: Datenanalyse**

**Aufgabe 1 | Beantworte die folgenden Fragen**

1. Was ist der beliebteste Film in dem Datensatz?
2. Aus welchem Jahr stammen die ältesten Filme in dem Datensatz? Aus welchem Jahr die jüngsten?
3. Wie hat sich die Verteilung der Bewertungen im Laufe der Erscheinungsjahre geändert?
4. Welcher Film wurde am häufigsten bewertet?
5. Bei welchem Film weicht die Bewertung der Nutzer am wenigsten vom durchschnittlichen Rating ab?

**Aufgabe 2 | Die Rating-Tabelle**

**Teil a**  
 Sammle Ideen, wie man die **bekanntesten Bewertungen** aus der Bewertungstabelle nutzen kann, um einem User Filme zu empfehlen, die er noch nicht gesehen hat.

**Teil b**  
 Welche Zusammenhänge zwischen verschiedenen Nutzern oder Filmen kannst du in der Rating-Tabelle finden? Beschreibe, was dies für die jeweiligen Nutzer bzw. Filme bedeutet.



**Wichtig: Antwortblatt downloaden und auf deinem PC ausfüllen!**  
**Antwortblatt zu Arbeitsblatt 4 (Langversion)**

**Aufgabe 1 | Schrittweise Entwicklung des Optimierungsverfahrens**  
 1. Schritt | Movie-Matrix festlegen

Fehlerfunktion  $error_U$ :

**2. Schritt | Einträge der User-Matrix optimieren**

Ableitung bzgl.  $u_{11}$ :

Ableitung bzgl.  $u_{12}$ :

Platz für das Lösen des linearen Gleichungssystems:

**3. und 4. Schritt | User-Matrix festlegen und Einträge der Movie-Matrix optimieren**

Fehlerfunktion  $error_M$ :

Ableitung bzgl.  $m_{11}$ :

Ableitung bzgl.  $m_{12}$ :

Ableitung bzgl.  $m_{21}$ :

Ableitung bzgl.  $m_{22}$ :

Platz für das Lösen des linearen Gleichungssystems:

**Antwortblatt zu Arbeitsblatt 4 (Kurzversion)** Langversion auf nächster Seite

**Aufgabe 1 | Anwendung des Verfahrens auf größere Ratingmatrizen**

**Teil a**

Beurteile, wie gut die **bekanntem** Bewertungen vorhergesagt werden. Ist die Zerlegung auch für die **unbekanntem** Bewertungen gut geeignet?

**Teil b | Der Einfluss des Parameters  $\lambda$ .**

Wie ändern sich die Werte in der Vorhersage-Matrix, wenn du  $\lambda$  größer oder kleiner wählst? Wie ändern sich die Vorhersagen der unbekanntem Bewertungen (Nullen in  $R_1$ )?

**Zusatzaufgabe**

Beantworte folgende Fragen:

- Für welchen Wert von  $\lambda$  entspricht die Funktion mit Regularisierungsterm der Funktion ohne Regularisierungsterm?

- Angenommen wir wählen einen großen Wert für  $\lambda$ , zum Beispiel  $\lambda = 10$ . Wie müssen die Einträge von  $U$  und  $M$  gewählt werden, damit die Fehlerfunktion mit Regularisierungsterm dennoch einen kleinen Wert liefert?

Wichtig: Antwortblatt downloaden und auf deinem PC ausfüllen!

Antwortblatt zu Arbeitsblatt 5

Aufgabe 2 | Parameterstudie

features	lambda	Fehler auf den Trainingsdaten	Fehler auf den Testdaten
8	0.00001		
8	0.001		
8	0.01		
8	0.1		
8	0.2		
8	0.5		
8	1		
8	5		
8	10		
8	15		
4	0.3		
6	0.3		
12	0.3		
20	0.3		
40	0.3		
Platz für weitere Ergebnisse			

Antwortblatt zu Arbeitsblatt 4 (Langversion)

Aufgabe 2 | Anwendung des Verfahrens auf größere Ratingmatrizen

Teil a |

Beurteile, wie gut die **bekanntesten** Bewertungen vorhergesagt werden. Ist die Zerlegung auch für die **unbekanntesten** Bewertungen gut geeignet?

Teil b | Der Einfluss des Parameters  $\lambda$ .

Beantworte folgende Fragen:

- Für welchen Wert von  $\lambda$  entspricht die Funktion mit Regularisierungsterm der Funktion ohne Regularisierungsterm?

- Angenommen wir wählen einen großen Wert für  $\lambda$ , zum Beispiel  $\lambda = 10$ . Wie müssen die Einträge von  $U$  und  $M$  gewählt werden, damit die Fehlerfunktion mit Regularisierungsterm dennoch einen kleinen Wert liefert?

**Einfluss des Parameters lambda auf den Trainings- und den Testfehler:**

**Einfluss des Parameters features auf den Trainings- und den Testfehler:**

**Zusatzaufgabe 1 | Cold Start Problem**

Unter dem Cold Start Problem versteht man, dass Nutzer, die sich neu bei Netflix anmelden noch gar keine Bewertungen abgegeben haben.

- *Wie können wir diesen Nutzern dennoch Filme vorschlagen? Notiert erste Ideen.*

**Zusatzaufgabe 2 | Andere Modellierungsansätze**

- *Überlegt, wie man Ähnlichkeiten zwischen verschiedenen Usern ausgehend von der Rating-Matrix messen kann!*

# F.9 Präsentationsfolien

## F.9.1 Folien für den Einstiegsvortrag



**Die Netflix Challenge... und was hat das mit Mathe zu tun?**

Einführung in die Problemstellung

Empfehlungssysteme | eure Einschätzung



- Welche Ziele verfolgen diese Unternehmen mit dem Einsatz von Empfehlungssystemen?
- Welche Informationen und Daten könnten hilfreich sein, um einem User neue Produkte (z. B. Filme, Bücher etc.) zu empfehlen?

Die Netflix Challenge | Der Datensatz

- 17.770 Filme 🎬
- 480.189 Nutzer 👤
- 100.480.507 Bewertungen (von 1 bis 5) ★★★★★

Beispiel:  
 $\text{B}_1$  = Dinosaur Planet,  $\text{B}_{28}$  = Lilo & Stich

	$\text{B}_1$	$\text{B}_2$	$\text{B}_3$	$\text{B}_4$	$\text{B}_5$	...
$\text{U}_1$	3	?	1	?	1	4 ...
$\text{U}_2$	?	2	4	1	3	1 ...
$\text{U}_3$	3	1	?	3	?	? ...
$\text{U}_4$	4	3	?	4	4	? ...
$\text{U}_5$	4	?	?	4	?	5 ...
...	...	...	...	...	...	...

Empfehlungssysteme

Screenshots der Empfehlungen verschiedener Plattformen (Netflix, Twitter, Google Playstore, Amazon Prime, Spotify)

**Definition:**  
 Ein Empfehlungssystem ist ein System, das versucht Bewertungen oder Präferenzen von Usern für Objekte vorherzusagen.

Die Netflix Challenge

- Aufgabe: Das Empfehlungssystem von Netflix um mindestens 10 % verbessern
- Hauptpreis: 1.000.000 USD\$
- Gewinner des Preises: Team *BellKor's Pragmatic Chaos* (2009)
- Netflix veröffentlichte im Oktober 2006 einen riesigen (anonymen) Datensatz

Foto des Gewinnerteams der Netflix Challenge  
 siehe <https://bits.blogs.nytimes.com/2009/09/21/netflix-awards-1-million-prize-and-starts-a-new-contest>

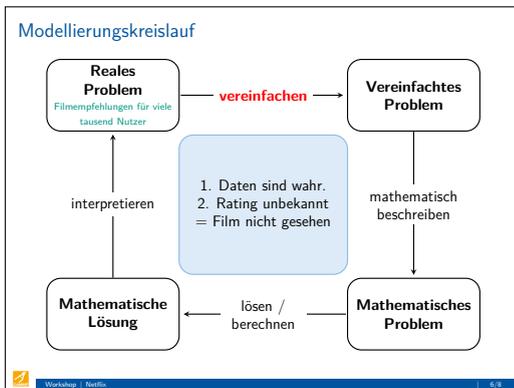
Die Netflix Challenge | Der Datensatz

- 17.770 Filme 🎬
- 480.189 Nutzer 👤
- 100.480.507 Bewertungen (von 1 bis 5) ★★★★★

Beispiel:  
 $\text{B}_1$  = Dinosaur Planet,  $\text{B}_{28}$  = Lilo & Stich

	$\text{B}_1$	$\text{B}_2$	$\text{B}_3$	$\text{B}_4$	$\text{B}_5$	...
$\text{U}_1$	3	2	1	5	1	4 ...
$\text{U}_2$	1	2	4	1	3	1 ...
$\text{U}_3$	3	1	3	3	2	1 ...
$\text{U}_4$	4	3	1	4	4	2 ...
$\text{U}_5$	4	2	2	4	3	5 ...
...	...	...	...	...	...	...

**Aufgabe: Fehlende Bewertungen bestmöglich vorhersagen!**



**Ziel und Ablauf des Workshops**

**Ziel: Entwicklung eines eigenen Empfehlungssystems!**

**Ablauf:**

- Die Daten verstehen.
- Modell zur Vorhersage von Bewertungen auf einem kleinen Beispieldatensatz entwickeln.
- Maß definieren, mit dem wir bewerten können, wie gut unser Modell funktioniert.
- Berechnungen dem Computer überlassen, indem wir einen Optimierungsalgorithmus anwenden.
- Empfehlungssystem auf Netflixdatensatz bewerten und Strategien des maschinellen Lernen diskutieren.
- Kritisch reflektieren: Grenzen des Modells? Vor- und Nachteile von Empfehlungssystemen?

Workshop | Netflix | 7/8

Jetzt seid ihr dran!

- Bearbeitet die Arbeitsblätter!
- Aufgabenstellung sorgfältig lesen!
- Teamwork!
- Hilfekarten!
- Nutzt das Internet!
- Fragt die Betreuer/innen!
- Nutzt die Antwortblätter!

Workshop | Netflix | 8/8

## F.9.2 Folien für die Zwischen- und Abschlussdiskussionen

**Die Netflix Challenge ... und was hat das mit Mathe zu tun?**

Zwischendiskussionen

calamus

**Diskussion nach AB 1 | Der Datensatz**

Beliebtester Film?      Ältester Film? Neueste Filme?

Am häufigsten bewerteter Film?      Geringste Abweichung vom durchschn. Rating?

- Wie hat sich die Verteilung der Bewertungen im Laufe der Erscheinungsjahre geändert?

Workshop | Netflix | 2/25

**Diskussion nach AB 1 | Der Datensatz**

- Was sind eure Ideen für die Entwicklung eines Empfehlungssystems aus den Bewertungsdaten?

							...
	3	?	1	?	1	4	...
	?	2	4	1	3	1	...
	3	1	?	3	?	?	...
	4	3	?	4	4	?	...
	4	?	?	4	?	5	...
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

- Zusatz: Wie viel Prozent der Bewertungen sind bekannt?

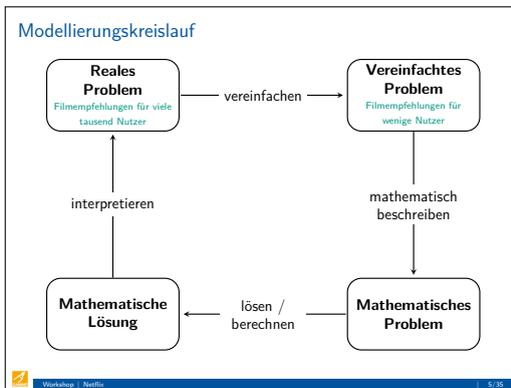
Workshop | Netflix | 3/25

**Diskussion nach AB 1 | Zusammenhänge in den Daten**

- Welche Zusammenhänge kann man in den Daten erkennen?

$$R = \begin{pmatrix} \text{film}_1 & \text{film}_2 & \text{film}_3 & \text{film}_4 & \text{film}_5 & \text{film}_6 \\ \text{person}_1 & 3 & 1 & 1 & 3 & 1 & 4 \\ \text{person}_2 & 1 & 2 & 4 & 1 & 3 & ? \\ \text{person}_3 & 3 & 1 & 1 & 3 & 1 & 4 \\ \text{person}_4 & ? & 3 & 5 & 4 & 4 & 5 \\ \text{person}_5 & 2 & 3 & 1 & 2 & 2 & 1 \end{pmatrix}$$

Workshop | Netflix | 4/25



Your turn!

Ihr seid dran!  
Beginnt mit der Bearbeitung von AB2.

**Diskussion nach AB 2 | Zerlegung der Rating-Tabelle**

- Welche Infos liefern User-Tabelle  $U$  und Movie-Tabelle  $M$ ?

$$R = \begin{matrix} & \text{f}_1 & \text{f}_2 & \text{f}_3 & \text{f}_4 & \text{f}_5 & \text{f}_6 \\ \begin{matrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{matrix} & \begin{pmatrix} 3 & 2 & 1 & 3 & 2 & 3 \\ 4 & 4 & 5 & 4 & 3 & 5 \\ 3 & 2 & 1 & 3 & 2 & 3 \\ 4 & 4 & 5 & 4 & 3 & 5 \\ 1 & ? & ? & 1 & 1 & 2 \end{pmatrix} \end{matrix}$$

$$U = \begin{matrix} & A & C \\ \begin{matrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{matrix} & \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{pmatrix} \end{matrix} \quad M = \begin{matrix} & \text{f}_1 & \text{f}_2 & \text{f}_3 & \text{f}_4 & \text{f}_5 & \text{f}_6 \\ \begin{matrix} A & C \\ C & \end{matrix} & \begin{pmatrix} 3 & 2 & 1 & 3 & 2 & 3 \\ 1 & 2 & 4 & 1 & 1 & 2 \end{pmatrix} \end{matrix}$$

**Diskussion nach AB 2 | Zerlegung der Rating-Tabelle**

- Wie können fehlende Bewertungen berechnet werden?

$$R = \begin{matrix} & \text{f}_1 & \text{f}_2 & \text{f}_3 & \text{f}_4 & \text{f}_5 & \text{f}_6 \\ \begin{matrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{matrix} & \begin{pmatrix} 3 & 2 & 1 & 3 & 2 & 3 \\ 4 & 4 & 5 & 4 & 3 & 5 \\ 3 & 2 & 1 & 3 & 2 & 3 \\ 4 & 4 & 5 & 4 & 3 & 5 \\ 1 & ? & ? & 1 & 1 & 2 \end{pmatrix} \end{matrix}$$

$$U = \begin{matrix} & A & C \\ \begin{matrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{matrix} & \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{pmatrix} \end{matrix} \quad M = \begin{matrix} & \text{f}_1 & \text{f}_2 & \text{f}_3 & \text{f}_4 & \text{f}_5 & \text{f}_6 \\ \begin{matrix} A & C \\ C & \end{matrix} & \begin{pmatrix} 3 & 2 & 1 & 3 & 2 & 3 \\ 1 & 2 & 4 & 1 & 1 & 2 \end{pmatrix} \end{matrix}$$

**Diskussion nach AB 2 | unbekannte Ratings**

$$R = \begin{matrix} & \text{f}_1 & \text{f}_2 & \text{f}_3 & \text{f}_4 & \text{f}_5 & \text{f}_6 \\ \begin{matrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{matrix} & \begin{pmatrix} 3 & 2 & 1 & 3 & 2 & 3 \\ 4 & 4 & 5 & 4 & 3 & 5 \\ 3 & 2 & 1 & 3 & 2 & 3 \\ 4 & 4 & 5 & 4 & 3 & 5 \\ 1 & 2 & ? & 1 & 1 & 2 \end{pmatrix} \end{matrix}$$

$$U = \begin{matrix} & A & C \\ \begin{matrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{matrix} & \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{pmatrix} \end{matrix} \quad M = \begin{matrix} & \text{f}_1 & \text{f}_2 & \text{f}_3 & \text{f}_4 & \text{f}_5 & \text{f}_6 \\ \begin{matrix} A & C \\ C & \end{matrix} & \begin{pmatrix} 3 & 2 & 1 & 3 & 2 & 3 \\ 1 & 2 & 4 & 1 & 1 & 2 \end{pmatrix} \end{matrix}$$

$2 = 0 \cdot 2 + 1 \cdot 2$

**Diskussion nach AB 2 | unbekannte Ratings**

$$R = \begin{matrix} & \text{f}_1 & \text{f}_2 & \text{f}_3 & \text{f}_4 & \text{f}_5 & \text{f}_6 \\ \begin{matrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{matrix} & \begin{pmatrix} 3 & 2 & 1 & 3 & 2 & 3 \\ 4 & 4 & 5 & 4 & 3 & 5 \\ 3 & 2 & 1 & 3 & 2 & 3 \\ 4 & 4 & 5 & 4 & 3 & 5 \\ 1 & 2 & 4 & 1 & 1 & 2 \end{pmatrix} \end{matrix}$$

$$U = \begin{matrix} & A & C \\ \begin{matrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{matrix} & \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{pmatrix} \end{matrix} \quad M = \begin{matrix} & \text{f}_1 & \text{f}_2 & \text{f}_3 & \text{f}_4 & \text{f}_5 & \text{f}_6 \\ \begin{matrix} A & C \\ C & \end{matrix} & \begin{pmatrix} 3 & 2 & 1 & 3 & 2 & 3 \\ 1 & 2 & 4 & 1 & 1 & 2 \end{pmatrix} \end{matrix}$$

$2 = 0 \cdot 2 + 1 \cdot 2$   
 $4 = 0 \cdot 1 + 1 \cdot 4$

**Diskussion nach AB 2 | unbekannte Ratings**

$$R = \begin{matrix} & \text{f}_1 & \text{f}_2 & \text{f}_3 & \text{f}_4 & \text{f}_5 & \text{f}_6 \\ \begin{matrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{matrix} & \begin{pmatrix} f_{11} & f_{12} & f_{13} & f_{14} & f_{15} & f_{16} \\ f_{21} & f_{22} & f_{23} & f_{24} & f_{25} & f_{26} \\ f_{31} & f_{32} & f_{33} & f_{34} & f_{35} & f_{36} \\ f_{41} & f_{42} & f_{43} & f_{44} & f_{45} & f_{46} \\ f_{51} & f_{52} & f_{53} & f_{54} & f_{55} & f_{56} \end{pmatrix} \end{matrix}$$

$$U = \begin{matrix} & E1 & E2 \\ \begin{matrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{matrix} & \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \\ u_{31} & u_{32} \\ u_{41} & u_{42} \\ u_{51} & u_{52} \end{pmatrix} \end{matrix} \quad M = \begin{matrix} & \text{f}_1 & \text{f}_2 & \text{f}_3 & \text{f}_4 & \text{f}_5 & \text{f}_6 \\ \begin{matrix} E1 & E2 \\ E2 & \end{matrix} & \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} & m_{15} & m_{16} \\ m_{21} & m_{22} & m_{23} & m_{24} & m_{25} & m_{26} \end{pmatrix} \end{matrix}$$

$r_{34} = ?$

**Diskussion nach AB 2 | unbekannte Ratings**

$$R = \begin{matrix} & \text{f}_1 & \text{f}_2 & \text{f}_3 & \text{f}_4 & \text{f}_5 & \text{f}_6 \\ \begin{matrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{matrix} & \begin{pmatrix} f_{11} & f_{12} & f_{13} & f_{14} & f_{15} & f_{16} \\ f_{21} & f_{22} & f_{23} & f_{24} & f_{25} & f_{26} \\ f_{31} & f_{32} & f_{33} & f_{34} & f_{35} & f_{36} \\ f_{41} & f_{42} & f_{43} & f_{44} & f_{45} & f_{46} \\ f_{51} & f_{52} & f_{53} & f_{54} & f_{55} & f_{56} \end{pmatrix} \end{matrix}$$

$$U = \begin{matrix} & E1 & E2 \\ \begin{matrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{matrix} & \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \\ u_{31} & u_{32} \\ u_{41} & u_{42} \\ u_{51} & u_{52} \end{pmatrix} \end{matrix} \quad M = \begin{matrix} & \text{f}_1 & \text{f}_2 & \text{f}_3 & \text{f}_4 & \text{f}_5 & \text{f}_6 \\ \begin{matrix} E1 & E2 \\ E2 & \end{matrix} & \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} & m_{15} & m_{16} \\ m_{21} & m_{22} & m_{23} & m_{24} & m_{25} & m_{26} \end{pmatrix} \end{matrix}$$

$r_{34} = u_{31} \cdot m_{14} + u_{32} \cdot m_{24}$

Diskussion nach AB 2 | unbekannte Ratings

$$R = \begin{matrix} & \text{E1} & \text{E2} & \text{E3} & \text{E4} & \text{E5} & \text{E6} \\ \begin{matrix} \text{U1} \\ \text{U2} \\ \text{U3} \\ \text{U4} \\ \text{U5} \end{matrix} & \begin{pmatrix} r_{11} & r_{12} & r_{13} & r_{14} & r_{15} & r_{16} \\ r_{21} & r_{22} & r_{23} & r_{24} & r_{25} & r_{26} \\ r_{31} & r_{32} & r_{33} & r_{34} & r_{35} & r_{36} \\ r_{41} & r_{42} & r_{43} & r_{44} & r_{45} & r_{46} \\ r_{51} & r_{52} & r_{53} & r_{54} & r_{55} & r_{56} \end{pmatrix} \end{matrix}$$

$$U = \begin{matrix} & \text{E1} & \text{E2} \\ \begin{matrix} \text{U1} \\ \text{U2} \\ \text{U3} \\ \text{U4} \\ \text{U5} \end{matrix} & \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \\ u_{31} & u_{32} \\ u_{41} & u_{42} \\ u_{51} & u_{52} \end{pmatrix} \end{matrix}$$

$$M = \begin{matrix} & \text{E1} & \text{E2} & \text{E3} & \text{E4} & \text{E5} & \text{E6} \\ \begin{matrix} \text{M1} \\ \text{M2} \end{matrix} & \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} & m_{15} & m_{16} \\ m_{21} & m_{22} & m_{23} & m_{24} & m_{25} & m_{26} \end{pmatrix} \end{matrix}$$

$$r_{34} = u_{31} \cdot m_{14} + u_{32} \cdot m_{24} = \begin{pmatrix} u_{31} \\ u_{32} \end{pmatrix} \cdot \begin{pmatrix} m_{14} \\ m_{24} \end{pmatrix}$$

**Skalarprodukt!**

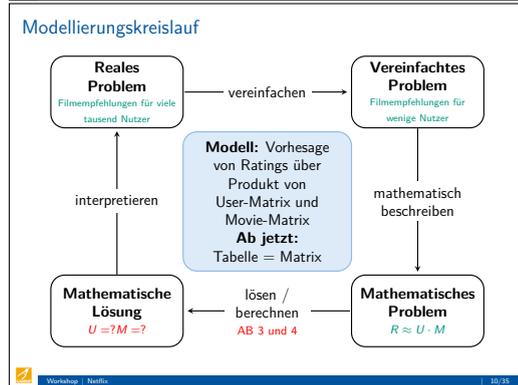
Diskussion nach AB 2 | unbekannte Ratings

$$R = \begin{matrix} & \text{E1} & \text{E2} & \text{E3} & \text{E4} & \text{E5} & \text{E6} \\ \begin{matrix} \text{U1} \\ \text{U2} \\ \text{U3} \\ \text{U4} \\ \text{U5} \end{matrix} & \begin{pmatrix} r_{11} & r_{12} & r_{13} & r_{14} & r_{15} & r_{16} \\ r_{21} & r_{22} & r_{23} & r_{24} & r_{25} & r_{26} \\ r_{31} & r_{32} & r_{33} & r_{34} & r_{35} & r_{36} \\ r_{41} & r_{42} & r_{43} & r_{44} & r_{45} & r_{46} \\ r_{51} & r_{52} & r_{53} & r_{54} & r_{55} & r_{56} \end{pmatrix} \end{matrix}$$

$$U = \begin{matrix} & \text{E1} & \text{E2} \\ \begin{matrix} \text{U1} \\ \text{U2} \\ \text{U3} \\ \text{U4} \\ \text{U5} \end{matrix} & \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \\ u_{31} & u_{32} \\ u_{41} & u_{42} \\ u_{51} & u_{52} \end{pmatrix} \end{matrix}$$

$$M = \begin{matrix} & \text{E1} & \text{E2} & \text{E3} & \text{E4} & \text{E5} & \text{E6} \\ \begin{matrix} \text{M1} \\ \text{M2} \end{matrix} & \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} & m_{15} & m_{16} \\ m_{21} & m_{22} & m_{23} & m_{24} & m_{25} & m_{26} \end{pmatrix} \end{matrix}$$

kurz:  $U \cdot M$



Diskussion nach AB 2 | offene Probleme

**Offene Probleme:**

- Nur die Rating-Matrix  $R$  ist bekannt.
- Wie können  $U$  und  $M$  berechnet werden?
- Reicht es 2 Eigenschaften (z.B. Action und Comedy) zu betrachten? Welche?

