

Alexander Schwarz

**Integration von Messdaten in die Simulation zur  
multikriteriellen, zeiteffizienten,  
versuchsbasierten Optimierung technischer  
Systeme**

Integration of measurement data in the simulation  
for the multicriteria time-efficient testbased  
optimization of technical systems

Band 71

Systeme ■ Methoden ■ Prozesse

Hrsg.: Prof. Dr.-Ing. Dr. h.c. A. Albers

Forschungsberichte



Alexander Schwarz

**Integration von Messdaten in die Simulation zur  
multikriteriellen, zeiteffizienten  
versuchsbasierten Optimierung technischer  
Systeme**

Integration of measurement data in the simulation for  
the multicriteria time-efficient testbased optimization  
of technical systems

Band 71

Herausgeber: Prof. Dr.-Ing. Dr. h.c. A. Albers

Copyright: IPEK ■ Institut für Produktentwicklung, 2013  
Karlsruher Institut für Technologie (KIT)  
Universität des Landes Baden-Württemberg und  
nationales Forschungszentrum in der Helmholtz-Gemeinschaft

Alle Rechte vorbehalten

Druck: Stolzenberger Druck und Werbung GmbH & Co. KG, Leimen  
06224-7697915

ISSN 1615-8113

# **Integration von Messdaten in die Simulation zur multikriteriellen, zeiteffizienten versuchsbasierten Optimierung technischer Systeme**

Zur Erlangung des akademischen Grades  
**Doktor der Ingenieurwissenschaften**  
der Fakultät für Maschinenbau  
Karlsruher Institut für Technologie (KIT)

genehmigte  
**Dissertation**

von

Dipl.-Ing. Alexander Schwarz  
aus Pforzheim

Tag der mündlichen Prüfung: 29. November 2013  
Hauptreferent: o. Prof. Dr.-Ing. Dr. h.c. A. Albers  
Korreferent: Prof. Dr. sc. techn. Thomas Koch



## **Vorwort des Herausgebers**(Stand: März 2013)

Wissen ist einer der entscheidenden Faktoren in den Volkswirtschaften unserer Zeit. Der Unternehmenserfolg wird in der Zukunft mehr denn je davon abhängen, wie schnell ein Unternehmen neues Wissen aufnehmen, zugänglich machen und verwerten kann. Die Aufgabe eines Universitätsinstitutes ist es, hier einen wesentlichen Beitrag zu leisten. In den Forschungsarbeiten wird ständig Wissen generiert. Dieses kann aber nur wirksam und für die Gemeinschaft nutzbar werden, wenn es in geeigneter Form kommuniziert wird. Diese Schriftenreihe dient als eine Plattform zum Transfer und macht damit das Wissenspotenzial aus aktuellen Forschungsarbeiten am IPEK - Institut für Produktentwicklung Karlsruhe<sup>[1]</sup> am Karlsruher Institut für Technologie (KIT) verfügbar.

Die Forschungsfelder des Institutes sind die methodische Entwicklung und das Entwicklungsmanagement, die rechnergestützte Optimierung von Strukturen und Systemen, die Antriebstechnik mit einem Schwerpunkt auf den Gebieten Antriebsstrang-Engineering und Tribologie und Monitoring von Lager- und Funktionsreibsystemen, die NVH mit dem Fokus auf Schwingungen und Akustik an Komponenten und am Gesamtfahrzeug, die Mikrosystemtechnik mit dem Fokus auf die zugehörigen Entwicklungsprozesse sowie die Mechatronik. Die Forschungsberichte werden aus allen diesen Gebieten Beiträge zur wissenschaftlichen Fortentwicklung des Wissens und der zugehörigen Anwendung – sowohl den auf diesen Gebieten tätigen Forschern als auch ganz besonders der anwendenden Industrie – zur Verfügung stellen. Ziel ist es, qualifizierte Beiträge zum Produktentwicklungsprozess zu leisten.

Albert Albers

---

<sup>[1]</sup> Eh.: Institut für Maschinenkonstruktionslehre und Kraftfahrzeugbau, Universität Karlsruhe (TH)



## Vorwort zu Band 71

Moderne Produkte des Maschinen- und Fahrzeugbaus sind gekennzeichnet durch eine große Komplexität und Kompliziertheit, die insbesondere durch die Mechatronisierung hervorgerufen worden ist. Zwar können mit diesen Produkten zusätzliche Kundenwünsche erfüllt und neue Produktleistungen realisiert werden, allerdings ist deren Entwicklung eine große Herausforderung. In den letzten zehn Jahren sind neue Ansätze zur Verbesserung der Entwicklungsprozesse erforscht worden. Diese basieren u. a. auf neuen Prozessmodellen, der Verwendung von Ontologien, dem Zurückführen der Vorgehensweisen auf die Systemtheorie, sowie dem Einsatz mathematischer Methoden in einer ganzheitlichen Optimierung. Im Kontext der Fahrzeugentwicklung ist die Abstimmung der vielfältigen Parameter in ihren Wechselwirkungen bei z. B. Antriebslösungen nur noch unter größtem Aufwand über reine manuelle „try and error“-Vorgehensweise möglich. Ziel der Forschung ist daher, bereits in frühen Phasen der Produktentstehung, auf der Basis von Modellen und unter der Anwendung von Optimierungsmethoden, möglichst mit Rechnerunterstützung, die Designparameter einem Optimum zuzuführen. Grundlage hierfür ist allerdings das Vorhandensein von Modellen, die nicht immer vollständig physikalisch abgebildet werden können. Um diese Modelle zu erstellen, ist es notwendig, diese mit experimentellen Methoden zu entwickeln und zu qualifizieren. Hierzu können verschiedenste Ansätze verwendet werden. Albers forscht mit seiner Gruppe unter dem Oberbegriff „XiL-Framework“ an einer grundlegenden neuen Methodik zur Synthese und Analyse komplexer mechatronischer Systeme. Hierbei wird durch eine Systembetrachtung das jeweilige betrachtete Produkt oder Teilsystem immer im Kontext des Übersystems entwickelt, wobei die Simulation sowohl virtuell als auch in Teilen durch Integration von physischen Modellen erfolgen kann. So ist es z. B. möglich, die Energieeffizienz oder auch die Kraftstoffverbräuche von Fahrzeugen mit hoher Genauigkeit zu untersuchen und zu definieren. Im Kontext dieses XiL-Konzepts kommt der geeigneten Anordnung von Teilversuchen und Ansätzen für die Identifizierung physikalischer Modelle anhand von Versuchsdaten eine große Bedeutung bei. An dieser Stelle setzt die Arbeit von Herrn Dr.-Ing. Alexander Schwarz an. Er hat in seiner Arbeit eine neue Methodik zur versuchsbasierten Optimierung technischer Systeme als Teil des XiL-Frameworks erforscht. Dabei berechnet er die optimalen Designparameterwerte, so dass die betrachteten Systeme hinsichtlich der definierten Kriterien optimales Verhalten zeigen. Ein weiteres Ziel der entwickelten Methodik ist, den Versuchs- und Optimierungsaufwand soweit als möglich zu reduzieren. Insgesamt ist die Methodik in den Kontext des Produktentstehungsprozesses als Handlungssystem nach Ropohl eingeordnet und in den übergreifenden Ansatz zur Beschreibung von Produktentstehungsprozessen iPeM – integriertes Produktentstehungsmodell nach



Albers integriert. Herr Dr.-Ing. Alexander Schwarz leistet mit seiner Arbeit einen wertvollen Beitrag zur weiteren Verbesserung der Entwicklungsprozesse für Fahrzeuge und zum Ausbau der „Karlsruher Schule für Produktentwicklung“.

November, 2013

Albert Albers

## Kurzfassung

Das Systemverhalten technischer Systeme hängt oft von einer Vielzahl von Parametern und deren Wechselwirkungen ab. Damit diese Systeme auf dem Markt wettbewerbsfähig sein können, müssen die Parameter so bestimmt werden, dass sich das System hinsichtlich mehrerer Kriterien optimal verhält. In frühen Phasen des Produktentstehungsprozesses werden, um das sogenannte Frontloading (vgl. Kapitel 3.1) zu erzielen, physikalische Modelle für die Optimierung der Designparameter verwendet. Eine Herausforderung bei der Erstellung dieser Modelle ist, dass oft nicht alle Parameter, wie zum Beispiel die Reibung in den relevanten Wirkflächenpaaren, bekannt sind. Deswegen müssen häufig Annahmen getroffen werden, die dann im Versuch abgesichert werden müssen. In späteren Phasen werden dann Versuche mit einzelnen Komponenten bis zum fertigen Prototyp durchgeführt. In diesem Stadium werden vor allem nicht-physikalische Modelle, wie künstliche neuronale Netze (KNN) oder polynomiale Modelle, verwendet, um die optimalen Werte für die Designparameter zu bestimmen. Eine Methode die hier angewendet wird, ist die statistische Versuchsplanung (Design-of-Experiments, DOE) nach Taguchi, bei der die Modellstruktur schon bei der Versuchsplanung berücksichtigt wird.

Diese Arbeit stellt ein Teilhandlungssystem vor, das beide Ansätze in sich vereint. Dabei werden Versuchsdaten verwendet, um physikalische Modellstrukturen zu parametrieren. Anschließend werden diese physikalischen Modelle für die Optimierung der Designparameter eingesetzt. Der Vorteil dieses Ansatzes ist, dass auch Parameter, die nur langsam im Versuch verstellt werden können, mit Hilfe des Simulationsmodells optimiert werden können. Außerdem ist es möglich Kriterien, die lange Versuchszeiten erfordern, schnell mit dem Modell zu berechnen. Ein Beispiel für ein solches Kriterium ist der Verbrauch im NEFZ-Zyklus (neuer europäischer Fahrzyklus), welcher ca. 20 Minuten für einen Durchlauf benötigt. Im Versuch müsste dieser Zyklus für jede Parametervariation einzeln abgefahren werden. Mit dem in dieser Arbeit vorgestellten Ansatz wird zunächst anhand von Versuchsdaten ein physikalisches Fahrzeugmodell gebildet, welches sich hinsichtlich der betrachteten Kriterien hinreichend genau verhält. Anschließend wird dieses Modell für die Optimierung der Parameter verwendet, wodurch entsprechend Versuchszeit eingespart werden kann. Ein anderer Vorteil des hier vorgestellten Ansatzes ist, dass die physikalischen Grenzen des Modells bekannt sind. Außer für die Optimierung, können diese Modelle verwendet werden, um bei gegebenen Eingangsgrößen und dem Ausgangszustand, das Systemverhalten vorherzusagen. Neben verschiedenen Ansätzen für die Identifizierung physikalischer Modelle anhand von Versuchsdaten wird eine Methode zur Optimierung der Messpunktfolgenfolge für die zeiteffiziente

Durchführung der Versuche untersucht. Diese Ansätze stellen dabei einen Baustein zum XiL-Framework vgl. Kapitel 3.1 dar.



## **abstract**

The system behaviour of technical systems often depends on a multitude of parameters and their interdependencies. These parameters must be determined in a way which optimizes the system behaviour with respect to the considered criteria to keep the system competitive in market. Therefore in early phases of the product development process, physical models are used for the optimization of design parameters to realize the so-called frontloading c.f. 3.1. One challenge during the creation of those models is, that often not all needed parameter, such as the friction in relevant working surface pairs, are known. That's why, presumptions have to be made first and reassured in tests later on. In later phases, tests with single components or even complete prototypes are executed. In this stage, mainly non-physical models like artificial neural networks or polynomial models are used to determine the optimal values for the design parameters. One method, that is applied here, is the statistical test planning, the Design-of-Experiments method (DOE) according to Taguchi. Here the structure of the model is already considered during the planning of the test.

This draft presents a part of the operation system, which combines both approaches. Thereby, test data are used to parameterise physical model structures. Afterwards these physical models are used for the optimization of the design parameters. The advantage of this approach is that also parameters, which can only be changed slowly during a test, can be optimized with the help of the simulation models. Furthermore, criteria, which need long testing times, can be quickly calculated using this model. The consumption in a NEDC (new european driving circle), which needs app. 20 minutes for one test cycle, is an example for such a criterion. In an experiment this cycle would have to be performed particularly for each set of parameters. With the approach, which is presented in this work, first a physical model with a sufficient model quality, regarding the considered criterions, is created using testdata. Afterwards this model is used for the optimization of the parameters, which saves time compared with the real tests. Another advantage of this approach is, that the physical boundaries of the models are known. These models also can be used to predict the system behaviour, when input quantities and initial state are known. Along with several approaches for the identification of physical models on the basis of test data, an approach for optimizing the order of measurement points for time efficient test execution is analysed. Those approaches are one brick for the XiL-Framework see chapter 3.1.





## Danksagung

Diese Arbeit ist im Rahmen meiner Tätigkeit als wissenschaftlicher Mitarbeiter am IPEK- Institut für Produktentwicklung am Karlsruher Institut für Technologie (KIT) entstanden.

Ich möchte mich für die wissenschaftliche Begleitung bei meinem Doktorvater o. Prof. Dr.-Ing. Dr. h. c. Albert Albers bedanken.

Außerdem möchte ich mich bei meinen allen bedanken, die mich direkt oder indirekt bei dieser Arbeit unterstützt haben, beziehungsweise mir den Rücken freigehalten und mich motiviert haben:

Vielen Dank!

Insbesondere möchte ich den meinen Eltern, meiner Familie, meiner Partnerin, meinen Freunden, meinen Kollegen sowie den Industriepartnern während dieser Zeit danken.

Karlsruhe, den 21.02.2013

Alexander Schwarz





Die Realität ist das, was man nicht erkennt, wenn man sie erkennt

Niklas Luhmann



# Inhalt

<b>1</b>	<b>Einleitung</b> .....	<b>1</b>
<b>2</b>	<b>Grundlagen und Stand der Forschung</b> .....	<b>4</b>
2.1	Grundlagen zur Produktentstehung.....	4
2.1.1	Das integrierte Produktentstehungsmodell (iPeM).....	4
2.1.2	X-in-the-Loop Framework (XiL-Framework) .....	9
2.2	Modelle .....	11
2.2.1	Klassifizierung von Modellen .....	13
2.2.2	Grundbegriffe Modellstruktur, Modellparameter, Modellbildung .....	14
2.3	Optimierung/Optimierungsverfahren .....	17
2.3.1	Parameteroptimierung.....	19
2.3.1.1	Axial-Iteration .....	19
2.3.1.2	Evolutionäre bzw. genetische Optimierungsverfahren .....	20
2.3.1.3	Monte-Carlo-Methode.....	21
2.3.1.4	Simplex Optimierungsstrategie .....	21
2.3.1.5	Simulated Annealing.....	22
2.3.1.6	Schätzer .....	23
2.3.2	Reihenfolgenoptimierung .....	25
2.3.2.1	Nächster Nachbar Heuristik (next-neighbor- oder NN-Heuristik).....	26
2.3.2.2	Doppelter Nächster-Nachbar-Heuristik .....	26
2.3.2.3	Minimalaufspannende Baum Heuristik (minimum spanning tree- oder MST-Heuristik) nach Hußmann .....	26
2.3.2.4	Christofides-Heuristik .....	28
2.4	Statistische Versuchsplanung - Design of Experiments (DOE).....	29
<b>3</b>	<b>Motivation und Zielsetzung</b> .....	<b>33</b>
3.1	Motivation .....	33
3.2	Zielsetzung .....	34
<b>4</b>	<b>Teilhandlungssystem zum versuchsbasierten Optimieren</b> .....	<b>38</b>
4.1	Einordnung in das integrierte Produktentstehungsmodell (iPeM) .....	38
4.2	Prozess.....	39
4.3	Definition der Optimierungsaufgabe .....	44
4.4	Variablen definieren .....	44
4.5	Manöver, Kriterien, Zielfunktion definieren .....	45
4.6	Versuchsdesign und Durchführung im XiL-Framework.....	47
4.6.1	Definition XiL-Validierungsumgebung.....	48
4.6.2	Auswahl des Verfahrens .....	48
4.7	Design of Experiments (DOE) – Verfahren.....	50

4.8	Design of Experiments (DOE) mit Ordnung der Messpunkte.....	51
4.8.1	Verfahren .....	51
4.8.2	Analyse DOE mit Ordnung der Manöver .....	53
4.9	Einsatz frei definierbarer Modellstrukturen .....	58
4.9.1	Art des Modells .....	58
4.9.2	Modellbeschreibungsform .....	60
4.9.3	Versuchsdurchführung und Auswertung.....	63
4.9.4	Identifikation.....	65
4.9.4.1	Identifikation offene Modellstrukturen .....	65
4.9.4.2	Identifikation offene Modellstrukturen - mehrere (gewichtete) Zustandsgrößen .....	67
4.9.4.3	Identifikation offene Modellstrukturen - stückweise definierte Funktionen ..	68
4.9.4.4	Identifikation offene Modellstrukturen - Berücksichtigung bekannter (konstanter) Parameter.....	70
4.9.4.5	Identifikation offene Modellstrukturen - rechnergestützte Umsetzung .....	71
4.9.4.6	Identifikation geschlossene Modellstrukturen mittels Optimierer .....	74
4.9.5	Modellverifikation .....	77
4.9.6	Bestimmung der optimalen Designparameterwerte .....	79
4.9.7	Verifikation Auslegung .....	80
4.9.8	Analyse Einsatz frei definierbarer Modellstrukturen.....	81
4.9.8.1	Der Einmassenschwinger als Beispiel für die Vorgehensweise .....	82
4.9.8.2	Kennwerte für die Leistungsfähigkeit des Verfahrens .....	86
4.9.8.3	Einflussfaktoren – Optimierungsaufgabe .....	87
4.9.8.4	Einflussfaktoren –Verfahren .....	88
<b>5</b>	<b>Bewertung der Verfahren .....</b>	<b>91</b>
<b>6</b>	<b>Weitere Anwendungsmöglichkeiten.....</b>	<b>97</b>
<b>7</b>	<b>Zusammenfassung und Ausblick.....</b>	<b>99</b>
7.1	Zusammenfassung .....	99
7.2	Ausblick .....	100
<b>8</b>	<b>Literaturverzeichnis.....</b>	<b>102</b>
8.1	Quellen .....	102
8.2	Co-betreute Studien-/Diplom-/Bachelor-/Masterarbeiten .....	105
<b>9</b>	<b>Anhang.....</b>	<b>106</b>
9.1	Maple Code für Generierung der Schätzermatrix .....	106
9.2	Leistungsdaten Rechner .....	107
9.3	Rechnergestütztes Werkzeug .....	108

# Formelzeichen

Formelzeichen	Einheit	Bezeichnung
$\vec{b} = b_k = (:) = o * w$	-	Gewichteter Normierungsfaktor
c	N/m	Federkonstante
d	N*s/m	Dämpferkonstante
F	N	Kraft
$F_N$	N	Normalkraft
GK	-	Gütekriterium
$\vec{H} = H_{kl} = (:::)$	-	Designmatrix
$H_{zust} = (:::)$	-	Designmatrix der einzelnen Zustände
I	-	Übersetzung
J	-	Schätzgrößen
k=1..ke	-	Laufvariable Messpunkte
l=1..le	-	Laufvariable Zustandsgrößen
M	Nm	Drehmoment
$m_i$	-	Geradensteigung
N	-	Konten, Anzahl
$n_{Designparameter}$	-	Anzahl Designparameter
$n_{Variationen}$	-	Anzahl Variationen
$n_{Variationsstufen}$	-	Anzahl Variationsstufen
n	$min^{-1}$	Motordrehzahl
$n_{turbo}$	$min^{-1}$	Turboladerdrehzahl
$o_j$	-	Normierungsfaktor
P	N/mm <sup>2</sup>	Pressung, Druck
$\vec{p}$	-	Parametervektor
$\vec{q} = 1..qe$	-	Laufvariable Parameter
$R^2$	-	Bewertung Modellabweichung
$R^2_{adj}$	-	Bewertung der Vorhersagekraft des Modells
R	mm	Radius
$t_{Bewertungsmanöver}$	s	Dauer Bewertungsmanöver
$t_{ges}$	s	Versuchszeit
$t_{Konditionierung}$	s	Konditionierzeit
$t_{variation}$	s	Dauer der Variation
$t_{Verstell}$	s	Dauer der Verstellung der Designparameter
$w_k$	-	Gewichtungsfaktor

$x_a$	-	Grenzen
$x_{re}$	-	Absolute Position Intervallgrenze
$\vec{x}$	-	Zustandsgrößen
$\vec{y} = y_k$	-	Schätzgröße(n) der gefilterten Messdaten
$\vec{\alpha} = \alpha_l = \begin{pmatrix} f_1(\vec{p}) \\ \vdots \\ f_2(\vec{p}) \end{pmatrix}$	-	kombinierter Parametervektor
$\hat{\alpha}_i$	-	bekannter Schätzparameter
$\vec{\epsilon}_a$	-	Abweichung (Residuen)
$\vec{\epsilon}_a = \epsilon_{ak}$	-	Absolute Abweichung von Modellwert zu Messwert
$\vec{\epsilon}_k = \epsilon_k$	-	Gewichteter relativer Fehler
$\lambda$	-	Nachkommen
$\mu$	-	Eltern
$\rho$	-	Anzahl der Gruppen
$\Sigma$	-	Standardabweichung
$\alpha$	-	Kombinierter Parametervektor

# 1 Einleitung

Während des Branchenlebenszyklus müssen Produkte stetig weiterentwickelt werden, damit diese auf dem Markt wettbewerbsfähig bleiben können (Produktgenerationsentwicklung vgl. Albers). Dabei hängen die wesentlichen Eigenschaften und das Systemverhalten des Produkts (siehe Bild 1) oft von einer Vielzahl an Parametern ab. Die Parameter beschreiben dabei die Eigenschaften der Wirkflächenpaare und Leitstützstrukturen (vgl. Kapitel 2.1.1.). Um das optimale Verhalten dieser Systeme zu erreichen, müssen die entsprechenden optimalen Designparametereinstellungen (vgl. Kapitel 4.4) gefunden werden. Dabei kann das Systemverhalten durch Kriterien bewertet werden. Moderne technische Systeme müssen sich oft hinsichtlich mehrerer Kriterien bzw. mehrere Gruppen von Kriterien optimal verhalten, die teilweise bezüglich der betrachteten Designparameter in Wechselwirkung stehen. Diese Optimierung wird als Teilaktivität der Aktivität "Modellierung von Prinzip und Gestalt" im iPeM (integriertes Produktentwicklungsmodell) nach Albers<sup>1, 2</sup> (vgl Kapitel 2.1.1) benötigt.



Bild 1: technische Systeme und Gruppen von Kriterien

<sup>1</sup> Albers 2010a

<sup>2</sup> Albers/ Braun 2011



Ein Beispielsystem in dieser Arbeit ist die Betriebsstrategie des Doppelkupplungsgetriebes eines Sportwagens. In diesem System sind teilweise über 200 Parameter in den Steuergeräten an der Durchführung einer Schaltung beteiligt. Die Einstellung dieser Parameter beeinflusst verschiedene Gruppen von Kriterien, wie zum Beispiel den Komfort, die Performance sowie die Energieeffizienz der Schaltung. Sollen diese Steuergeräteparameter bzw. ein Teil davon optimiert werden sind dies die Designparameter für diese Optimierungsaufgabe.

Im Rahmen dieser Arbeit wird ein Teilhandlungssystem erarbeitet, mit dem die optimalen Designparameterwerte effizient bestimmt werden können. Dazu werden durch Versuchsdaten parametrisierte physikalische Modellen verwendet. Dieses Teilhandlungssystem setzt sich aus vier wesentlichen Bestandteilen zusammen:

- Versuchsplanung und -durchführung
- Identifizierung
- Optimierung
- Verifikation

Im ersten Schritt wird basierend auf einer gezielt abgeleiteten Modellstruktur und entsprechenden Kriterien ein geeigneter Versuchsaufbau sowie entsprechende Messpunkte definiert. Für den Versuchsaufbau wird das XiL-Framework vgl. Kapitel 2.1.2 verwendet, um den in dieser Arbeit beschriebenen Ansatz auch schon in frühen Entwicklungsphasen einsetzen zu können. Bereits die Versuchsdurchführung kann durch den Einsatz von Modellen effizienter gestaltet werden. Indem die Messpunkte, mit Hilfe des Modells, in eine zeiteffiziente Reihenfolge gebracht werden, wird die Gesamtdauer der Konditionierungsphasen zwischen den Versuchen reduziert. Als Beispiel wird in Kapitel 4.8 die zeiteffiziente Erfassung der Schaltqualität eines Fahrzeugs mit Doppelkupplungsgetriebe auf dem Rollenprüfstand verwendet. Bereits durch diesen Schritt können erhebliche Zeiteinsparungen erzielt werden. Beim Beispiel der Getriebekalibration können dadurch gegenüber dem herkömmlichen Ansatz (mit einem erfahrenen Testfahrer) bei den gleichen Messpunkten ca. 80% Zeiteinsparung erzielt werden.

Der zentrale Schritt des Teilhandlungssystems ist die Identifizierung physikalischer Modellstrukturen. Dazu werden mit Hilfe von Messdaten die unbekannt Parameter der Modellstruktur so berechnet, dass sich das Modell hinsichtlich der definierten Kriterien so wie der Prüfling (Unit-Under-Development) verhält. In Kapitel 4.8.2 werden dazu verschiedene Ansätze am Beispiel eines Fahrzeugs mit Automatikgetriebe vorgestellt und untersucht. Da das resultierende Modell, zumindest in den relevanten Subsystemen physikalisch ist, darf das Modell im

folgenden Schritt zur Optimierung der Designparameter herangezogen werden. Deshalb müssen keine Variationen der Parameter, wie bei bestehenden Ansätzen mit dem physischen System, durchgeführt werden. Dadurch kann bei Optimierungsaufgaben, bei denen die Erfassung der Kriterien oder die Verstellung der Parameter viel Zeit in Anspruch nimmt, eine große Zeitersparnis erzielt werden. So können auch in späten Phasen der Entwicklung noch Parameter als Designparameter definiert werden, die sonst zu aufwendig für eine Variation wären.

Eine zentrale Aktivität in dieser Arbeit ist die Verifikation des identifizierten Modells. Dazu werden Methoden vorgestellt mit denen beurteilt werden kann, ob sich das Modell mit der geforderten Genauigkeit verhält (Verifikation) aber auch ob die richtige Modellstruktur verwendet wird.

## 2 Grundlagen und Stand der Forschung

In diesem Kapitel wird der relevante Stand der Forschung zu der aufgeworfenen Forschungsfrage zusammengefasst. Dazu wird zunächst auf den Produktentstehungsprozess eingegangen und die Grundbegriffe geklärt. Basierend auf diesen Grundlagen wird in Kapitel 3 die Forschungshypothese sowie die Zielsetzung der Arbeit abgeleitet.

### 2.1 Grundlagen zur Produktentstehung

Eine wichtige Anforderung für neue Methoden und deren effizienten Einsatz ist, dass diese sich in vorhandene Produktentwicklungsprozesse eingliedern lassen bzw. der Entwicklungsprozess muss entsprechend gestaltet werden. Deshalb wird im Folgenden auf die Modellierung des Produktentstehungsprozesses eingegangen. Außerdem wird das in dieser Arbeit als Entwicklungs- bzw. Validierungsumgebung verwendete XiL-Framework erläutert (siehe Kapitel 2.1.2.)

#### 2.1.1 Das integrierte Produktentstehungsmodell (iPeM)

Albers<sup>3,4</sup> formuliert auf Basis von Erfahrungen aus erfolgreichen Projekten und der Forschung an Prozessmodellen ein Metamodell für die Produktentstehung – das integrierte Produktentstehungsmodell (iPeM). Dieses integriert Erkenntnisse bekannter Prozessmodelle wie dem V-Modell der VDI Richtlinie 2206<sup>5</sup> bzw. der VDI Richtlinie 2223<sup>6</sup>, dem Prozess nach Pahl<sup>7</sup>, dem Stage-Gate-Model von Cooper<sup>8</sup> sowie den Prozessabläufen nach Ehrlenspiel<sup>9</sup>. Dadurch wird ein Modell geschaffen, mit dem es möglich ist, reaktionsfähige Prozesse mit komplexen Produkten/Systemen mit sich ändernden Randbedingungen abzubilden.

Dieses Metamodell basiert nach Albers<sup>10</sup> auf den folgenden fünf Grundhypothesen:

---

<sup>3</sup> Albers 2010a

<sup>4</sup> Albers/Braun 2011

<sup>5</sup> VDI-Richtlinie 2206

<sup>6</sup> VDI-Richtlinie 2223

<sup>7</sup> Pahl 2005

<sup>8</sup> Cooper 1990

<sup>9</sup> Ehrlenspiel

<sup>10</sup> Albers 2010a

**Hypothese 1:** Individualität von Produktentstehungsprozessen

Der Entstehungsprozess jedes Produkts ist einzigartig und wird sich nicht identisch wiederholen, da sich Randbedingungen wie Wirtschaftslage, Wissen oder Kundenforderungen ändern.

**Hypothese 2:** System der Produktentstehung

Mit Hilfe des Handlungssystems (siehe Bild 2) wird das Objektsystem (Istzustand) sowie das Zielsystem kontinuierlich während des Produktentstehungsprozesses weiterentwickelt vgl. Ropohl<sup>11</sup>, Meboldt<sup>12</sup>, Albers<sup>13</sup>

Im Zielsystem sind dabei neben der mentalen Vorstellung alle Beschreibungen, Anforderung und Restriktionen des fertigen Produkts enthalten. Dazu zählen alle entwickelten Kriterien sowie deren Bewertungskenngrößen. Schwerpunkt dieser Arbeit ist die Beurteilung der Energieeffizienz und des Komforts von Personenkraftwagen mit dem Fokus auf antriebsstranginduzierte Phänomene. Dazu werden in verschiedenen Phasen des Entwicklungsprozesses verschiedene Kriterien, wie der Verbrauch im NEFZ (neuer europäischer Fahrzyklus) oder der A-gewichtete Schalldruckpegel im Innenraum, bei einem Fahrmanöver verwendet. Im Objektsystem ist der aktuelle Stand der Entwicklung mit allen Zeichnungen, Zwischenergebnissen, virtuellen oder physikalischen Prototypen enthalten.

Mit Hilfe des Handlungssystems wird ein Zielsystem in ein Objektsystem überführt. Während dieses Prozesses wird das Zielsystem kontinuierlich erweitert, geändert und konkretisiert. Das Handlungssystem ist dabei ein sozio-technisches System in dem alle Aktivitäten, Methoden und Prozessbeschreibungen enthalten sind um dieses Ziel zu erreichen. Dabei ist der Mensch der zentrale Akteur im Produktentstehungsprozess.

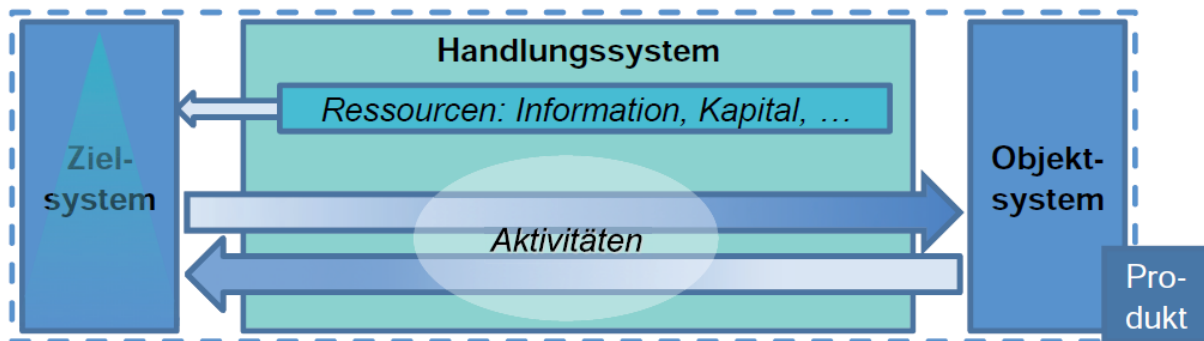
Ein anderer wichtiger Einfluss auf den Produktentstehungsprozess ist das Ressourcensystem. Hier sind Informationen wie Kapital, Material, Energie aber auch Personalressourcen enthalten. Dieses System kann sich, zum Beispiel durch die wirtschaftliche Lage, auch zeitlich ändern. Deshalb hat das Ressourcensystem als Teil des Handlungssystems einen Einfluss auf das Zielsystem.

---

<sup>11</sup> Ropohl 2009

<sup>12</sup> Meboldt 2008

<sup>13</sup> Albers / Sadowski 2013

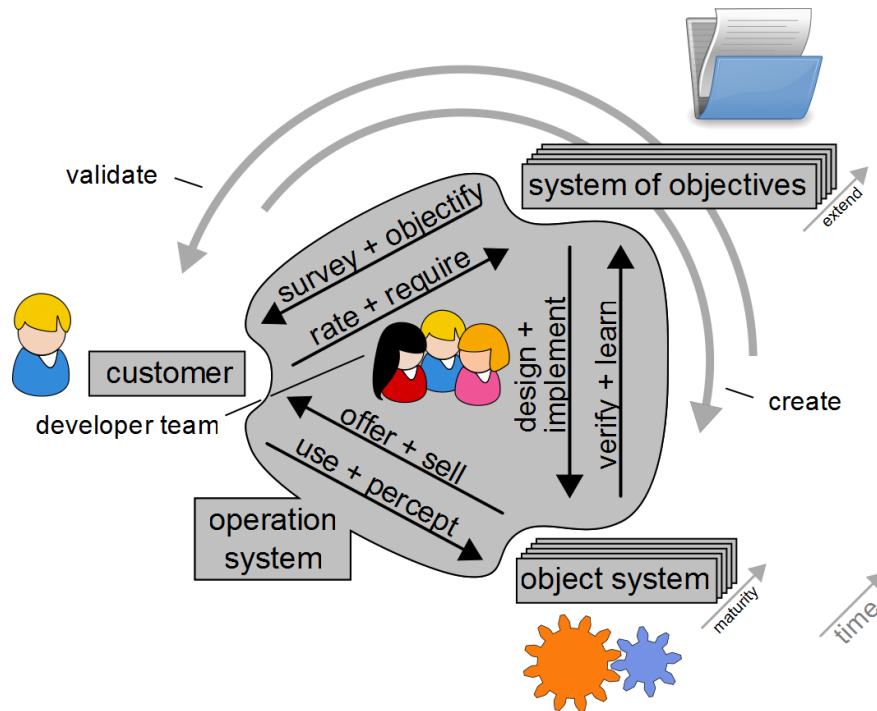
Bild 2: System der Produktentstehung<sup>14</sup>

### Hypothese 3: Validierung

Die zentrale Aktivität während des Produktentstehungsprozess ist die Validierung. Diese Aktivität wird kontinuierlich in mehreren Phasen im Produktentstehungsprozess durchgeführt. In diesen Phasen wird das aktuelle Objektsystem mit dem Zielsystem abgeglichen. Dieser Abgleich dient zum einen dazu, den Prozessverlauf im Sinne des Controlling zu messen, aber auch um Wissen zu erzeugen vgl. Meboldt<sup>15</sup>. Dabei kann die Validierung in zwei Fragestellungen aufgeteilt werden (siehe Bild 3). Die Frage "Wurde das Produkt richtig, nach den Zielsystem, entwickelt" (Verifikation) und "Passt das System zur Kundengruppe/ Marktsituation" (Validierung). Das heißt, die Validierung enthält die Fragestellung wie gut das initiale Zielsystem definiert wurde und wie gut dieses umgesetzt wurde.

<sup>14</sup>Albers / Braun 2011

<sup>15</sup>Meboldt 2008

Bild 3: Validierung im Produktentwicklungsprozess<sup>16</sup>**Hypothese 4:** Zielbeschreibung in der Problemlösung.

Der Zielerreichungsprozess kann in der Entwicklung als Problemlösungsprozess verstanden werden. In der Analogie des Problemlösungsprozesses stellt das Ziel den Sollzustand und die Ausgangssituation den Istzustand des Problems dar. Das Handlungssystem stellt dabei Methoden zur Überführung des Istzustandes in den Sollzustand bereit.

**Hypothese 5:** Beschreibung von Funktionen

Nach Albers<sup>17,18</sup> wird in technischen Systemen für die Realisierung von Funktionen immer die Wechselwirkung von mindestens zwei Wirkflächenpaaren, einer Leitstützstruktur und zwei Konnektoren benötigt. Dabei modellieren die Konnektoren die Wirkung der Umwelt, die für die betrachtete Funktion des Systems relevant ist. Die Leitstützstrukturen, die Kräfte, Stoff und Information leiten, müssen in diesem Fall nicht zwingend Festkörper sein. Weitere Beispiele für Leitstützstrukturen sind Flüssigkeiten, elektrische- sowie magnetische Felder. Zwischen den Leitstützstrukturen befinden sich die Wirkflächenpaare.

<sup>16</sup> Albers et al. 2011c

<sup>17</sup> Albers / Braun 2011

<sup>18</sup> Albers / Sadowski 2013

Auf Basis dieser Hypothesen definiert Albers<sup>19</sup> das integrierte Produktentstehungsmodell iPeM (siehe Bild 4). Dabei enthält dieses Metamodell alle Aktivitäten, die im Produktentstehungsprozess durchgeführt werden (Aktivitätenmatrix). Aus dieser Aktivitätenmatrix wird dann das individuelle Phasenmodell entwickelt. Dabei können einzelne Aktivitäten, wie die Validierung, mehrfach im Phasenmodell auftreten. In jeder Aktivität bzw. Teilaktivität findet jeweils mindestens ein SPALTEN (siehe Albers<sup>20</sup>) zur Problemlösung im Sinne der Zielerreichung statt. Dieser Prozess wechselwirkt wiederum mit dem Objektsystem als Istzustand, dem Zielsystem als Sollzustand und dem Ressourcensystem.

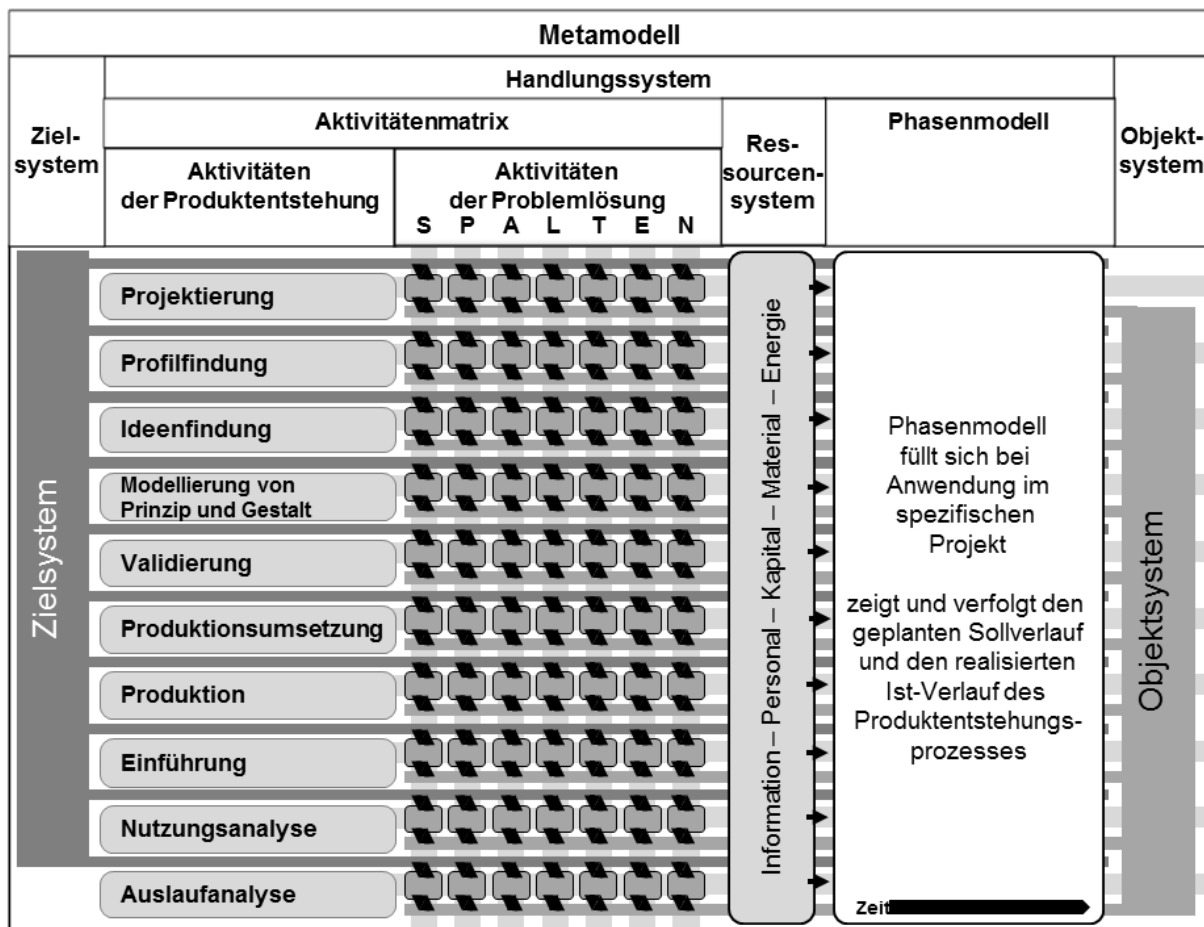


Bild 4: integriertes Produktentstehungsmodell (iPeM) nach<sup>19</sup>

<sup>19</sup> Albers / Braun 2011

<sup>20</sup> Albers et al 2005

### 2.1.2 X-in-the-Loop Framework (XiL-Framework)

Das X-in-the-Loop-Framework (XiL-Framework) nach Albers<sup>21, 22, 23</sup> stellt ein Rahmenwerk dar, mit dem es möglich ist, das Objektsystem zu unterschiedlichen Zeitpunkten und Reifegraden zu validieren. Während des Entwicklungsprozesses können Simulations- und Berechnungsmodelle, einzelne Wirkflächenpaare (WFP), Subsysteme, aber auch fertige Prototypen für die Validierung/Entwicklung zu Verfügung stehen. Mit Hilfe des XiL-Framework, siehe Bild 5, werden diese Prüflinge durch eine entsprechende Restfahrzeugsimulation zu einem Gesamtfahrzeug ergänzt. Dabei werden die einzelnen Ebenen (Wirkflächenpaare, Subsystem, Gesamtsystem), als Layer im XiL-Framework bezeichnet.

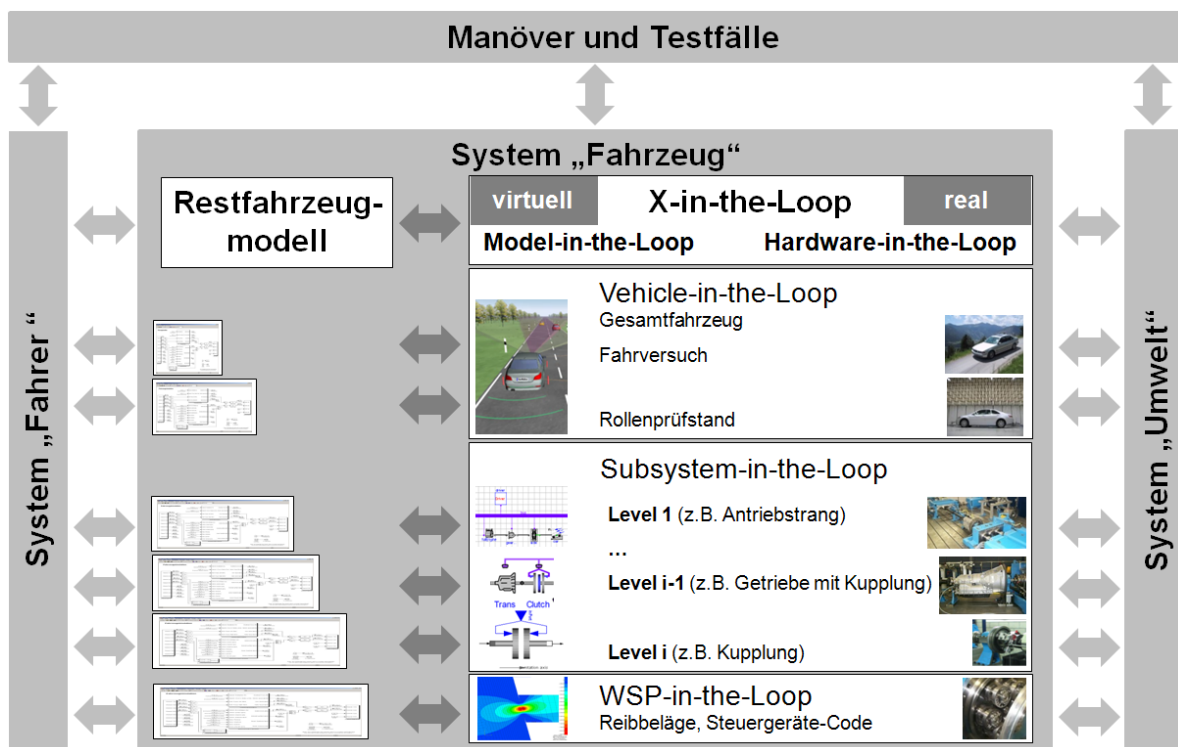


Bild 5: XiL-Framework nach Albers<sup>24</sup>

Die Restfahrzeugsimulation kann dabei aus physischen Komponenten, z.B. Prüfstandskomponenten bestehen, aber auch Simulationsmodelle beinhalten (siehe

<sup>21</sup> Albers et al 2008

<sup>22</sup> Albers et al 2009

<sup>23</sup> Albers et al. 2010c

<sup>24</sup> Albers et al 2008



Bild 6). Dies ermöglicht die Untersuchung von einzelnen Wirkflächenpaaren (WFP) über einzelne Subsysteme bis zum Fahrzeug im Gesamtsystemkontext. So können Bewertungsmanöver (vgl Kapitel 4.5), wie der NEFZ (neuer europäischer Fahrzyklus) mit einzelnen physischen Komponenten/Subsystemen durchgeführt werden. Damit ist es möglich, die Auswirkungen von Änderungen am Prüfling auf das Gesamtsystem bzw. auf die betrachteten Bewertungskriterien zu untersuchen. Außerdem werden auch die Wechselwirkungen zwischen Fahrer, Umwelt sowie dem System Fahrzeug für die jeweilige Untersuchung (Manöver und Testfälle) durch eine entsprechende Simulation berücksichtigt (siehe Bild 5).

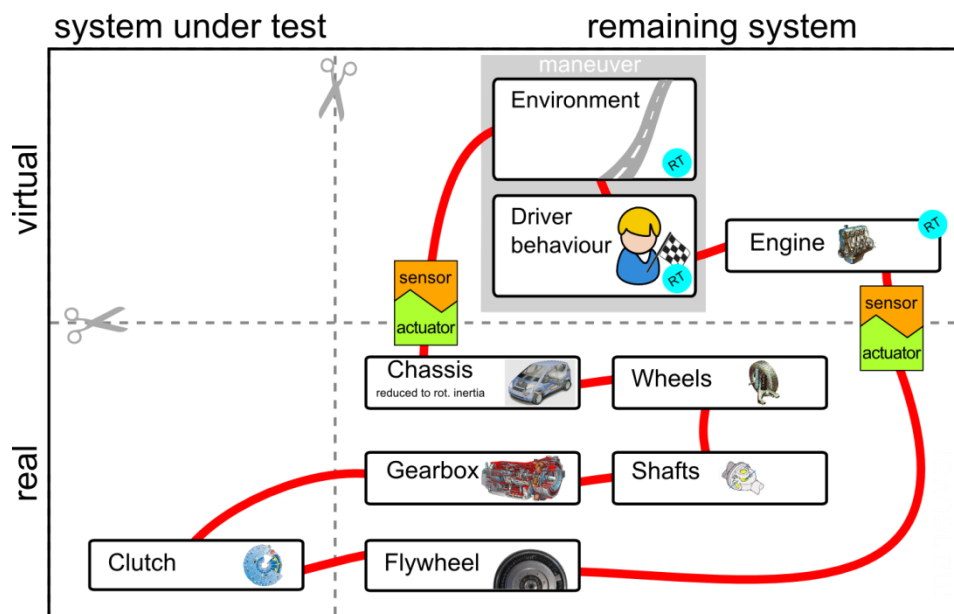


Bild 6: Einordnung in physische und virtuelle Modelle für eine Kupplung als UUD<sup>25</sup>

Dadurch können schon in frühen Phasen des Produktentstehungsprozesses schon Bewertungsmanöver bzw. Testmanöver auf Gesamtfahrzeugebene durchgeführt werden. Dies ermöglicht ein entsprechendes Vorverlagern (Frontloading), (siehe Bild 17), entsprechender Änderungen/Entscheidungen.