Modellannahme

**Annahme:** Es sind weiterhin nur 2 Eigenschaften relevant. Wir legen nicht fest, welche Eigenschaften wir betrachten.

$$U = \begin{matrix} & \text{E1} & \text{E2} \\ \begin{matrix} \text{U1} \\ \text{U2} \\ \text{U3} \\ \text{U4} \\ \text{U5} \\ \vdots \end{matrix} & \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \\ u_{31} & u_{32} \\ u_{41} & u_{42} \\ u_{51} & u_{52} \\ \vdots & \vdots \end{pmatrix} \end{matrix}$$

$$M = \begin{matrix} & \text{E1} & \text{E2} & \text{E3} & \text{E4} & \text{E5} & \dots \\ \begin{matrix} \text{M1} \\ \text{M2} \end{matrix} & \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} & m_{15} & \dots \\ m_{21} & m_{22} & m_{23} & m_{24} & m_{25} & \dots \end{pmatrix} \end{matrix}$$

Ziel: Zerlegung für eine beliebige Rating-Matrix  $R$  berechnen!

Ihr seid dran!  
Beginnt mit der Bearbeitung von AB3.

Diskussion nach AB 3 | Zerlegung berechnen

$$R = \begin{matrix} & \text{E1} & \text{E2} \\ \begin{matrix} \text{U1} \\ \text{U2} \end{matrix} & \begin{pmatrix} 5 & 3 \\ 3 & 1 \end{pmatrix} \end{matrix}$$

$$U = \begin{matrix} & \text{E1} & \text{E2} \\ \begin{matrix} \text{U1} \\ \text{U2} \end{matrix} & \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{pmatrix} \end{matrix}$$

$$M = \begin{matrix} & \text{E1} & \text{E2} \\ \begin{matrix} \text{M1} \\ \text{M2} \end{matrix} & \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix} \end{matrix}$$

- Wie seid ihr vorgegangen?
- Was ist euch beim Vergleich in der Gruppe aufgefallen?

**Fazit:**

- Zerlegung nicht eindeutig.
- Berechnung einer Zerlegung per Hand aufwendig

Ziel: Verfahren entwickeln, sodass der PC die Zerlegung berechnet!

Diskussion nach AB 3 | Das Fehlermaß

**Rating-Matrix**

$$R = \begin{matrix} & \text{E1} & \text{E2} & \text{E3} & \text{E4} \\ \begin{matrix} \text{U1} \\ \text{U2} \end{matrix} & \begin{pmatrix} 3 & 2 & 1 & 4 \\ 1 & 2 & 3 & 4 \end{pmatrix} \end{matrix}$$

**Vorhersage-Matrix**

$$P = \begin{matrix} & \text{E1} & \text{E2} & \text{E3} & \text{E4} \\ \begin{matrix} \text{U1} \\ \text{U2} \end{matrix} & \begin{pmatrix} 3 & 2 & 2 & 4 \\ 1.5 & 4.0 & 2.5 & 5 \end{pmatrix} \end{matrix}$$

Wie habt ihr den Fehler berechnet?

Diskussion nach AB 3 | Vergleich von Fehlermaßen

**Rating-Matrix**

$$R = \begin{matrix} & \text{E1} & \text{E2} & \text{E3} & \text{E4} \\ \begin{matrix} \text{U1} \\ \text{U2} \end{matrix} & \begin{pmatrix} 3 & 2 & 1 & 4 \\ 1 & 2 & 3 & 4 \end{pmatrix} \end{matrix}$$

**Vorhersage-Matrix**

$$P = \begin{matrix} & \text{E1} & \text{E2} & \text{E3} & \text{E4} \\ \begin{matrix} \text{U1} \\ \text{U2} \end{matrix} & \begin{pmatrix} 3 & 2 & 2 & 4 \\ 1.5 & 4.0 & 2.5 & 5 \end{pmatrix} \end{matrix}$$

Was sind Vor- und Nachteile der folgenden Fehlermaße?

- Summe der Abweichungen  
 $error = (3 - 3) + (2 - 2) + (1 - 2) + \dots + (4 - 5)$
- Summe der absoluten Abweichungen  
 $error = |3 - 3| + |2 - 2| + |1 - 2| + \dots + |4 - 5|$
- Summe der quadratischen Abweichungen  
 $error = (3 - 3)^2 + (2 - 2)^2 + (1 - 2)^2 + \dots + (4 - 5)^2$

Diskussion nach AB 3 | Formel für den Fehler

**Rating-Matrix**

$$R = \begin{matrix} & \begin{matrix} \text{B}_1 & \text{B}_2 \end{matrix} \\ \begin{matrix} \text{A}_1 \\ \text{A}_2 \end{matrix} & \begin{pmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{pmatrix} \end{matrix}$$

**Vorhersage-Matrix**

$$P = \begin{matrix} & \begin{matrix} \text{B}_1 & \text{B}_2 \end{matrix} \\ \begin{matrix} \text{A}_1 \\ \text{A}_2 \end{matrix} & \begin{pmatrix} u_{11} \cdot m_{11} + u_{12} \cdot m_{21} & u_{11} \cdot m_{12} + u_{12} \cdot m_{22} \\ u_{21} \cdot m_{11} + u_{22} \cdot m_{21} & u_{21} \cdot m_{12} + u_{22} \cdot m_{22} \end{pmatrix} \end{matrix}$$

**Formel für den Fehler**

$$(r_{11} - (u_{11} \cdot m_{11} + u_{12} \cdot m_{21}))^2 + (r_{12} - (u_{11} \cdot m_{12} + u_{12} \cdot m_{22}))^2 + \dots$$

Workshop | Nestlé | 16/35

Diskussion nach AB 3 | Formel für den Fehler

**Rating-Matrix**

$$R = \begin{matrix} & \begin{matrix} \text{B}_1 & \text{B}_2 \end{matrix} \\ \begin{matrix} \text{A}_1 \\ \text{A}_2 \end{matrix} & \begin{pmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{pmatrix} \end{matrix}$$

**Vorhersage-Matrix**

$$P = \begin{matrix} & \begin{matrix} \text{B}_1 & \text{B}_2 \end{matrix} \\ \begin{matrix} \text{A}_1 \\ \text{A}_2 \end{matrix} & \begin{pmatrix} u_{11} \cdot m_{11} + u_{12} \cdot m_{21} & u_{11} \cdot m_{12} + u_{12} \cdot m_{22} \\ u_{21} \cdot m_{11} + u_{22} \cdot m_{21} & u_{21} \cdot m_{12} + u_{22} \cdot m_{22} \end{pmatrix} \end{matrix}$$

**Formel für den Fehler**

$$(r_{11} - (u_{11} \cdot m_{11} + u_{12} \cdot m_{21}))^2 + (r_{12} - (u_{11} \cdot m_{12} + u_{12} \cdot m_{22}))^2 + \dots$$

**Frage:** Kann der Fehler von zwei Vorhersage-Matrizen mit unterschiedlicher Größe verglichen werden?

Workshop | Nestlé | 16/35

Optimierung

**Ziel:**  
Entwicklung eines mathematischen Optimierungsverfahrens mit dem eine Zerlegung computergestützt berechnet werden kann.

Workshop | Nestlé | 17/35

Optimierung

**Alternierendes Optimierungsverfahren**

$$R = \begin{pmatrix} 2 & ? & 1 & 5 \\ ? & 2 & 4 & 1 \\ 1 & 2 & ? & 2 \end{pmatrix} \quad P = \begin{pmatrix} ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \end{pmatrix}$$

$$U = \begin{pmatrix} ? & ? \\ ? & ? \\ ? & ? \end{pmatrix} \quad M = \begin{pmatrix} ? & ? & ? & ? \\ ? & ? & ? & ? \end{pmatrix}$$

**Gemittelte Summe der Fehlerquadrate:**



**Ziel:** Finde Einträge von U und M, sodass der Fehler minimal wird.

Workshop | Nestlé | 17/35

Optimierung

**Alternierendes Optimierungsverfahren**

$$R = \begin{pmatrix} 2 & ? & 1 & 5 \\ ? & 2 & 4 & 1 \\ 1 & 2 & ? & 2 \end{pmatrix} \quad P = \begin{pmatrix} ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \end{pmatrix}$$

$$U = \begin{pmatrix} ? & ? \\ ? & ? \\ ? & ? \end{pmatrix} \quad M = \begin{pmatrix} 0.59 & 0.27 & 0.16 & 0.87 \\ 0.05 & 0.11 & 0.47 & 0.62 \end{pmatrix}$$

**Gemittelte Summe der Fehlerquadrate:**



- Schritt:** Festlegung der Einträge von M auf beliebige Werte.

Workshop | Nestlé | 17/35

Optimierung

**Alternierendes Optimierungsverfahren**

$$R = \begin{pmatrix} 2 & ? & 1 & 5 \\ ? & 2 & 4 & 1 \\ 1 & 2 & ? & 2 \end{pmatrix} \quad P = \begin{pmatrix} 2.47 & 1.28 & 1.49 & 4.59 \\ -2.04 & -0.18 & 3.49 & 1.77 \\ 1.31 & 0.64 & 0.61 & 2.21 \end{pmatrix}$$

$$U = \begin{pmatrix} 4.07 & 1.73 \\ -4.27 & 8.86 \\ 2.18 & 0.52 \end{pmatrix} \quad M = \begin{pmatrix} 0.59 & 0.27 & 0.16 & 0.87 \\ 0.05 & 0.11 & 0.47 & 0.62 \end{pmatrix}$$

**Gemittelte Summe der Fehlerquadrate:**

 0.91451

- Schritt:** Festlegung der Einträge von M auf beliebige Werte.
- Schritt: Optimierung** Einträge von U so wählen, dass der Fehler minimal wird.

Workshop | Nestlé | 17/35

Optimierung

**Alternierendes Optimierungsverfahren**

$$R = \begin{pmatrix} 2 & ? & 1 & 5 \\ ? & 2 & 4 & 1 \\ 1 & 2 & ? & 2 \end{pmatrix} \quad P = \begin{pmatrix} 2.0 & 4.17 & 1.0 & 4.82 \\ -0.15 & 2.0 & 4.0 & 1.01 \\ 1.0 & 2.0 & 0.35 & 2.36 \end{pmatrix}$$

$$U = \begin{pmatrix} 4.07 & 1.73 \\ -4.27 & 8.86 \\ 2.18 & 0.52 \end{pmatrix} \quad M = \begin{pmatrix} 0.41 & 0.77 & 0.04 & 0.94 \\ 0.18 & 0.60 & 0.47 & 0.57 \end{pmatrix}$$

**Gemittelte Summe der Fehlerquadrate:**

 0.01781

- Schritt:** Festlegung der Einträge von M auf beliebige Werte.
- Schritt: Optimierung** Einträge von U so wählen, dass der Fehler minimal wird.
- Schritt:** Festlegung der Einträge von U auf gefundene Werte.
- Schritt: Optimierung** Einträge von M so wählen, dass der Fehler minimal wird.

Workshop | Nestlé | 17/35

Optimierung

**Alternierendes Optimierungsverfahren**

$$R = \begin{pmatrix} 2 & ? & 1 & 5 \\ ? & 2 & 4 & 1 \\ 1 & 2 & ? & 2 \end{pmatrix} \quad P = \begin{pmatrix} 2.07 & 4.3 & 1.01 & 4.97 \\ -0.15 & 1.99 & 4.0 & 1.01 \\ 0.85 & 1.92 & 0.35 & 2.13 \end{pmatrix}$$

$$U = \begin{pmatrix} 4.22 & 1.74 \\ -4.29 & 8.87 \\ 1.46 & 1.33 \end{pmatrix} \quad M = \begin{pmatrix} 0.41 & 0.77 & 0.04 & 0.94 \\ 0.18 & 0.60 & 0.47 & 0.57 \end{pmatrix}$$

**Gemittelte Summe der Fehlerquadrate:**

 0.00583

- Schritt:** Festlegung der Einträge von M auf gefundene Werte.
- Schritt: Optimierung** Einträge von U so wählen, dass der Fehler minimal wird.
- ...

Workshop | Nestlé | 17/35

### Optimierung

**Alternierendes Optimierungsverfahren**

$$R = \begin{pmatrix} 2 & ? & 1 & 5 \\ ? & 2 & 4 & 1 \\ 1 & 2 & ? & 2 \end{pmatrix} \quad P = \begin{pmatrix} 2.0 & 4.69 & 1.0 & 5.0 \\ 2.21 & 2.0 & 4.0 & 1.0 \\ 1.0 & 2.0 & 0.82 & 2.0 \end{pmatrix}$$

$$U = \begin{pmatrix} 4.22 & 1.74 \\ -4.29 & 8.87 \\ 1.46 & 1.33 \end{pmatrix} \quad M = \begin{pmatrix} 0.30 & 0.77 & 0.04 & 0.94 \\ 0.18 & 0.60 & 0.47 & 0.57 \end{pmatrix}$$

Gemittelte Summe der Fehlerquadrate: 0.00001

- Schritt: Festlegung der Einträge von M auf gefundene Werte.
- Schritt: Optimierung Einträge von U so wählen, dass der Fehler minimal wird.
- ...

Wiederholung 8: U fest, M wird optimiert.

### Optimierung | Ein Blick in die Schulmathe...

**Optimierung**

$$R = \begin{pmatrix} E1 & E2 \\ 2 & 3 \end{pmatrix}$$

$$U = \begin{pmatrix} E1 & E2 \\ u_{11} & u_{12} \end{pmatrix} \quad M = \begin{pmatrix} E1 & E2 \\ 1 & 1 \\ E2 & 1 & 1 \end{pmatrix}$$

In unserem Beispiel: Fehlerfunktion error mit  $error(u_{11}, u_{12}) = (2 - (u_{11} \cdot 1 + u_{12} \cdot 1))^2 + (3 - (u_{11} \cdot 1 + u_{12} \cdot 1))^2$

Gesucht: Minimum

Schulnahes Beispiel: Funktion f mit  $f(x) = 0.1 \cdot x^3 + 2 \cdot x^2$

### Diskussion nach AB 4 | Modellierungskreislauf

Wie gut ist die Vorhersage von bekannten und unbekannt Daten?

```

    graph TD
        RP[Reales Problem  
Filmpfehlungen für viele  
tausend Nutzer] -- vereinfachen --> VP[Vereinfachtes Problem  
Filmpfehlungen für  
wenige Nutzer]
        VP -- mathematisch beschreiben --> MP[Mathematisches Problem  
R ≈ U · M  
Optimierungsproblem]
        MP -- lösen / berechnen  
Optimierungsverfahren --> ML[Mathematische Lösung  
P = U · M]
        ML -- interpretieren --> RP
    
```

### Diskussion nach AB 4 | Interpretieren und Validieren

Die Vorhersage der **bekannt** Bewertungen funktioniert schon ziemlich gut!

Rating-Matrix R:

3.0	1.0	0.0	2.0	1.0	0.0
1.0	0.0	4.0	1.0	3.0	1.0
0.0	1.0	1.0	0.0	1.0	4.0
0.0	0.0	5.0	0.0	4.0	0.0

Vorhersage-Matrix P:

3.0	1.0	1.665	2.0	1.082	11.584
1.0	5.34	3.946	1.0	3.044	1.0
1.135	1.0	1.051	0.798	0.743	4.0
-0.133	7.493	5.031	0.413	3.992	-4.778

### Diskussion nach AB 4 | Interpretieren und Validieren

Die Vorhersage der **unbekannt** Bewertungen sieht weniger gut aus...

Rating-Matrix R:

3.0	1.0	0.0	2.0	1.0	0.0
1.0	0.0	4.0	1.0	3.0	1.0
0.0	1.0	1.0	0.0	1.0	4.0
0.0	0.0	5.0	0.0	4.0	0.0

Vorhersage-Matrix P:

3.0	1.0	1.665	2.0	1.082	11.584
1.0	5.34	3.946	1.0	3.044	1.0
1.135	1.0	1.051	0.798	0.743	4.0
-0.133	7.493	5.031	0.413	3.992	-4.778

Die Vorhersagen liegen zum Teil gar nicht zwischen 1 und 5.

### Diskussion nach AB 4 | Modellierungskreislauf

Modellverbesserung: Regularisierung

```

    graph TD
        RP[Reales Problem  
Filmpfehlungen für viele  
tausend Nutzer] -- vereinfachen --> VP[Vereinfachtes Problem  
Filmpfehlungen für  
wenige Nutzer]
        VP -- mathematisch beschreiben --> MP[Mathematisches Problem  
R ≈ U · M  
Optimierungsproblem]
        MP -- lösen / berechnen  
Optimierungsverfahren --> ML[Mathematische Lösung  
P = U · M]
        ML -- interpretieren --> RP
        subgraph MB [Modellverbesserung: Regularisierung]
            direction TB
            M1[ ] --> M2[ ]
        end
    
```

### Diskussion nach AB 4 | Regularisierung

Welchen Einfluss hat der Parameter  $\lambda$  auf die Einträge von U und M?

$$(r_{11} - (u_{11} \cdot m_{11} + u_{12} \cdot m_{21}))^2 + (r_{12} - (u_{11} \cdot m_{12} + u_{12} \cdot m_{22}))^2 + \dots + \lambda \cdot (u_{11}^2 + u_{12}^2 + m_{11}^2 + m_{12}^2 + m_{21}^2 + m_{22}^2 + \dots) \rightarrow \text{minimal}$$

### Zusatzblatt: Diskussion der Fehlerfunktionen

Was sind Vor- und Nachteile der folgenden Fehlermaße

- Summe der absoluten Abweichungen
- Summe der quadratischen Fehler

bei der Optimierung?

### Diskussion nach AB 4 | Zwischenfazit

- Modell zur Vorhersage von Ratings: Matrix-Faktorisierung ✓
- Verfahren zur Berechnung von User-Matrix  $U$  und Movie-Matrix  $M$  ✓
- Validierung, wie gut eine gefundene Zerlegung bekannte Bewertungen darstellt ✓

**Aber... ?**

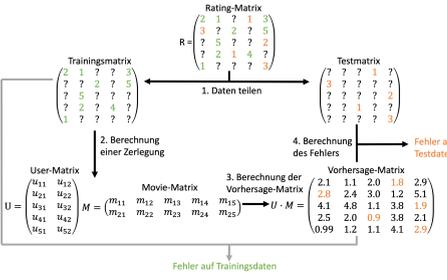
- Wie gut lässt sich unser Modell zur Vorhersage von **fehlenden bzw. unbekannt**en Bewertungen anwenden?



Workshop | Netflix | 24/35

### Unterteilung der bekannten Bewertungen

**Wie gut können wir unbekannte Daten vorhersagen?**



1. Daten teilen

2. Berechnung einer Zerlegung

3. Berechnung der Vorhersage-Matrix

4. Berechnung des Fehlers

Fehler auf Trainingsdaten

Fehler auf Testdaten

Workshop | Netflix | 25/35

### Maschinelles Lernen, was ist das?

- Beim maschinellen Lernen sollen Muster und Zusammenhänge aus großen Datenmengen erkannt und neues Wissen gewonnen werden.  
Millionen Bewertungsdaten sollen genutzt werden, um Präferenzen der User vorherzusagen. 👤
- Dazu werden Modelle und Algorithmen entwickelt...  
Matrix-Faktorisierung & Optimierungsverfahren ✍️
- und mithilfe von Computern umgesetzt.  
Jupyter Notebooks und Julia 📄
- Das aus den Daten gewonnene Wissen wird auf ein Problem (Vorhersage, Klassifizierung, ...) angewendet.  
Aussprechen von Filmempfehlungen 📺

Workshop | Netflix | 26/35

### Diskussion nach AB 4 | Maschinelles Lernen

Ihr seid dran!  
Beginnt mit der Bearbeitung von AB5.