Neben der Möglichkeit physische Prüflinge zu betrachten, gibt es auch die Möglichkeit Simulationsmodelle mit Hilfe des XIL-Framework zu optimieren (siehe Albers<sup>26</sup>).

<sup>25</sup> Albers et al 2013

<sup>26</sup> Albers et al 2012c

## 2.2 Modelle

Von zentraler Bedeutung in dieser Arbeit ist der Modellbegriff, da selbst einfache Strategien zum versuchsbasierten Optimieren nicht ohne eine Form von Modellbildung bzw. Modellvorstellung auskommen. In dieser Arbeit wird der Modellbegriff von Stachowiak<sup>27</sup> verwendet. Nach Stachowiak<sup>27</sup> kennzeichnen sich Modelle im Allgemeinen durch drei wesentliche Merkmale. Diese sowie ihre wesentlichen Bedeutungen für diese Arbeit werden im Folgenden dargestellt:

- **Abbildungsmerkmal**  
“Modelle sind stets Modelle von etwas, nämlich Abbildungen, Repräsentationen natürlicher oder künstlicher Originale, die selbst wieder Modelle sein können.“<sup>27</sup> Eine wichtige Klasse von Modellen in dieser Arbeit sind Simulationsmodelle, die das Verhalten der dazugehörigen physischen Systeme abbilden sollen. Jedoch können auch physische Komponenten und Subsysteme diese Eigenschaft übernehmen. Bild 7 zeigt ein Beispiel dafür vom Antriebsbaugruppenprüfstand des IPEKs zur Untersuchung von Antriebssträngen vgl. Albers<sup>28</sup>. Hier wird eine Torsionswelle mit einer angeschlossenen Drehmasse zur Abbildung der Restfahrzeugsimulation (Antriebsstrangsteifigkeit und der Fahrzeugmassenträgheit) verwendet.

---

<sup>27</sup> Stachowiak 1973

<sup>28</sup> Albers et al. 2011d

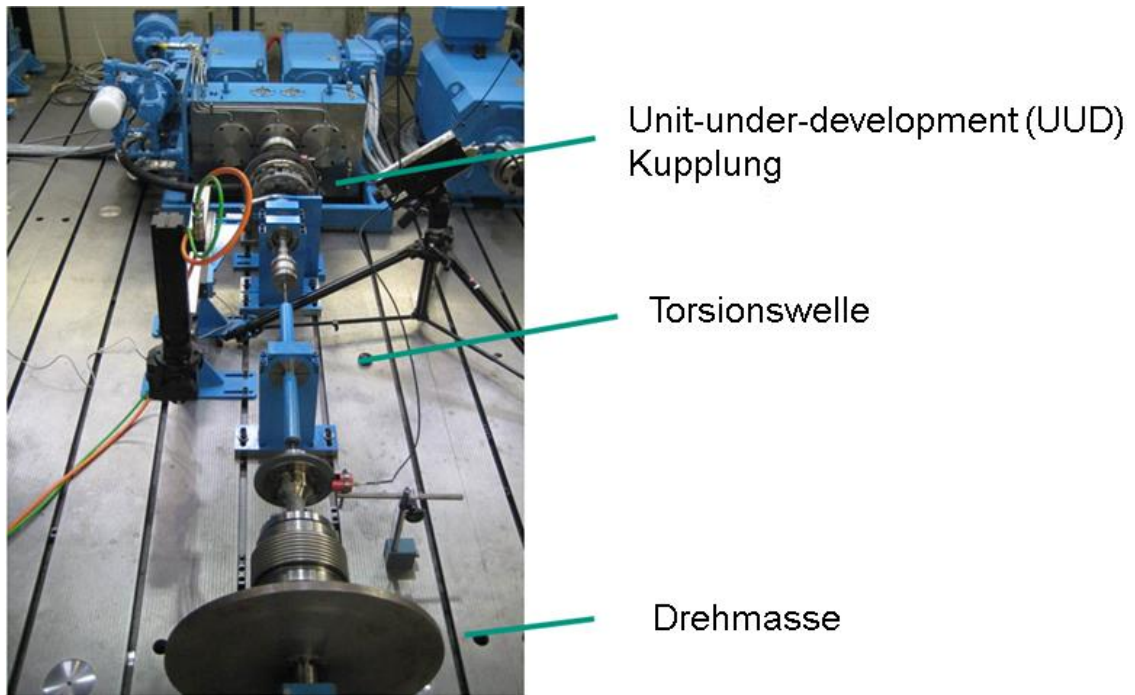


Bild 7: Antriebsbaugruppenprüfstand (Kupplungsaufbau) nach Albers<sup>29</sup>

- Verkürzungsmerkmal  
“Modelle erfassen im Allgemeinen nicht alle Attribute des durch sie repräsentierten Originals, sondern nur solche, die den jeweiligen Modellerschaffern und/oder Modellbenutzern relevant scheinen.“<sup>30</sup> Explizit bedeutet das, dass Modelle nur für die Aspekte zuverlässig verwendet werden können, für die sie erstellt und verifiziert wurden. Auf der anderen Seite müssen die Berechnungsmodelle nur die Aspekte abbilden, für die diese erstellt werden. Das reduziert den Modellbildungsaufwand, die Modellgröße sowie die Rechenzeit, welche ein wichtiges Kriterium für die Echtzeitfähigkeit darstellt.
- Pragmatisches Merkmal  
“Modelle sind ihren Originalen nicht per se eindeutig zugeordnet. Sie erfüllen ihre Ersetzungsfunktion a) für bestimmte – erkennende und/oder handelnde, modellbenutzende - Subjekte, b) innerhalb bestimmter Zeitintervalle und c) unter Einschränkung auf bestimmte gedankliche oder tatsächliche Operationen.“<sup>30</sup> Für die Klasse der Simulationsmodelle bedeutet das, dass bei der Modellnutzung auch die Grenzen der Modelle beachtet werden müssen.

---

<sup>29</sup> Albers et al. 2011d

<sup>30</sup> Stachowiak 1973

### 2.2.1 Klassifizierung von Modellen

In diesem Abschnitt soll gezielt auf die Klasse der Simulationsmodelle und deren Hauptaufgaben eingegangen werden. Dabei sind die Hauptaufgaben von Simulationsmodellen für diese Arbeit im Folgenden formuliert:

- Zur Restfahrzeugsimulation bzw. als Umweltsimulation im XiL-Kontext (siehe Kapitel 2.1.2)
- Zur Berechnung der Systemreaktion auf eine bestimmte Anregung
- Zur Optimierung der Designparameter
- Zur Abschätzung von Flussvariablen eines physischen Systems, welche messtechnisch schwer oder nicht erfassbar sind

Modelle lassen sich in eine Vielzahl von Klassen einteilen. An dieser Stelle sollen daher nur die wichtigsten Klassierungsmöglichkeiten nach Ortlieb<sup>31</sup> genannt werden:

- mathematische Struktur  
(z.B. Differenzialgleichungen, Differenzengleichungen, Transferfunktionen, Zustandsraummodelle, Polynome, Splines)
- physikalische/teilweise physikalische/nicht physikalische Modelle  
(z.B. Hammerstein-Wiener-Modell ARMAX oder ARMA<sup>32</sup>)
- geschlossene/offene Modelle
- spezifisch/allgemein
- diskret/kontinuierlich
- numerisch/analytisch
- stochastisch/deterministisch
- quantitativ/qualitativ
- makroskopisch/mikroskopisch
- induktiv/deduktiv
- statisch/dynamisch
- linear/nichtlinear
- Aufgabe (z.B. Systembeschreibung, Systemanalyse, Systemsimulation und Vorhersage.)

---

<sup>31</sup>Ortlieb 2009

<sup>32</sup>Iserman 1992

### 2.2.2 Grundbegriffe Modellstruktur, Modellparameter, Modellbildung

In diesem Abschnitt werden zunächst einige Grundbegriffe zu Modellstruktur, Modellparameter sowie Modellbildung geklärt. Simulationsmodelle können in Form von Gleichungen bzw. mathematische Beziehungen dargestellt werden. Dabei hängt die Systemantwort im Wesentlichen von der Systemanregung, das heißt den Eingangsgrößen, sowie der Modellstruktur und den Modellparametern ab. Im Gegensatz zu Eingangs-/Ausgangsgrößen, sind die Modellparameter keine Flussgrößen, das heißt sie ändern sich während der Simulation nicht. Im physischen System können sich diese Parameter jedoch, zum Beispiel durch Alterungseffekte, über die Zeit verändern. Ist dies der Fall und ist dieser Effekt hinsichtlich der Aufgabe des Modells relevant, muss dieses Verhalten durch entsprechende Flussgrößen modelliert werden. In dieser Arbeit werden Modelle zum versuchsbasierten Optimieren verwendet. Dabei werden die Parameter der Modellstruktur als Designparameter bezeichnet, die in diesem Entwicklungsschritt optimiert werden sollen. Die Anzahl der Designparameter hat einen großen Einfluss auf den Versuchsaufwand, der benötigt wird. Deshalb sollte die Anzahl dieser Designparameter so gering wie möglich aber nicht geringer gehalten werden. Üblicherweise gibt es physikalische Grenzen bzw. andere Restriktionen, in denen die Designparameterwerte gewählt werden dürfen. Der erlaubte Bereich bildet den sogenannten Designraum für die Optimierungsaufgabe.

Die Modellbildung hängt sowohl von der Art des Modells als auch von der Aufgabe des Modells ab. Zwei große Klassen von Modellen sind die physikalischen Modelle bzw. die nichtphysikalischen Modelle. Beide Formen erfordern spezielle Methoden bei der Modelbildung. Physikalische Modelle basieren auf der mathematischen Beschreibung der physikalischen Vorgänge des betrachteten Systems. Diese werden nach Schlecht<sup>33</sup> häufig in einem deduktiven Verfahren aus einzelnen physikalischen Teilmodellen/Effekten erstellt. Dressig<sup>34</sup> und Bub<sup>35</sup> stellen neben anderen eine Systematik für diese Art von Modellbildung bereit (siehe Bild 8/Bild 9).

---

<sup>33</sup> Schlecht 2006

<sup>34</sup> Dressig 2006

<sup>35</sup> Bub 1992

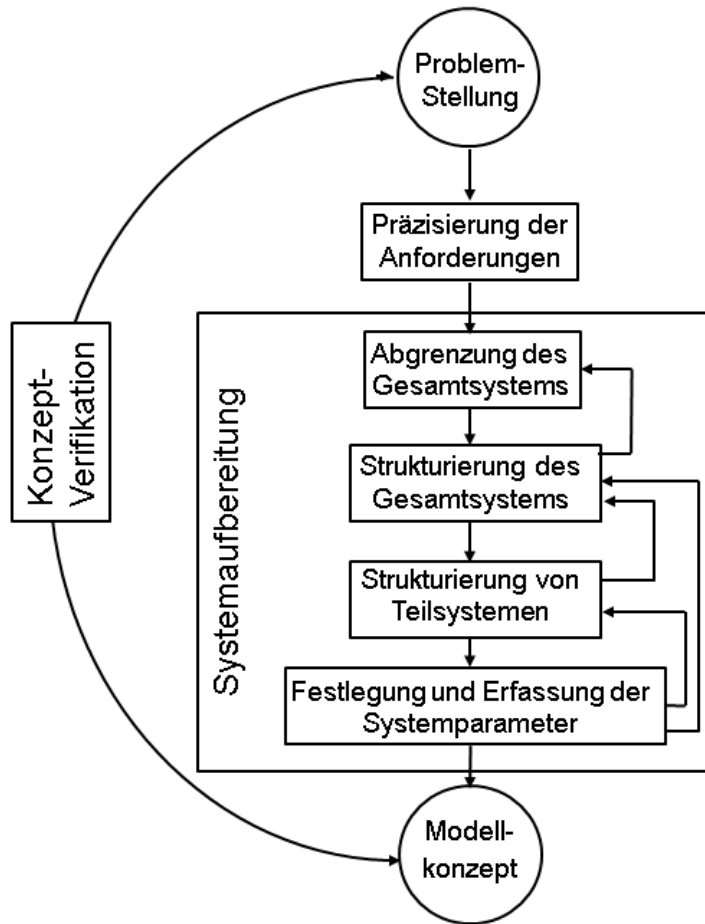


Bild 8: Systematik der Modellbildung nach Bub<sup>36</sup>

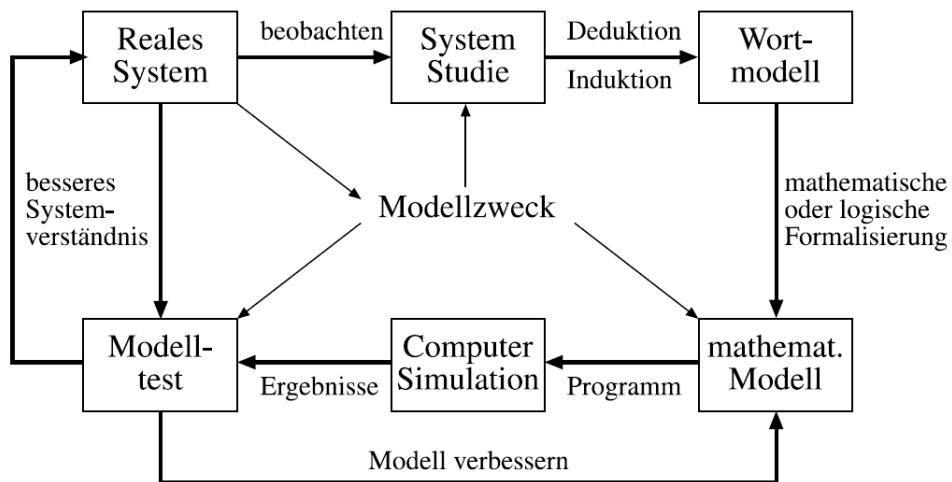


Bild 9: Modellbildungsprozess nach Dresig<sup>37</sup>

<sup>36</sup> Bub 1992

<sup>37</sup> Dresig 2006

Ein andere Klasse von Modellen sind die Modelle die nicht auf einer physikalischen Modelstruktur basieren. Diese Modelle haben unabhängig von dem physischen Prozess, den sie abbilden, eine feste Modellstruktur. Die Parameter der Struktur werden dabei mit Hilfe von Versuchsdaten berechnet. Dabei lassen die Parameter keinen direkten Rückschluss auf physikalische Eigenschaften der Wirkflächenpaare und Leitstützstrukturen zu. Beispiele für definierte nichtphysikalische Modelle sind Wiener/ Hammersteinmodelle, künstliche neuronale Netze oder polynomiale Modelle, die häufig zur Beschreibung des menschlichen Komfortempfindens oder für den Reglerentwurf verwendet werden. Außerdem findet diese Klasse Anwendung in der DOE-Methode nach Taguchi (siehe Kapitel 2.4). Eine Sammlung der häufigsten Modellstruktur sowie den dazugehörigen Verfahren liefern Iserman<sup>38, 39</sup> und Münz<sup>40</sup> (siehe Bild 10).

Eingangssignal	Model	Ausgangssignal	Identifik. Methode	Mess/ Ausgangswert/Gerät	zuläss. Stör-signal	Dig. R. Kopplg.				Erreichbare Genauigkeit	Erweit.-barkeit			Anwendungsbeispiel
						Offl.	Onl.	Block	Echtz.		ZVS	MGS	NLS	
			Kennwert-ermittlung	Schreiber	sehr klein	-	-	-	-	klein	-	-	-	- Grobes Modell - Reglereinstellung
			Fourier Analyse	Schreiber	klein	x	-	x	-	mittel	-	x	-	- Überprüfung - theor. Modelle
			Frequenz-gang-messung	Schreiber F.G. Mess-Gerät	mittel	-	-	-	-	sehr gross	-	x	-	- Überprüfung - theor. Modelle - Entwurf Klass. Regler
			Korrela-tion	Korrelator Prozess-rechner	gross	-	-	-	x	gross	x	x	x	Erkennung Signalzus.hänge Laufzeitidentifikati-on
			Modell-Abgleich	Analogr.	gross	-	-	-	x	mittel	x	-	-	Analoge adapt. Regelung
			Param. Schätz. G.	Prozessr.			x	x	x	x	gross	x	x	x

Bild 10: Modellarten/Identifikationsverfahren<sup>41</sup>

<sup>38</sup> Iserman 1992

<sup>39</sup> Iserman 2006

<sup>40</sup> Münz 2009

<sup>41</sup> Iserman 1992

## 2.3 Optimierung/Optimierungsverfahren

Nach Schneider<sup>42</sup> ist das Ziel einer Optimierung das Berechnen optimaler Designparameterwerte hinsichtlich einer Zielfunktion bei gegebenen Randbedingungen. Dabei drückt das Optimum das minimal oder maximal erreichbare Ergebnis im Sinne eines Kompromisses zwischen den verschiedenen Bewertungskriterien aus. Nach Schneider<sup>42</sup> gibt es folgende Arten von Optimierungsaufgaben:

- **Parameteroptimierung**  
Bei der Parameteroptimierung wird die Einstellung der Designvariablen z.B. die optimalen Getriebeübersetzungen gesucht, bei der eine Zielfunktion z.B. der Verbrauch in einem gegeben Zyklus optimal wird.
- **Reihenfolgenoptimierung**  
Bei der Reihenfolgenoptimierung wird die optimale Reihenfolge von Elementen einer Grundmenge optimiert. Bekannte Unterprobleme sind die Frage des kürzesten Wegs von einem Start zu einem Zielpunkt oder die Frage nach dem kürzesten Rundweg, der alle Punkte verbindet (travelling-salesmen-problem). Ein technisches Beispiel ist die Berechnung der optimalen Abfolge mit der ein computergesteuerter Bohrer unregelmäßig verteilte Löcher in eine gegebene Leiterplatte Platte, siehe Bild 11, bohrt.

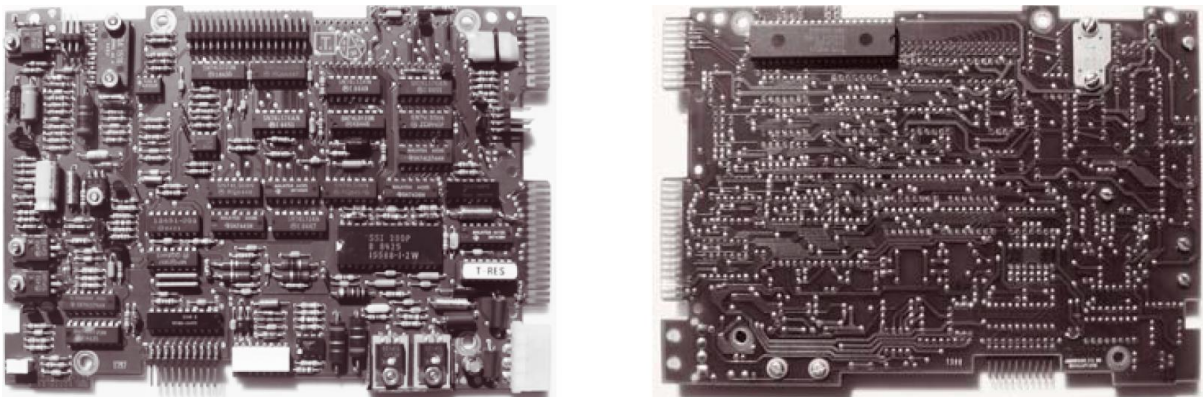


Bild 11: Beispiel Leiterplatte<sup>43</sup>

<sup>42</sup> Schneider 2001

<sup>43</sup> Hußmann 2007



- Auswahloptimierung

Bei der Auswahloptimierung muss die optimale Möglichkeit aus einer Grundgesamtheit ausgewählt werden. Zum Beispiel die Bestimmung des optimalen Gangs im aktuellen Fahrzustand.

In dieser Arbeit sind jedoch nur die ersten beiden Aufgabenstellungen relevant, da nur diese für die Aufgabenstellung der versuchsbasierten Optimierung einsetzbar sind. Für diese Arten von Optimierungsaufgaben gibt es wiederum verschiedene Optimierungsverfahren die nach folgenden Eigenschaften klassifiziert werden können nach<sup>44</sup>:

- numerisch/analytisch

Teilweise können Problemstellungen, die in Form von Gleichungen beschrieben werden können, noch analytisch gelöst werden. Bei komplexeren Systemen müssen Lösungen oft numerisch gefunden werden. Das heißt, durch Einsetzen von mehreren Punkten wird versucht, das Optimum möglichst gut zu bestimmen.

- Stochastisch/deterministische Verfahren

Bei stochastischen Verfahren werden Zufallsprozesse zur Optimierung verwendet. Ein Vertreter ist die Monte-Carlo-Methode, bei der zufällige Punkte getestet werden. Bei deterministischen Verfahren hängen die getesteten Punkte nur vom System und dem Startvektor ab. Der Optimierungspfad ist auch bei mehrmaliger Ausführung immer derselbe.

- diskret/kontinuierlich

Bei manchen Optimierungsaufgaben sind nur bestimmte diskrete Werte für die Parameter, wie zum Beispiel ganzzahlige Lösungen, zugelassen. In diesem Fall handelt es sich um einen diskreten Parameterraum. Für diese Art von Optimierungsaufgaben gibt es entsprechende diskrete Optimierungsverfahren.

- statisch/dynamisch

Bei dynamischer Optimierung ändert sich das Optimum mit der Zeit, d.h. es ist zeitvariant. Ziel der dynamischen Optimierung ist eine Steuerfolge die das System im gewünschten Zustand hält. Bei einer statischen Optimierung ist das Optimum zeitinvariant.

---

<sup>44</sup> Sauter 2000

### 2.3.1 Parameteroptimierung

Es gibt eine Vielzahl von Optimierungsalgorithmen. Dabei hängt die Wahl des Optimierungsalgorithmus von der Aufgabe ab. Im Folgenden wird auf die für diese Arbeit relevanten Algorithmen eingegangen.

#### 2.3.1.1 Axial-Iteration

Bei der Axial-Iteration nach Schneider<sup>45</sup>, (siehe Bild 12) testet der Algorithmus solange Punkte in einer achsparallelen Richtung bis er auf einen schlechteren Wert trifft. Dann wechselt er die Richtung. Dieser Algorithmus ist nicht in der Lage sich von lokalen Optima zu entfernen. Deshalb spielt der Startpunkt bei diesem Verfahren eine wichtige Rolle.

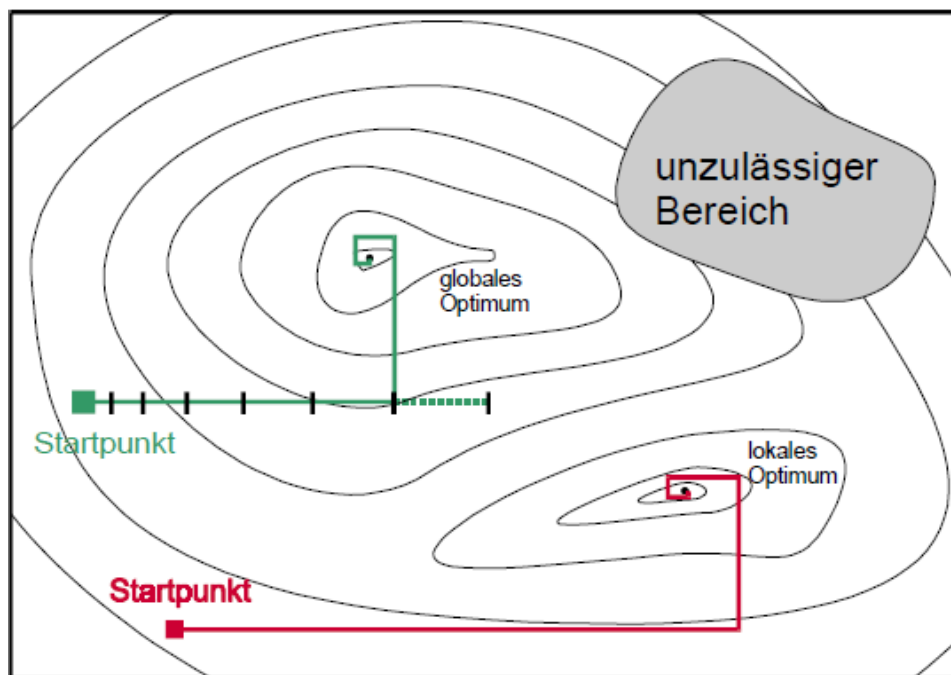


Bild 12: Axial-Iteration<sup>46</sup>

<sup>45</sup> Schneider 2001

<sup>46</sup> Schneider 2001

### 2.3.1.2 Evolutionäre bzw. genetische Optimierungsverfahren

Die evolutionären bzw. genetischen Optimierungsverfahren sind sehr verbreitete Optimierungsverfahren. Dabei nutzen diese Analogien aus der Evolutionstheorie als Vorbild (siehe Naumann<sup>47</sup> Duddeck<sup>48</sup> Furch<sup>49</sup>). Im Allgemeinen werden immer wieder Gruppen (Generationen) von Punkten (im Designraum) generiert. Für diese Designparametereinstellungen wird die Zielfunktion ausgewertet. Anschließend werden die besten Punkte (Individuen) dieser Generation für die Bestimmung der Punkte der nächsten Generation verwendet. Ziel ist es eine sich ständig verbessernde Population zu erreichen.

Nach Schneider<sup>46</sup> ist die einfachste Form einer Evolutionsstrategie (ES) die zweigliedrige (1+1)-ES. Hier erzeugt jeder Punkt (Individuum) einen mutierten Nachkommen. Dabei bildet der Bessere der beiden das Ausgangsindividuum für die nächste Generation. Bei einer  $(\mu + \lambda)$ -ES erzeugen  $\mu$  Eltern  $\lambda$  Nachkommen wobei nur die Besten  $\mu$  die nächsten Generationen mutierte Nachkommen erzeugen.

Eine Variation dieser Strategie ist die  $(\mu, \lambda)$ -ES. Hier überleben nur die besten Nachkommen. Damit wird dem entgegengewirkt, dass sich die Gruppe in lokalen Optima festläuft. Eine weitere Variation dieser Strategie ist die  $(\mu/\rho \# \lambda)$ -ES. Diese Strategie hat die Möglichkeit zur Rekombination der Gene. Dabei wird ein Wert des Designparametervektors als Gen bezeichnet. Bei dieser Strategie werden die Gene der Eltern addiert und durch Anzahl der Werte  $\rho$  geteilt. Eine andere Möglichkeit ist die zufällige Auswahl der Werte der Eltern.

Dabei ist die Mutation von einzelnen Individuen die zufällige Verschiebung des Punktes. Dazu wird auf den ursprünglichen Punkt eine normalverteilte Zufallsvariable mit dem Mittelwert Null und einer definierten Standardabweichung  $\sigma$  addiert.

Eine Möglichkeit ist dabei auch diese Standardabweichung zu mutieren, sodass die definierte Standardabweichung nicht statisch bleibt.

Schneider<sup>50</sup> empfiehlt diese Methode für un stetige bzw. multimodale Zielfunktionen, da sich die evolutionären Algorithmen mit einer Mutation der Standardabweichung weitestgehend nicht in lokalen Optima festlaufen.

---

<sup>47</sup> Naumann 2002

<sup>48</sup> Duddeck 2005

<sup>49</sup> Furch 2005

<sup>50</sup> Schneider 2001

### 2.3.1.3 Monte-Carlo-Methode

Bei der Monte-Carlo-Methode handelt es sich um eines der einfachsten stochastischen Optimierungsverfahren. Die Basisstrategie dieses Verfahrens ist eine große Anzahl von Punkten, die zufällig im Designraum gewählt werden, zu untersuchen Duddeck<sup>51</sup>. Der beste Punkt ist dann das Optimum für den Algorithmus.

Da der Algorithmus die Zielfunktion in keiner Weise nachmodelliert (indem bereits untersuchte Punkte mit einbezogen werden), kann dieses Verfahren nicht konvergieren. Deshalb muss eine hohe Anzahl an Punkten untersucht werden, damit der Satz an gezogenen Punkten repräsentativ ist und ein guter Punkt gefunden wird. Dadurch ist vor allem bei vielen Designparametern eine hohe Rechenzeit notwendig.

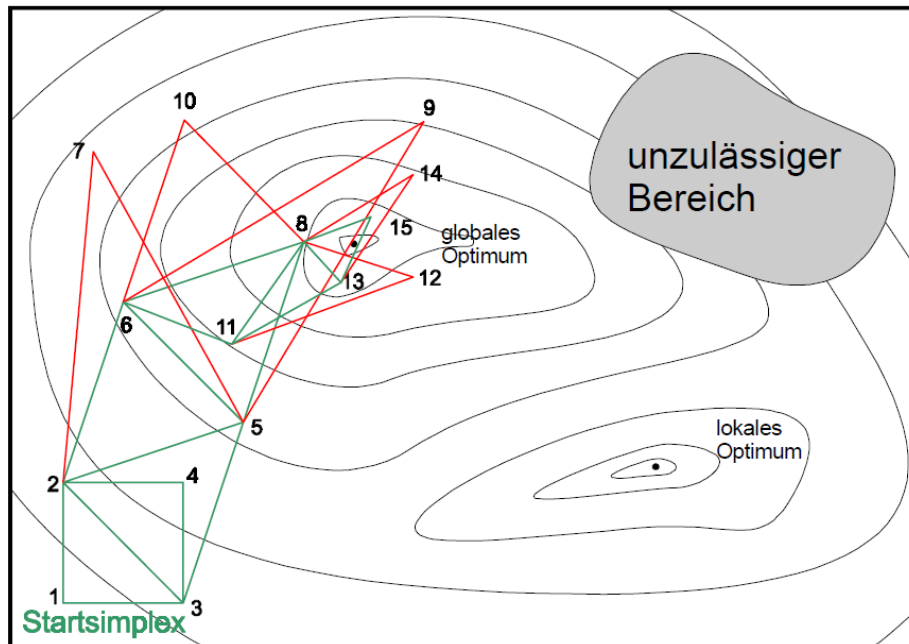
Eine Möglichkeit ist das Optimum dieses Verfahren als Startpunkt für ein anders Verfahren zu verwenden. Von diesem guten Startpunkt kann dann ein anderes Verfahren (z. B. ein gradienten-basiertes Verfahren) das Optimum genauer bestimmen.

### 2.3.1.4 Simplex Optimierungsstrategie

In diesem Kontext hat die Simplex Optimierungsstrategie nichts mit der Simplexrechnung der linearen Programmierung zu tun. Bei der Simplex Optimierungsstrategie werden Punkte in Form einer geometrischen Figur (Polygons) im Versuchsraum platziert und ausgewertet. Dabei ist die kleinste mögliche Figur ein Dreieck. In einem Iterationsschritt wird jeweils der schlechteste Punkt an der Verbindungsgeraden der verbleibenden Eckpunkte gespiegelt, um einen besseren Punkt zu erhalten. Dieser Punkt ist dann der neue Eckpunkt anstelle des schlechtesten. Ist dieser Punkt schlechter, wird der zweit schlechteste Punkt gespiegelt. Ist der Folgepunkt besser wird der Eckpunkt noch mal in diese Richtung verschoben. So passt sich die geometrische Figur (Simplex) über die Iterationen an die Zielfunktion an (siehe Bild 13).

---

<sup>51</sup> Duddeck 2005

Bild 13: Simplex Optimierungsstrategie<sup>52</sup>

### 2.3.1.5 Simulated Annealing

Beim Simulated Annealing nach Scheider<sup>52</sup> wird der Abkühlungsprozess von Metallen nachempfunden, da hier das System immer einen Zustand geringer Gesamtenergie einnimmt. Analog zu den Atomen werden Punkte im Designraum bewegt. Atome mit hoher Temperatur sind dabei nicht so ortsgebunden wie Atome mit tieferen Temperaturen und erfahren Stöße aus allen möglichen Richtungen. Über diese Analogie bewegen sich Punkte entlang des größten Gradienten Richtung des nächsten lokalen Optimums, werden jedoch zusätzlich durch die Stöße in Form von Zufallsvektoren beeinflusst. Die Idee ist, dass diese Stöße die Punkte wieder aus kleinen lokalen Minima herausbefördern und somit das globale Optimum gefunden wird. Die Stöße nehmen dabei mit sinkender Temperatur ab.

<sup>52</sup> Schneider 2001

## 2.3.1.6 Schätzer

Häufig wird ein Trend in Kalkulationen für Prognosen oder Ausgleichskurven durch streuungsbehaftete Messungen gesucht. Für diese Aufgaben werden häufig Schätzverfahren verwendet.

Das bekannteste Schätzverfahren ist der Least-Squares (kleinsten Fehlerquadrate) Schätzer. Die Aufgabe des Schätzers ist es, die "besten" Parameterwerte eines Parametervektors  $\vec{p}$  für ein Modell (häufig eine Gerade) zu finden. Dazu wird die Modellkurve zunächst allgemein definiert. Für eine Gerade ergibt sich folgende Gleichung:

$$\vec{y} + \vec{\epsilon}_a = p_0 + p_1 \cdot \vec{x} \quad (2.3.1)$$

Dabei steht  $\vec{\epsilon}_a$  für die Abweichung (Residuen) der einzelnen Punkte von dem Modell. Im nächsten Schritt wird diese Gleichung in folgende Matrixform gebracht:

$$\vec{y} + \vec{\epsilon}_a = \begin{pmatrix} 1 & x_1 \\ \vdots & \vdots \end{pmatrix} \vec{p} \quad (2.3.2)$$

Im Anschluss wird dieser Ausdruck nach  $\vec{\epsilon}_a$  aufgelöst und um die Summe der Fehlerquadrate zu erhalten mit  $\vec{\epsilon}_a^T$  multipliziert:

$$\sum_{k=1}^{ke} \epsilon_{a,k} = \vec{\epsilon}_a^T \cdot \vec{\epsilon}_a = \begin{pmatrix} 1 & x_1 \\ \vdots & \vdots \end{pmatrix} \vec{p} = \vec{H} \cdot \vec{p} \quad (2.3.3)$$

Dieser Ausdruck soll durch das Least-Squares Verfahren minimiert werden. Das heißt, es soll der Parametervektor bestimmt werden, der die kleinste Summe der Fehlerquadrate ergibt. Dazu wird der Ausdruck  $\vec{\epsilon}_a^T \vec{\epsilon}_a$  nach  $\vec{p}$  abgeleitet und anschließend mit Null gleichgesetzt. Im allgemeinen Fall ergibt das ein Gleichungssystem. Bei Modellen die linear in den Parameter sind, ist es jedoch möglich, direkt nach  $\vec{p}$  aufzulösen. Im Fall der Geraden ergibt sich folgende Gleichung:

$$\vec{p}_{optimal} = (\vec{H}^T \vec{H})^{-1} \vec{H} \vec{y} \quad (2.3.4)$$

Neben dem klassischen Least-Squares-Schätzer gibt es noch Modifikationen die aber nach demselben Prinzip arbeiten. Ein Beispiel ist der weighted (gewichteter) Least-Squares-Schätzer (siehe Isermann<sup>53</sup>). Hier wird der Fehler der einzelnen Messpunkte gewichtet. Dadurch können Punkte, die (z. B. durch Messunsicherheiten) nicht so vertrauenswürdig sind, weniger gewichtet werden.

Wenn eine die Messung mit Ausreißern behaftet ist, kann die Ausgleichskurve stark verzerrt werden, wenn die Ausreißer weit von der richtigen Kurve entfernt sind. Deshalb senkt der iterated Least-Squares Schätzer die Gewichte von weit entfernten Punkten. So werden Ausreiser in der nächsten Iteration nicht mehr so hoch gewichtet. Der rechnerische Aufwand steigt dadurch mit der Anzahl der verwendeten Iterationen.

Der Ransac-Schätzer Isermann<sup>54</sup> verfolgt ein anders Prinzip. Hier wird nach dem Zufallsprinzip genau die Teilmenge der möglichen Messpunkte gezogen, sodass die Parameter genau bestimmt werden können. Dann wird berechnet, wie viele Punkte in einem gewissen Abstand zu dieser Ausgleichskurve liegen. Diese Schritte werden mehrfach wiederholt. Die Schätzung mit den meisten Punkten innerhalb dieses Bereiches ist dann das Ergebnis. Teilweise wird zum Schluss noch mit allen Punkten, die innerhalb des Bereiches liegen, eine Least-Square Schätzung durchgeführt, um das Ergebnis weiter zu verbessern. Dieser Schätzer ist sehr robust gegen eine hohe Zahl an Ausreißern, jedoch ist dessen Ausführung auch mit einem hohen Rechenaufwand verbunden.

---

<sup>53</sup> Isermann 1992

<sup>54</sup> Isermann 1992

### 2.3.2 Reihenfolgenoptimierung

Nach Schneider<sup>55</sup> ist die Reihenfolgenoptimierung ein Problem der diskreten Mathematik. Wobei es hier mehrere verschiedene Aufgabenstellungen gibt:

- Das Finden der kürzesten Wegstrecke von einem Knoten zu einem anderen bei einem gegeben Netz
- Das sogenannte Problem des Handlungsreisenden (traveling-salesman), der jeden Knoten im Netz einmal besuchen muss und dann zum Ausgangspunkt zurück muss, und dafür den kürzesten Gesamtweg sucht

Im Fokus dieser Arbeit steht dabei nur eine Form des Problems des Handlungsreisenden. Eine Möglichkeit dieses Problem zu lösen ist die vollständige Enumeration. Hier werden alle möglichen Reihenfolgen berechnet. Für eine kleinere Anzahl von Knoten ist das rechnergestützt möglich. Dabei werden in der Kombinatorik die Orte des Handlungsreisenden als Knoten und die Verbindungslinien als Graphen bezeichnet. Für eine größere Anzahl steigt der Aufwand stark an (fakultativ). Dabei existieren  $(n - 1)!$  mögliche Reihenfolgen, die für  $n$  Knoten berechnet werden müssen. Bei symmetrischen Problemen ist die Anzahl aller Möglichkeiten  $\frac{1}{2}(n - 1)!$ , steigt jedoch ebenfalls fakultativ. Bei einem symmetrischen Problem sind die Graphen ungerichtet, das heißt, dass der Graph zwischen zwei Knoten in beide Richtungen den gleichen Wert hat (siehe Hußmann<sup>56</sup>).

Ausgangslage für die kombinatorische Optimierung ist eine sogenannte Kostenmatrix, in der die Kosten für die Reise von jedem Knoten zu jedem anderen abgelegt sind (siehe Bild 14). Um Kombinationen auszuschließen, die unerwünschte Graphen (Reiseschritte) enthalten, werden diese Positionen in der Matrix mit sehr hohen Werten belegt. Beispiele sind die Graphen, die von einem Knoten zu sich selbst führen.

---

<sup>55</sup> Schneider 2001

<sup>56</sup> Hußmann 2007



Kosten		Nach Knoten				
		1	2	3	4	5
Von Knoten	1	$\infty$	$\Delta t_{1 \rightarrow 2}$	$\Delta t_{1 \rightarrow 3}$	$\Delta t_{1 \rightarrow 4}$	$\Delta t_{1 \rightarrow 5}$
	2	$\Delta t_{2 \rightarrow 1}$	$\infty$	$\Delta t_{2 \rightarrow 3}$	$\Delta t_{2 \rightarrow 4}$	$\Delta t_{2 \rightarrow 5}$
	3	$\Delta t_{3 \rightarrow 1}$	$\Delta t_{3 \rightarrow 2}$	$\infty$	$\Delta t_{3 \rightarrow 4}$	$\Delta t_{3 \rightarrow 5}$
	4	$\Delta t_{4 \rightarrow 1}$	$\Delta t_{4 \rightarrow 2}$	$\Delta t_{4 \rightarrow 3}$	$\infty$	$\Delta t_{4 \rightarrow 5}$
	5	$\Delta t_{5 \rightarrow 1}$	$\Delta t_{5 \rightarrow 2}$	$\Delta t_{5 \rightarrow 3}$	$\Delta t_{5 \rightarrow 4}$	$\infty$

Bild 14: Kottenmatrix

Bei vielen realen Aufgabenstellungen existieren mehrere (gleich gute) optimale Lösungen. Wird nur eine gute bzw. eine optimale Lösung benötigt, werden Heuristiken verwendet (siehe folgende Kapitel) um diese zu finden.

### 2.3.2.1 Nächster Nachbar Heuristik (next-neighbor- oder NN-Heuristik)

Nach Hußmann<sup>57</sup> wird bei der NN-Heuristik zunächst ein Startknoten gewählt. Dann wird der nächstliegende unverbundene Knoten ausgewählt. Das wird solange durchgeführt bis die Route komplett ist. Dieser Algorithmus ist sehr schnell, da die Anzahl der notwendigen Operationen nur quadratisch mit der Knotenanzahl steigt, jedoch ist dieser Algorithmus auch sehr ungenau.

### 2.3.2.2 Doppelter Nächster-Nachbar-Heuristik

Diese Heuristik ist ähnlich zu der NN-Heuristik vgl. Hußmann<sup>57</sup>. Jedoch wird nach der Markierung der ersten Kante immer an beiden Enden nach dem nächsten Nachbarn gesucht und entsprechend mit dem Graph verbunden.

### 2.3.2.3 Minimalaufspannende Baum Heuristik (minimum spanning tree- oder MST-Heuristik) nach Hußmann<sup>58</sup>

Bei dieser Heuristik muss zunächst ein sogenannter minimalaufspannender Baum (siehe Bild 15a) konstruiert werden. Dazu gibt es mehrere Algorithmen, welche

<sup>57</sup> Hußmann 2007

<sup>58</sup> Hußmann 2007

dasselbe Ergebnis liefern. Bei einer Methode werden solange die kürzesten Kanten markiert, die keinen Kreis bilden, bis alle Punkte verbunden sind. Der sich bildende Graph ist der minimalaufspannende Baum.

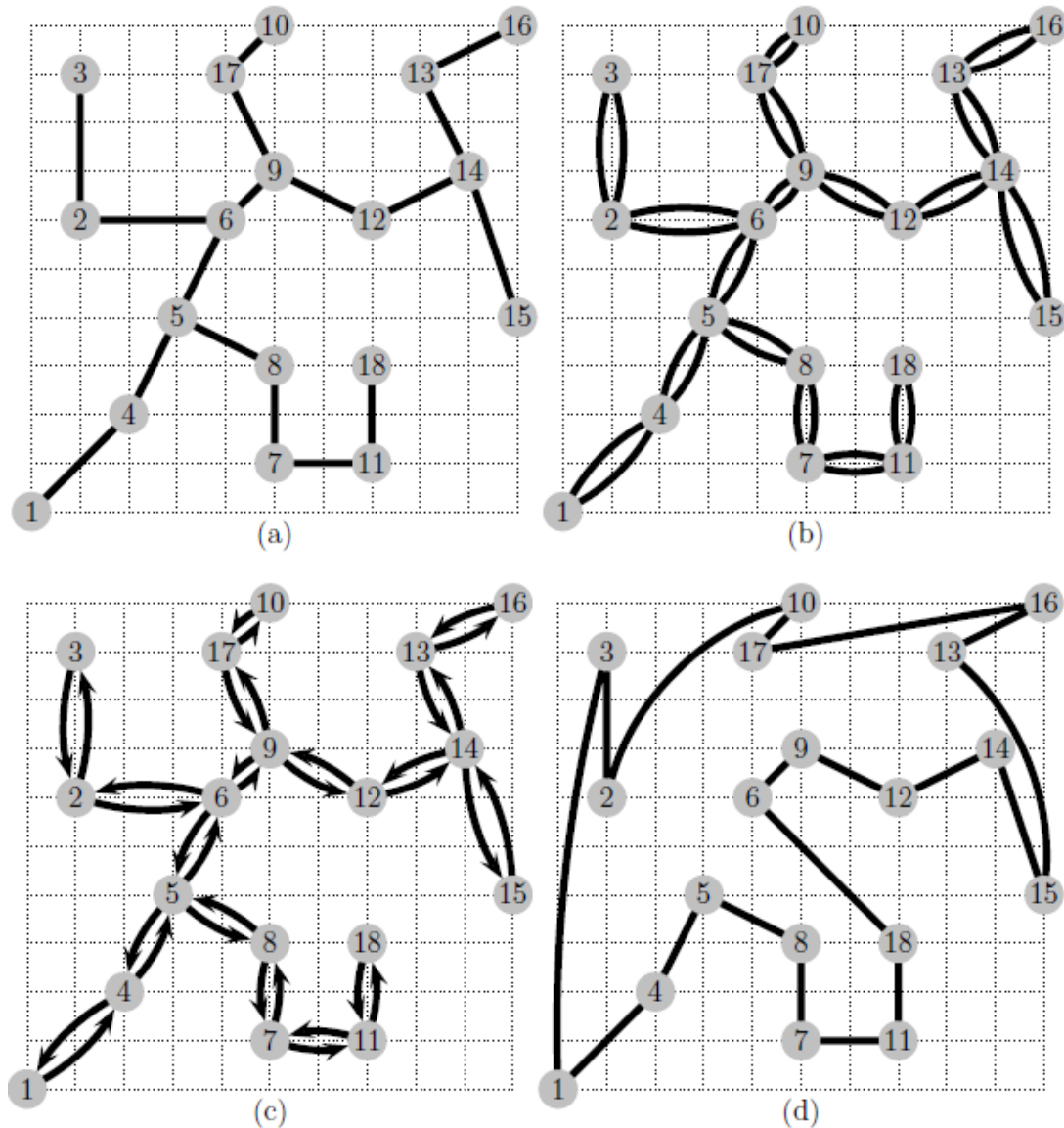


Bild 15: Beispiel Minimalaufspannender Baum (MST) Heuristik nach Hußmann<sup>58</sup>

Bei der MST- Heuristik wird, ausgehend von einem Startknoten, die sogenannte Eulertour (siehe Bild 15 b,c) abgelaufen. Das heißt, der Algorithmus umrundet den minimalen aufspannenden Baum. Jedoch werden bereits besuchte Knoten weggelassen und die direkte Verbindung zwischen den restlichen Knoten eingefügt (siehe Bild 15 d). Für einen Spezialfall des Problems, nämlich wenn die Dreiecksungleichung erfüllt wird, kann die Güte der Lösung abgeschätzt werden. Demnach kann die Gesamtlänge dieser Reise maximal doppelt so hoch sein, wie die

optimale Rundreise. Die Dreiecksungleichung gilt dabei, wenn der direkte Weg von einem Knoten nicht länger ist als der über einen anderen Knoten.

#### 2.3.2.4 Christofides-Heuristik

Nach Hußmann<sup>59</sup> ist die Christofides-Heuristik eine Erweiterung der MST-Heuristik. Jedoch wird hier eine Kombination aus dem minimalaufspannenden Baum und dem sogenannten Perfekt Matching als Grundlage für die Eulertour verwendet. Das heißt, im ersten Schritt wird wieder ein minimalaufspannender Baum erzeugt. Dann werden die geraden Knoten (Knoten mit einer geraden Anzahl von Verbindungen) markiert. Anschließend werden aus den restlichen Knoten wieder die kürzesten Kanten gesucht, die nicht im ursprünglichen minimalaufspannenden Baum enthalten sind. Diese Kanten werden dann zu dem minimalaufspannenden Baum hinzugefügt. Anschließend wird auf diesem Graph die Eulertour erzeugt.

Das Interessante an dieser Heuristik ist, dass diese eine Gütegarantie von 50% hat. Das heißt, im schlechtesten Fall, bestimmt man einen Weg der 50% länger als der optimale Weg ist.

---

<sup>59</sup> Hußmann 2007

## 2.4 Statistische Versuchsplanung - Design of Experiments (DOE)

Eine der Hauptaufgaben der statistischen Versuchsplanung ist die versuchsbasierte Bestimmung der optimalen Werte der Designparameter hinsichtlich der betrachteten Verlustfunktion. Dabei können die Designparameter sowohl Eigenschaften des Produkts als auch Verfahrensgrößen in der Produktion sein.