Workshop | Netflix | 27/35

### Diskussion nach AB 5 | Eure Ergebnisse

- Mit welchen Parametern habt ihr das beste Ergebnis auf den Netflixdaten erhalten?
  - Ausgangspunkt der Challenge: 0.976 (Netflix)
  - Gewinnerteam: 0.734
- Weitere Modellverbesserungsideen für unser Modell?

Workshop | Netflix | 28/35

### Diskussion nach AB 5 | Ergebnisse der Parameterstudie



- Welchen Einfluss hat der Regularisierungsparameter  $\lambda$  auf
  - den Trainingsfehler?
  - den Testfehler?

Workshop | Netflix | 29/35

### Diskussion nach AB 5 | Ergebnisse der Parameterstudie



- Welchen Einfluss hat die Anzahl der Eigenschaften (Parameter features) auf
  - den Trainingsfehler?
  - den Testfehler?

Workshop | Netflix | 30/35

### Abschlussdiskussion | Teil 1

Viele große Unternehmen (Netflix, Amazon, Zalando und Co) nutzen Empfehlungssysteme, um unsere Interessen und Vorlieben zu analysieren und vorherzusagen.

**Was ist eure persönliche Meinung:**

- Findet ihr Empfehlungssysteme gut / hilfreich?
- Findet ihr es gut oder schlecht, dass eure Nutzervergangenheit ausgenutzt wird, um euch neue Produkte vorzuschlagen?

Workshop | Netflix | 31/35

Abschlussdiskussion | Teil 2

*Im Jahr 2008 haben zwei Forscher von der Uni Texas den anonymen Netflixdatensatz mit einem weiteren öffentlichen Datensatz (Internet Movie Database) kombiniert. Durch die Kombination beider Datensätze konnten die Forscher die Identität von verschiedenen Nutzern und deren Watch-History öffentlich machen.*

**Haltet ihr diese De-Anonymisierung für problematisch? Warum?**

Abschlussdiskussion | Kurz zusammengefasst ...

In diesem Projekt habt ihr ...

- mit einem realen Datensatz gearbeitet
- ein Modell zur Vorhersage unbekannter Bewertungen entwickelt (Matrix-Faktorisierung).
- ein Optimierungsverfahren erarbeitet / angewandt.
- Strategien des maschinellen Lernens genutzt.
- einen Einblick in die mathematische Modellierung erhalten.
- (hoffentlich) Neues gelernt und dabei Spaß gehabt.

Abschlussdiskussion | Teil 3

**Wie könnte man ein Empfehlungssystem manipulieren? Wer könnte daran interessiert sein?**

**Wie gehen wir mit Usern um, die noch keine Bewertung abgegeben haben?**

Evaluation

Eure Meinung ist gefragt.



[www.umfrageonline.com/s/39fa586](http://www.umfrageonline.com/s/39fa586)

Vielen Dank!

# F.10 Material für Modellierungswochen

## F.10.1 Problembeschreibung

### Entwicklung eines Empfehlungssystems für Filme

Stephane Hofmann  
&  
Netflix Challenge – gemittelt von Sarah Schölkopf

Netflix, Amazon und Co setzen bei der Kundenbindung vor allem auf eins: gute, auf den Nutzer zugeschnittene Empfehlungen für neue Produkte, Filme, Kleidung etc. Dazu entwickeln diese Unternehmen Empfehlungssysteme, die möglichst gut vorhersagen sollen, was dem/jeder jeweiligen Nutzer/in gefallen könnte. Um das eigene Empfehlungssystem weiter zu verbessern richtete Netflix 2006 eine Challenge aus: Das Team, was mindestens 10 Prozent genauer vorhersagen kann, welche Filme einem Nutzer gefallen, hatte Chancen auf den Hauptpreis von 1.000.000 \$. Netflix veröffentlichte im Rahmen der Challenge einen Datensatz mit den Bewertungen tausender Nutzer. Die Bewertungen reichen von 1 (schlechteste) bis 5 (beste). Zahlreiche Filme hatten die Nutzer noch nicht bewertet.

3	?	1	4	...		
?	2	4	1	3	1	...
3	1	?	3	?	?	...
4	3	?	4	4	?	...
4	?	?	4	?	5	...
...	...	...	...	...	...	...

Ziel der Challenge war es, bestmöglich vorherzusagen, wie viele Nutzer/in einen noch nicht bewerteten Film bewerten würde, um dann entsprechende Empfehlungen auszusprechen. Zum Beispiel:

3	2	1	5	1	4	...
1	2	4	1	3	1	...
3	1	3	3	2	1	...
4	3	1	4	4	2	...
4	2	2	4	3	5	...
...	...	...	...	...	...	...

**Eure Aufgabe:**  
Entwickelt basierend auf dem Netflix-Datensatz euer eigenes Empfehlungssystem – und zwar so, dass der Geschmack der Nutzer/innen bestmöglich getroffen wird. Entwickelt euer Modell zunächst für einen kleinen Beispieldatensatz. Testet es anschließend auf den gegebenen Testdatensätzen und verbessert euer Modell, wenn möglich. Nutzt zur Bewertung, wie gut euer Modell die Bewertungen der Nutzer/innen vorhersagen kann, das Fehlermaß mit dem auch Netflix die Teams bei der Challenge bewertet hat:  
*Wurzel der gemittelten Fehlerquadratsumme* (engl. Root Mean Squared Error)

**Wenn noch Zeit ist:**

- Was schlägt ihr Nutzern/innen vor, die noch nie eine Bewertung abgegeben haben?
- Wer könnte Interesse daran haben, euer Empfehlungssystem zu manipulieren? Wie könnte er/sie das anstellen? Und wie könnt ihr euer System vor solchen Manipulationen schützen?

**Ausgangsdaten:**

- Bewertungsdaten der Nutzer/innen aus dem Datensatz der Netflix Challenge
- Verschiedene Testdatensätze
- Titel und Erscheinungsjahr der Filme

Viel Erfolg!

## F.10.2 Jupyter Notebooks

### Die Rating-Tabelle

Wir können alle Bewertungen, die die Nutzer für die Filme abgegeben haben, in einer Art Tabelle bzw. in einem rechteckigen Zahlenschema notieren. Jede Zeile steht für einen User und jede Spalte für einen Film. Die Einträge der Tabelle geben an, welche Bewertung ein Nutzer für einen Film abgegeben hat. Jedoch sind nicht alle Bewertungen bekannt.

Ein Auszug aus der Tabelle könnte wie folgt aussehen:

	1	2	3	4	5	6	...
1	3	?	1	?	1	4	...
2	?	2	4	1	3	1	...
3	3	1	?	3	?	?	...
4	4	3	?	4	4	?	...
5	4	?	?	4	?	5	...
...	...	...	...	...	...	...	...

Der Netflixdatensatz besteht aus:

- 17770 Filmen
- 480189 Nutzern
- 100480507 Bewertungen (engl. *ratings*) von 1 - 5

### Auswahl eines Teildatensatzes

Ihr braucht während der CAMMP week nicht mit dem kompletten, sehr großen Datensatz zu arbeiten. Stattdessen liegen 3 kleinere Datensätze bereit, auf denen ihr euer Empfehlungsmodell entwickeln und testen sollt.

```
In [ ]: # Pakete laden
import numpy as np
import json
```

Die Werte in der Tabelle 'r\_train' sind bekannt. Diese könnt ihr nutzen um eine Empfehlungsmodell zu entwickeln.

```
In [ ]: # Trainingsdaten
r_train_1 = np.load('../data/dataset_6/r_train_6.npy', allow_pickle=True)
# np.set_printoptions(threshold=np.inf)
print(r_train_1)

print("Anzahl User (=Zeilen) und Anzahl Movies (=Spalten) im Teildatensatz:")
np.shape(r_train_1)
```

Achtung: Es sind nicht alle Einträge Null, wie es auf den ersten Blick scheint!

```
In [ ]: # Anzahl Einträge größer 0 (= Anzahl der Trainingsbewertungen)
np.sum(r_train_1 > 0)
```

Jede Spalte in der Tabelle 'r\_train' steht für einen Film. Die Zuordnung der Spalten zu den

Filmen und deren Erscheinungsjahr haben wir in zwei sogenannten *dictionary* (eine Art Lexikon) gespeichert. Ihr könnt euch mit Hilfe dieser den Titel und das Erscheinungsjahr anzeigen lassen.

```
In [ ]: # Laden des dictionary für die Zuordnung Spalte -> Erscheinungsjahr
with open('../data/dataset_1/map_release_year_1.json') as f:
    dict_movie_year = json.load(f)
dict_movie_year = {int(k): v for k, v in dict_movie_year.items()}

# Laden des dictionary für die Zuordnung Spalte -> Titel
with open('../data/dataset_1/map_title_1.json') as f:
    dict_movie_title = json.load(f)
dict_movie_title = {int(k): v for k, v in dict_movie_title.items()})
```

```
In [ ]: # Auswahl einer Filmspalte
film_spalte = 2

# Zugriff auf Einträge im dictionary
dict_movie_title[film_spalte]

print(f'Erscheinungsjahr und Titel von Film in Spalte {film_spalte}: \n(dict_m
```

### Validierung eures Modells

Mit den Werten in der Tabelle 'r\_test' könnt ihr eure Ergebnisse vergleichen und bewerten, wie gut euer Modell die Bewertungen vorhergesagt hat!

```
In [ ]: r_test_1 = np.load('../data/dataset_1/r_test_1.npy', allow_pickle=True) # Tes
```

**⚠ Hinweis:** Die Gänsefüßchen " " " müssen um die Bezeichnung der Spalte stehen bleiben.

```
In [ ]: # Sortierung nach Spalte "sortI"
sortI = "durchschnittl. Rating"
# True = aufsteigende Sortierung, False = absteigende Sortierung
aufsteigend = False
# Ab hier nichts ändern!
df_table.sort_values(by=[sortI], ascending = aufsteigend)
```

**⚠ Hinweis zur Tabelle:** Jeder Film hat eine eindeutige Kennzeichnung bzw. Nummer: seine ID.

## Netflix Datensatz

Auf diesem Notebook kannst du dir einen ersten Eindruck von dem gesamten Datensatz der Netflix Challenge verschaffen.

```
In [ ]: # Hier nichts ändern!
import sys; sys.path.append("../code"); from setup_data_netflix import * ;

In [ ]: I = np.load('../data/npys/I.npy', allow_pickle=True, fix_imports=1) #Movie ID
J = np.load('../data/npys/J.npy', allow_pickle=True) #User ID
V = np.load('../data/npys/V.npy', allow_pickle=True) #Rating of user. J for mov

# Show some stats
print("Nicht-Null Einträge der Bewertungsmatrix: ")
print(I.size)
print("Number of unique movies in the training set, i.e. columns of the matrix:")
print(np.unique(I).size)
print("Number of unique users in the training set, i.e. rows of the matrix:")
print(np.unique(J).size)
print("The highest user id is:")
print(np.amax(J))
```

## Infos über Filme im Datensatz

Wir haben bereits ein paar Infos zu den Filmen im Datensatz zusammengefasst und visualisiert.

**Grafische Darstellung: Erscheinungsjahr und durchschnittliches Rating**

```
In [ ]: ab_jahr = 1900
create_rls_av_std_plot(ab_jahr)
```

**Grafische Darstellung: Erscheinungsjahr und Anzahl Ratings**



Führe den folgenden Code aus. Dir wird eine Grafik angezeigt, in der das Erscheinungsjahr auf der x-Achse und die Anzahl der Ratings eines jeden Filmes auf der y-Achse aufgetragen sind.

```
In [ ]: # Hier nichts ändern!
create_rls_count_av_plot(ab_jahr)
```

## Tabellarische Übersicht

Du kannst die Tabelle nach einer beliebigen Spalte sortieren. Lege fest, nach welcher Spalte (z. B. Anzahl Ratings, Erscheinungsjahr ...) sortiert werden soll. Trage dazu den Namen der Spalte für die Variable `sortI` ein. Du kannst außerdem festlegen, ob die Tabelle aufsteigend (`aufsteigend = True`) oder absteigend (`aufsteigend = False`) sortiert werden soll.



# G Evaluationen der Durchführungen

## G.1 Evaluationsergebnisse zum Thema Solarenergie

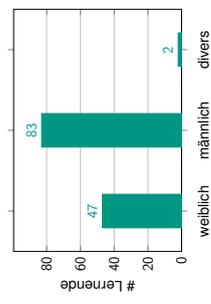
### Evaluationen der Workshops zum Thema Solarenergie

*Hinweis:* Die Antworten zu den offenen Fragestellungen sind nach den Themen gruppiert worden. Dabei ist Thema I = Optimierung der Rohrhöhe, Thema II = Optimierung der Spiegelpositionen, Thema III = Bearbeitung offener Modellierungsprobleme. Die Lernenden haben nicht immer alle Fragen beantwortet, weshalb die Anzahl der Antworten je Frage variiert.

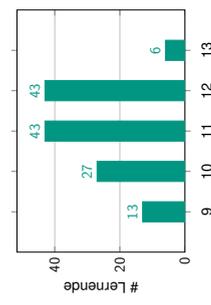
#### 1 Angaben zur eigenen Person und Schule

Anzahl Teilnehmende: 132

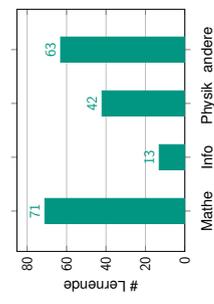
##### Geschlecht:



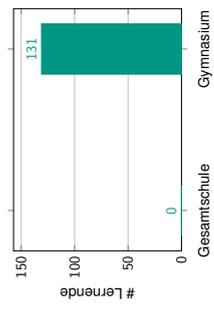
##### Jahrgangsstufe

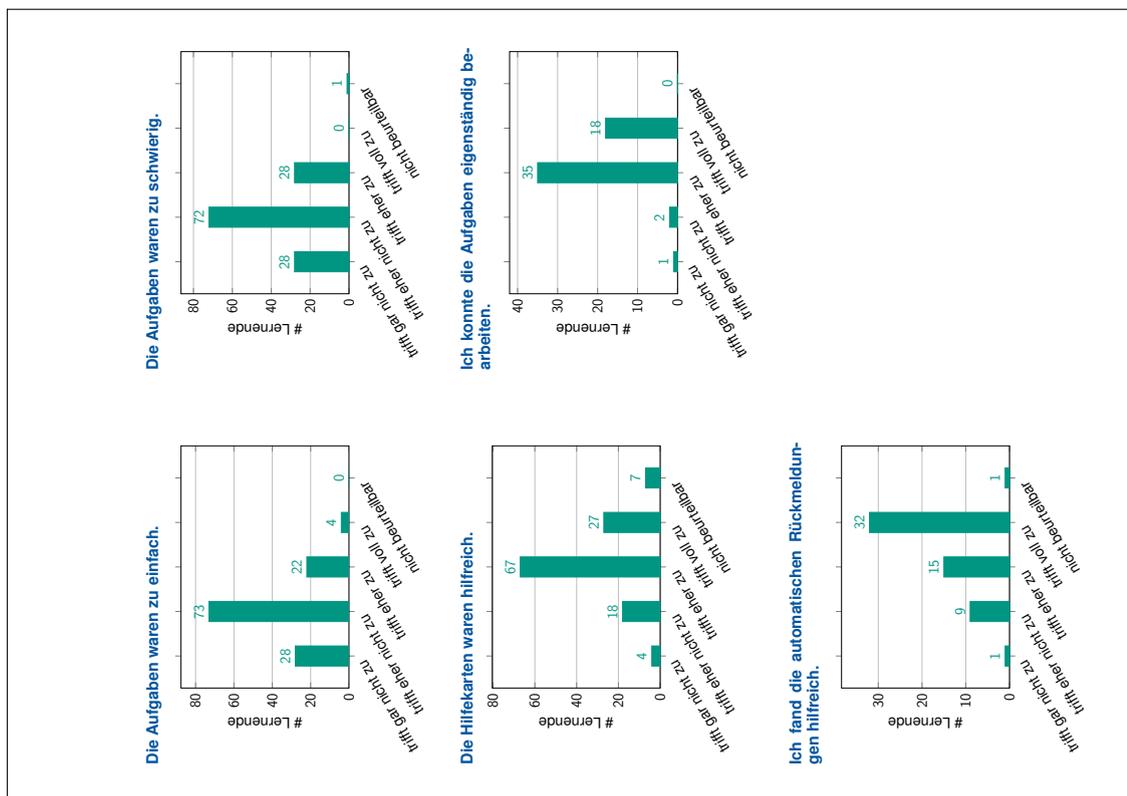
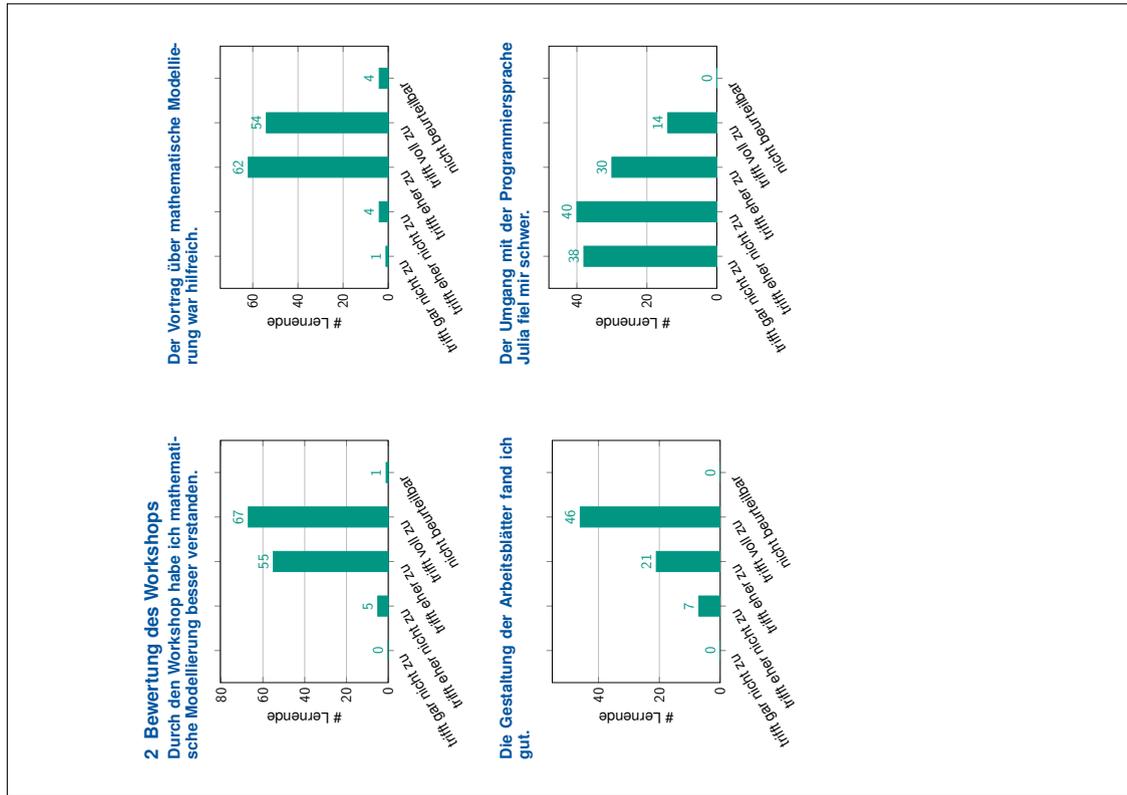


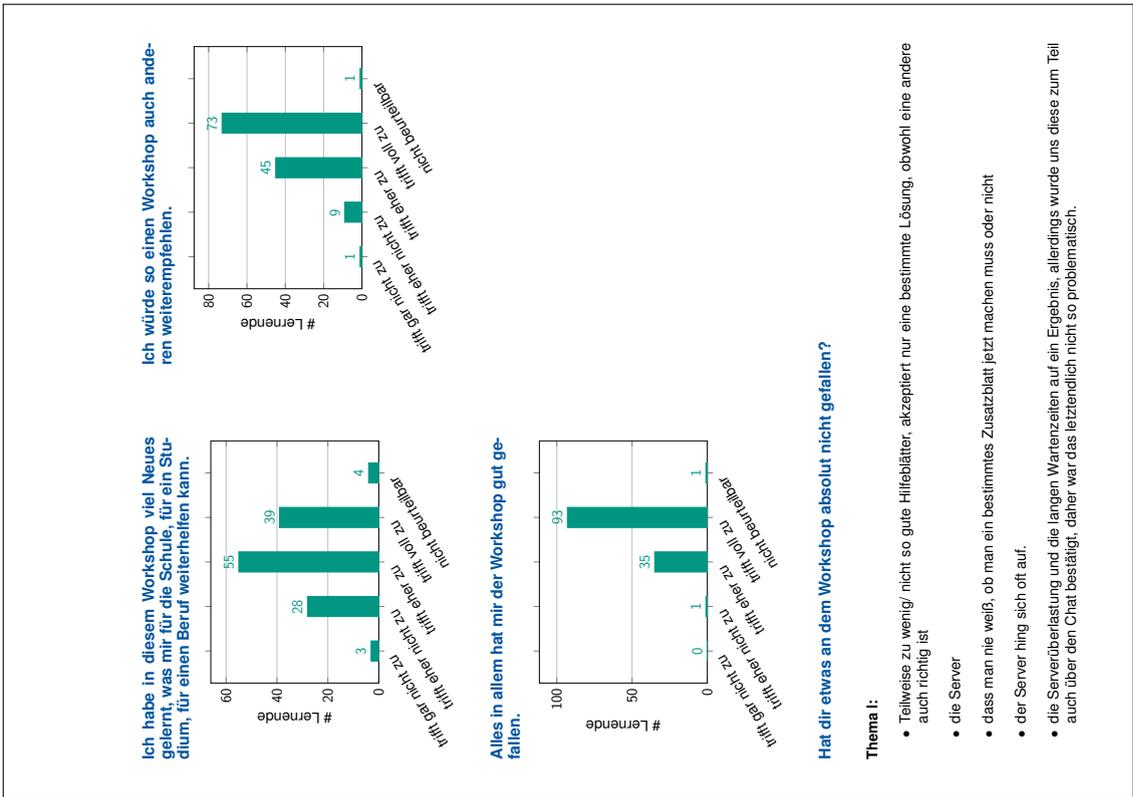
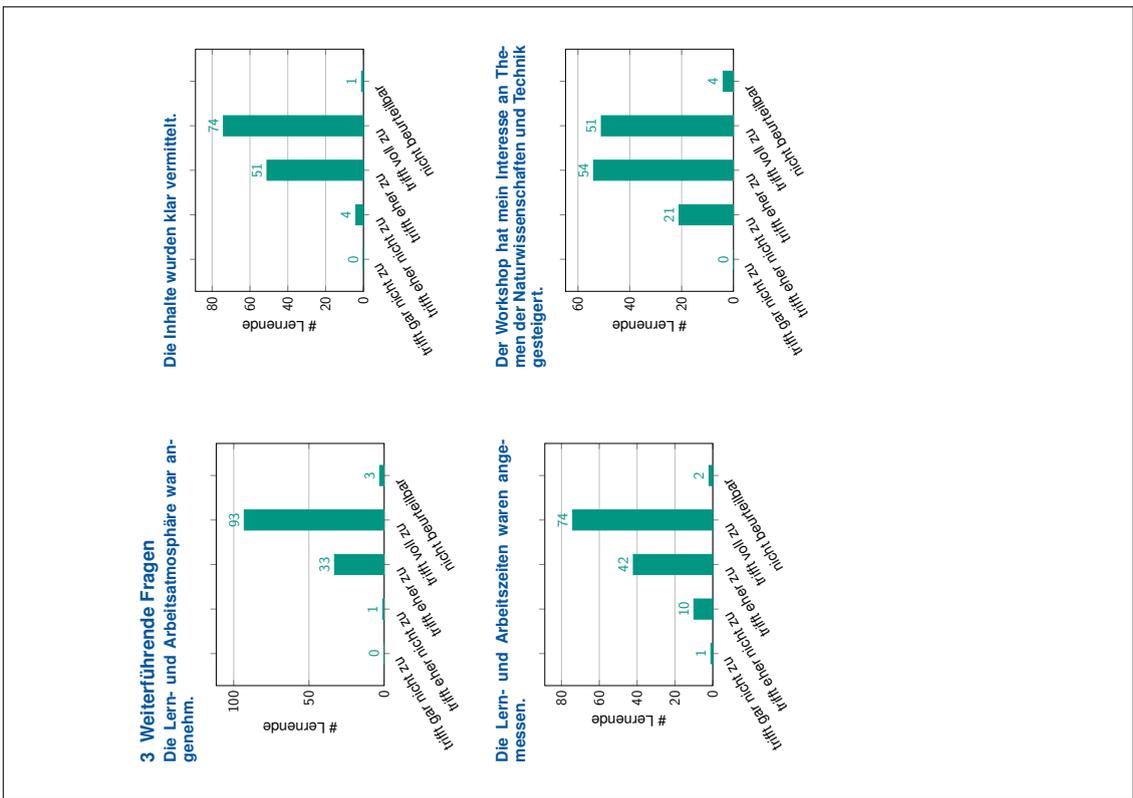
##### Leistungskurse



##### Schulform







- dass es ein sehr tolles Thema ist und, dass es auch sehr gut erklärt war.

- die gute Arbeit der Betreuer

- der Umgang mit einer neuen Programmiersprache und Infos über das Solarkraftwerk.

- Ein wenig über die Programmiersprache Julia zu lernen.

- dass man in Gruppen gearbeitet hat

- Die Lernatmosphäre war sehr gut

- Das eigenständige Lösen von Herausforderungen

- die Gruppenarbeit

- die Atmosphäre

- Die freie Gruppenarbeit.

- Die anwendungsnahen Fragestellungen

- das selbständige anwenden von Formeln auf reale Probleme.

- Erkenntnisgewinn

- das selbständige Arbeiten in Kleingruppen.

- Es war schön, dass es so viele Grafiken gab, die wir auch vorstellen konnten. Es ist gut so ein bisschen visuelle Hilfe zu haben.

- der Bezug zur Realität und Aktualität, schließlich sind Solarkraftwerke wichtig für die Zukunft und nichts aus vergangenen Zeiten.

- Das eigenständige Arbeiten

## Thema I + III

- Jetzt am letzten Tag endlich in Gruppen selbstständig arbeiten zu können

- Das selbständige Programmieren am Ende

- Offene Gruppenarbeit am Ende

- Ich finde einerseits das übermittelte Wissen über die Funktionsweise von Solarkraftwerken sehr interessant und, dass man gesehen hat wie viel mit ein paar trigonometrischen Tricks alles lösen konnte.

- Dass ich viel Neues gelernt habe, was ich später anwenden kann.

- Das freie Arbeiten und das die Tutoren sich so flexibel auf uns angepasst haben

## Thema II

- Die Vorbereitungen der Aufgaben war super

- Essen

- Oft war der Server langsam

- die Leistung des Servers

## Thema I + III

- Zu viele Themen - lieber ein großes, wenn man mehrere Tage dran arbeitet

- Nein

- Teilweise haben sich die Aufgaben so sehr gemüht, dass es meistens immer das gleiche war, was irgendwann ein bisschen langweilig wurde.

- Viele Wiederholungen im ersten nicht freien Teil des Workshops

- Man soll vielleicht doch MATLAB benutzen. Mir gefällt Julia, leider nicht...

## Thema II

- Ich fand es schade das die Arbeit am Ende abgebrochen wurde.

- Zum Ende hin ging die Motivation verloren

- Wo sind die Donuts?

## Thema II + III

- Corona (kann man aber nichts gegen machen)

- dass Lösungen, die richtig waren als falsch angezeigt wurden und man nur mit mehrmaligen Durchlaufen eine richtige Antwort bekommen hat

- Zu einfache Aufgaben

- Das Jitsi verwendet wurde, obwohl da schon früher oft Probleme aufgetreten sind. Das hat viel Zeit gestohlen und es hätte direkt Zoom verwendet werden können.

- nächstes Mal vielleicht eine kurze Pause zwischendurch

- In der ersten Gruppenarbeitsphase fiel die Kommunikation etwas schwer.

- Das Whiteboard und die Technischen Probleme

- Nicht absolut nicht gefallen aber an den technischen Feinheiten (Jitsi z.B.) gab es Schwierigkeiten und die willkommensrunde am Anfang hätte man besser in breakour räumen oder kombiniert machen können, finde ich.

- Freies Programmieren bei schulischen Programmierkenntnissen

- Teilweise die Gruppen

**Hat dir etwas an dem Workshop besonders gut gefallen?**

Thema I

<ul style="list-style-type: none"> <li>• Die Arbeitsblätter / Arbeitsumgebung</li> <li>• Die Arbeitsblätter, die sehr gut strukturiert waren und angenehm zu bearbeiten</li> <li>• Die Freiheiten sich in bestimmten Themen besonders einzulesen.</li> <li>• Dass die Teilnehmer sich vieles mithilfe der Worksheets selbst erarbeiten konnten</li> <li>• Das Heranführen an Programmiersprache Julia.</li> <li>• Die interaktiven Arbeitsblätter</li> <li>• viel Interaktion, man konnte selber mit anderen an den Problemen arbeiten. Dabei wurde man aber durch die Arbeitsblätter systematisch unterstützt.</li> <li>• Das eigene Projekt</li> <li>• Die Möglichkeiten, frei und nach dem eigenen Tempo, sowie an eigenen Projekten zu arbeiten hat mir persönlich gut gefallen.</li> <li>• das freie Arbeiten</li> <li>• Die Motivation der Vermittelnden über den Workshop hinaus Interesse zu wecken (durch weiterführende Materialien, wertschätzender Umgang etc.)!</li> <li>• Die Arbeitsblätter</li> <li>• Das eigenständige bzw. das Arbeiten in sehr kleinen Gruppen, aber dennoch mit der Möglichkeit bei Schwierigkeiten Hilfe zu bekommen.</li> <li>• Der praktische Bezug/Hintergrund zur mathematischen Modellierung</li> <li>• Ich fand die Zeiten sehr gut geregelt</li> <li>• Bearbeiten der Arbeitsblätter im Team</li> <li>• Hilfsbereitschaft</li> </ul> <p><b>Hättest du gerne noch etwas anderes gesehen oder erfahren?</b></p> <p><b>Thema I</b></p> <ul style="list-style-type: none"> <li>• Noch detaillierter modellieren</li> <li>• die anderen Arbeitsblätter</li> <li>• mehr Programmieren</li> <li>• Die anderen Fragestellungen (Arbeitsblätter) bearbeitet</li> </ul> <p><b>Thema I + III</b></p> <ul style="list-style-type: none"> <li>• Gerne noch mehr über die genauere Funktionsweise der Solarkraftwerke</li> </ul>	<ul style="list-style-type: none"> <li>• Diskussion am Ende der Arbeitsphase</li> <li>• Dass wir in Gruppen selbstständig arbeiten konnten</li> <li>• Das Essen</li> <li>• Selbstständige Arbeiten</li> <li>• Kompetente Durchführung, Atmosphäre</li> <li>• Arbeitsblätter waren anschaulich, schön das man selber arbeiten konnte und immer jemand da war zum helfen</li> <li>• Herangehensweise an das Projekt</li> <li>• Viel</li> <li>• Das Modellieren hat Spaß gemacht.</li> </ul> <p><b>Thema II + III</b></p> <ul style="list-style-type: none"> <li>• Programmieren an sich</li> <li>• Die Gestaltung der Arbeitsblätter, der Umgang mit den Schüler*innen + Beratung/Hilfe, der Schwierigkeitsgrad</li> <li>• Abbildungen</li> <li>• Struktur der Worksheets + Hilfestellungen</li> <li>• die Vorstellung und der Vergleich der Algorithmen und dann auch das freie Arbeiten an einem eigenen.</li> <li>• die Vorstellung der Algorithmen</li> <li>• Der Umgang mit dem Distanzlernen</li> <li>• Das man direkt Hilfe bekommt wenn man sie braucht</li> <li>• Die vorgegebenen Codefelder, in den man seine eigenen Formeln/Algorithmen einfügen muss, finde ich sehr angenehm und intuitiv und die direkte Kontrolle ist sehr hilfreich.</li> <li>• die Optimierungen</li> <li>• die Algorithmen</li> <li>• die gemeinsamen Diskussionsrunden</li> <li>• Das freie Arbeiten</li> <li>• Die Möglichkeit selber mal mit Algorithmen herumzuprobieren statt sozusagen nur Formeln auszufüllen</li> <li>• Die Erklärungen waren anschaulich und sehr verständlich</li> <li>• anwendungsnahe Seite der Mathematik; gute, übersichtliche Gestaltung der ABs</li> <li>• Weitere Programmiersprache, Simulationen besser verstehen lernen</li> </ul>
--	--

**Thema II**

- Aktueller Bezug
- Ich hätte es gut gefunden, wenn man bei den anderen Beispielen, wo man solche Modellierungen macht, noch erzählt hätte, was bzw. wie man Optimierungen durchführt oder generell, dass man weitere Optimierungsverfahren kennenlernt.
- Mehr Theorie
- Andere Optimierungsverfahren, mathematische Einblick
- Die (physikalischen) Prozesse in einem Solarkraftwerk im Sinne von Wirkungsgrad, Vergleichbarkeit mit anderen Energieformen etc
- Eventuell Funktionsweisen anderer Solarzell Alternativen
- Konkretere Ergebnisse/Vorstellungen.

**Thema II + III**

- Rechenleistung der KIT - Server
- vlt noch eine kurze Übersicht gegen Ende, was alles getan bzw. versucht wurde.
- Ich hätte gern die Optimierungsverfahren eigens implementiert
- Konkretere Berufsaussichten, Mehr Infos zu Julia
- Gerne mehr Informationen zu realen Projekten und optimierungs"problemen". Die Anordnung einiger weniger Spiegel ist ja ziemlich vereinfacht.
- Die Technik hinter dem Fresnelkraftwerk mit .B. Wirkungsgrad etc. Andere Solarkraftwerke wie einen Solarturm
- Wie das Wetter den Energieverlust beeinflusst, und ob bestimmte Materialien (Absorberroh) besser geeignet wären als andere.
- Wie genau der Strom mit dem Satzwasser gespeichert werden kann (auch physikalische Gesetze, welche dahinter stecken)
- in welchen Berufen oder Tätigkeitsfeldern so eine Arbeit alles zum Einsatz kommt
- Podiumsdiskussion zum tatsächlichen Einsatz, Vergleich mit anderen Solarthermekraftwerken
- Wenn man eine falsche Formeln geschrieben hat, fällt es erheblich schwerer diese zu korrigieren, wenn man nicht den Output des Codes kennt. Deshalb wünsche ich mir, dass, zumindest wenn man eine falsche Formel angegeben hat, das Output angezeigt wird.
- Mehr unterschiedliche Varianten solcher Kraftwerke und auch eine Vertiefung in die Technik und nicht in den optimalen Aufbau eines einzelnen Kraftwerks.
- Wie weit ist die Wissenschaft in diesem Bereich? Forscht das KIT auch daran?
- Beispielrechnungen, Beispiel aus der aktuellen Forschung und Entwicklung

**4 Lernzuwachs**  
**Was hast du für dich persönlich durch die Teilnahme am Workshop gelernt?**

**Thema I:**

- Wie Solarenergie hergestellt wird und welche Bedingungen dafür erfüllt werden müssen um viel Strom daraus zu gewinnen. Außerdem habe ich gelernt wie strom funktioniert.
- Umgang mit Julia
- Funktion eines solarkraftwerks
- Wenn man mathematisch modelliert, macht man es so einfach wie möglich, und lässt es dann immer schwerer werden.
- Das es schwieriger ist solche Dinge (z.B. Solar) umzusetzen, als man auf den ersten Blick denkt.
- wie man mathematisch modelliert
- wie man sich gut konzentrieren kann
- Wie ich Herausforderungen bestehe, dass man mit ein bisschen Nachdenken viel erreichen kann, das unsere Welt voller faszinierender Technik steckt, die nur durch Mathematik möglich wird.
- Sinus, Cosinus und Tangens, die hatten wir noch nicht in der Schule
- Ich habe viele mathematischen Fähigkeiten, vertieft und verbessert.
- Ich habe vieles Neues gelernt.
- Ich habe gelernt, wie man bestimmte Mathematische Funktionen anwendet.
- Ich habe neue Seiten der Mathematik kennengelernt und konkrete Einblicke in mathematische Anwendungsbereiche erhalten.
- Rechnen mit etwas komplexeren Aufgabenstellungen als in der Schule - was mathematische Modellierungen sind - arbeiten mit einer Programmiersprache - selbständig Probleme lösen - neue Formeln
- Das es zum Glück auch eine eher mehr spaßige Version von Mathematik gibt, auch bei der man eben normalerweise gegebene Hilfsmittel, wie Programme benutzt, ganz im gegensatz zu dem meist langweiligem Schulmathe.
- Ich kann etwas besser mit solchen Programmen wie Julia umgehen. Generell hab ich aber vieles mathematisches dazugelernt. In der Schule haben wir noch keine Trigonometrie gemacht, also hab ich das auch jetzt gelernt und etwas angewendet.
- Ein Problem ist leichter zu lösen, wenn man anfangs möglichst viele Faktoren außer Acht lässt und diese erst später in seine Überlegungen miteinbezieht.
- Nachdenken und Grübeln ist hilfreich und macht meistens auch Spaß, wie die Berechnungen zustande kommen. Und wo alles Mathematik (und Physik) im Alltag gebraucht wird (werden).
- Verknüpfung von Mathematik mit Physik und Informatik

**Thema I + III**

- Was mathematische Modellierung eigentlich ist und wie man vorgeht.
  - Julia ist böse
  - Das Julia (Programmiersprache) sehr schlecht ist...
  - Ich habe gelernt, wie man mit Julia umgeht. Da ich seit 2 Jahren kein Informatik mehr habe. Zudem hat es mein Interesse an Informatik gesteigert.
  - Funktionsweise von dieser Art von Solarkraftwerk
  - Ein bisschen über die Funktionsweise von Solarkraftwerken und praktische Anwendung von mathematischem Wissen.
  - Mathematisch zu modellieren und zu optimieren
  - Anwendung der mathematischen Methoden in realistischen Problem.
- Thema II**
- Ich mag Julia nicht
  - Mathematische Probleme lösen mit Hilfe des Programmierens
  - Umgang mit Julia
  - Dass Mathematik sehr spannend und anwendungsbezogen sein kann.
  - Dass ich nicht Informatik studieren möchte
  - Einfache Mathematische Modellierung und Optimierung
  - Ein paar neue Verfahren in Mathe, ein wenig Programmieren und viel über Solartechnik
  - Mein Horizont wurde etwas erweitert in eine Richtung, die ich gewöhnlich nicht einschläge. Dies hilft mir mehr Wissen in dem Bereich zu verstehen.
  - Schulmathe findet tatsächlich Anwendungen. Es macht Spaß und es ist interessant wie weit man Sachen berechnen kann.
  - Mathe ist gar nicht immer so leicht
  - Programmieren
  - Mit Mathe ist Alles möglich.
- Thema II + III**
- Ich habe gelernt wie man besser mit Modellierungsaufgaben umgeht, wie z.B. Formeln aufzustellen um bestimmte Sachen zu ermitteln.
  - Modellierung, Reduzierung eines komplexen Sachzusammenhangs (Solarkraftwerk) zu einer (mehr oder weniger) simplen Modellvorstellung
  - Umgang mit einer neuen Programmiersprache, Anwendung mathematischer Formeln in der "Praxis".

- Ich habe ein Verständnis für die Entstehung solcher Projekte im echten Leben entwickelt, was 1. eine gute Abwechslung zum gewöhnlichen Unterricht ist und 2. zu einer guten Orientierung im Bezug auf die spätere Berufslaufbahn beigetragen hat.
- Die Spiegel sind gar nicht so leicht optimal einstellbar.
- Einen Plan zu erstellen kann hilfreich sein und nicht nur Zeit verschwenden.
- Optimierung kann immer weitergeführt werden, automatische Optimierung kann optimiert werden (Effizienz).
- Wie wichtig mathematische Modellierung bei Entwicklungsprozessen ist
- Ich habe verschiedene Optimierungsverfahren zum Optimieren einer Zielfunktion mit Nebenfunktionen kennengelernt. Außerdem habe ich aktuelle Probleme / Fragen der Forschung zu Solarkraftwerken erfahren, was ich ebenfalls sehr interessant fand.
- Das Algorithmen echt viel effizienter sein können, als Brute-Force
- Wie man durch die Modellierung alltägliche Fragestellungen nicht nur lösen, sondern auch optimieren kann.
- Wie kompliziert es eigentlich ist Techniken zu optimieren die zur uns allen bekannnten Energiewende führen sollen
- Vieles über Physik und auch über Winkel, auch wenn dies eher eine Auffrischung war
- Das Bruteforce wirklich nicht so gut ist, Trigonometrie üben
- Wie viele Feinheiten, die man am Anfang total unterschätzt beachtet werden müssen.
- Etwas über Solarkraftwerke und Optimierungsalgorithmen
- Ich habe gelernt, wie man Schritt für Schritt ein math. Problem löst, und auch optimiert. Zudem habe ich gelernt, was bei der Optimierung wichtig ist (4 Fragen) und es am Beispiel der Algorithmen sehen können.
- Eine Menge neuer Dinge über Trigonometrie, über Julia, etc. und habe nun eine bessere Vorstellung davon, wie reale Projekte durch mathematische Modellierung unterstützt werden.
- Dass ich für die Aufgaben besser ausgeschlafen sein sollte als ich es für die Schule bin. Die Aufgaben sind zwar toll, aber beanspruchen stets Konzentration.
- Komplexität hinter vielen Themen
- Ich habe heute nochmals weitere Erfahrungen in mathematischer Modellierung gesammelt, inhaltlich jedoch nichts neues erfahren, da die Grundsteine für dieses Projekt schon in den letzten Tagen gelegt wurden.
- Es ist für mich sehr schwer nach so einem kurzen Projekt meinen Lernzuwachs zu erkennen. Zentral in der Lösung war für mich klare Vereinfachungen zu wählen, an Stellen wo sie vertretbar sind
- Dass modellieren manchmal echt anspruchsvoll sein kann.
- Kommunikation ist der Schlüssel zum Sieg
- Dass man viel Logik durch ein wenig Mathe einsetzen kann. Meine 1. Variante hatte sehr viel Logik und war

sehr kompliziert, aber als ich die komplett gelöst und neu angefangen habe kam ich auf eine Idee, die sehr viel einfacher war, und nur ein wenig mehr Mathematik benötigte.