Ein wesentliches Ziel bei der statistischen Versuchsplanung ist es den Versuchsaufwand möglichst gering zu halten. Hier ist der DOE-Ansatz nach Taguchi/Shainin gegenüber anderen Ansätzen, wie zum Beispiel den folgenden Methoden zur versuchsbasierten Optimierung, stark im Vorteil (siehe Klein<sup>60</sup>, Siebertz<sup>61</sup>):

- one factor at a time (OFAT)  
Hier wird immer ein Designparameter zwischen den Versuchen verändert.
- vollfaktorielle/teilmfaktorielle Versuchspläne  
Beim vollfaktoriellen Versuch wird zunächst ausgewählt wie viele Faktorstufen d.h. wie viele Variationsmöglichkeiten für jeden Designparameter getestet werden sollen. Anschließend werden alle Kombinationsmöglichkeiten im Versuch erfasst. Auf Grund des hohen Aufwands von vollfaktoriellen Versuchsplänen werden beim teilfaktoriellen Versuch nicht alle Variationen des vollfaktoriellen Versuchsplan durchgeführt.
- Box Behnken Design  
Mit dem Box-Behnken-Design wird versucht den Versuchsaufwand von  $3^n$  Faktorenversuchen, das heißt Versuche mit 3 möglichen Faktorstufen pro Designvariablen, zu reduzieren. Bild 16 links zeigt die Punkte Box Behnken Designs für die beiden Designparameter A und B. Bild 16 rechts zeigt die Anordnung der Versuchspunkte für 3 Designparameter (A, B, C). Bei diesem Design werden nur die Punkte verwendet die auf einem Kreis bzw. einer Sphärenoberfläche liegen. Das heißt, dass die Ecken werden nicht verwendet werden. So werden in den beiden Beispielen 5 statt 9 bzw. 7 statt 27 Versuche benötigt.

---

<sup>60</sup> Klein 2004

<sup>61</sup> Siebertz 2010

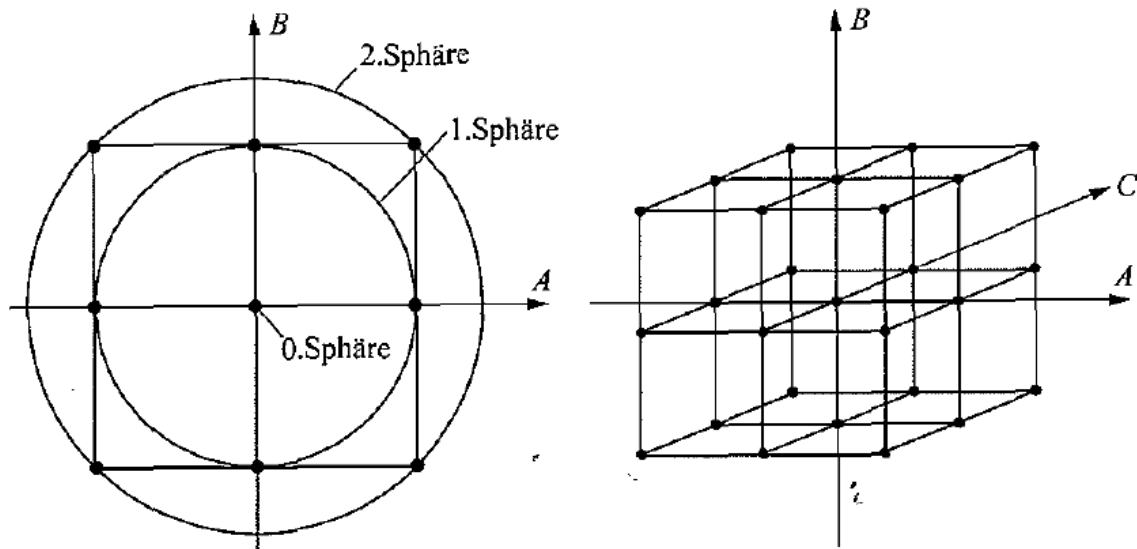


Bild 16: Anordnung Versuchspunkte bei Box Behnken Design<sup>62</sup>

- Zentral zusammengesetzte Versuchspläne – Central Composite Design (CCD)  
Bei  $2^n$  und  $2^{n-p}$  Faktorenversuchen wird der Zentralpunkt zwischen den beiden Variationsstufen beim zentral zusammengesetzten Versuchsplan dazugenommen. Damit können Aussagen über Nichtlinearitäten bzw. Streuungen bei der Annahme eines linearen Zusammenhangs getroffen werden.
- Latin Hypercube Sampling (LHS)  
Beim Latin Hypercube Sampling werden nacheinander zufällige Punkte im Versuchsraum gezogen. Jedoch wird nach jeder Ziehung ein gewisser Bereich um den Versuchspunkt gesperrt. Damit wird verhindert, dass viele Versuchspunkte nahe bei einander sind (siehe Siebertz<sup>63</sup>).

Dieser Vorteil wird mit einer höheren Komplexität bei der Versuchsgestaltung bzw. bei der Auswertung erkaufte. Da in dieser Arbeit besonderes Augenmerk auf den Versuchsaufwand gelegt wird, stellt die DOE-Methode nach Taguchi den relevanten Stand der Forschung dar. Andere Ziele der DOE-Methoden nach Taguchi/Shainin sind nach Klein<sup>62</sup>:

- Identifizierung der Haupteinflussgrößen

<sup>62</sup> Klein 2004

<sup>63</sup> Siebertz 2010

- Steigerung der Qualität
- Steigerung der Zuverlässigkeit und Nachvollziehbarkeit

Bei der DOE-Methode nach Taguchi/Shainin werden im Versuch verschiedene Designparametervariationen durchgeführt. Dabei wird für jede Variation das bzw. die Bewertungsmanöver durchgeführt. Anschließend wird anhand der Versuchsdaten eine Modellstruktur parametrisiert. Diese wird dann verwendet um die optimalen Designparameterwerte zu berechnen (Optimierungsaktivität). Danach werden im System die berechneten Designparameterwerte eingestellt und der Verifikationsversuch, welcher zur Bestimmung der Modellgüte dient, durchgeführt. Als Modellstruktur werden häufig polynomiale Modelle verwendet. Aber auch künstliche neuronale Netze (KNN) oder stückweise definierte polynomiale Modelle sind möglich. In Abhängigkeit der Modellstruktur bzw. der vermuteten Wechselwirkungen der Designparameter, welche in der Modellstruktur modelliert sind, erfolgt die Berechnung der Designparametervariationen (Versuchspunkte). Das heißt, dass die Versuchspunkte so im Designraum angeordnet werden müssen, dass diese möglichst viel Aussagekraft hinsichtlich der Parameter in der Modellstruktur haben. Das heißt, dass bei diesem Vorgehen für eine optimale Versuchsplanung die Modellstruktur vor dem 1. Versuch bekannt sein muss. Üblicherweise wird deshalb bei einer ganz neuen Optimierungsaufgabe ein vollfaktorieller Versuchsplan mit vielen Variationen durchgeführt, um die vermutete Modellstruktur zu verifizieren.

Generell kann man aber nach Klein<sup>64</sup> durch den DOE- Versuchsplan (D-optimales Design) folgende Vorteile gegenüber anderen Versuchsplänen erzielen:

- freie und unabhängige Wahl für die Anzahl der Stufen pro Designparameter
- freie Wahl der Stufenabstände
- freie Wahl der polynomialen Modellstruktur
- bereits durchgeführte Versuche können mit einbezogen werden
- der Designraum darf durch Neben- und Zwangsbedingungen restriktiert sein
- nachträglich können neue Designparameter hinzugefügt werden

Der wesentliche Vorteil des D-optimalen Designs ist, dass vor allem bei einer großen Anzahl von Designparametern, wesentlich weniger Versuchspunkte benötigt werden. Im vollfaktoriellen Versuch steigt der Versuchsaufwand mit  $2^n$  bei einem linearen

---

<sup>64</sup> Klein 2004

Ansatz wobei hier  $n$  die Anzahl der Designparameter symbolisiert. Nach Klein<sup>65</sup> gilt bei der D- optimaler Versuchsplanung die Faustregel:

$$\text{Anzahl Versuche} \sim = 1,5 * \text{Koeffizienten Modell} \quad (2.4.5)$$

Dabei hängt die Zahl der Koeffizienten von der Anzahl der gewählten Abstufungen der einzelnen Parameter ab. Für einen linearen Ansatz gilt:

$$\text{Koeffizienten Modell} = \text{Anzahl Designparameter} + 1 \quad (2.4.6)$$

Durch die Anzahl der Versuche kann dann die Versuchszeit aus der benötigten Zeit pro Messpunkt abgeschätzt werden. Hinzu kommen noch Konditionierungs-, Stabilisierungsphasen sowie Wiederholpunkte, um die Streuung und die Reproduzierbarkeit zu bewerten.

---

<sup>65</sup> Klein 2004

### 3 Motivation und Zielsetzung

#### 3.1 Motivation

Technische Systeme müssen stetig weiterentwickelt werden, damit sie im Markt wettbewerbsfähig bleiben können. Deshalb ist die Verkürzung der Entwicklungszeit ein wichtiges Ziel für ein Unternehmen. Um dieses Ziel erreichen zu können, werden neue Methoden und Prozesse benötigt.

Eine wichtige Strategie ist das sogenannte Frontloading (siehe Bild 17). Hier werden kostenintensive Prototypen in frühen Phasen durch geeignete virtuelle Prototypen (Simulationsmodelle) ersetzt, um so Entwicklungszeit und damit Kosten zu sparen. Dabei steigt der Modellierungsaufwand mit dem Projektfortschritt, da immer detailliertere Modelle benötigt werden. Deshalb werden in späteren Entwicklungsphasen reale Prototypen zur Validierung und Optimierung eingesetzt.

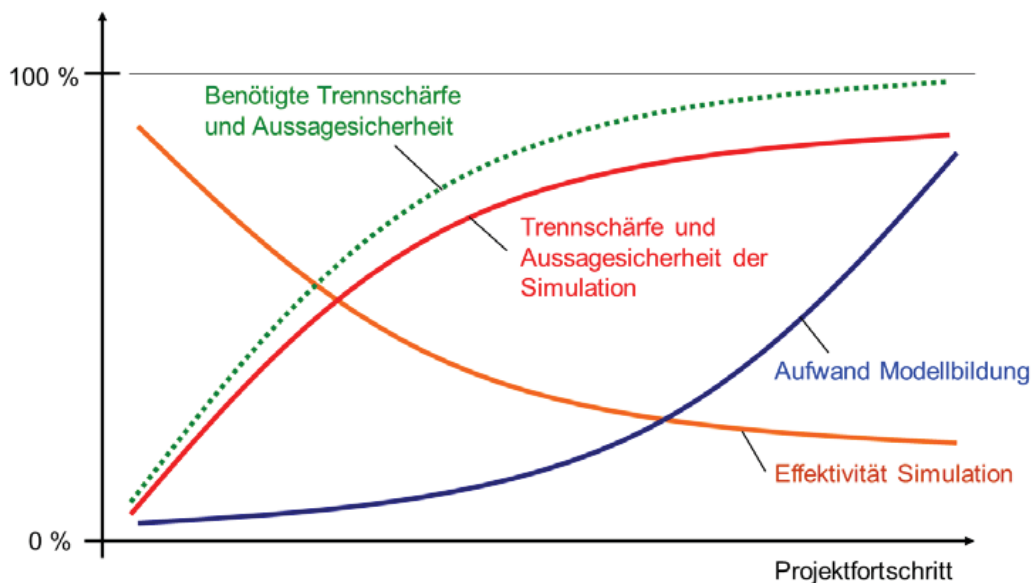


Bild 17: Frontloading<sup>66</sup>

In dieser Phase besteht die Hauptaufgabe darin, die optimalen Werte für die betrachteten Designparameter zu finden. Dazu müssen geeignete Versuche in einer

<sup>66</sup>Heißling 2002



entsprechenden Validierungsumgebung durchgeführt werden. Da die Prototypen oft um ein Vielfaches teurer sind als das spätere Produkt, lassen sich durch geeignete Methoden zur Verkürzung der Versuchszeiten erhebliche Kosteneinsparungen erzielen. Die wesentlichen Faktoren, die dabei zur Kostensenkung beitragen, sind:

- Verkürzung der Entwicklungszeit
- Einsparung Kosten für Prototypen/Prototypenzeit
- Einsparung Prüfstandskosten (zzgl. Messtechnik und Personal)

Ein weiterer Trend in der Technik ist, dass moderne technische Systeme immer mehr aus hochintegrierten, mechatronischen Anteilen bestehen. Eine Eigenschaft dieser Systeme ist, dass aufgrund von starken Wechselwirkungen oft mehrere Designparameter, und Kriterien gleichzeitig bei der Optimierung betrachtet werden müssen. Das hat zur Folge, dass der Versuchsaufwand deutlich ansteigt.

### 3.2 Zielsetzung

Zielsetzung dieser Arbeit ist daher die Erarbeitung eines neuen Teilhandlungssystems zur versuchsbasierten Optimierung technischer Systeme. Die Aufgabe dieses Teilhandlungssystems ist die Berechnung der optimalen Designparameterwerte, sodass sich das betrachtete System hinsichtlich der definierten Kriterien optimal verhält (siehe Bild 18).

Dabei soll dieses Ziel mit möglichst wenig Versuchs-/Optimierungsaufwand erreicht werden. Ein wesentlicher Punkt ist die Versuchszeit zu reduzieren, da diese einen erheblichen Anteil am Versuchsaufwand verursacht. Letztendlich sollen jedoch die Entwicklungskosten minimiert werden. Deshalb wird die Versuchszeit immer in Form von Kosten ausgedrückt. Nach Klein<sup>67</sup>, kann ein schlechteres Optimierungsergebnis ebenfalls in Form von Kosten ausgedrückt werden, da das Produkt in diesem Fall für den Kunden weniger wert ist. Damit ist es möglich, auch ein schlechteres Optimierungsergebnis zu akzeptieren, wenn der Versuchsaufwand für ein besseres Ergebnis sehr hoch ist, da diese Kosten ebenfalls auf das Produkt umgelegt werden müssen.

---

<sup>67</sup> Klein 2004

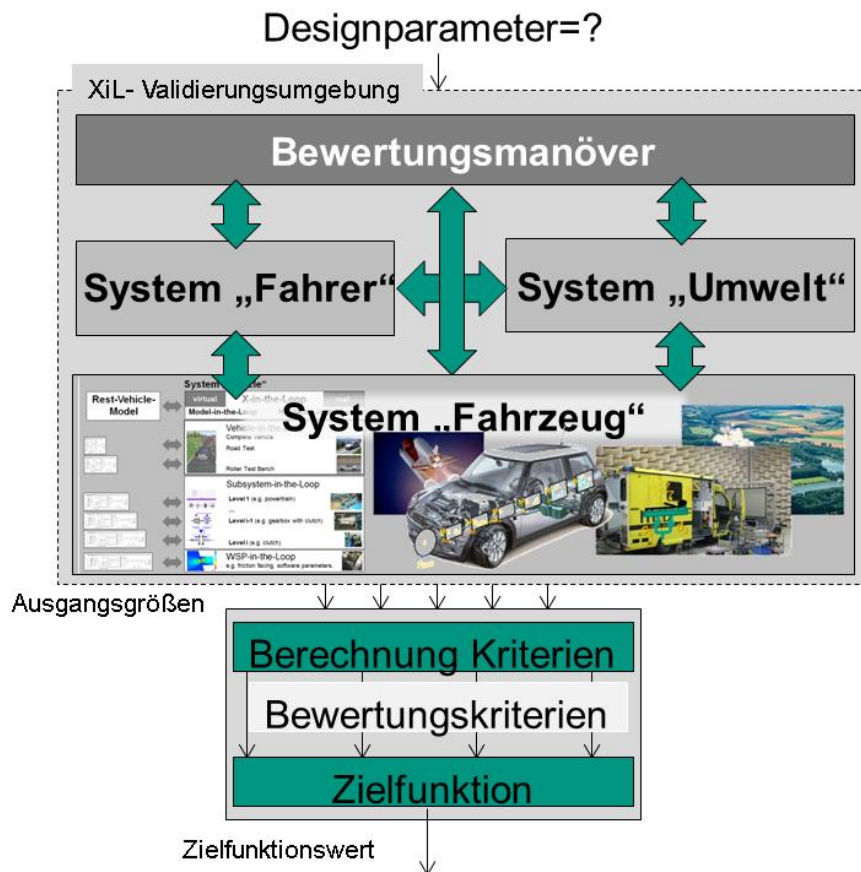


Bild 18: Aufgabe versuchs-basierte Optimierung am Beispiel der Fahrzeugentwicklung

Klein schlägt<sup>68</sup> daher anstelle einer abrupten Übergangsfunktion wie in Bild 19 oben einen weichen Übergang wie in Bild 19 unten vor, um den Verlust eines schlechteren Optimierungsergebnis zu modellieren.

Aktuelle versuchs-basierte Methoden basieren auf der direkten Erfassung der Kriterien für verschiedene Designparametereinstellungen. Das heißt, der Prüfling muss für jede Designparametereinstellung das Bewertungsmanöver absolvieren. Bei Kriterien wie dem Verbrauch im neuen europäischen Fahrzyklus (NEFZ) bedeutet das, dass pro Designparametereinstellung ca. 20 min benötigt werden. Sollen neben diesem Kriterium noch Kriterien, wie der Verbrauch in der Auto-Motorsport-Normalrunde<sup>69</sup> (202 km) berücksichtigt werden, bedeutet das einen sehr hohen Versuchsaufwand pro Variation. Deshalb müssen in diesem Fall die Anzahl der Variationen so gering wie möglich gehalten werden. Ziel ist es daher geeignete

<sup>68</sup> Klein 2004

<sup>69</sup> Auto-Motor-Sport 2011

Methoden zu finden, die auch bei einer großen Anzahl von Designparameter wenige Variationen benötigen um so den Versuchsaufwand zu reduzieren.

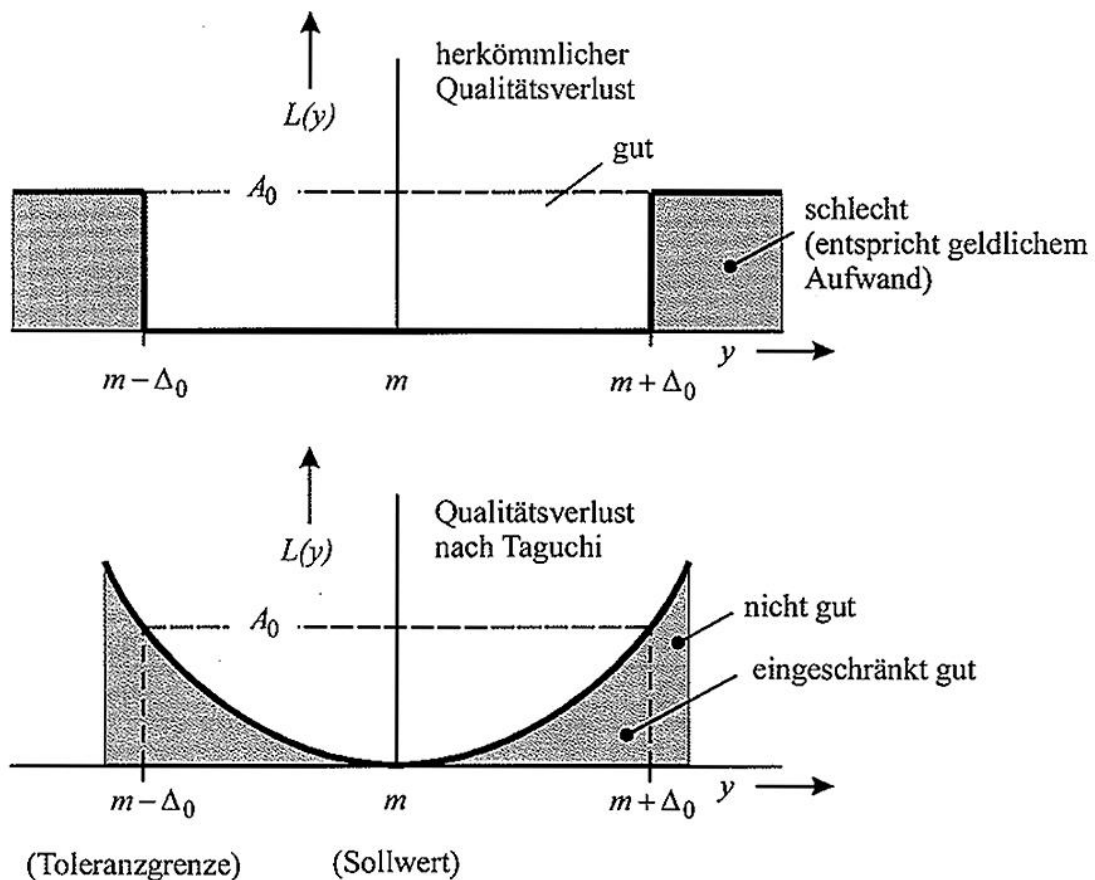


Bild 19: Formulierung Qualitätsverlust<sup>70</sup>

Ein wichtiger Aspekt dieser Arbeit ist die Herausstellung des Nutzens im Sinne der Wirtschaftlichkeit. Deshalb ist ein Ziel dieser Arbeit die Erarbeitung von Kennwerten, welche die Leistungsfähigkeit des Verfahrens charakterisieren. Beispiele für solche Kennwerte sind die benötigte Versuchszeit, Rechenzeit, Modellbildungsaufwand, Güte Optimierungsergebnis. Anschließend sollen Einflussgrößen auf diese Kennwerte identifiziert und quantifiziert werden. Dabei sollen sowohl Einflussgrößen, die die Optimierungsaufgabe betreffen (z.B. der Rauschanteil der im Versuch gemessenen Größen) als auch die Einflussgrößen die das Verfahren (Anzahl der Generation des verwendeten genetischen Algorithmus) beeinflussen, untersucht werden.

<sup>70</sup> Klein 2004

Im Folgenden sind die Teilziele nochmals explizit aufgeführt:

- Definition eines Teilhandlungssystems zur versuchsbasierten Optimierung für die Aktivitäten "Modellierung von Prinzip und Gestalt" sowie "Validierung" im integrierten Produktentstehungsmodell (iPeM).
- Erarbeiten von Methoden für die verschiedenen Arten von Optimierungsaufgaben.
- Exemplarische Anwendung des neuen Teilhandlungssystems auf verschiedene Optimierungsaufgaben
- Herausstellen des Nutzens und der Übertragbarkeit der erarbeiteten Methoden
- Definition der Anwendungsgrenzen der Verfahren
- Identifikation und Analyse von Einflussfaktoren des Optimierungsproblems auf Bewertungsgrößen des Verfahrens (wie z.B. die Versuchszeit)
- Multikriterielle Optimierung ermöglichen

## 4 Teilhandlungssystem zum versuchsbasierten Optimieren

In den folgenden Kapiteln wird das in Kapitel 3 skizzierte Teilhandlungssystem erarbeitet und dargelegt. Die Aufgabe dieses Teilhandlungssystems ist die Berechnung der optimalen Designparameterwerte, sodass sich das betrachtete System hinsichtlich der definierten Kriterien optimal verhält. Damit die Methode effizient im Produktentwicklungsprozess genutzt werden kann, wird diese im folgenden Kapitel in das integrierte Produktentstehungsmodell (iPeM) exemplarisch eingeordnet. Anschließend wird ein Überblick über den erarbeiteten Optimierungsprozess gegeben. Danach wird detailliert auf die einzelnen Teilprozesse und Methoden eingegangen.

Abschließend werden in Kapitel 4.9.8 die verschiedenen Teilmethoden und Prozesse des Teilhandlungssystems hinsichtlich der Wirtschaftlichkeit der einzelnen Verfahren in Abhängigkeit des Optimierungsproblems und der Einstellmöglichkeiten der Verfahren analysiert. Wesentliche Teile der Wirtschaftlichkeitsbetrachtung sind Versuchsdauer, Modellierungsaufwand aber auch die Rechenzeit.

Für die Erarbeitung dieses Teilhandlungssystems werden neue Teilverfahren erarbeitet. Teilweise können diese Methoden für weitere Entwicklungsaufgaben genutzt werden. Kapitel 6 beschreibt welche Methoden für welche Entwicklungsaufgaben wie genutzt werden können.

### 4.1 Einordnung in das integrierte Produktentstehungsmodell (iPeM)

Um das Frontloading zu realisieren, werden in frühen Entwicklungsphasen geeignete Simulationsmodelle statt Prototypen verwendet, um Entwicklungszeit einzusparen. Jedoch steigt der Modellierungsaufwand mit dem Projektfortschritt, da immer detailliertere Modelle benötigt werden. Mit dem XiL-Framework ist es möglich, auch Teilmodelle in Form physischer Komponenten in die Simulation einzubinden. Ab einem gewissen Zeitpunkt in der Entwicklung werden dennoch häufig physische Prototypen eingesetzt. Aufgabe des erarbeiteten Teilhandlungssystems ist die Bestimmung der optimalen Designparameter des technischen Systems. Das heißt, es müssen die zu diesem Zeitpunkt relevanten (veränderbaren) Eigenschaften der Wirkflächenpaare und Leitstützstrukturen so bestimmt werden, dass die Funktion des technischen Systems optimal erfüllt wird. Dazu werden im erarbeiteten

Teilhandlungssystem Methoden für die Aktivität "Modellierung von Prinzip und Gestalt" bereitgestellt (siehe Bild 20).

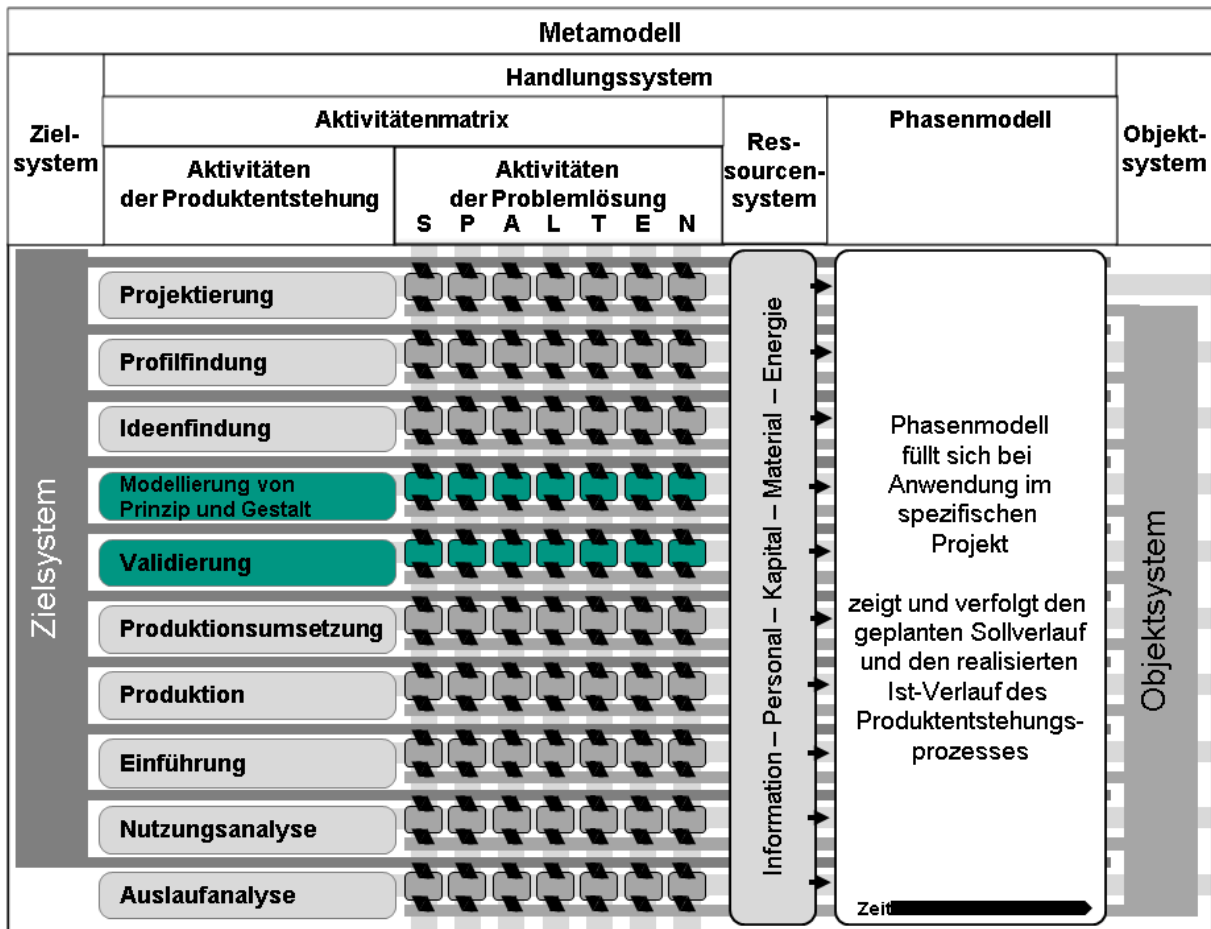


Bild 20: Einordnung in das integriertes Produktentstehungsmodell (iPeM)<sup>71</sup>

Außerdem stellt das Teilhandlungssystem entsprechende Methoden für die Validierung bereit, um das aktuelle Objektsystem mit dem Zielsystem abgleichen zu können. Diese werden in unterschiedlichen Ausprägungen während des Optimierungsprozesses benötigt um zum Beispiel die Modellgüte zu beurteilen.

## 4.2 Prozess

Stand der Forschung für die versuchsbasierte Optimierung ist die statistische Versuchsplanung, die nichtphysikalische Modelle, wie polynomiale Modelle oder künstliche neuronale Netze, für die versuchsbasierte Optimierung verwendet. Durch

<sup>71</sup> Albers / Braun 2011

das DOE-Verfahren (siehe Kapitel 2.4) werden die Designparameter gezielt variiert und die Bewertungskriterien für jede Variation bestimmt. Mit der Zielfunktion kann dann für jede Variation der Zielfunktionswert berechnet werden. Diese Daten (Designparameter und Zielfunktionswert) werden verwendet, um die Parameter des nicht physikalischen Modells zu bestimmen. Dieses Modell wird dann für die Optimierung verwendet. Abschließend müssen die gefunden Parametereinstellungen im Versuch verifiziert werden und die Modellstruktur gegebenenfalls angepasst bzw. weitere Variationsversuche durchgeführt werden. Deshalb muss bei diesem Ansatz die Art der Abhängigkeit des Zielfunktionswerts von den Designparametern hinreichend bekannt sein. Das ist die Voraussetzung, damit eine entsprechende Modellgüte sichergestellt werden kann. Dieses Vorgehen eignet sich besonders bei folgenden Randbedingungen:

- Eine geringe Anzahl von Designparameter
- Die Variation der Designparameter benötigt wenig Zeit
- Die Durchführung des Bewertungsmanövers erfordert wenig Zeit (Erfassung der Bewertungskriterien)

Werden eine oder mehrere Bedingungen nicht erfüllt, steigt die benötigte Versuchszeit stark an. Für den vollfaktoriellen Versuch gilt für die Dauer des Versuchs  $t_{Ges}$  folgender Zusammenhang:

$$t_{Ges} = n_{Variationen} * t_{Variation} \quad (4.2.1)$$

Dabei hängt die Zahl der Variationen  $n_{Variationen}$  stark von der Anzahl der Designparameter  $n_{Designparameter}$  und der Anzahl der dazugehörigen Variationsstufen  $n_{Variationsstufen}$  ab:

$$n_{Variationen} = n_{Variationsstufen}^{n_{Designparameter}} \quad (4.2.2)$$

Die Dauer einer Variation  $t_{Variation}$  hängt dabei von der benötigten Zeit für die Verstellung der Designparameter  $t_{Verstell}$ , der Konditionierzeit  $t_{Konditionierung}$  (d.h. der Zeit in der das System in den geforderten Anfangszustand für das Bewertungsmanöver gebracht wird) sowie von der Dauer des Bewertungsmanövers  $t_{Bewertungsmanöver}$  an sich ab:

$$t_{Variation} = t_{Verstell} + t_{Konditionierung} + t_{Bewertungsmanöver} \quad (4.2.3)$$

Zu dieser Zeitspanne muss noch der Verifikationsversuch dazugerechnet werden. Hier werden die berechneten (optimalen) Parameter eingestellt und das Bewertungsmanöver durchgeführt. Anhand eines Vergleichs zwischen dem Modell und dem Versuch kann anschließend die Modellqualität beurteilt werden.

Um die Zeiteffizienz, sowie die Qualität des Optimierungsprozesses zu verbessern, erweitert das erarbeitete Teilhandlungssystem diesen Ansatz um drei weitere Verfahren (siehe

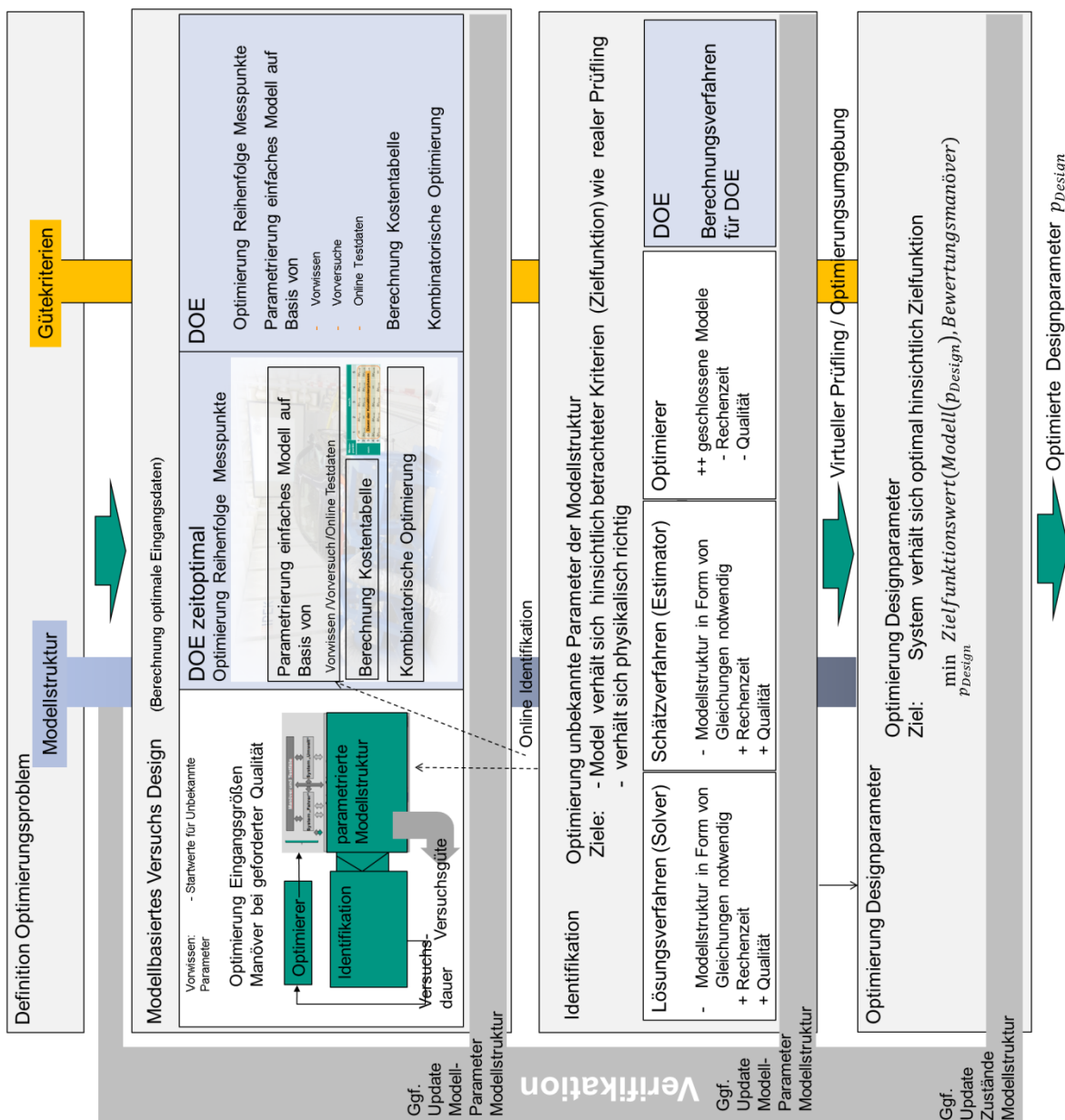


Bild 21):



- DOE mit zeiteffizienter Reihenfolge der Versuchspunkte  
Wenn unterschiedliche Ausgangs- und Endzustände bei den einzelnen Messpunkten benötigt werden, kann durch eine geschickte Reihenfolge der Messpunkte eine Verkürzung der Versuchszeit erreicht werden (siehe Kapitel 4.8). Für die Evaluation der Schaltqualität auf dem Rollenprüfstand können nach Albers<sup>72</sup> dadurch 80% Versuchszeit eingespart werden.
- Frei definierbare Modellstrukturen (offline)  
Hier werden physikalische Modelle, die jedoch unter bestimmten Voraussetzungen nicht physikalische Anteile enthalten dürfen, für die Optimierung verwendet. Dabei werden die in früheren Entwicklungsphasen entstanden physikalische Modellstrukturen bzw. Modellvorstellungen verwendet, um einen virtuellen Prüfling zu generieren. Dazu werden die unbekannt Parameter der Modellstruktur durch Versuchsdaten identifiziert. Der Vorteil ist, dass bei diesem Verfahren das Bewertungsmanöver nicht in jeder Variationsstufe als Versuch durchgeführt werden muss. Im Idealfall sind schon Messdaten von anderen Versuchen mit dem Prüfling vorhanden, um das Simulationsmodell zu erstellen.
- Frei definierbare Modellstrukturen (online)  
Eine Erweiterung dazu ist das Identifizierungsmanöver bzw. das optimale Versuchsdesign (Art und Anzahl der Sensoren) anhand der Modellstruktur zu berechnen, beziehungsweise das Modell zur Versuchszeit online zu verbessern und damit das Identifikationsmanöver zu verbessern.

In den folgenden Kapiteln werden die einzelnen Teilaktivitäten des Teilhandlungssystems im Detail erläutert.

---

<sup>72</sup> Albers et al. 2012

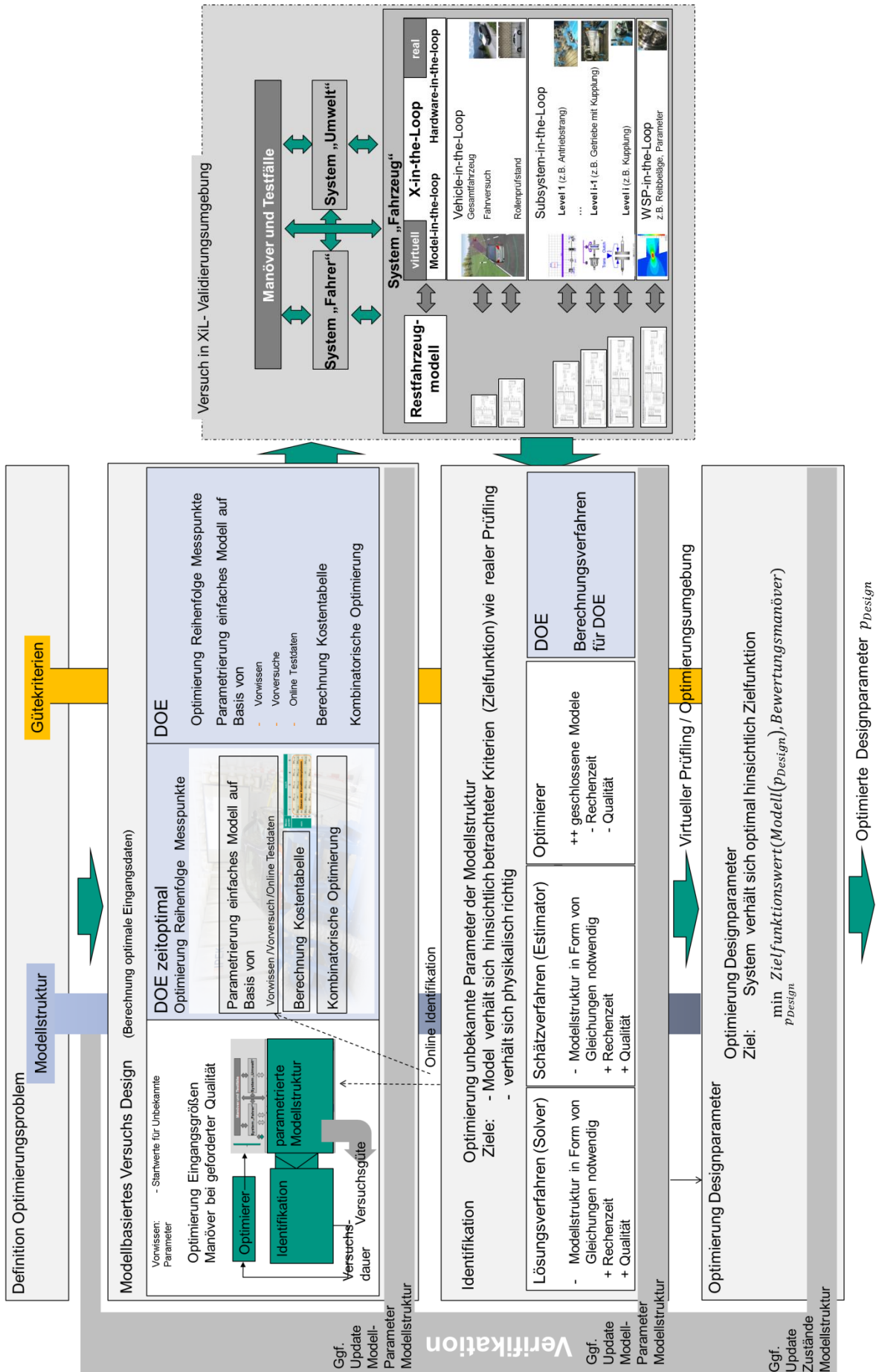


Bild 21: entwickeltes Teilhaltungssystem zum versuchsbasierten Optimieren

### 4.3 Definition der Optimierungsaufgabe

In dieser Phase wird die Optimierungsaufgabe in Form einer Situationsanalyse und Problemeingrenzung gemäß der Methode SPALTEN<sup>73</sup> definiert. Dabei müssen nachfolgende Aktivitäten definiert/durchgeführt werden.

### 4.4 Variablen definieren

Zunächst müssen die relevanten Größen der Optimierungsaufgabe in die folgenden Gruppen zugeordnet werden. Dabei kann dieselbe Variable in mehreren Gruppen vorkommen:

- **bekannte Parameter**  
Manche Parameter, wie Naturkonstanten die in der Modellstruktur enthalten sind, sind schon vor dem eigentlichen Versuch bekannt. Um den Versuchs- und Rechenaufwand zu senken ist es sinnvoll möglichst viele Parameter schon vorab zu definieren.
- **Eingangsgrößen**  
Die Eingangsgrößen sind Flussgrößen wie z.B. Sollwerte, die in das System gehen. Falls Systemparameter zum Beispiel durch einen DOE-Plan variiert werden, sind das ebenfalls Eingangsgrößen.
- **Ausgangsgrößen**  
Ausgangsgrößen sind alle Flussgrößen, mit denen das System auf die Eingangsgrößen antwortet.
- **Berücksichtigte Zustandsgrößen**  
Diese Flussgrößen werden bei der Identifizierung ebenfalls mit abgeglichen. Das verhindert, dass der virtuelle Prototyp sich zwar hinsichtlich der Ausgangsgrößen wie das reale System verhält, sich intern jedoch nicht-physikalisch verhält.

---

<sup>73</sup> Albers et al 2005

- **Unbekannte Parameter**  
Die unbekannt Parameter sind die Parameter, die während der Identifizierung berechnet werden müssen. Zusätzlich hilft es, die Grenzen in denen sich diese Parameter bewegen können, zu definieren. Je nach Algorithmus können zusätzlich Startwerte für diese Parameter angegeben werden. Dadurch kann die Rechenzeit deutlich reduziert werden.
- **Designparameter**  
Designparameter sind die Parameter die in dieser Phase des Entwicklungsprozesses so bestimmt werden sollen, dass sich das System hinsichtlich der definierten Kriterien (siehe Kapitel 4.5) optimal verhält. Zu der Definition der Designparameter gehört auch die Definition der Grenzen, in denen die Designparameter verändert werden dürfen, dem sogenannten Designraum. Dabei können diese Grenzen auch in Form von Abhängigkeiten zwischen den Designparametern vorliegen.

#### 4.5 Manöver, Kriterien, Zielfunktion definieren

Moderne technische Systeme müssen sich oft hinsichtlich einer Vielzahl von Kriterien optimal verhalten. Viele dieser Kriterien lassen sich in die in Bild 22 dargestellten Gruppen einordnen. Dabei ist es möglich, dass zwischen einem oder mehreren dieser Kriterien Wechselwirkungen bestehen und ein Kompromiss gefunden werden muss.



Bild 22: Übergruppen von Kriterien

Deswegen muss in der Definitionsphase der Optimierungsaufgabe neben den Kriterien auch eine Zielfunktion, die die Gewichtung der Kriterien zueinander enthält, definiert werden. Dabei hängen die Gewichtungen der Kriterien vom Zielsystem ab. So kann die beste Gesamtbewertung für das Gesamtsystem gefunden werden. In

Albers<sup>74</sup> wird eine solche Zielfunktion für die kombinierte Bewertung der Energieeffizienz und der Fahrbarkeit von Fahrzeugen beschrieben (siehe Bild 23).

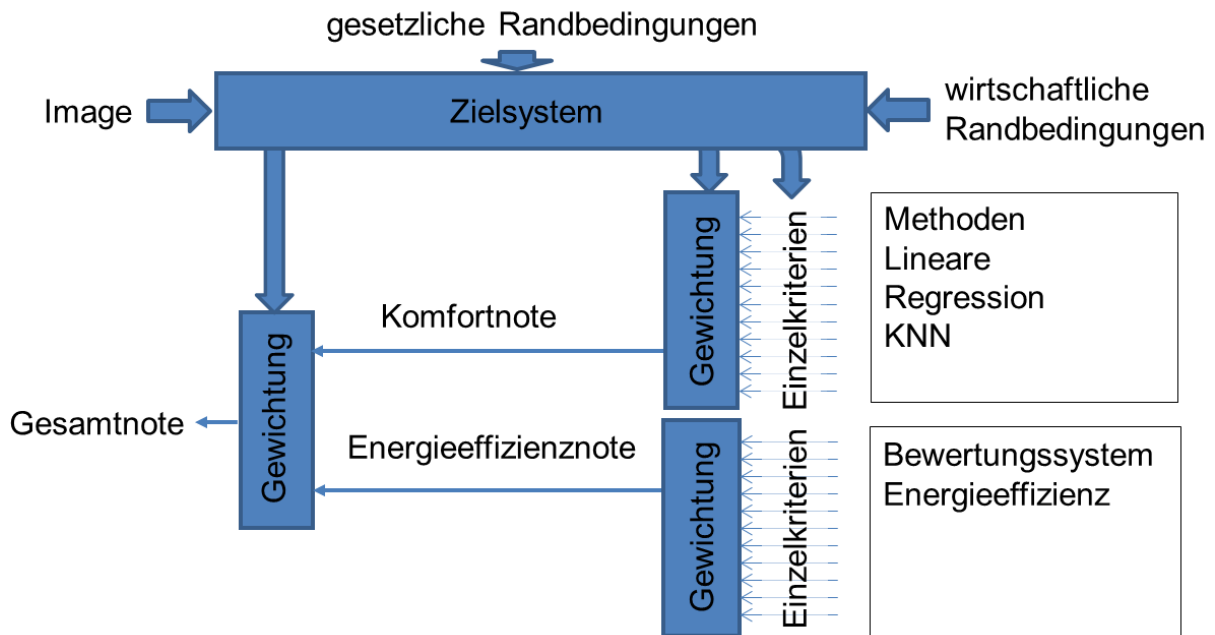


Bild 23: Methode zur kombinierten Bewertung der Energieeffizienz und der Fahrbarkeit von Fahrzeugen (siehe Albers<sup>75</sup>)

Dabei erfordert jedes Kriterium ein entsprechendes Manöver, in dem dieses Kriterium ausgewertet werden kann. Dieses Manöver mit allen Randbedingungen sowie der mathematischen Beschreibungen der Auswertungen der Kriterien werden in dieser Phase benötigt. Ein wichtiger Punkt bei der Definition der mathematischen Beschreibung der Kriterien ist, dass diese möglichst unabhängig vom Rauschanteil (signal-to-noise-ratio) der Messdaten definiert sind. Dadurch wird verhindert, dass sich unterschiedliche Kennwerte für die Kriterien z.B. bei einem anderen Sensor für die gleiche Untersuchung ergeben. Dieser Einfluss wird anhand eines Beispielsystems in Kapitel 4.9.8.4 exemplarisch untersucht und Maßnahmen zur Minimierung dieses Einflusses aufgezeigt. Häufig werden die Kriterien im 1. Schritt durch physikalische Kennwerte beschrieben, die dann in ein Notensystem überführt werden. Bei der Bewertung des Fahrkomforts werden die physikalischen Kennwerte, wie die maximale Beschleunigung oder peak-to-peak Werte, mittels vorher

<sup>74</sup> Albers et al 2010b

<sup>75</sup> Albers et al 2010b

gebildeten (trainierten) künstlichen neuronalen Netzen (siehe Lerspalungsanti<sup>76</sup>) oder lineare Regressionsmodelle in Noten umgerechnet. Im Falle der Komfortbeurteilung wird häufig die ATZ-Skala (siehe Bild 24) verwendet.

	nicht annehmbar				Grenzfall	annehmbar				
Bewertungsindex	1	2	3	4	5	6	7	8	9	10
Geräusche, Vibrationen	nicht annehmbar			unangenehm	Verbesserung erforderlich	mäßig	leicht	sehr leicht	Spuren	keine
Festgestellt von	Allen Kunden	Durchschnittskunden			kritischen Kunden		ausgebildeten Beobachtern		nicht wahrnehmbar	

Bild 24: ATZ-Skala nach Grzesik<sup>77</sup>

#### 4.6 Versuchsdesign und Durchführung im XiL-Framework

In dieser Teilaktivität wird das Versuchsdesign definiert. Dafür müssen die Manöver sowie die Validierungsumgebung für den Versuch definiert werden. Ziel ist es, das Versuchsdesign so zu wählen, dass möglichst geringe Optimierungskosten entstehen. Dabei beinhaltet das Versuchsdesign folgende Punkte:

- Definition der XiL-Validierungsumgebung
  - Art und Eigenschaften der Messstellen
  - Initialzustände
  - Restfahrzeug-/Restsystem-/Umgebungssimulation
- Auswahl des Verfahrens
- Versuchsdurchführung
  - Messpunkte/Manöver
  - Reihenfolge
  - Konditionierphasen
- Auswertung der Kriterien

<sup>76</sup>Lerspalungsanti 2010

<sup>77</sup>Grzesik 2009

In dieser Teilaktivität soll das hinsichtlich der Entwicklungs-/Optimierungskosten optimale Versuchsdesign bestimmt und der Versuch durchgeführt werden. Im Wesentlichen muss zunächst eine geeignete Validierungsumgebung nach Kapitel 4.6.1 definiert werden, in der das System hinsichtlich der betrachteten Kriterien untersucht werden kann. Anschließend muss eines von vier Verfahren für die eigentliche versuchsbasierte Optimierung nach den Kriterien in Kapitel 4.6.2 ausgewählt werden. In Abhängigkeit des Verfahrens werden die Manöver für den Versuch definiert bzw. berechnet. Anschließend wird der Versuch in der Validierungsumgebung durchgeführt. Die Messdaten aller Manöver sowie die aus den Bewertungsmanövern ermittelten Kennwerte (siehe Kapitel 4.9.7) bilden die Eingangsdaten für die Identifikation der Parameter der zugrunde liegenden Modellstruktur (siehe Kapitel 4.9.4).