- Weitere Möglichkeiten des Implementierens eines Codes in Julia.
- Dass es gar nicht so schwer ist, ein eigenes Modell zu entwickeln und selbst anzupassen.
- erweitertes Denken
- Selbst Probleme mit mathematischer Modellierung zu lösen ohne von den ABs daran geführt zu werden.
- Vieles über Sinusfunktionen und ähnliches, vieles zum herleiten. Mit einem Touchpad des Computers leserlich schreiben
- Ich habe gelernt, wie man beim Modellieren eines mathematischen Problemes vorgeht und dass man es immer weiter verbessern kann.
- Das sowas zu Programmieren erstaunlich einfach ist und man des ruhig mal machen kann
- Ich habe gelernt wie viele Sachen man beim modellieren beachten muss
- Viele Kleinigkeiten müssen beachtet werden, auch Sonderfälle. Am Anfang viele Bedingungen festsetzen und dann nach und nach variieren ist leichter.
- Festigung und Fortführung des bisher erlangten Wissens
- Ich habe mich im Rahmen des Schattenschwurfs mit Geradengleichungen beschäftigt und da ein bisschen dazu gelernt. Außerdem habe ich meine Fähigkeiten in der Sprache Julia ausgebaut.
- Wie noch komplizierte vollständiges Modellieren sein kann.
- Es gibt viele Faktoren die optimiert werden können, die abhängig von anderen Parametern einen großen Einfluss haben können. Es ist leicht den Überblick darüber zu verlieren, was sich gegenseitig beeinflusst und warum.
- Wie man eigenständig Optimierungsprobleme angeht
- Wie viele verschiedene Modellierungs-Ansätze es gibt
- Formel für Energieabsorption der Luft auf Bodenhöhe
- Ich habe mit der mathematischen Modellierung eine wichtige Einsatzmöglichkeit der Mathematik kennengelernt, die ich vorher nicht kannte. Nun habe ich konkretere Vorstellungen von der Bedeutung der Mathematik für andere Bereiche.
- Die Prinzipien / Das Konzept von Mathematischer Modellierung, ein wenig zu Optimierung
- Wie toll Julia und Jupyter sind :D Um ehrlich zu sein habe ich heute durchaus einiges gelernt, nur war das eher bezogen auf die Techniken, Algorithmen, Optimierung von Parametern etc als die eigentliche mathematische Modellierung. Wobei das (für mich) natürlich auch nichts schlechtes ist. :)
- Ich habe vor allem mehr über das Programmieren mit Julia gelernt, aber auch mehr über die mathematische Modellierung und das Optimieren.
- Die Umsetzung erneuerbarer Energiegewinnungsverfahren finde ich sehr Interessant

- Wie ein Solarkraftwerk funktioniert und in Grundzügen aufgebaut ist. Außerdem habe ich gelernt, welche Probleme mit einem Solarkraftwerk einher gehen.
- Das man allgemein alles optimieren kann und dass modellieren durch Mathematik zur Lösung führen kann
- Bischen besser programmieren & simulieren
- Die Funktion eines Solarkraftwerks
- Ich konnte das erste Mal sehen, wie Informatik im Leben umgesetzt und gebraucht wird.
- Wie man beim mathematischen Modellieren vorgeht und wie wichtig es ist, ein großes Problem auf kleinere Überlegungen aufzuteilen und auch mit Skizzen zu arbeiten.
- Mathematische Modellierung, Programmierung, Optik
- Ich bin zum ersten Mal mit dem Thema der mathematischen Modellierung in Kontakt gekommen und habe mehr über Fresnel Kraftwerke und ihre Funktion gelernt.
- Ich habe gelernt, dass es sinnvoll ist sich zusammenzuschließen und mit Interessierten es besonders Spaß macht an einem Thema zu arbeiten. Außerdem habe ich viel im Bezug auf Studiengänge und Möglichkeiten für die Zukunft gelernt, was mir sehr weitergeholfen hat.
- Das fast alles mathematisch beschrieben werden kann und wie man dabei vorgehen kann.
- Das kleinschrittige Herangehen an ein komplexes Problem
- Mathematische Modulation wurde vertieft
- die Mathformeln kommen doch nicht nur im Schulbuch vor
- Ich habe das Programmieren mit Julia kennengelernt
- Neben neuen Informationen zum Thema, habe ich persönlich durch den Workshop das Interesse an der mathematischen Modellierung gewonnen.
- wie man ein problem analytisch angeht
- Was mathematisches Modellieren ist
- Kennenlernen von Solarkraftwerken, übersichtliche Darstellung der mathematischen Herangehensweise
- Optimierungsprobleme mathematisch angehen
- Ich habe einen besseren Einblick in Teile der Mathematik bekommen, die man so nicht in der Schule bekommt
- Strukturen von Programmiersprachen, Herangehensweise zur Lösung mathematischer Probleme
- Ich habe mathematisches modellieren nun besser verstanden
- Einführung in das mathematische Modellieren

**Was fandest du an dem Teil des Workshops zur Optimierung besonders interessant?**

**Thema I:**

- Wo man den optimalen wert für "h" bestimmen musste.
- Das kleinste Veränderungen eines Parameters große Unterschiede im Ergebnis verursacht haben.
- Alles
- die atmosphärische Abschwächung des Lichts (AB 6)
- Wie man Schritt für Schritt von einem simplen Problem zu einer komplexen Lösung kommt.
- Kann ich nicht sagen, da ich noch nicht so weit kam
- Zu sehen, wie man sich Schritt für Schritt vorantastet, um alle wichtigen Faktoren zu entdecken.
- Das Feststellen der Fehler beim Arbeitsblatt 7.
- zu sehen wie sich die Ergebnisse ändern, wenn man die ganzen Faktoren mit einbezieht (z.B. die Leistung mit und ohne atmosphärische Effekte)
- Dinge selber zu überlegen, die nicht nur rein Matheformel A Matheformel B sind, sondern sich etwas mehr mit logischem Denken erschließen muss.
- Sehr interessant fand ich, was für einen großen unterschied Variablen wie die Schrittweite machen, auch bei minimaler Veränderung.
- Ich fand besonders interessant, dass atmosphärische Einflüsse die Lichtstrahlen vergleichsweise stark beeinflussen können.
- Logisches Denken
- Nichts wirklich

**Thema I + III**

- Das eigene Programmieren und nicht mehr stumpf nach Anleitung zu arbeiten
- Anwendung der mathematischen Methode in einem realistischen Problem
- Die freie Arbeit am letzten Tag
- Ich fand besonders interessant, dass man sehr offen an Optimierungsproblemen arbeiten kann
- Ich fands gut, dass wir am Ende mehr Freiheiten hatte (auch weil ich gerne programmiere). Gut fand ich auch, dass ich einiges gelernt habe, vor allem Sachen die ich noch nicht in der Schule hatte aber noch haben werde (die kann ich jetzt schon)
- Dass man das Problem immer um weitere Faktoren erweitern konnte
- Die Arbeit in Teams und in Gruppen
- die Mathematik dahinter

- Inhalt (Solarkraftwerk-Funktionsweise, Solarturmkratwerk), eigenes Umsetzen im Code
- Es handelt sich um ein aktuelles Forschungsthema
- Das es viele verschiedene Möglichkeiten gab. Außerdem das wir freier Arbeiten konnten.
- Die Einführungen in die verschiedenen Themen sowie in MM
- Ich fand es sehr interessant, dass die optimale Lösung sich veränderte, als wir verschiedene Aspekte bedenkt haben
- Verbindung von mathematischen Überlegungen und Programmieren, reale Systeme auf Modelle herunterbrechen
- besonders interessant fand ich die verschiedenen Problemstellungen sowie differenzierten Ansichten der Gruppen zur Lösung eines Problems
- selber, freier Programmieren
- besonders der Einfluss von Schatten zu berechnen
- das Thema im Zusammenhang mit den nötigen Rechnungen
- interessanter Einstieg (klar strukturiert), guter erster Eindruck

**Thema II:**

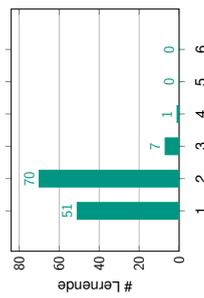
- Mal wieder mit alten Themen auseinandergesetzt
- Wie viel man bei der Optimierung zu beachten hat
- Ein neues Lösungsverfahren, das ich noch nicht kannte.
- Die Theorie am Anfang
- Die Arbeitsphasen
- Die Annäherung an das Problem durch Vereinfachen und diese Vereinfachungen nach und nach zu beseitigen
- Knobelaufgaben
- Das es auch mit einfachen Laptops in diesem Teilbereich geht und man keine Superrechner braucht
- Das Umsetzen, Rechnerische.

**Thema II + III**

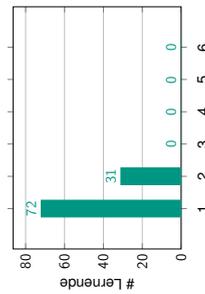
- Optimierung ging ins Unrealistische hinein -> Optimierung mit Nebenbedingungen
- Die Erstellung des eigenen Optimierungsblattes. Allerdings weiß man da nicht so recht, was genau man optimieren soll. Eine so offene Aufgabenstellung ist schön, aber man könnte zusätzlich vielleicht noch ein paar Beispiele inklusive Datensätzen geben.
- Was die Spiegelposition für einen Einfluss auf die Gesamtenergie ausmacht.

- Wie es auf das echte Leben übertragbar ist.
- Dass man eine neue Technik kennengelernt hat und dass es mehrere Möglichkeiten gab.
- Die freien Projekte
- Den Schattenschwurf mathematisch zu beschreiben
- Die Gruppenarbeit am Ende.
- Die verschiedenen Optimierungsalgorithmen und deren Vor- und Nachteile
- es gibt einige Hindernisse/ Probleme, die zu beachten sind
- Die Analyse und der Vergleich verschiedener Algorithmen.
- freie Arbeiten
- alles
- War halt einfach alles interessant, ohne, dass etwas heraussticht
- Gegensätzliche Optima
- Ich fand die Verbindung zu einem aktuellen Thema sehr gut
- Arbeitsblätter zur Bearbeitung im Zweierteam
- Modellierung von Beschattungseffekten
- Das Thema. Es war neu und interessant, besonders durch das arbeiten/modellieren.

**5 Abschließende Bewertung**  
**Ich gebe dem Workshop die Schulnote.**



**Ich gebe den Betreuer:innen die Schulnote.**



**Abschließender persönlicher Kommentar (Lob, Kritik, Verbesserungsvorschläge, ...)**

**Thema I**

- Tipp: nicht nur Text.

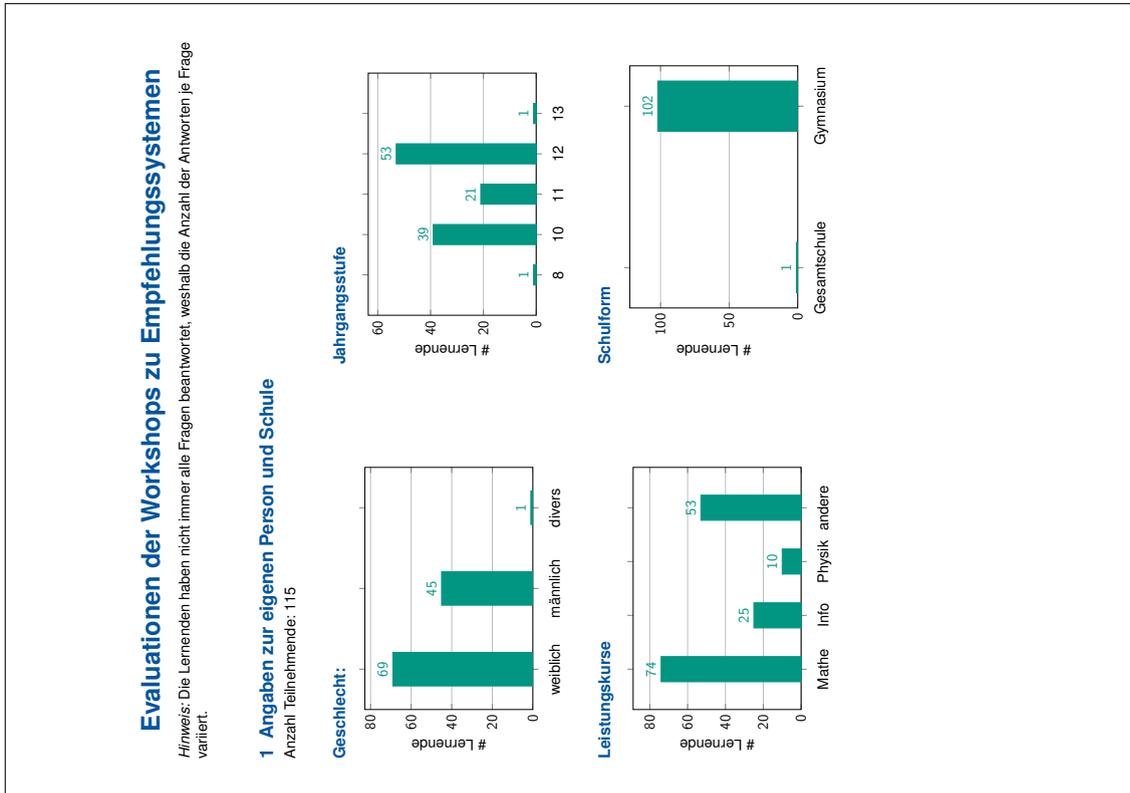
- Algorithmen zur Automatisierung der Optimierung analysieren + implementieren
- Alles Inhalte waren interessant.
- Die Entwicklung eines eigenen Algorithmus.
- Die verschiedenen Algorithmen
- Zu sehen, wie viele unterschiedliche Verfahren es gibt und dass ein Großteil nicht alle Probleme die entstehen könnten auffangen kann. Dementsprechend fand ich es interessant zu sehen, wie schwierig die Optimierungsfrage häufig sein kann und wie man sie dennoch beantworten kann.
- Wie man selbst mit veränderten Eingaben Annäherungen machen konnte
- Ich fand es einfach interessant, mehr darüber zu erfahren.
- Algorithmen
- Die Vor- und Nachteile verschiedener Techniken.
- Die verschiedenen Algorithmen und Herangehensweisen, um ein Problem zu lösen.
- Ich fand es interessant die verschiedenen Algorithmen zu sehen und sie anhand der 4 Fragen zu vergleichen.
- Den Teil, bei dem es darum ging, den Winkel der Spiegel bei einer beweglichen x-Position zu errechnen.
- Nichts. Alles.
- Das eigenständige Arbeiten an Optimierungsmöglichkeiten
- Die vielen Faktoren
- das aufstellen der mathematischen Modelle und auch die Verbesserung / Erweiterung davon
- die Diagramme und die Veränderungen, die wir darin vornehmen können
- zu sehen, wie durch mathematische Vorgehensweisen eine komplexe Optimierung mit auch mehreren Ideen gelöst werden kann.
- Den Vergleich der verschiedenen Methoden
- der Workshop war für mich insgesamt sehr interessant
- Die Vielseitigkeit an Möglichkeiten, welche offen stehen und dass bei jeder neuen Idee direkt wieder Probleme und Grenzen aufkommen.
- Wie viele verschiedene Arten der Optimierung gibt
- Praktische Anwendung eines Optimierungsproblems
- Eigene Programmierung
- komplexes problem
- Selber Ansätze finden

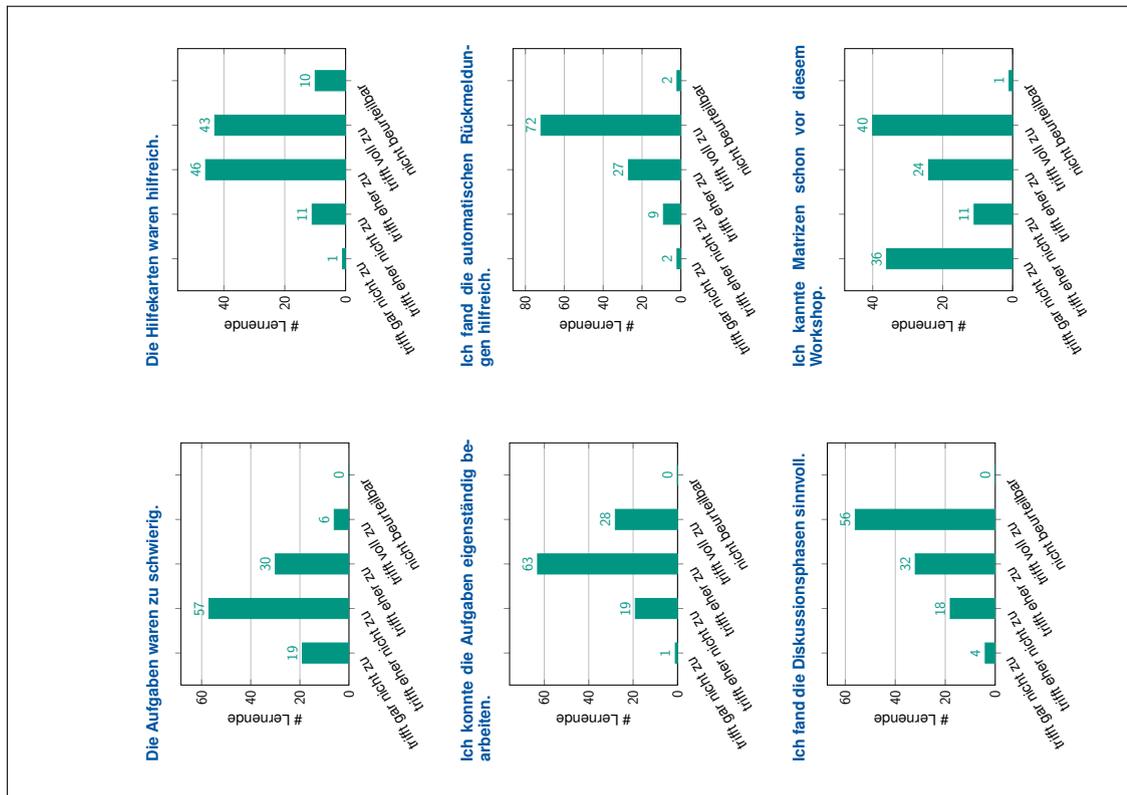
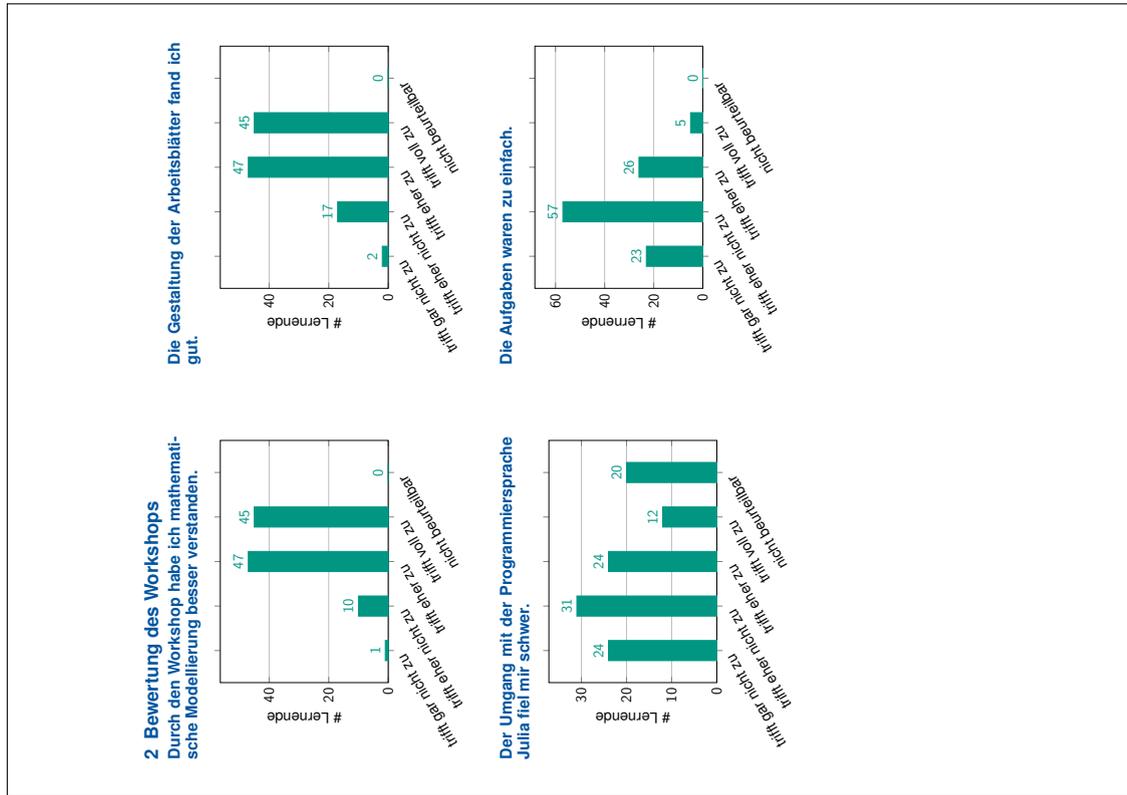
<ul style="list-style-type: none"> <li>• Meiner Meinung nach, finde ich diese Idee eines Online- Kurses sehr gut ausgeführt worden</li> <li>• und ich möchte auch die Betreuer sehr doll loben, da sie wirklich immer sehr gut erklären konnten.</li> <li>• Mir hat das Projekt viel Spaß gemacht, aber ich fand die ständigen Besprechungen nervig, weshalb ich diese nur für Schüler, die Fragen haben oder nicht weiterkommen machen würde. Man könnte Sinus und Kosinus für Schüler in Klasse 9 besser erklären</li> <li>• Es hat mir sehr viel Spaß gemacht, ich habe sehr viel gelernt ohne, dass es eine strenge Unterrichtsstruktur gab. Es war sehr interessant und ich würde gerne wieder an solch einem Projekt teilnehmen.</li> <li>• War ein sehr interessantes Projekt, die Themen wurden gut erklärt und es hat Spaß gemacht.</li> <li>• Sehr gut, es gibt nix zu sagen :-)</li> <li>• Das Programm hat mir viel Spaß gemacht, da man nicht unter einem Notendruck stand. Im Nachhinein bin ich sehr froh, das Angebot von der Schule empfohlen bekommen zu haben.</li> <li>• Sehr übersichtlicher und verständlicher Workshop.</li> <li>• die Worksheets waren gut strukturiert und man wusste immer was zu tun war. Es wäre an manchen Stellen hilfreich gewesen, wenn man z.B. Formeln die auf einem anderen Worksheet erarbeitet wurden nochmal anzeigt wenn sie erneut gebraucht werden. Sonst muss man ständig von Worksheet zu Worksheet wechseln. Alles in allem ein gutes Nachmittagsprojekt.</li> </ul> <p><b>Thema I + III</b></p> <ul style="list-style-type: none"> <li>• Wenn Gruppen mehrtägig da sind, bitte nicht jeden Tag einfach ein anderes Modellierungsproblem vorgeben. Lieber ein großes Thema für die ganze Woche anbieten</li> <li>• Bitte Matlab anstelle von Julia nutzen!!!</li> <li>• Hat sehr viel Spaß gemacht, das uns so viele Freiheiten gelassen wurden.</li> <li>• Es war gut, dass die Betreuer (scheinbar) gut über das Thema informiert waren</li> <li>• Ich finde das Thema sehr schön gewählt und auch die Aufgabenstellungen gut gewählt. Allerdings manchmal etwas engmaschig, wobei die offene Aufgabe am Ende sehr schön war.</li> <li>• Besser im Vorhinein auf das Vorwissen einstellen und anpassen</li> </ul> <p><b>Thema II + III</b></p> <ul style="list-style-type: none"> <li>• Gut erklärtes und geleitetes Projekt</li> <li>• (Für die Klassenstufe) verständliche Zusammenhänge bzw. Inhalte</li> <li>• Alles in allem bin ich dem Projekt sehr positiv gestimmt. Nur fehlte mir teilweise etwas die Übersicht.</li> <li>• Vielen Dank für das Projekt, wie gesagt es war eine tolle Abwechslung zum normalen Unterricht und hat mir viel Spaß gemacht :-). Ich würde gerne wieder an solch einem Projekt teilnehmen M</li> <li>• Sehr interessant und lehrreich, auch wenn es auch anstrengend war (kommt möglicherweise aber auch durch die 3 Tage intensives Denken davor zustande)</li> <li>• Die Zeiten sind eigentlich zu großzügig bemessen. Anstatt der zu vielen Zeit für Abs lieber mehr Zeit für</li> </ul>	<p>die Diskussionen einplanen. Dann ist man auch pünktlich.</p> <ul style="list-style-type: none"> <li>• Mit hat der Kurs sehr viel Spaß gebracht :-)</li> <li>• Gutes Camp</li> <li>• N6 Heute fällt mir nichts ein</li> <li>• Bei Julia gab es ein paar wenige Programmier Fehler, wobei man von den Betreuer da super durch geleitet wurde.</li> <li>• Teilweise hätte ich mir etwas mehr Zeit für die Aufgaben gewünscht</li> <li>• Alles sehr interessant und eine gute Mischung aus Theorie und Praxis, Nachdenken und Implementieren, usw</li> <li>• Das Solarprojekt als Ganzes hat mir viel Spaß gemacht, vor allem das freie Arbeiten am Schluss. Es gab ungefähr die richtige Mischung an gegebenen Informationen und solchen, die man erst herleiten musste. Dass sich dieses Verhältnis immer weiter angepasst hat, war angenehm.</li> <li>• Es hat mir viel Spaß gemacht. Noch schöner wäre es wahrscheinlich offline gewesen.</li> <li>• Ich fand die Auseinandersetzung mit der Problematik sehr interessant und werde den Workshop in positiver Erinnerung behalten.</li> <li>• Alles in allem hat es mir Spaß gemacht beim Workshop teilzunehmen :-)</li> <li>• Eine sehr nette Betreuerin die alles super erklärt und jedem gut weitergeholfen hat :)</li> <li>• Ich bin leider nicht so Technik interessiert, weswegen ich mit solchen Arbeitsblättern nicht so viel anfangen kann dennoch hat mir das Projekt Spaß gemacht, auch wenn ich vieles nicht so schnell verstanden habe.</li> <li>• Sehr gut organisiert. Habe mich immer Sicher gefühlt bei dem was wir tun</li> <li>• Ich fand den Workshop sehr gut strukturiert, habe viel gelernt und hatte Spaß. Vielen Dank.</li> <li>• Mir hat das MINT-Camp sehr gut gefallen, obwohl es nur online stattgefunden hat. Die Arbeitsblätter waren sehr schön gestaltet und super strukturiert. Der ganze Workshop war wirklich sehr interessant!</li> <li>• Hat sehr viel Spaß gemacht, auch wenn es manchmal ziemlich kompliziert war :-)</li> <li>• Ich fand es sehr schön das wir immer in Gruppen gearbeitet haben und das wir bei Problemen auch immer einen Ansprechpartner hatten.</li> <li>• Ich hatte eine sehr schöne Zeit und habe einiges dazugelernt, wofür ich mich bedanken möchte.</li> <li>• Mehr Zeit zum freien Arbeiten wäre schön</li> <li>• Es hat mir sehr gut gefallen und ich habe viel gelernt, aber teilweise wären Musterlösungen gut gewesen</li> <li>• Ich möchte ein großes Lob gegenüber der Veranstalter/-innen aussprechen und würde mich sehr freuen, wenn beim nächsten Mal eine Podiumsdiskussion integriert werden könnte.</li> <li>• Es kriegten zwar beide eine 1, allerdings ist die 1 der Betreuer*innen deutlich stärker als die, des Workshops.</li> </ul>
---	--