#### **4.6.1 Definition XiL-Validierungsumgebung**

Kundenrelevante Kriterien beziehen sich in der Regel auf das Gesamtsystem und können nicht immer direkt auf die Komponenten- beziehungsweise auf die Subsystemebene übertragen werden. Die XiL-Validierungsumgebung wird verwendet (siehe Kapitel 2.1.2), um Komponenten und Subsysteme schon in frühen Entwicklungsphasen, in denen das Gesamtsystem noch nicht verfügbar ist, optimieren zu können. Die Unit-under-Development (UUD) wird in dieser Umgebung durch eine entsprechende Restsystemsimulation zu einem Gesamtsystem ergänzt. Dabei kann die Restfahrzeugsimulation physische Anteile z.B. Prüfstandsteile aber auch Simulationsmodelle beinhalten. Das ermöglicht die Auswertung der Kriterien bzw. der Zielfunktion von einzelnen Wirkflächenpaaren über einzelne Subsysteme bis zum Gesamtsystem. Dabei werden auch die Wechselwirkungen zwischen Fahrer, Umwelt sowie dem System Fahrzeug, soweit es die betrachteten Kriterien erfordern, abgebildet.

#### **4.6.2 Auswahl des Verfahrens**

Es existieren verschiedene Ansätze zum versuchsbasierten Optimieren (siehe Kapitel 2.4). Dabei ist das DOE-Verfahren nach Taguchi ein sehr zeiteffizientes Verfahren, wenn der Zusammenhang (die polynomiale Modellstruktur) zwischen Designparametern und den Kriterien bekannt ist. Dies kann durch einen vollfaktoriellen Versuchsplan mit vielen Faktorstufen, einem sogenannten Screeningversuch bzw. durch analytische Überlegungen erreicht werden. Ist die Art des

Zusammenhangs bekannt, können nach dem DOE-Verfahren viele Messpunkte im Vergleich zum vollfaktoriellen Versuch gespart werden. Deshalb werden die in dieser Arbeit erarbeiteten Erweiterungen bzw. neuen Ansätze mit dem DOE-Verfahren verglichen.

Im Rahmen dieser Arbeit wird der DOE-Ansatz um eine Methode, (siehe Kapitel 4.8), mit der die ermittelten Messpunkte in eine zeiteffiziente Reihenfolge gebracht werden können, erweitert. Außerdem wird eine Methode (siehe Kapitel 4.9) geschaffen, bei der freidefinierbare Modellstrukturen eingesetzt werden können, um den Aufwand des versuchsbasierten Optimierungsprozess zu reduzieren. Diese Methoden (siehe

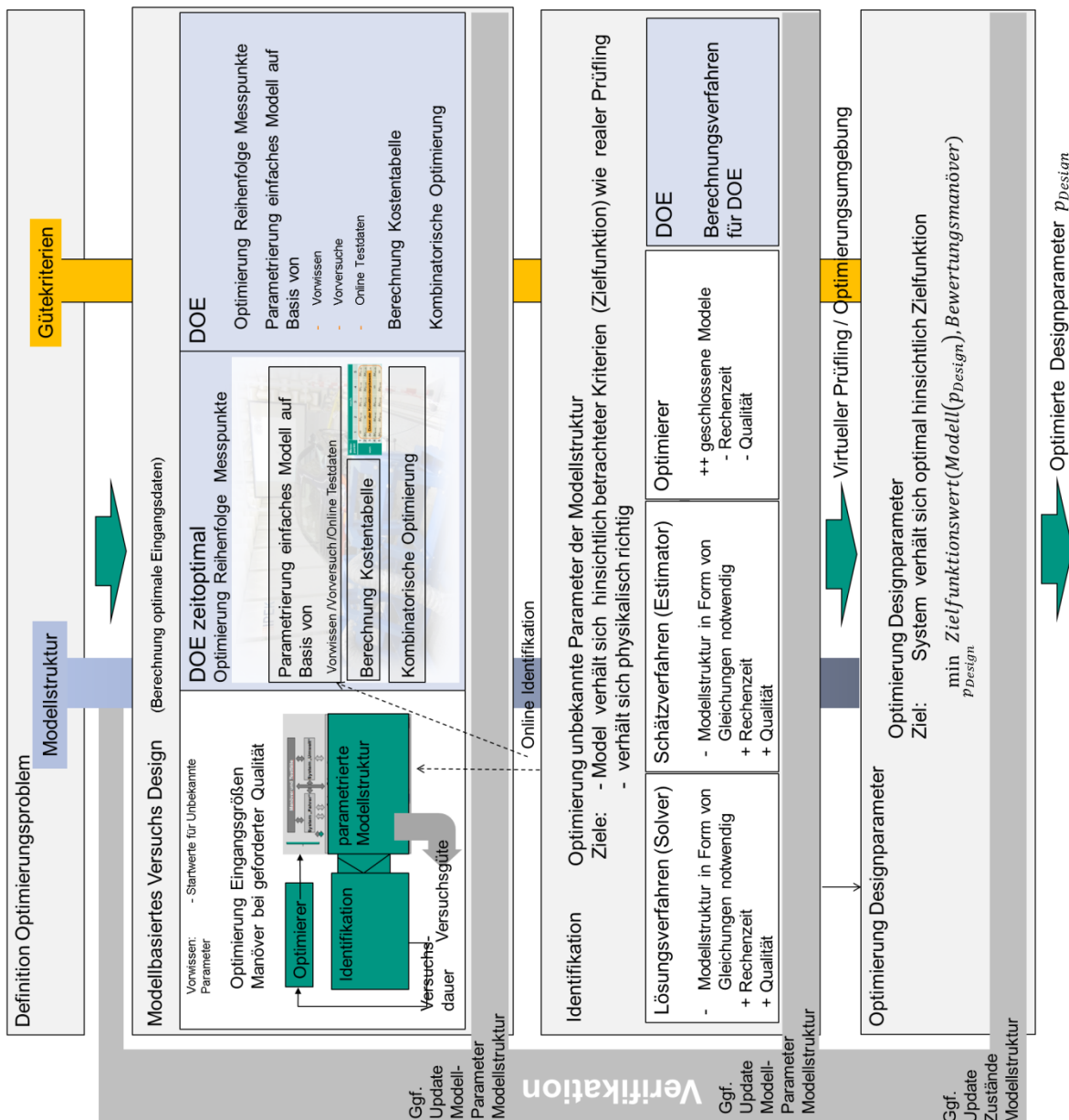


Bild 21) werden im Folgenden kurz erläutert. Danach werden die erarbeiteten Methoden im Detail vorgestellt. In Kapitel 4.9.8 werden dann die Einflussfaktoren des Versuchsaufwand der Verfahren gezielt analysiert. Das Ergebnis dieser Betrachtung folgt in Kapitel 5.



## 4.7 Design of Experiments (DOE) – Verfahren

Beim dem klassischen DOE-Verfahren nach Taguchi wird das Bewertungsmanöver mit verschiedenen Designparametersätzen durchgeführt. Dabei wird zunächst die Anzahl der Variationsstufen für jeden Designparameter gewählt. Danach kann ein vollfaktorieller Versuchsplan aufgestellt werden. Nach dem Vorgehen (siehe Kapitel 2.4) werden dann die Parametersätze ausgewählt, die am meisten für die Parameteridentifikation der Modellstruktur beitragen. Im Versuch werden anschließend die Bewertungsmanöver für alle ausgewählten Designparametersätze durchgeführt. Anschließend werden die Kriterien für jeden Parametersatz ausgewertet. Beim klassischen DOE nach Taguchi werden polynomiale Modelle verwendet, um den Zusammenhang zwischen den Kriterien vgl. Kapitel 2.4 und den Designparametern zu modellieren. Mittlerweile können jedoch auch andere Formen von definierten Funktionen verwendet werden, wie künstliche neuronale Netze oder stückweise definierte Funktionen. Nach der Identifikation der Parameter der Modellstruktur wird das Modell verwendet, um die optimalen Designparametereinstellungen innerhalb des Designraums zu berechnen. Diese Designparametereinstellungen werden anschließend experimentell verifiziert.

## 4.8 Design of Experiments (DOE) mit Ordnung der Messpunkte

Eine Erweiterung zum klassischen DOE Verfahren, die im Rahmen dieser Arbeit entwickelt wurde, ist ein DOE- Verfahren bei dem die Bewertungsmanöver in eine zeiteffiziente Reihenfolge gebracht werden. Voraussetzung für eine Zeiteinsparung mit diesem Verfahren ist, dass unterschiedliche Anfangs- bzw. Endzustände des Systems bei den verschiedenen Manövern benötigt werden bzw. sich ergeben.

Ziel der Ordnung der Manöver ist es, die Konditionierungszeiten zwischen den einzelnen Manövern zu reduzieren. Dabei können die Manöver der Einzelkriterien aber auch die Designparametervariationen geordnet werden. Je nach System/Optimierungsaufgabe kann dadurch viel Versuchszeit bei den gleichen Ergebnissen (da dieselben Manöver durchgeführt werden) eingespart werden. Das heißt, neben der versuchsbasierten Optimierung kann dieses Verfahren auch für die Optimierung der Zeiteffizienz anderer Manöver eingesetzt werden (vgl. 4.8.2).

### 4.8.1 Verfahren

Für die Ordnung der Manöver wird ein Modell benötigt, das in der Lage ist, die notwendige Konditionierungszeit zwischen den Manövern zu berechnen. Für die Ordnung der Manöver kann dieses Modell relativ grob sein, da es nur darauf ankommt, wie lange die Konditionierungszeiten relativ zueinander sind. Aber je genauer das Modell ist, desto genauer kann die Gesamtversuchszeit abgeschätzt werden. Diese Größe ist unter anderem für die Planung der Prüfstandsverfügbarkeit wichtig.

Neben den klassischen Modellbildungsmethoden (siehe Kapitel 2.2.2) können auch Modelle verwendet werden, die mit Hilfe von Versuchsdaten parametrisiert werden (siehe Kapitel 4.9). Dabei gibt es folgende Möglichkeiten, die notwendigen Versuchsdaten zu erhalten:

- Versuchsdaten aus vorangegangenen Versuchen  
Im Idealfall stehen Versuchsdaten aus andern Versuchen zu Verfügung, die für die Identifikation verwendet werden können. Bei konsequenter Anwendung des iPeM, können jedoch häufig Daten bzw. Modelle aus dem Zielsystem und dem Objektsystem von Vorgängerprodukten als Grundlage für die aktuelle Optimierungsaufgabe verwendet werden.
- Aus einem Teil der Versuchsdaten

Wenn ein Teil der Manöver die gleichen Anfangs- bzw. Endzustände enthält, können diese vorab durchgeführt werden. (Da durch eine Ordnung der Messpunkte keine Ersparnis erzielt werden kann). Diese Versuchsdaten können dann verwendet werden, um das Modell zu identifizieren, um damit die restlichen Manöver zu ordnen.

- Identifikationsversuch

Alternativ zu den ersten beiden Möglichkeiten können auch Versuchsdaten für die Modellbildung identifiziert werden. In diesem Fall muss betrachtet werden, ob der Zeitaufwand für den Identifikationsversuch noch kleiner ist als die Zeitersparnis. Alternativ kann die Reihenfolge mit einem Initialmodell, das während der Versuche genauer parametrisiert wird, berechnet werden. Dabei werden die bereits durchgeführten Versuche verwendet, um das Modell zu verbessern um die beste Reihenfolge der verbleibenden Manöver zu berechnen.

Nach der Modellidentifikation wird das Modell mithilfe von Versuchsdaten verifiziert. Dann wird das Modell eingesetzt, um die Endzustände der einzelnen Teilmanöver zu berechnen. Anschließend werden die notwendigen Konditionierungsphasen (vgl. Kapitel 4.5) zwischen den Teilmanövern simuliert. Das heißt, es wird für jedes Manöver die Zeit berechnet, die benötigt wird um das System vom Endzustand des Manövers zu den restlichen Manövern zu bringen. Diese Ergebnisse werden dann in Form der Tabelle (siehe Bild 25) abgelegt.

Messpunkt	nach					
	1	2	3	4	5	
von	1	$\infty$	$\Delta t_{1 \rightarrow 2}$	$\Delta t_{1 \rightarrow 3}$	$\Delta t_{1 \rightarrow 4}$	$\Delta t_{1 \rightarrow 5}$
	2	$\Delta t_{2 \rightarrow 1}$	$\infty$	$\Delta t_{2 \rightarrow 3}$	$\Delta t_{2 \rightarrow 4}$	$\Delta t_{2 \rightarrow 5}$
	3	$\Delta t_{3 \rightarrow 1}$	$\Delta t_{3 \rightarrow 2}$	$\infty$	$\Delta t_{3 \rightarrow 4}$	$\Delta t_{3 \rightarrow 5}$
	4	$\Delta t_{4 \rightarrow 1}$	$\Delta t_{4 \rightarrow 2}$	$\Delta t_{4 \rightarrow 3}$	$\infty$	$\Delta t_{4 \rightarrow 5}$
	5	$\Delta t_{5 \rightarrow 1}$	$\Delta t_{5 \rightarrow 2}$	$\Delta t_{5 \rightarrow 3}$	$\Delta t_{5 \rightarrow 4}$	$\infty$

Bild 25: Tabelle für die Konditionierungszeiten

Diese Tabelle bildet die Grundlage für eine kombinatorische Optimierung vom Typ des "travelling salesman" Problems. Das heißt, jedes Manöver muss einmal durchgeführt werden. Gesucht wird die Reihenfolge, die die kürzeste Gesamtdauer der Konditionierungsphasen verursacht. Für die Optimierung selbst können



- Manöver 1 Hochschaltungen im Automatikmodus

Das Fahrzeug wird bei verschiedenen konstanten Gaspedalstellungen beschleunigt und wieder auf Kriechgeschwindigkeit verzögert (siehe Bild 27). Die Schaltungen finden dann während der Beschleunigungsphasen statt.

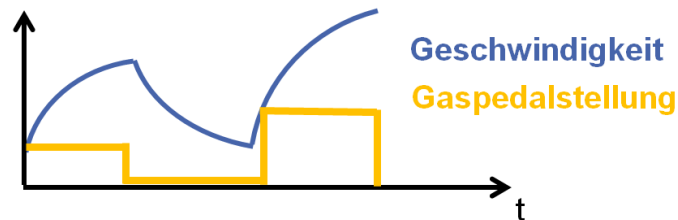


Bild 27: Manöver 1

- Manöver 2 getippte Schaltungen

Das Fahrzeug wird mit einer definierten Gaspedalstellung und Gang betrieben. Sobald die definierte Drehzahl erreicht ist, wird die Schaltung durch manuelles Tippen ausgelöst (siehe Bild 28).

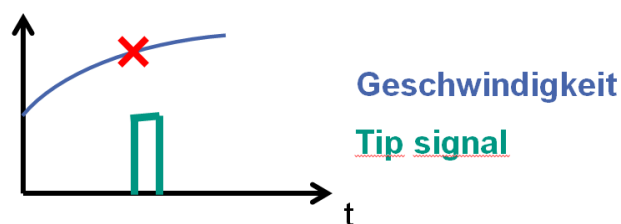


Bild 28: Manöver 2

- Manöver 3/4 Hoch/Runterschaltungen nach einem Tipout/Tipin

Das Fahrzeug wird in einen bestimmten Zustand (Gaspedalstellung, Gang) gebracht und gehalten. Bei der definierten Geschwindigkeit erfolgt der Tipin (positiver Gaspedalsprung) beziehungsweise der Tipout (negativer Gaspedalsprung). Dieser löst die gewünschte Schaltung aus (siehe Bild 29)

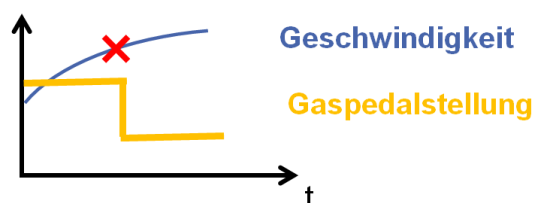


Bild 29: Manöver 3 Tipout

Ziel dieser Optimierungsaufgabe ist es, die optimale Reihenfolge der Schaltungen innerhalb dieser Manöver zu berechnen und auf dem Rollenprüfstand mit einem virtuellen Fahrer (Fahrroboter) durchzuführen.

Dazu wird folgendermaßen, wie Bild 30 zeigt, vorgegangen. Zunächst wird Manöver 1 durchgeführt. Da hier immer der gleiche Anfangszustand (Fahrzeug fährt mit Kriechgeschwindigkeit) eingestellt werden muss, bringt eine Änderung der Reihenfolge der einzelnen Hochläufe keine Reduzierung der Konditionierungszeit. Deshalb wird dieses Manöver zur Bildung eines einfachen Fahrzeugmodells verwendet. Hierzu reicht es in diesem Fall aus, jede dieser Hochfahrten durch ein Polynom zweiten Grades zu beschreiben. Für die Gaspedalstellungen zwischen den gemessenen Abstufungen wird zwischen diesen Kurven interpoliert. Dieses einfache Fahrzeugmodell wird verwendet, um zunächst die Anfangs und Endzustände der Manöver 2-4 zu berechnen. Dabei wird berücksichtigt, dass die Gaspedalstellung vor und nach dem Schaltpunkt eine gewisse Zeit konstant gehalten werden muss, damit mögliche Schwingungen aus der Konditionierungsphase abgeklungen sind. Das ist wichtig für eine korrekte automatische Komfortbewertung der Schaltung (siehe Albers<sup>78</sup>). Dann wird mit Hilfe des Modells berechnet, wie lange die Konditionierungszeiten zwischen den einzelnen geplanten Schaltungen wären. Ergebnis dieser Berechnungen ist eine Kostentabelle (im Sinne der kombinatorischen Optimierung). Diese wird dann verwendet, um mit Hilfe eines kombinatorischen Optimierers die optimale Reihenfolge zu berechnen. Im Kapitel 2.3.2 werden die Eigenschaften erläutert der möglichen Optimierer anhand, derer die Auswahl individuell getroffen werden muss. In diesem Fall wird eine MST-Heuristik für die kombinatorische Optimierung verwendet.