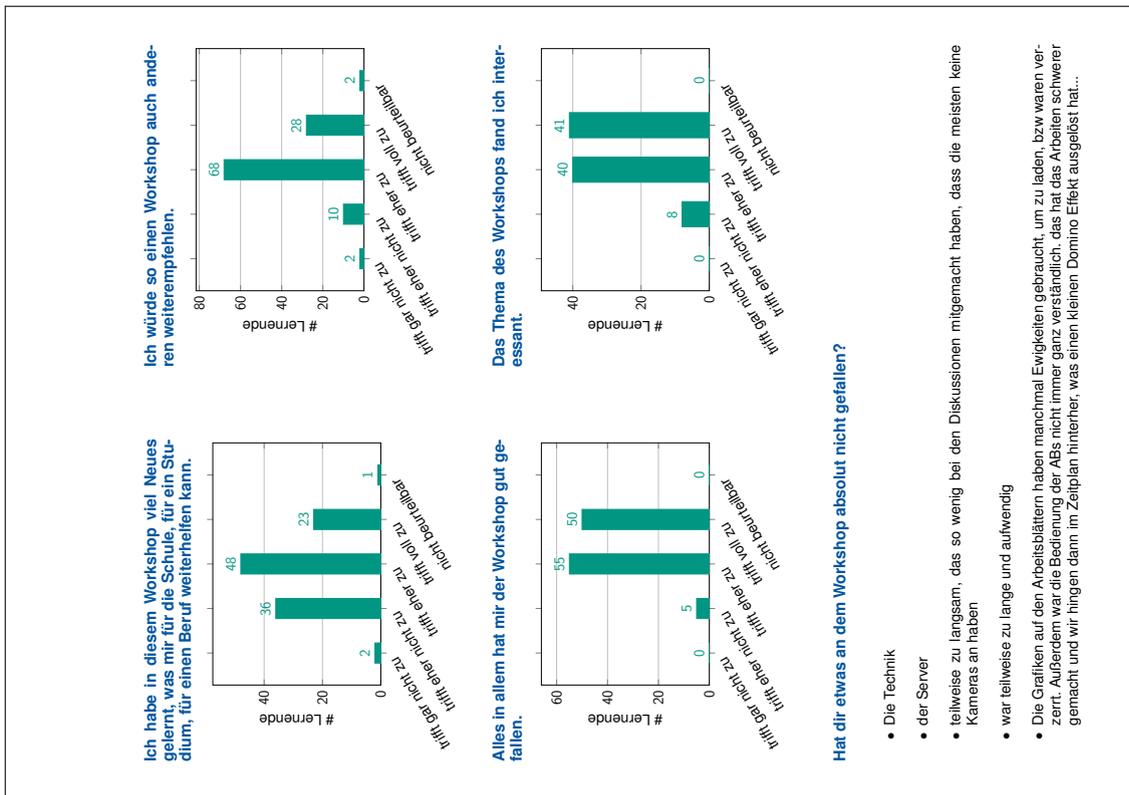
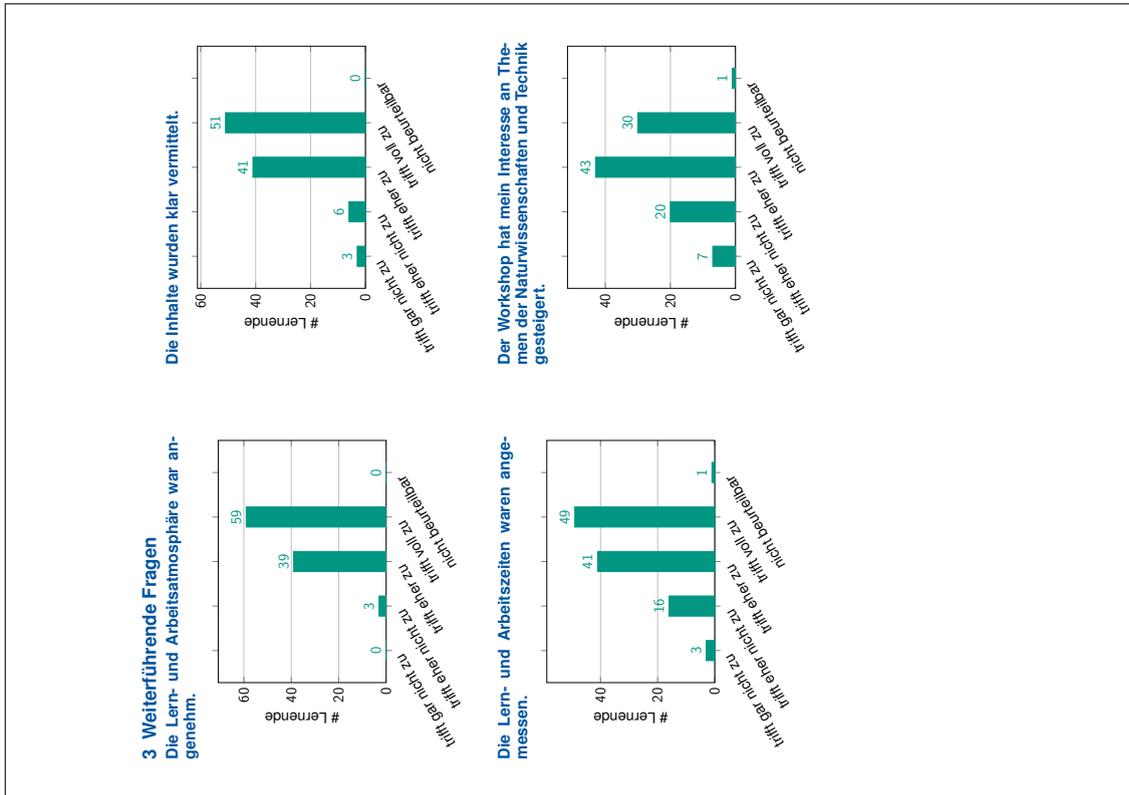
- Ich fand es sehr gut das auch auf Fragen geachtet wurden und diese sehr gut erklärt wurden ohne die Lösung vorzusagen
- Insgesamt finde ich die Idee des CAMMP Workshops sehr gut, allerdings ist es für Schüler/-innen mit Schulprogrammerkennissen schwierig sich schnell in einer Programmiersprache zurechtzufinden. Aus meiner Sicht hätten noch mehr Vorträge von Experten aus der aktuellen Forschung und Entwicklung in den Workshop eingebracht werden können, um noch mehr über die Anwendung in der Praxis zu erfahren.

## G.2 Evaluationsergebnisse zum Thema Empfehlungssysteme

### G.2.1 Angleitete Modellierungsprojekte







- Manche Aufgaben waren zu einfach, andere einen Tick zu schwer. Bessere Aufteilung der einzelnen Schritte hätte vielleicht geholfen. Nicht schlimm insgesamt, nur sollten Sie ein Auge darauf haben.
- Die Bearbeitungszeit in der Gruppe hat manchmal nicht ganz ausgereicht. Va. bei AB 4 (Langversion) habe ich mich etwas unter Druck gesetzt und überfordert gefühlt.
- Das der Workshop online war, hat ein bisschen gestört, da ich mich online nicht so gut konzentrieren kann. (Dadurch das es wegen Corona so ist, ist es nicht so schlimm.)
- Ein kleiner Hinweis, dass im Code statt Kommata Punkte genutzt werden müssen
- Zu kurze Zeit [2x von verschiedenen Teilnehmer:innen angegeben]
- Ja, Julia

**Hat dir etwas an dem Workshop besonders gut gefallen?**

- Anwendung am echten Datensatz
- Das ich neue Sachen gelernt habe, Ableiten von einer Funktion mit 2 Variablen
- Die Partnerarbeiten mit dem direkten Feedback durch die Arbeitsblätter
- dass Inhalte noch einmal wiederholt wurden
- die umfangreichen Erklärungen und der Bezug zu aktuellen Themengebieten. Dadurch wurde der gesamte Workshop spannender.
- Die Problemstellung ist na an der Realität
- die Organisator:innen waren sehr freundlich. Das zusammentragen der Ergebnisse und die Diskussionen waren interessant. ;)
- die Gruppenarbeit [6 x von verschiedenen Teilnehmer:innen angegeben]
- das eigenständige Erarbeiten und die Diskussionen
- die breakout rooms
- Dass alle sehr hilfsbereit sind und uns bei Problemen immer geholfen haben und natürlich generell das Thema mit den Bewertungen und zu sehen wie viel Arbeit und Daten da dahinter steckt
- gemeinsames Besprechen
- Das aktuelle Thema /Netflix
- Die Abwechslung von Gruppenarbeiten und Besprechungen im Plenum
- der Aufbau der Arbeitsblätter
- Die Betreuer\*innen waren sehr hilfsbereit
- Die Strukturierung mit den Code-Feldern war sehr gelungen und brachte Dynamik

- Klein-Gruppen
- Die Mischung aus selber arbeiten und gemeinsam besprechen
- Dass man selber Aufgaben machen konnte
- Matrizen rechnen
- die schrittweise Einführung in ein Thema, das nicht allen Teilnehmern bekannt ist sowie Plenumsdiskussionen
- Ich war positiv überrascht von der guten Organisation und Strukturierung.
- Die Offenheit der Betreuer:innen für Fragen, Unverständnis von Zusammenhängen.
- Das selbstständige Erarbeiten der Aufgaben und der hohe Praxisanteil.
- Wir haben sehr viel interessantes in sehr kurzer Zeit gelernt
- die Abwechslung von Plenum und Gruppenphasen. So ging es immer weiter, auch wenn man mal hängengeblieben ist. Die Einführung und die Überleitungen im Plenum fand ich sehr hilfreich.
- Dass Sie sich so viel Mühe gegeben haben und sehr viel Arbeit reingesteckt haben. Vielen Dank dafür.
- Die Arbeitsblätter und besonders das Verändern der Parameter im Code, sodass man deren Einfluss nachvollziehen konnte.
- die Arbeitsblätter
- alles
- Gruppeneinteilung konnte man sich aussuchen = macht mehr Spaß
- Ein Fallbeispiel zur verbildlichung des Wissens
- Neue Methoden und Verfahren kennengelernt
- Alles
- Die Alltagsnähe
- Die Diskussionsmöglichkeiten
- Mir hat es besonders gut gefallen, dass wir quasi einen Einblick in das Empfehlungssystem von Netflix bekommen haben und vorallem das mithilfe der Arbeitsblätter selber herausgearbeitet haben. Ich finde es ebenfalls sehr schön mit meine Kollegen zu arbeiten und neue Menschen kennenzulernen
- Die informative und reaktionelle Kontrolle der Antworten
- Die Benutzerfreundliche Oberfläche (Julia)
- Coden
- Das man viele Dinge selber durchführen konnte.
- Das Thema war interessant

<ul style="list-style-type: none"> <li>• Eigenes Erproben</li> <li>• Vielfältigkeit</li> <li>• Die Diskussionsrunden, da man viel vergleichen konnte</li> <li>• Das Programm hat gut funktioniert</li> <li>• das Thema [5 x von verschiedenen Teilnehmer:innen angegeben]</li> <li>• Nice</li> <li>• mit Julia zu arbeiten</li> <li>• Alles war ehrgeizig und es hatte viel mit teamwork zu tun</li> </ul>	<p><b>Hättest du gerne noch etwas anderes gesehen oder erfahren?</b></p> <ul style="list-style-type: none"> <li>• Die Auflösung, wie die Gewinner gewonnen haben; wir sind aber noch nicht ganz fertig, also was noch nicht ist kann noch werden :)</li> <li>• Andere Anwendungen für Matrizen und mathematische Modellierung, mit mehr Zeit vielleicht sogar ein paar nicht zum Hauptthema passende Probleme zum Anschauen, zumindest, wenn nicht Lösen.</li> <li>• weitere Modellierungsbeispiele</li> <li>• Ob es in der mathematischen Modellierung auch Bereiche gibt, in denen großer Schwerpunkt auf der Differential- / Integralrechnung liegt.</li> <li>• Am Ende die Lösung Programmieren..</li> <li>• ob es auch einen anderen, ähnlich erfolgreichen Ansatz gibt. Ob Netflix heute immer noch dieses System nutzt zum Beispiel.</li> <li>• Vielleicht noch andere Themen run um Netflix. Aber ist eigentlich nicht zwingend notwendig. Man kann vielleicht auch die Gruppen durchmischen, um mit anderen Kontakte knüpfen zu können.</li> <li>• Alternative Optimierungsverfahren (z.B. ein sehr kurzer Exkurs, was für Alternativen es gegeben hätte)</li> <li>• Ich hätte noch sehr gerne gesehen, wie das Empfehlungssystem in der Praxis dann tatsächlich umgesetzt wird. Hier reicht aber denke ich auch einfach ein Video oder eine PDF-Datei</li> <li>• Andere mathematischen Modellierungen</li> <li>• Zur Netflix-Challenge und der Lösung dazu</li> </ul> <p><b>4 Lernzuwachs</b>  <b>Was hast du für dich persönlich durch die Teilnahme am Workshop gelernt?</b></p> <ul style="list-style-type: none"> <li>• Die Theorie hinter mathematischer Modellierung.</li> </ul>
--	--

<ul style="list-style-type: none"> <li>• Ich habe einen interessanten Einblick erhalten wie aus vielen Daten weitere Daten vorhergesagt werden können</li> <li>• siehe letzte Frage</li> <li>• Wie man mit Matrizen am besten umgeht.</li> <li>• Es war für mich gut mal den Bezug von der Mathematik mit wirklichen Problemen zu sehen.</li> <li>• Das Mathematik sehr vielfältig ist und das man dadurch sehr viel neues lernen kann und vieles gut erklären kann.</li> <li>• Den Umgang mit Matrizen. [3x von verschiedenen Teilnehmer:innen angegeben]</li> <li>• etwas besser wie matrizen funktionieren</li> <li>• Wie große Datenmengen mit Matrizen analysiert werden.</li> <li>• Ich habe gelernt, dass man sein Wissen auf interessante und abwechslungsreiche Weise in Zusammenarbeit mit anderen Menschen und schulunabhängigen Themen sehr spannend erweitern kann</li> <li>• Ich habe das Modell, das hinter den Vorschlägen für Filme steckt besser nachvollziehen und verstehen können.</li> <li>• Dass Matrizen auch Anwendung in alltäglichen Dingen finden und dort eine wichtige Rolle spielen. Dass die Grundlagen des maschinellen Lernens relativ gut zu verstehen sind und sich mit Hilfe von vereinfachten Beispielen erklären lassen.</li> <li>• Das dies nicht nur bei Netflix so ist sondern auch bei vielen anderen Plattformen genutzt werden</li> <li>• Das das Internet viel komplexer ist als gedacht</li> <li>• Wo überall maschinelles lernen vorhanden ist und wie es in groben Zügen funktioniert</li> <li>• Matrizen besser zu verstehen</li> <li>• Umgang mit Matrizen, Wege, Probleme zu modellieren</li> <li>• Wie man mit solche Probleme umgeht</li> <li>• Matrizenrechnung, Anwendungsmöglichkeit(en) für Mathematik</li> <li>• Mathematische Modellierung als interessantes Thema</li> <li>• Matrizen, Optimierungsverfahren</li> <li>• Ich habe gelernt, kleinschrittig und logisch an große mathematische Probleme heranzugehen sowie das Rechnen mit Matrizen und allgemein den Umgang mit großen Datensätzen.</li> <li>• Die Möglichkeiten mathematischer Problemlösung; Umgang mit Matrizen</li> <li>• Matrizenmultiplikation</li> <li>• Dass Netflix verdeckt sehr viel über mich und mein Nutzerverhalten weiß.</li> <li>• Neu Bereiche und Anwendungsmöglichkeiten der Mathematik (Auch die Mathematik die man dazu nutzt Ma-</li> </ul>	
---	--

trizen, ...)

- was Matrizen sind
- Matrizen, Matrix, Kettenregel, ...
- Was eine Matrix ist, was man damit macht und wie man Funktionen partiell ableitet.
- Kettenregel für Ableitungen, Maß der Fehlerquadrate, Matrizenrechnen
- Umgang mit Matrizen, Vorgehen bei Erstellung von Bewertungsempfehlungen /Problemfragen
- Ich habe gelernt wie man mathematische Probleme auch anhand von Programmiersprachen lösen kann. Außerdem habe ich gelernt, was eigentlich für riesige Daten hinter Netflix stehen. Man hat einfach einmal tiefer in das System Netflix geblickt.
- Ja, ich habe vieles neues gelernt.
- Partielle Ableitung, Optimierungsverfahren, Maschinelles Lernen
- Alles kann mit gewisser Präzision mathematisch modelliert werden.
- Matrizen besser verstanden
- Mathe
- Das sehr viel von den Bewertungen von Filmen abhängt
- Dass Mathematik wichtiger ist, als ich anfangs annahm.
- ja
- Vieles [wurde 3x von unterschiedlichen Teilnehmer:innen genannt]
- Wie das mit den Empfehlungen funktioniert
- wie das Empfehlungssystem funktioniert
- Etwas über Netflix und wie man etwas empfohlen bekommt
- Was Netflix mit Mathe zu tun hat
- Matrix
- Wie solche Empfehlungen funktionieren
- Dass Mathe in alltäglichen Dingen wie sogar Netflix enthalten ist
- Matrix und so
- Wie ein Bürojob aussehen würde
- Wie Empfehlungen in Streaming-Websites zB funktionieren
- Worum geht es bei mathematischer Modellierung; Was kann ich von einem Studium in diesem Feld erwarten

- Mathematische Modellierung
- Wie Mathematik im Alltag nutzbar ist
- Mathematik im realen Leben verwenden
- wie man Matrizen in die realen Welt benutzen kann
- Matrizen, Vorgehensweise um ein Problem zu lösen, wie man Matrizen einsetzen kann
- Die Möglichkeit, dass aufgrund vieler Datensätze das Menschliche Verhalten/Vorfälle vorhergesagt werden können
- Ich habe mein mathematisches Verständnis in Hinblick auf Lineare Algebra besonders gut verbessert
- Modellieren
- Algorithmus von Netflix
- Wie Empfehlungssysteme funktionieren
- Mathe spielt in vielen Bereichen im Alltag eine wichtige Rolle.
- Die Funktionsweise von Empfehlungssystemen
- Nichts neues
- Wie kompliziert manche Themen sind, die natürlich erscheinen
- Empfehlungssystem-wie entsteht sowas
- Wissen über Matrizen aufgefrischt
- Komplexität von Modellierungsprozessen
- Wie Empfehlungssysteme mathematisch funktionieren, Basic-Umgang mit Matrizen
- Was sind Matrizen.
- Sind nicht immer spannend
- Julia kennt lernen
- Wie Empfehlungen entstehen
- Wie man Empfehlungen auf Netflix macht
- Was Matrizen sind.
- Teamwork
- Wissen über Matrizen aufgefrischt

**Hast du im Workshop irgendetwas gar nicht verstanden?**

- Nein / Nö / Ne [26x von verschiedenen Teilnehmer:innen angegeben]
- Nein, da wir immer Fragen stellen konnten und Hilfe bekommen haben.
- Am Anfang waren einige Dinge unverständlich, doch durch eine genauere Erklärung wurden die meisten Aufgaben klar.
- Nein es war mir alles klar, zwar nicht immer sofort aber durch die Besprechungen dann.
- Nein, ich müsste nur noch einmal alles gründlich durch arbeiten, damit ich es auch korrekt wieder geben könnte.
- Da der letzte Teil (maschinelles Lernen) etwas knapp beschrieben war, bin ich hier nicht sicher.
- Dreidimensionale Funktionen waren etwas komplex.
- Wenn ja dann habe ich dass gar nicht verstanden.
- Welche Bedeutung hat der Parameter Lambda?
- nur ein paar Sachen zwischen drin nicht, ich weiß aber gerade nicht mehr was
- Eigentlich nicht.
- Manchmal schon, jedoch hat es sich meistens im Plenum geklärt.
- Es war alles, wenn auch erst nach etwas Zeit & Überlegen, verständlich
- Eher nicht.
- Durch die Diskussionen wurden einige Themen verständlicher.
- Sehr schön
- Die Aufgaben mit der matrix
- Das oben genannte.
- Doch
- das maschinelle lernen viel mir doch schwerer als gedacht
- Matrizen oder so war schwer
- nein im großem ganzen alles verstanden, das eine mehr das andere weniger
- Ja [2x von verschiedenen Teilnehmer:innen angegeben]
- Kleinigkeiten, aber das ist normal, denke ich
- ja blatt 4
- Die Herleitung mancher Formeln.

**5 Abschließende Bewertung**  
**Ich gebe dem Workshop die Schulnote.**

Grade	# Lernende
1	21
2	61
3	18
4	1
5	0
6	0

**Ich gebe den Betreuer:innen die Schulnote.**

Grade	# Lernende
1	33
2	62
3	4
4	2
5	0
6	0

**Abschließender persönlicher Kommentar (Lob, Kritik, Verbesserungsvorschläge, ...)**

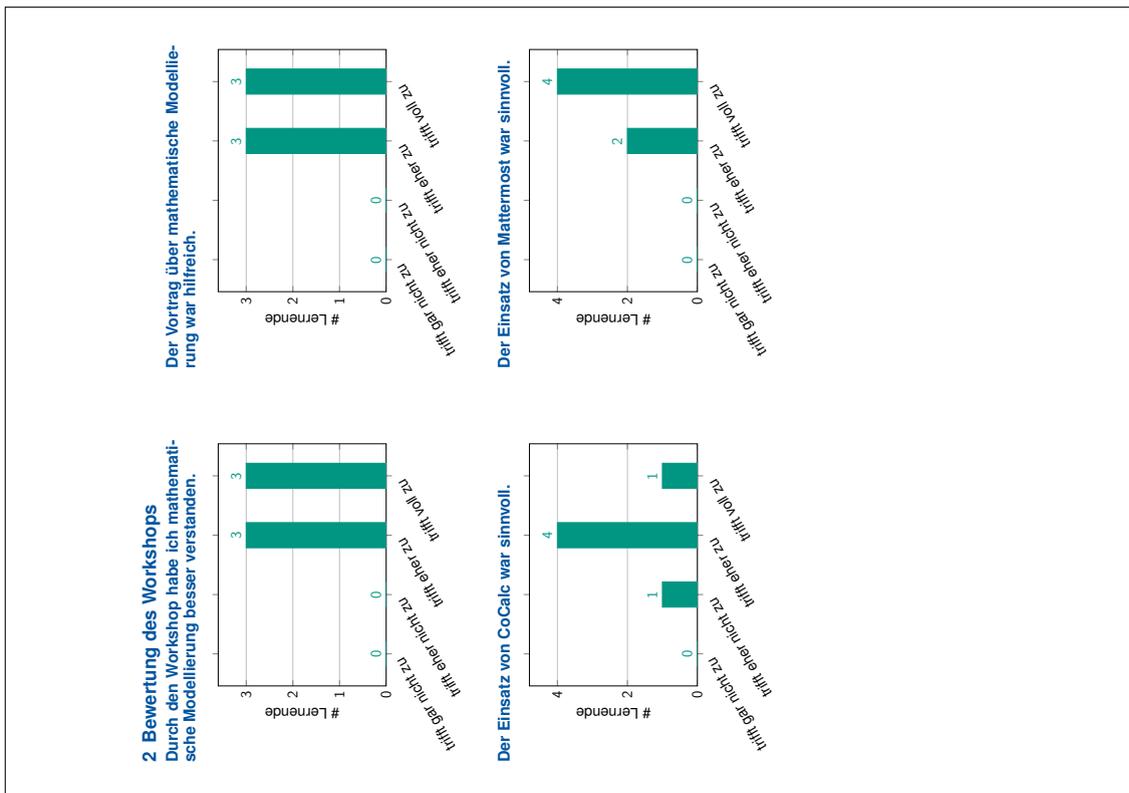
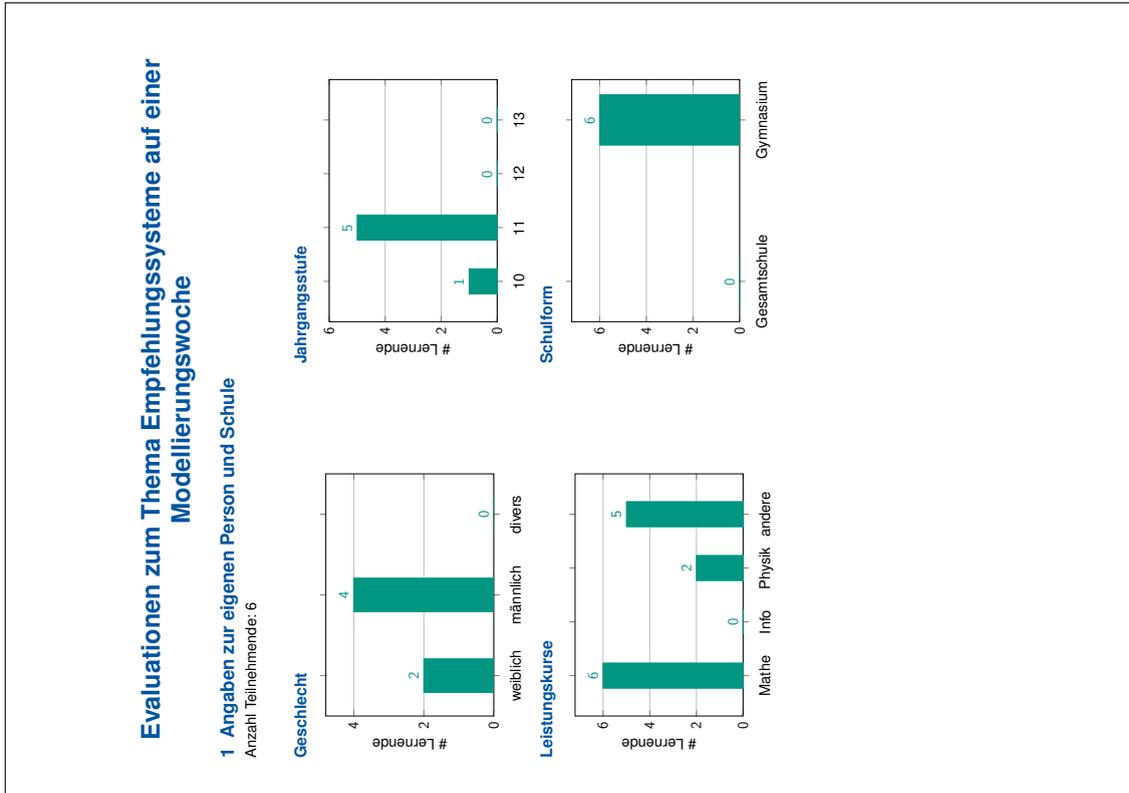
- Leute mehr animieren Kamera an zu machen und zu reden bei Diskussionen, und teilweise schneller voran kommen
- ;)
- Alles war toll und sehr spannend. Super gemacht.
- Sehr guter Workshop.
- Man muss bei der Umfrage viel zu viel schreiben. Aber sonst war alles super
- Ich fand's cool, weiter so ;)
- Ich fand die Arbeitsblätter sehr interessant und gut gestaltet. Die Gruppenarbeit gestaltete sich interessant und abwechslungsreich. Alles in allem hat es mir sehr viel Spaß gemacht und ich würde auch wieder daran teilnehmen.
- Der Workshop war sehr interessant gestaltet und hat auch Spaß gemacht. Ich war überrascht wie gut das Programm auch von Zuhause aus funktioniert hat.
- Danke für den schönen Workshop und die Zeit. Ich fand es wurde alles immer gut erklärt und ich habe alles verstanden. Manchmal waren die Aufgaben recht schwer, aber mit den Hilfekarten war alles machbar.
- Ich fand den Workshop insgesamt sehr gelungen, jedoch fand ich es zeitlich teilweise sehr knapp und ein bisschen stressig, was aber wahrscheinlich auch ein wenig der Online Situation geschuldet war. Darum sollte man für den Online Workshop vielleicht etwas mehr Zeit einplanen, aber ich fand die Durchführung

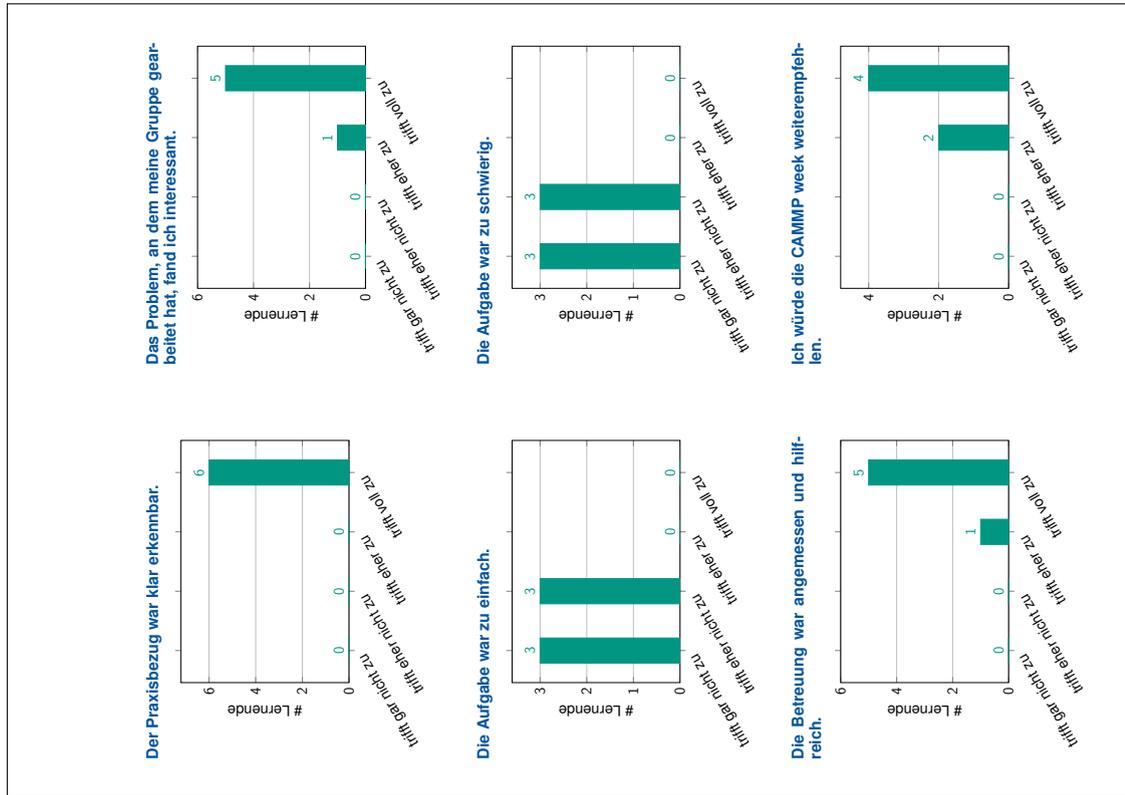
insgesamt sehr angenehm.

- Sehr gut, es war mir nur teilweise zu viel PC-Arbeit. Vor allem freitags nachmittags.
- Es war sehr gut, jedoch waren manchmal die Einführungsphasen etwas länger und man hätte sich gewünscht schneller zum Arbeiten zu kommen (Manches hat sich dort auch wiederholt)
- Das spannendste fand ich in kleinen Gruppen die Abs. auszufüllen, weil man durch die Website eigentlich ganz gut durch das Thema kam.
- Mir hat der Workshop gut gefallen :
- Ich fand es sehr interessant, weiter so!
- War etwas anders ich dachte, aber sehr gut Vielen Dank!
- Die Technik könnte noch besser sein
- Dieser Schnupperkurs war sehr bereichernd und von allen Mitwirkenden sehr schön gestaltet :
- Vielen Dank, hat Spaß gemacht!
- Corona sucks (Schulnoten auch)
- Ich möchte mich bei Ihnen für die spannenden und herausfordernden online-meetings bedanken. Das haben Sie sehr gut gemacht!
- Es hat Spaß gemacht und war sehr interessant.
- Ich finde der Kurs könnte etwas länger sein oder die Tage einzeln am Stück und nicht einmal die Woche
- Die technischen Probleme sind der einzige Grund für eine 2. Dass teilweise einzelne Arbeitsblätter nur bei einer von 3 Personen funktioniert ist recht unpraktisch. Initalich fand ich den Workshop sehr interessant und lehrreich. Was mich gefreut hat war dass nicht so viele Grundlagen aus der Schulmathematik gebraucht wurden, da wir im Stoff etwas hinterherhängen und ich befürchtet hatte, nicht mitzukommen. Im Plenum und bei den Raumbesuchen war das Arbeitsklima stets angenehm, die Diskussionen hatten am Ende aber ruhig etwas länger sein können. Die Betreuenden waren freundlich und die Erklärungen nicht zu lang und nicht zu kurz.
- Am Anfang hätte ich mir eine längeres und intensiveres Kennenlernen/ Vorstellen der Teilnehmer untereinander gewünscht. Insgesamt fand ich den Kurs aber sehr gut gelungen.
- Ich danke Ihnen vielmals, dass Sie und Ihre Kolleginnen sich so viel Mühe gegeben haben und Sie wirklich mit Engagement und Spaß dabei waren. Vielen Dank für Ihre Zeit und Mühe. Ich werde Sie auf jeden Fall weiterempfehlen. Ich bin froh dabei gewesen sein zu dürfen. PS: Meine Lehrerin Frau Reimer hat mir diesen Workshop empfohlen, ohne Sie hätte ich wohl nichts davon gewusst. Viele Grüße und bis bald!
- Ich fand den Kurs sehr schön und alles erklärt. Die Gruppeneinteilung war sehr hilfreich, sodass man mit Personen, die ein ähnliches Wissen teilen zusammengearbeitet hat, was für eine angenehme Atmosphäre gesorgt hat. Die Diskussionen und anfänglichen Wiederholungen war auch super.
- Der Kurs hat mir gefallen, ging sehr schnell vorbei. War überrascht festzustellen dass heute der letzte Termin ist. Aber ja, ich habe viel gelernt und ich finde die Inhalte wurden gut überbracht und präsentiert.
- Sehr ruhige Atmosphäre zum Lernen, informativ

- Etwas langweilig
- Sehr gut gemacht
- Ich fand die Betreuer gut
- war toll
- Ich finde ihr habt das super vorgestellt und denke dass das thema gut vermittelt und vor allem interessant vermittelt wurde
- Hat echt Spaß gemacht
- die Inhalte waren gut jedoch von 8 uhr bis 16 uhr vor dem Laptop zu sitzen ist sehr anstrengend
- Sehr nettes Personal :D
- Georg und Annika sind sehr sympathisch. Sie haben sich eine Gehaltserhöhung verdient.
- Weiter so hat sehr Spaß gemacht
- Alles genannt
- einfach mehr mit die Studenten arbeiten und zusammen durch gehen
- Vielleicht ein kleiner Verbesserungsvorschlag: man könnte die Aufgaben auf dem Arbeitsblatt stärker hervorheben (z.B.: eine andere Farbe oder andere Dicke)
- Es war eine der schönsten Erfahrungen! Danke sehr -3
- Ich finde der Workshop wurde gut durch Zoom durchgesetzt und verständlich vorgestellt
- Etwas mehr Zeit für die Aufgaben
- Nice
- Die Umfrage war zu lang
- Sehr nett, freundlich. Haben sofort bei Fragen geholfen und versucht alles gut zu erklären und dies auch geschafft.

## G.2.2 Modellierungswoche





**Nenne drei Gründe, warum du gerne erneut an der CAMMP week teilnehmen möchtest oder warum du nicht erneut teilnehmen möchtest:**

- selbständiges, wissenschaftliches Arbeiten jenseits vom Lehrplan an der Schule
- Einführung in wissenschaftliche Modellierungsmethoden und direkte praktische Umsetzung in der eigenen Gruppe (sehr hilfreich für Studien- und Berufsorientierung)
- Austausch mit Wissenschaftler/-innen und Studierenden
- Es hat sehr viel Spaß gemacht
- Ich konnte zum ersten mal ein größeres Programm mit anderen zusammen schreiben
- Ich verstehe Bewertungssysteme jetzt viel besser
- interessant
- Herausforderung
- Spaß
- sehr interessante Themen
- gute Gruppenarbeit
- gute Organisation und Zeitplanung
- Interessantes Projekt
- Man lernt über Programmieren, Modellieren usw. dazu
- Das Treffen neuer Leute
- Das Zusammenarbeiten im Team um das Ziel zu erreichen
- Die Realitätsnähe der Aufgaben, was sie deutlich sinnvoller erscheinen lässt

**Was hast du für dich persönlich durch die Teilnahme an der CAMMP week gelernt?**

- Ich habe für mich persönlich gelernt, dass mir wissenschaftliches Modellieren und Problemlösen wirklich Spaß macht. Das hat mich in meinem Wunsch gestärkt, eine Naturwissenschaft (wahrscheinlich Mathematik oder Physik) zu studieren. Außerdem war es eine tolle Erfahrung, in einer Gruppe von interessierten Menschen zu arbeiten, bei der jeder und jede seine/ihre Stärken mit eingebracht hat (z. B. Personen, die (nicht) programmieren können). Die Informationsveranstaltungen zum Thema Studieren waren ebenfalls hilfreich, da sich bei denen auch organisatorische Fragen geklärt haben. Abschließend hat die Teilnahme an der CAMMP week auch mein Selbstbewusstsein im Hinblick auf das Engagieren in der Gruppe gesteigert.
- Wie funktionierender Empfehlungssysteme
- Probleme zu lösen/einfacher zu verstehen

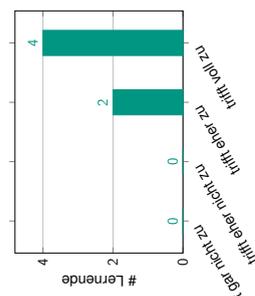
hat. Auch die Erfahrung, dass man bei mathematischen Aufgaben nicht immer alles auf dem Silberblett serviert bekommt, wie das oft in der Schule ist, war eine neue und sehr gute Erfahrung. Schwierigkeiten hatte ich mit dem Ausragsort. Ich bin es nicht gewöhnt den ganzen Tag so lange, konzentriert, vor dem Bildschirm zu sitzen, was dann leider dazu geführt hat, dass ich nach fast jedem Tag Abends ziemlich kaputt war und Kopfschmerzen hatte.

**Was würdest du an der CAMMP week verändern bzw. verbessern wollen?**

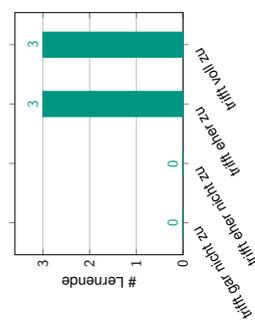
- Ich fände es toll, wenn gerade in der digitalen CAMMP week noch mehr soziale Aktivitäten in der großen Gruppe stattfinden würden.
- Nicht mehr CoCalc nutzen, da es... ..oft einfach so das Programm auf den Stand von vor 10min zurückversetzt (Einiges gelöscht) ...das Programm oft einmäch abbricht (zum Teil 5 Versuche, bis es einmal durchläuft) (Abbruch nicht wegen Fehlern)"
- gar nicht
- Durch die digitale CAMMP week fand ich es manchmal sehr anstrengend über einen so langen Zeitraum ohne Pause zu arbeiten. Ich würde dazu tendieren entweder weitere Pausen in den Zeitplan einzubauen oder die Zeiten, in denen am Tag gearbeitet wird, zu verkürzen. Vielleicht könnte man stattdessen noch einen weiteren Tag anhängen, um trotzdem die gleiche oder eine ähnliche Arbeitszeit zu erhalten.
- Ich weiß zwar nicht ob es viele bessere Plattformen gibt, aber mit CoCalc gab es schon einige male Probleme weshalb man vielleicht alternativen benutzen könnte.
- Natürlich war es schade, dass die CAMMP-Week dieses Jahr nur online stattfinden konnte. Neben ein paar technischen Problemen ist es online aber wirklich ganz gut gelaufen.

**3 Weiterführende Fragen**

**Die Lern- und Arbeitsatmosphäre war angenehm.**



**Die Dozenten haben die Inhalte plausibel und klar dargestellt.**



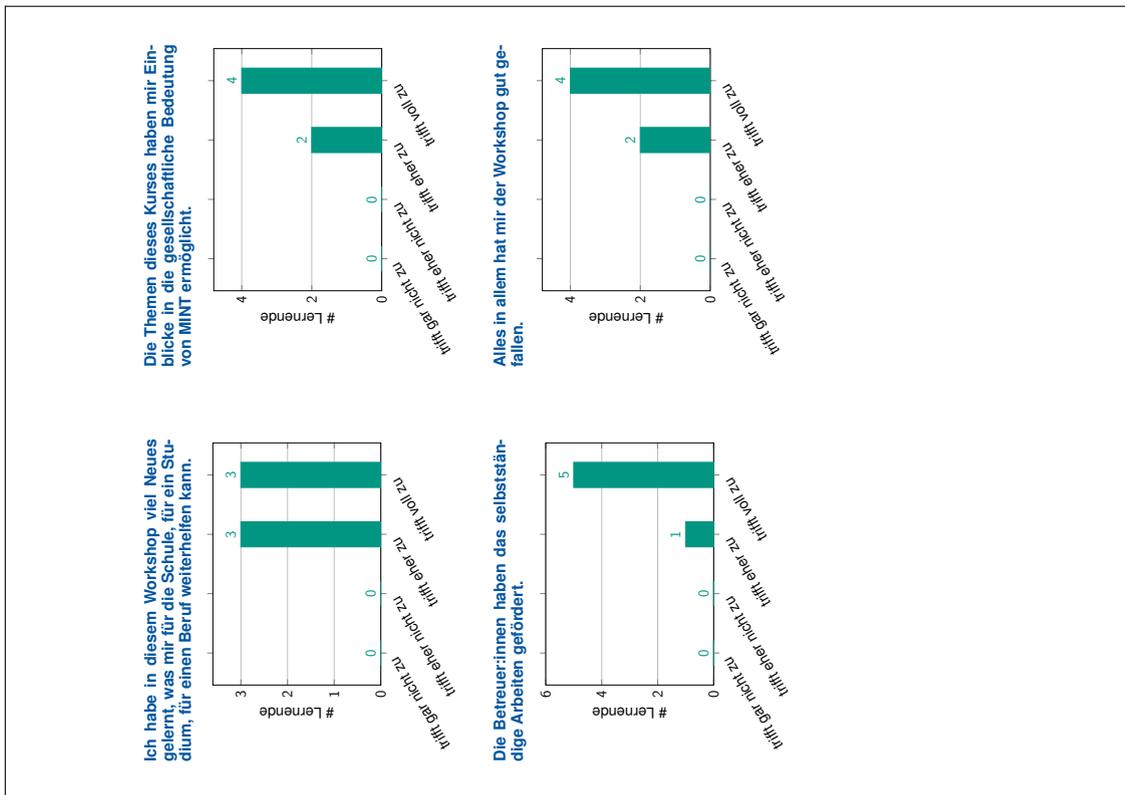
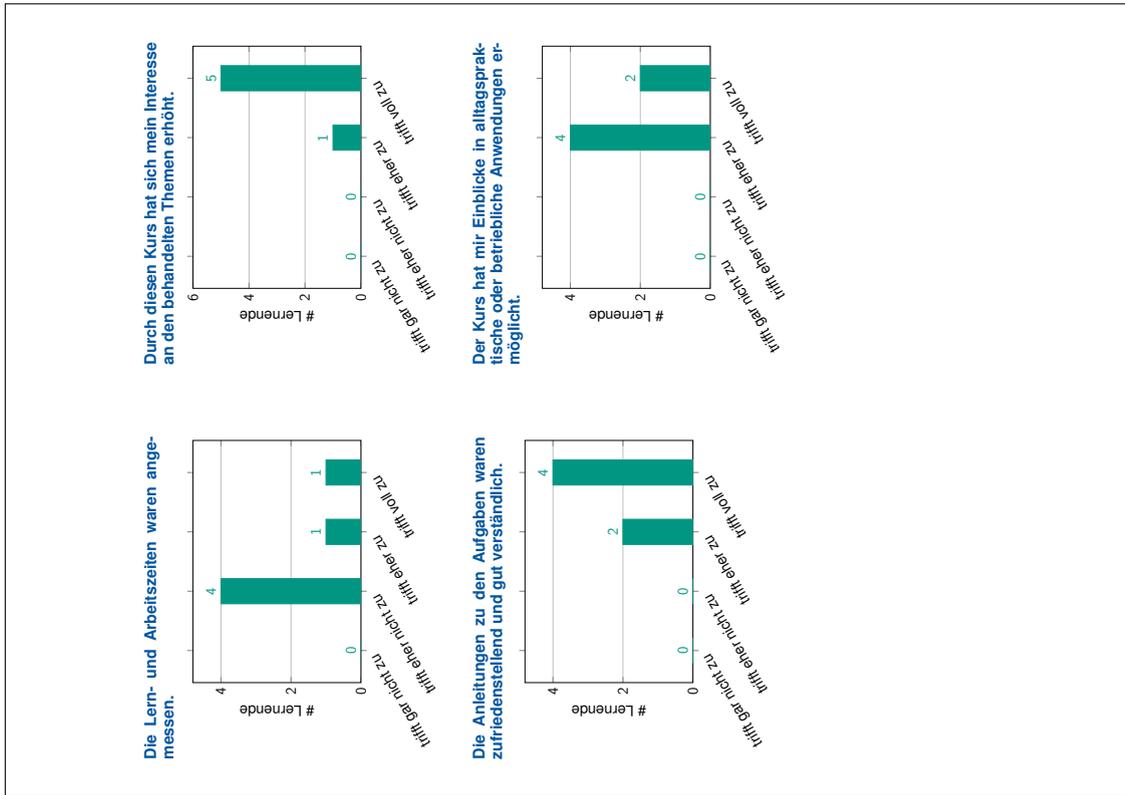
- Ich habe gelernt, dass es möglich ist, in einem kurzen Zeitraum sehr viel Neues zu lernen und dass man auch mit vielen fremden Personen in kurzer Zeit sehr gut zusammenarbeiten kann und sie auch kennenlernt.
- Ich habe einiges in Python und vorallem die Arbeit mit größeren Datensätzen kennengelernt.
- Zusammenarbeit im Team ist für solche Aufgaben essenziell, genauso wie gute Planung.

**Wo lagen für dich Schwierigkeiten im Modellierungs- und/oder Problemlöseprozess?**

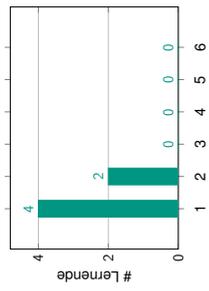
- Die größte Schwierigkeit bestand darin, mit den stark begrenzten Daten, die uns zur Verfügung standen umzugehen. Dadurch war es jedoch auch recht übersichtlich und man konnte sich auf weniger Teilspekte des Problems beschränken. Insgesamt fand ich die Aufgabenstellung recht herausfordernd, aber keineswegs überfordernd.
- CoCalc hat oft einfach so das Programm auf den Stand von vor 10min zurückversetzt (Einiges gelöscht) ). CoCalc bricht das Programm oft einmäch ab (zum Teil 5 Versuche, bis es einmal durchläuft) (Abbruch nicht wegen Fehlern)"
- keine
- Für mich lag anfangs die Schwierigkeit darin, dass ich zuvor noch nie programmiert habe und wir in unserer Gruppe nach einer kurzen Einführung sofort angefangen haben zu programmieren und ich zu dieser Zeit nicht wusste, was ich zur Gruppenarbeit beitragen kann. Das hat sich aber nach kurzer Zeit gelegt, da es dann auch für mich einige Aufgaben gab.
- Das Programmieren mit mehreren Personen, da man schnell durcheinander Kommt. Nicht genug Rechenleistung der CoCalc Plattform."
- Ohne große Hilfe sich zu organisieren. Abends versuchen über etwas anderes Nachzudenken M

**Wie gut konntest du dich in der Gruppenarbeit einbringen? Wo lagen für dich Schwierigkeiten? Wo bist du über dich hinaus gewachsen?**

- Zu Anfang war ich überfordert mit der Einführung in CoCalc, da ich zuvor keine Programmiererfahrungen gemacht hatte. Daher hatte ich zu Beginn die Sorge, dass ich mich nicht einbringen könnte. In unserer Gruppe haben wir uns jedoch entsprechend aufgeteilt. Mein Teil der Gruppe hat den mathematischen Ansatz ausgearbeitet und weitere Ansätze für erweiterte Problemstellungen entwickelt. In diesem Bereich hatte ich den Eindruck, dass ich einen guten Beitrag zu unserem Gruppenprojekt leisten konnte.
- Gut, die Schwierigkeiten lagen oft darin, dass wir uns Organisiert bekommen (nicht jeder macht alles). Wir haben es aber geschafft uns zu organisieren.
- sehr gut einfügen
- Ich habe mich am Anfang aufgrund meiner fehlenden Programmierkenntnisse nicht ganz so gut einbringen können, aber als dann klar war, dass ich mich um weitere Ansätze, Fragestellungen und um den Bericht und die Präsentation kümmern konnte, konnte ich mich auch sehr gut einbringen.
- Ich konnte mich gut einbringen, Schwierigkeiten waren, dass ich noch nicht viel mit python bzw. jupyter gearbeitet habe und ich dementsprechend vieles im Internet suchen musste.
- Anfänglich waren alle ein wenig schüchtern, was sich im Verlauf der Woche aber ziemlich schnell geändert



**4 Abschließende Bewertung**  
Ich gebe dem Workshop die Schulnote



**Abschließender persönlicher Kommentar (Lob, Kritik, Verbesserungsvorschläge, ...)**

- Vielen Dank. Ich hatte sehr viel Spaß und es war auch unglaublich interessant (: Meine Gruppe war auch super und die Betreuer waren alle sehr gut. Es war einfach super (:



# H Offene Problemstellungen aus Modellierungswochen

## H.1 Optimale Energieversorgung basierend auf drei erneuerbaren Energiequellen

### Optimal power supply with three renewable energies

Name Betreuer:in  
&

Namen der Problemsteller:innen

The city of Aachen has more than 260,000 inhabitants. It is possible to supply electricity for the city using 100% of renewable energies. There are at least three possibilities to generate green energy in this area: wind turbines, photovoltaic cells and water storage. Our goal is to generate enough energy for Aachen by just using sun light, wind and water. But, this should be as cheap as possible!

#### Problem

Find the best number of wind turbines and photovoltaic cells, such that Aachen has enough energy over the whole year. But sun and wind have different behavior over the time and it could happen that for some time of a day there is no wind or solar power. Therefore, we additionally can use a pumped-storage hydroelectricity system with an upper reservoir and a lower reservoir (see figure below with the locations in Aachen). If there is a high electric demand, water is released from the upper reservoir into the lower reservoir. During this process a turbine is powered which generates electricity. But the other way around, at times of low electric demand, the electricity generated with sun and wind is used to pump water into the upper reservoir. What is the perfect number of wind turbines and photovoltaics for Aachen? And how huge should be the volume of the upper reservoir? Please find the cheapest configuration.

#### Data

We have the hourly energy demand of the city of Aachen and the measured wind and sun data over a whole year. As costs for wind and photovoltaic energy use the current EEG rates of remuneration<sup>1</sup>, whereby, for the pumped-storage hydroelectricity you can assume costs of 5 Eurocent for each generated kWh. Also you can consider different cost scenarios for the future. The upper reservoir is a rectangular pool with a maximum surface of  $6 \text{ km}^2$ . You can select out of six different types of wind turbines which fits best for the given problem.

**Help us to transform Aachen into an energetically self-sufficient city!**



Abbildung 1: Possible locations of water reservoirs

<sup>1</sup> Vergütungssätze des Erneuerbare-Energien-Gesetz (EEG)

## H.2 Parkplatzwahl auf Automobilterminals

### Parkplatzwahl auf Automobilterminals

Name Betreuer:in

&

Namen der Problemsteller:innen

Auf ihrem Weg vom Werk zum Händler und schließlich zum Käufer passieren die meisten Fahrzeuge spezialisierte Automobilterminals. Im Regelfall werden sie hier eine Zeitlang zwischen geparkt und dann für den Weitertransport per Schiff, Zug oder LKW vorbereitet und zur Abholung bereitgestellt. Darüber hinaus bieten größere Terminals auf ihrem Gelände weitere Dienstleistungen an. Autos können hier repariert oder neu lackiert werden, es kann die Bereifung getauscht, eine bessere Musikanlage eingebaut oder eine anderweitige Modifikation vorgenommen werden. Die Fahrzeuge im Bestand eines solchen Terminals müssen somit unter Umständen mehrfach umgeparkt werden, bis sie alle vorgesehenen Schritte durchlaufen haben. Dies erledigen Fahrer, die auf dem gesamten Terminal im Einsatz sind. Ihre konkrete Aufgabe ist es, die Autos rechtzeitig zu den richtigen Parkblöcken zu bringen, von wo sie zu den Stationen abgerufen oder zum Weitertransport verladen werden. Innerhalb der Parkblöcke kann das Auto grundsätzlich auf jedem freien Parkplatz abgestellt werden.

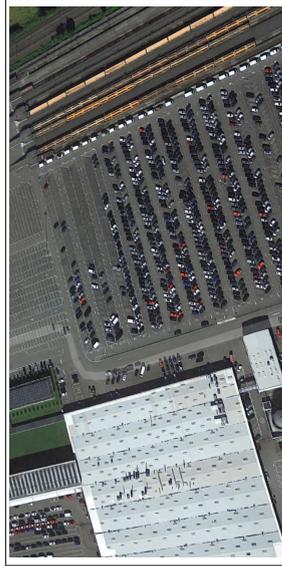


Abbildung 1: Beispiel für die Parksituation auf einem Autoterminal, Quelle: Inform

Die Parkplatzsuche soll im Folgenden näher betrachtet werden. Es lässt sich zunächst feststellen, dass die Parkblöcke groß sein können, manchmal Reihen sich hier einige Dutzend Parkplätze auf einer Strecke von mehr als einhundert Metern aneinander. Nachdem die Fahrer ein Auto abgestellt haben, müssen sie zu Fuß das nächste Auto erreichen. Die Wahl eines Parkplatzes bestimmt also hierbei, wie lange der Fahrer im Auto fahren muss, um ihn zu erreichen, und wie lange er anschließend zu Fuß zu seinem nächsten Ziel unterwegs ist. Die Summe aus beiden Zeiten soll so klein wie möglich sein, damit der Fahrer eine möglichst hohe Produktivität erreicht.

Abbildung 2 gibt ein Beispiel für den zurückzulegenden Weg für die Erledigung eines Fahrauftrags bis zum Antritt des nächsten Fahrauftrags. Der Ausschnitt zeigt ein Terminal in Belgien. Ein – in der Halle wartendes – blaues Fahrzeug soll in den markierten Block gestellt werden. Vom gewählten Parkplatz aus muss der Fahrer dann zu Fuß das rote Auto erreichen. Wie in der Abbildung angedeutet, ist davon auszugehen, dass der Fahrer mit dem Auto rechtwinklig angeordneten Straßen folgt, zu Fuß aber den direkteren diagonalen Weg nehmen kann. Es wird ferner angenommen, dass ein Fahrer mit dem Auto etwa 15 km/h schnell fahren kann, da aus Sicherheitsgründen auf dem Terminal Schrittgeschwindigkeit geboten ist. Zu Fuß erreicht er eine Geschwindigkeit von etwa 5 km/h.

**Problem:** Es stellen sich nun die folgenden Fragen:

1. Welcher Parkplatz soll in einem Block gewählt werden, um in möglichst kurzer Zeit den Fahrer

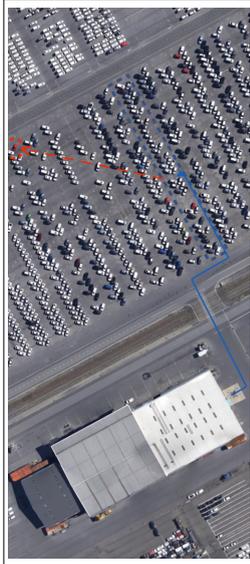


Abbildung 2: Beispiel für einen Fahrauftrag, Quelle: Inform

auftrag zu erledigen und das nächste Fahrzeug zu erreichen?

2. Welche Parkplätze sollen jeweils gewählt werden, wenn es zeitgleich mehrere Fahrer gibt, die ihre Autos in den gleichen Block stellen sollen?
3. Ändert sich die Strategie von 2.), wenn zur Parkplatzwahl die Auswahl des nächsten Fahrzeugs hinzugenommen wird? In diesem Fall gibt es bspw. drei Fahrer, die alle das Auto ihres aktuellen Auftrags erreicht haben, oder damit bereits unterwegs sind. Ferner gibt es drei Autos, die von diesen drei Fahrern im Anschluss übernommen werden sollen, wobei jedem der drei Fahrer jedes der drei Autos zugewiesen werden darf.

Zu jeder dieser drei Fragen wird ein möglichst schnelles Verfahren gesucht, das die optimale Lösung findet. Die Laufzeitänderung ergibt sich insbesondere daraus, dass die Probleme als Subprobleme in einem größeren Kontext mit längerem Zeithorizont und zahlreichen weiteren Beschränkungen sehr oft gelöst werden sollen.

## H.3 Optimierung eines Windparks

### Optimierung von Windparks

Name Betreuer:in

&

Namen der Problemsteller:innen

Nach dem Willen der Bundesregierung soll der Anteil erneuerbarer Energien an der Stromerzeugung bis zum Jahr 2025 von derzeit etwas über 30% auf 40-45% im Jahr 2025 und 55-60% im Jahr 2035 steigen. Gleichzeitig verleiht die Bundesregierung mit ihrer neuesten Novelle des Erneuerbare-Energien-Gesetz (EEG) aus dem Jahr 2016 ihrem Interesse Nachdruck, mit Hilfe von Ausschreibungsmodellen die Kosten für die Energiewende zu verringern. Um dem starken Wettbewerb Rechnung zu tragen müssen auch die Entwicklungskosten von erneuerbaren Technologien zwangsläufig weiter verringert werden.



Abbildung 1: Wirbelschleppen in einem Offshore Windpark, Quelle: Vattenfall

Der Konstruktion eines Windparks gehen viele Überlegungen voraus. Grob gesprochen produziert jede Windenergieanlage (WEA) eine bestimmte Leistung. Je mehr Anlagen an einem Standort installiert werden, desto höher ist zwar die installierte Leistung, aber durch mehr Anlagen erhöht sich auch die gegenseitige Abschattung (WAKE) (s. Abbildung). Je höher die Abschattung, desto größer sind Energieverlust und desto größer sind die Turbulenzen innerhalb des Systems. Diese Turbulenzen führen zu einer Eigenschwingung einzelner Teile der Anlage. Eine WEA ist in der Regel für 20-25 Jahre Lebenszeit ausgerichtet. Je höher die Eigenschwingungen, desto kürzer ist die Lebenszeitwartung der WEA. Eigentlich geht es also um einen komplexen zusammenhängenden Sachverhalt: Welches Layout hält am längsten durch und generiert dabei die meiste Energie? Wie kann so ein Szenario aussehen? Je mehr installierte Anlagen, desto besser? Oder doch weniger Anlagen dafür aber längere Lebenszeit und weniger Wartung?

**Problem:** Entwickelt ein Layout, sodass die installierte Leistung maximiert wird unter Einhaltung von Laskriterien. Überlegt nur für ein gegebenes/ener selbst entwickeltes Layout, wie die WAKE reduziert werden kann. Kann eine Drosselung einzelner WEA zu einer Steigerung auf der Gesamtpark-Ebene führen?

## H.4 Optimierung eines Fresnelkraftwerks

### Optimierung der Spiegel in einem solarthermischen Kraftwerk

Name Betreuer:in

&

Namen der Problemsteller:innen

Zur Sicherstellung der Energieversorgung in Zeiten von steigendem Energiebedarf und zunehmender Knappheit fossiler Brennstoffe wie Kohle, Öl und Gas, wird der Ausbau der regenerativen Energie immer schneller vorangetrieben. Innerhalb der EU bestehen bereits Pläne, in der Sahara riesige Solarkraftwerke zu errichten und den Strom über Hochspannungslleitungen in das europäische Stromnetz einzuspeisen.



Abbildung 1: Absorberrohr und planare Spiegel, Quelle: ebl

Das Prinzip solarthermischer Kraftwerke erscheint recht simpel: Spiegel bündeln Sonnenstrahlen auf ein Rohr, durch das Wasser geleitet wird. Auf diese Weise wird das Wasser erhitzt und durch den so entstehenden Dampf werden Turbinen angetrieben, welche dadurch Strom erzeugen. Idealerweise würde man gerne einen großen Spiegel in Parabolform verwenden, da dieser das eingefangene Licht auf genau einen Brennpunkt fokussiert.

Die Herstellung eines solchen Spiegel ist aber viel zu aufwändig und teuer. Zudem wäre ein solcher Spiegel stark windanfällig. Daher wird in aktuellen Forschungsprojekten der vielversprechende Ansatz verfolgt, einen Parabolspiegel durch mehrere ungewölbte Spiegelstreifen zu ersetzen. Da sich die Position der Sonne im Laufe des Tages ändert, wird der Neigungswinkel der Spiegel über den Tag hinweg stets angepasst.

Dadurch entstehen aber Probleme: Wird der Abstand zwischen den einzelnen Spiegelstreifen zu gering gewählt, so können Spiegel bei flach stehender Sonne Schatten auf andere Spiegel werfen, so dass nicht die komplette Fläche dieser Spiegel genutzt wird. Wird der Abstand allerdings größer gewählt, so muss man beachten, dass die äußersten Spiegelreihen nicht zu weit vom Rohr entfernt sind, da sich deren Neigungswinkel nicht so fein einstellen lassen und somit die gebündelten Strahlen das Rohr verfehlen können.

Wie kann man also den Abstand der Spiegel, deren Größe, die Höhe des Wasserrohres oder andere Parameter wählen, damit man die Sonnenenergie optimal ausnutzen kann?