---

<sup>78</sup> Albers et al. 2012

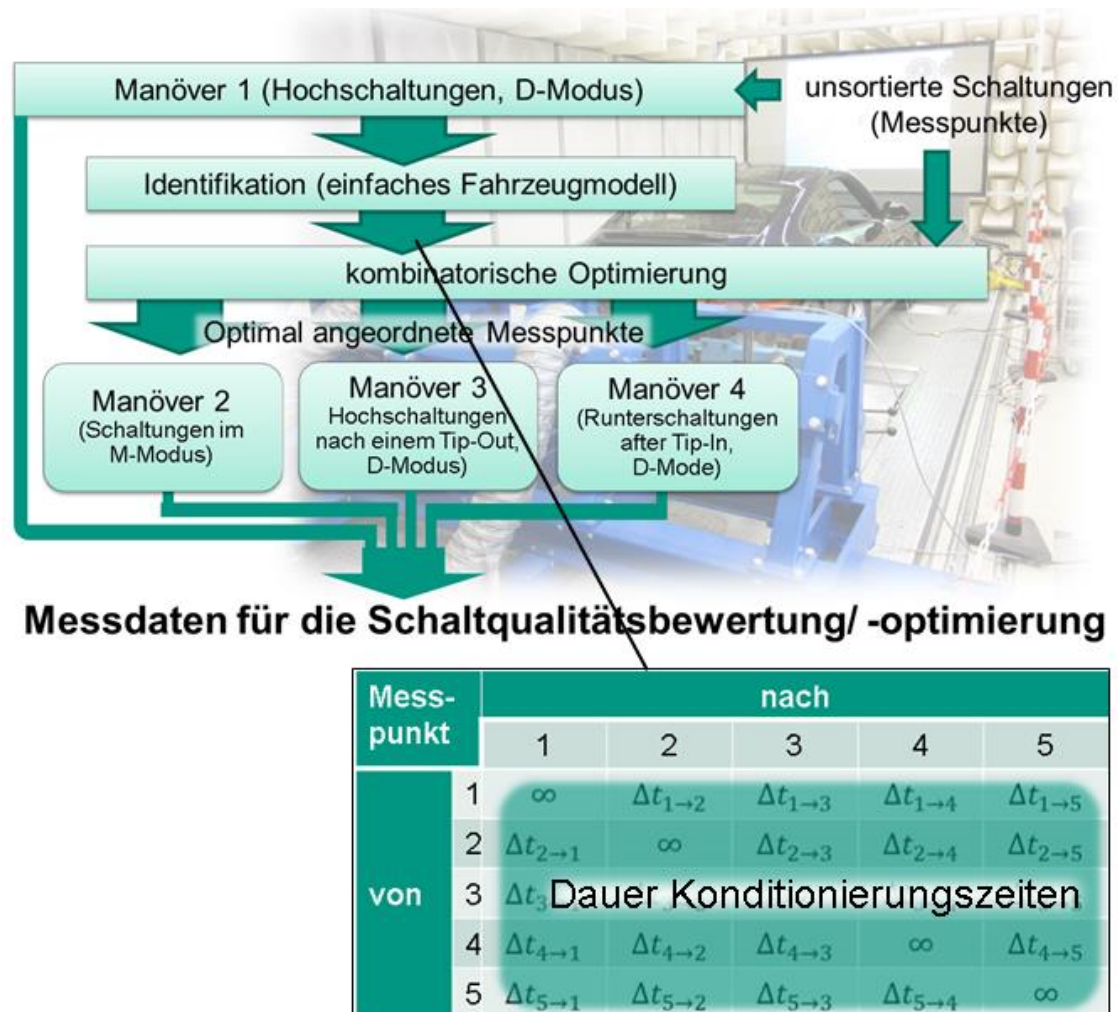
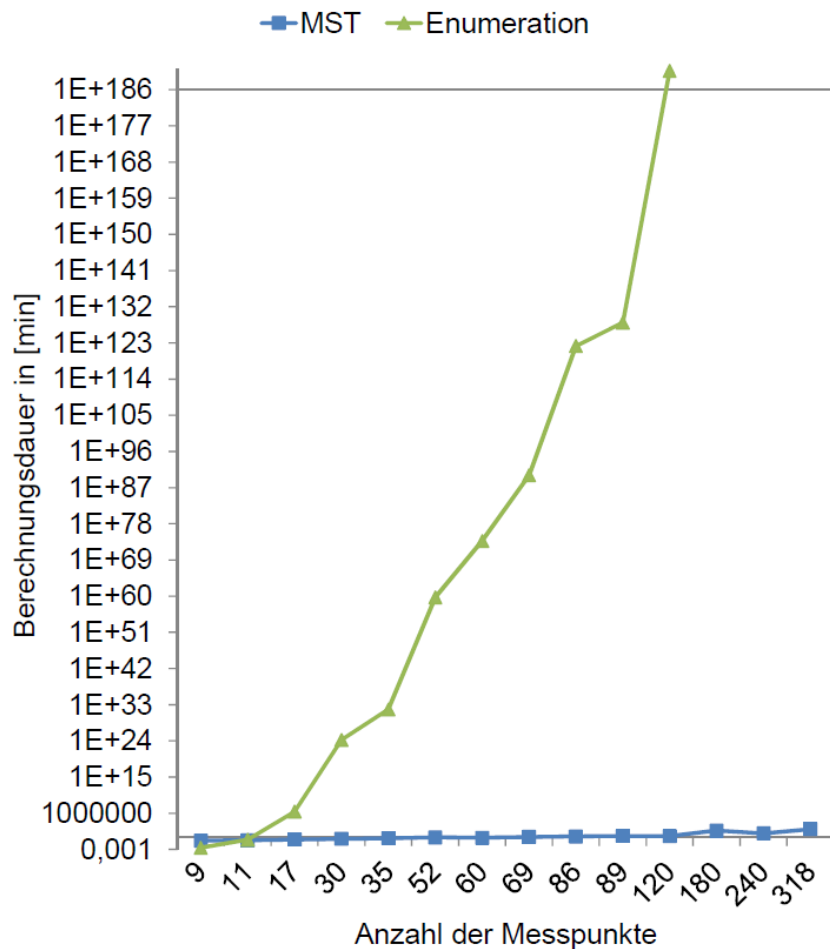


Bild 30: Methode zur Verkürzung von messpunktbasierten Prüfläufen

Experimentelle Analysen nach Römer<sup>79</sup> zeigen eine mittlere Abweichung der Beschleunigungszeiten von 13,5% mit einer Standardabweichung von 6,27%. Außerdem vergleicht Römer die vollständige Enumeration mit der verwendeten MST-Heuristik. Die vollständige Enumeration liefert dabei garantiert das optimale Ergebnis. Dies ist jedoch aufgrund des Rechenaufwands nur bis ca. 20 Messpunkte möglich. Die MST-Heuristik benötigt bei 318 Messpunkten 115 min (siehe Bild 31) zur Berechnung (konventioneller Rechner, Leistungsdaten siehe Anhang 9.2). Bei der MST-Heuristik wird zwar nicht das Optimum erreicht, jedoch lässt sich durch den Einsatz dieser Methode im Vergleich zu einem Expertenfahrer auf der Teststrecke, dem die Schaltungen nicht genau vorgegeben wurden, 6% Zeit einsparen. Diese Einsparung konnte obwohl der virtuelle Fahrer nicht bremsen konnte erzielt werden (was zu einer entsprechenden Verlängerung der Konditionierungszeit führt).

<sup>79</sup> Römer 2011 (betreute Abschlussarbeit)

Bild 31: Rechenzeit kombinatorische Optimierung<sup>80</sup>

Wird in den Konditionierungsphasen der Prüfstand verwendet, um das Fahrzeug zu verzögern, können zusätzlich ca. 75% Zeitersparnis erzielt werden. So können mit Hilfe dieser Methode im Vergleich zum Versuch auf der Teststrecke gezielt die gewünschten Schaltungen durchgeführt werden. Inklusive der Aufspannung und Kalibrierung des Fahrzeugs auf dem Prüfstand ist so eine komplette Bewertung der Schaltungen in ca. vier Stunden möglich. Im Vergleich dauert eine Bewertung auf der Teststrecke ungefähr zwei Tage.

Im Allgemeinen kann a priori keine Aussage gemacht werden, wie viel Zeitersparnis mit dem Verfahren erzielt werden kann. Dabei sind die wesentlichen Einflussgrößen für die Einsparung von Prüfstandszeit die Lage der Anfangs- und Endzustände, sowie der zu erwartete Anteil der Konditionierungszeiten.

<sup>80</sup> Römer 2011(betreute Abschlussarbeit), S.71



Der Aufwand für das Verfahren hängt stark davon ab, wie hoch der Modellbildungsaufwand für die Reihenfolgenoptimierung ist. Dieser wird im Wesentlichen von der Optimierungsaufgabe bzw. von der Dauer des Prüflaufs zur Identifikation der notwendigen Parameter des Modells beeinflusst. Werden dafür extra Versuchsdaten benötigt, muss die Zeit zur Generierung dieser ebenfalls zum Versuchsaufwand addiert werden.

Bei der Optimierung ist eine wesentliche Einflussgröße die Startreihenfolge sowie die Anzahl der zu ordnenden Manöver. Bei einer kleinen Anzahl (<20) kann die vollständige Enumeration verwendet werden. Für eine größere Anzahl sollte ein kombinatorischer Optimierer aus Kapitel 2.3.2 verwendet werden.

## **4.9 Einsatz frei definierbarer Modellstrukturen**

Bei diesem Verfahren werden freidefinierbare Modellstrukturen verwendet, um das Verhalten des Prüflings (UUD) zu modellieren. Dazu wird zunächst eine geeignete Modellstruktur aufgestellt (vgl. Kapitel 4.9.1). Anschließend wird auf Basis dieser Modellstruktur ein geeignetes Identifikationsmanöver definiert. Ziel dieses Manövers ist es die unbekannt Parameter (vgl. Kapitel 4.4) mit Hilfe der Versuchsdaten zu bestimmen. Dazu werden die in Kapitel 4.9.4 Algorithmen erläuterten verwendet. Ergebnis dieser Teilaktivität ist ein Simulationsmodell des Prüflings, der Unit under development (UUD), mit einem vollständigen Parametersatz das für die Optimierung (siehe Kapitel 4.9.6) verwendet werden kann.

### **4.9.1 Art des Modells**

In der Identifikationsphase werden die unbekannt Parameter der Modellstruktur berechnet. Dabei kann das Modell auf physikalischen Gleichungen, nicht physikalisch begründeten Strukturen oder Mischungen aus beidem bestehen (siehe Bild 32).

Vorwissen	Ergebnis	Beispielanwendungen	Anwendungsgrenzen
mögliche Einflussgrößen	Nicht physikalisches Modell	Komfortobjektivierung	Nur in den Grenzen der Eingangsgrößenbereiche
mögliche Einflussgrößen + Struktur	Physikalisches Modell für Struktur + Restfehlermodell	z.B. Antriebsstrang/ Komponentenmodellierung	Anwendung mit den Grenzen der Modellstruktur
physik. Modellstruktur	Parameter für physik. Modellstruktur	z.B. Antriebsstrang, Komponentenmodellierung	

Bild 32: Art der Modelle

Nicht physikalische Strukturen werden häufig verwendet, um das menschliche Komfortempfinden zu modellieren. In diesem Fall ist das parametrisierte Modell nur in dem Bereich gültig, in dem Versuchsdaten zur Identifikation zu Verfügung stehen (Interpolation). Physikalische Modellstrukturen können auch außerhalb dieser Bereiche verwendet werden. Diese unterliegen nur der Modellgenauigkeit. Das heißt, dass das Modell nur die Effekte abbilden kann, die auch modelliert wurden. Ein Vorteil der physikalischen Modellierung ist, dass diese Grenzen bekannt sind. Außerdem können, bei einer physikalischen Modellierung, alle Parameter als Designparameter bei einer Optimierung des Systemverhaltens verwendet werden. Bei Modellen mit nichtphysikalischen Anteilen (Teilsysteme) können die physikalischen Parameter optimiert werden, wenn die nicht-physikalisch modellierten Teilsysteme nur innerhalb deren Grenzen eingesetzt werden (siehe Bild 33). Vorteil bei Modellen mit nicht-physikalischen Anteilen ist, dass Teilsysteme die aufgrund der Komplexität schwer modelliert werden können nicht physikalisch modelliert werden können. So kann der Modellierungsaufwand deutlich reduziert werden. Außerdem ist auch ein geringerer Versuchsaufwand notwendig, da weniger unbekannte Parameter in der Modellstruktur vorhanden sind.

Vorwissen	Aufgabe
mögliche Einflussgrößen	Prädiktion des Ausgangs bei gegebenem Eingang
mögliche Einflussgrößen + Struktur	Zur Prädiktion des Ausgangs bei gegebenem Eingang + <b>Optimierung Parameter</b> nur möglich wenn Einflussgrößen nicht mit Designparameter wechselwirken
Modell-Struktur	Zur Prädiktion des Ausgangs bei gegebenem Eingang + <b>Optimierung Parameter</b>

Bild 33: Einsatzarten der Modelle

#### 4.9.2 Modellbeschreibungsform

Je nach individuellem Produktentwicklungsprozess können schon zu Beginn der versuchsbasierten Optimierung Modelle bzw. Modellstrukturen oder Teilmodelle in verschiedenen Beschreibungsformen bzw. Simulationsumgebungen vorliegen. Im Sinne der durchgängigen Nutzung der Modelle während des Produktentwicklungsprozesses müssen diese auch zur versuchsbasierten Optimierung verwendet werden können. Jedoch können nicht alle Arten von Algorithmen (siehe Kapitel 4.9.4) für alle Modellbeschreibungsformen verwendet werden. Im Folgenden werden die möglichen Beschreibungsformen erläutert:

- geschlossene Modelle  
Häufig liegen Modelle von Dritten, wie Zulieferern, in Form von geschlossenen Modellen vor, da hier das Firmen Know-How enthalten ist. Bei geschlossenen Modellen kann die genaue Modellstruktur nicht eingesehen werden. Nur Eingangssignale und teilweise auch festgelegte physikalische Parameter können vom Nutzer bzw. von den Identifizierungsalgorithmen variiert werden. Bedingung für die Verwendung dieser Modelle ist, dass sämtliche unbekannte Parameter bzw. Designparameter von dem Identifikationsalgorithmus gesteuert werden können. Damit ist es möglich auch domänenspezifische Teilmodelle in teilweise anderen Simulationsumgebungen zu nutzen. Bild 34

zeigt die Kopplung zwischen einem Fahrer-Modell in Matlab- Simulink, einem Antriebsstrangmodell in AVL-Cruise<sup>81</sup> und einem Modell zur Fahrbarkeitsbewertung (AVL-DRIVE<sup>82</sup>). So können Optimierungsalgorithmen von Matlab verwendet werden, um die Fahrbarkeitskriterien zu optimieren.

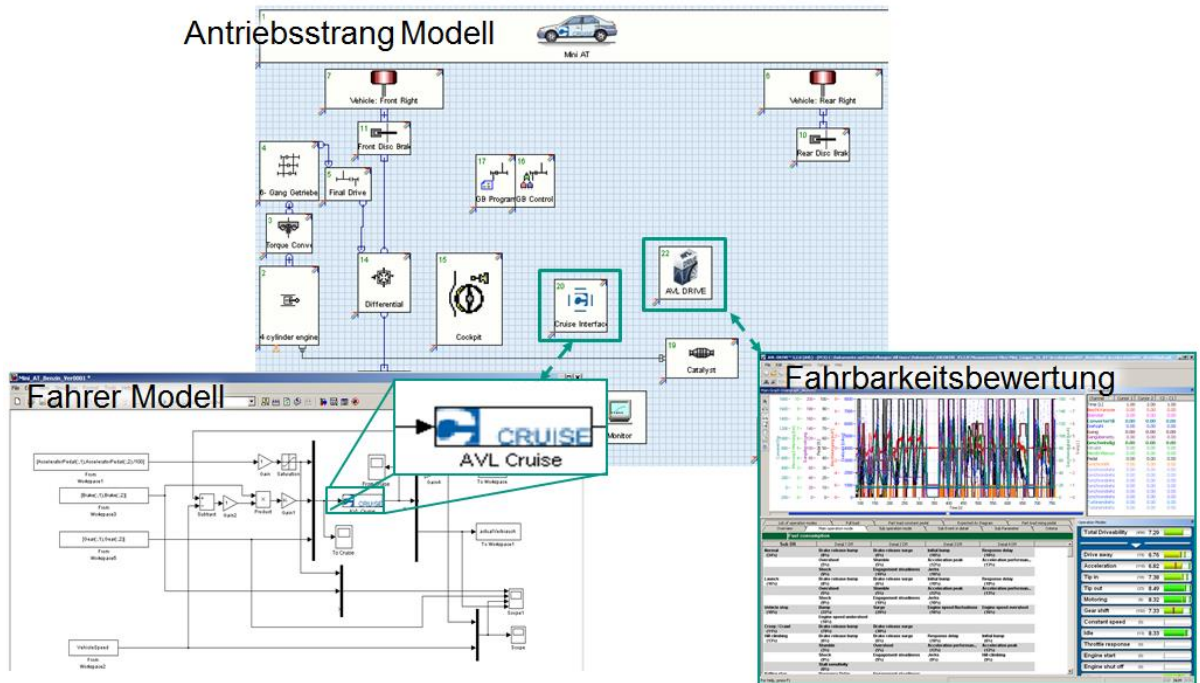


Bild 34: Kopplung Simulationsumgebungen

- offene Modelle  
 Bei offenen Modellen können die Modellstruktur bzw. die Parameter des Modells eingesehen bzw. verändert werden. Diese Parameter stehen dann auch für die Identifikation zu Verfügung. Genau wie die geschlossenen Modelle erfordern offene Modelle Algorithmen, die auch ohne Zugriff auf die genaue Modellstruktur zu einem Ergebnis kommen. Deshalb können für diesen Fall nur Optimierer eingesetzt werden (siehe Kapitel 4.9.4.6). Mit entsprechendem Aufwand können aber auch die Gleichungen aus diesen Modellen extrahiert werden. Dann können deutlich effizientere Algorithmen zur Identifizierung eingesetzt werden (siehe Bild 35)

<sup>81</sup> Gesamtfahrzeugsimulationsumgebung von der AVL LIST GMBH

<sup>82</sup> Softwar zur Fahrbarkeitsbewertung von der AVL LIST GMBH

- definierte Modellstrukturen  
Eine weitere Möglichkeit ist, dass die Modellstruktur in Form definierter (fester) Gleichungen vorliegt die das Verhalten des Systems nicht physikalisch abbilden. Beispiele für solche Strukturen sind polynomiale Modelle, künstliche neuronale Netze oder auch Wiener Hammersteinmodelle. Für diese mathematischen Strukturen existieren spezialisierte Lösungsverfahren, mit denen die entsprechenden Parameter identifiziert werden können (siehe Isermann<sup>83</sup>.)
- Gleichungen/Gleichungssysteme  
Die effizienteste Möglichkeit hinsichtlich des Identifikationsalgorithmus bietet eine offene Modellstruktur in Form von Gleichungen. Hier kann der Algorithmus direkt auf die Modellstruktur zugreifen. Dadurch lassen sich häufig bessere Ergebnisse in wesentlich weniger Rechenzeit erreichen. Bild 35 zeigt einen Vergleich zwischen einem genetischen Optimierungsalgorithmus und einem Least-Squares-Schätzer bei einem quadratischen Modell. In diesem Beispiel kann der genetische Optimierungsalgorithmus trotz einer wesentlich höheren Rechenzeit nicht die gleiche Ergebnisqualität wie die des Least-Squares-Schätzers erreichen.

---

<sup>83</sup>Isermann 1992

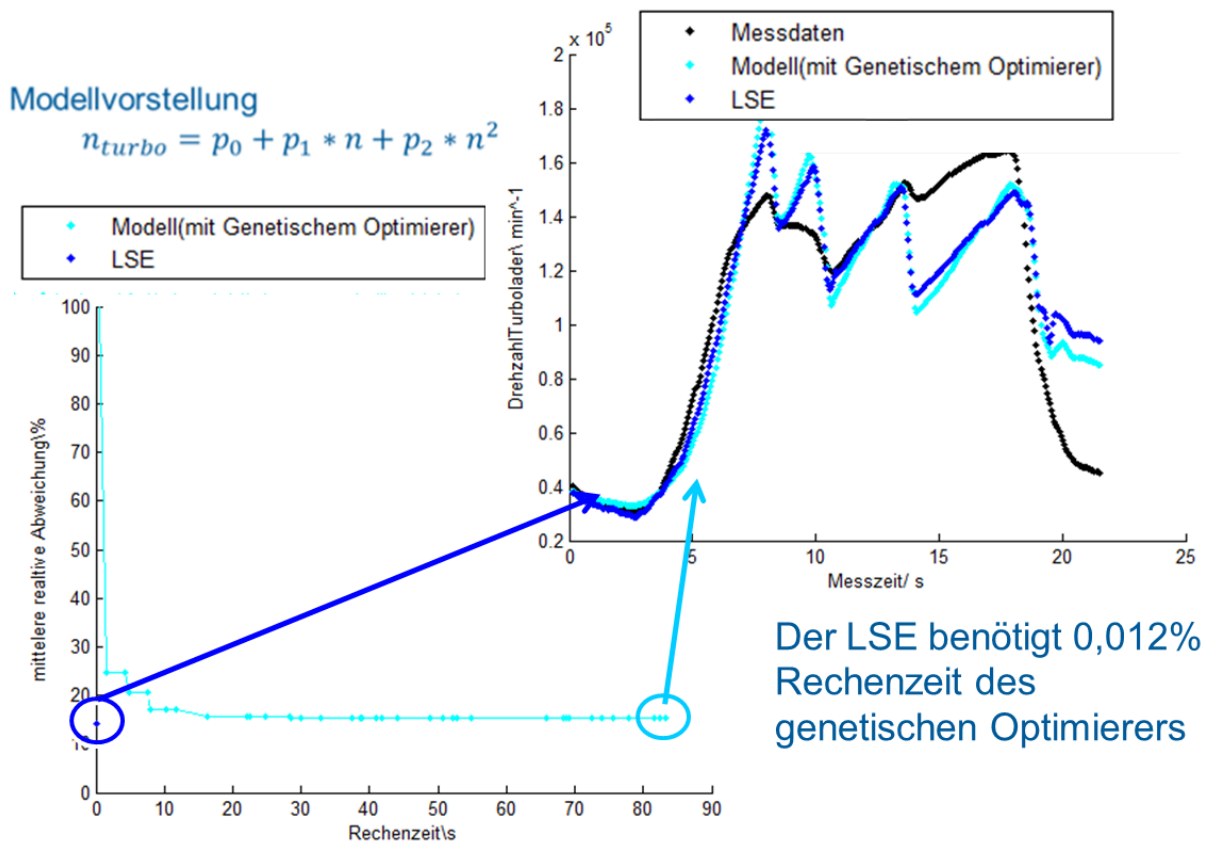


Bild 35: Vergleich genetischer Optimierer mit Least-Squares-Schätzer

### 4.9.3 Versuchsdurchführung und Auswertung

Im Gegensatz zum Einsatz der klassischen DOE-Methode wird bei dem hier verwendeten Verfahren ein Identifikationsmanöver und nicht das bzw. die Bewertungsmanöver für verschiedene Designparametersätze durchgeführt. Dabei können im Idealfall Messdaten aus anderen Versuchen für die Identifikation verwendet werden. In diesem Fall wird nur noch einmal das Bewertungsmanöver mit dem optimal berechneten Designparametersatz zur Verifikation hinsichtlich der betrachteten Kriterien des Modells benötigt.

Stehen keine Versuchsdaten aus anderen Versuchen zur Verfügung, muss zunächst ein Identifikationsmanöver definiert werden. Hier ist zunächst einmal eine Vielzahl von Manövern möglich. Wichtig bei der Wahl des Identifizierungsmanövers ist, dass alle Zustände/Zustandsübergänge, die auch bei den Bewertungsmanövern angeregt werden, auch im Identifizierungsmanöver ausreichend vorhanden sind. Nur so können die unbekannt Parameter der Modellstruktur identifiziert werden. Dies wird



#### 4.9.4 Identifikation

In der Teilaktivität Identifikation werden die unbekannt Parameter so bestimmt, dass sich das Modell wie der reale Prüfling verhält. Das heißt, ein Algorithmus berechnet die Modellparameter so, dass sich das Modell hinsichtlich der Bewertungsgrößen aber auch hinsichtlich der Zustandsgrößen wie der reale Prüfling verhält. Damit kann sichergestellt werden, dass sich das Modell physikalisch korrekt verhält, wenn eine physikalische Modellstruktur gewählt wurde.

Der Rechenaufwand für die Identifikation kann deutlich reduziert werden, wenn das Gesamtmodell in Teilmodelle unterteilt wird. Jedoch müssen sämtliche Ein- und Ausgangsgrößen im Versuch erfasst werden können. Je nachdem, in welcher Form die Modellstruktur vorliegt, zeigt diese Arbeit verschiedene Ansätze zur Identifikation. Für diese Ansätze werden im Rahmen dieser Arbeit die entsprechenden Schnittstellen geschaffen und diese hinsichtlich ihrer Limitierungen untersucht.

##### 4.9.4.1 Identifikation offene Modellstrukturen

Liegt die Modellstruktur in Form von offen zugänglichen Gleichungen vor, können die Parameter durch einen entsprechenden Lösungsalgorithmus bestimmt werden. Die effizienteste Art von Algorithmen für die Systemidentifikation sind tendenziell Schätzer (vgl. Kapitel5). Für den direkten Einsatz von Schätzalgorithmen muss das System in die folgende Form gebracht werden:

$$\vec{y} + \vec{\epsilon}_a = \vec{H} * \vec{p} \quad (4.9.4)$$

$\vec{y} = y_k = (:)$	Schätzgröße(n) (gefilterte Messdaten)
$k = 1..ke$	Laufvariable Messpunkte
$\vec{\epsilon} = \epsilon_{a_k} = (:)$	absolute Abweichung von Modellwert zu Messwert
$\vec{H} = H_{kl} = (:::)$	Designmatrix
$l = 1..le$	Laufvariable Zustandsgrößen
$\vec{\alpha} = \alpha_l = \begin{pmatrix} f_1(\vec{p}) \\ \vdots \\ f_l(\vec{p}) \end{pmatrix}$	Kombinierter Parametervektor



$\vec{p} = p_q = (:$	Parametervektor
$\vec{q} = 1..qe$	Laufvariable Parameter

Im Rahmen dieser Arbeit wird eine allgemeine Form (siehe Gleichung (4.9.5)) sowie eine entsprechende Erweiterung für zusätzliche Randbedingungen, wie zum Beispiel mehrere Zustände oder stückweise definierte Funktionen erarbeitet. Diese Erweiterungen werden in Kapitel 4.9.4.2 bis Kapitel 4.9.4.5 dargestellt. Dadurch lässt sich eine Vielzahl von Systemen bzw. Optimierungsaufgaben darstellen. Für den Fall, dass sich das System nicht in dieser Form darstellen lässt oder nicht auf die Gleichungen der Modellstruktur zugegriffen werden kann, ist es möglich, Optimierer (siehe Kapitel 4.9.4.6) für die Identifikation einzusetzen.

Technische Systeme sind häufig nicht linear in ihren Parametern. Das führt dazu, dass die Designmatrix nicht mit einem Vektor, sondern mit einem kombinierten Parametervektor  $\vec{\alpha}$ , der verschiedene Kombinationen der unbekannt Parameter enthält, multipliziert wird:

$$\vec{y} + \vec{\epsilon}_a = \vec{H} * \vec{\alpha} \quad (4.9.5)$$

Hier liegen in jeder Zeile Kombinationen aus den gesuchten Modellparametern  $\vec{p}$  vor. Ziel ist es, die Parameter  $\vec{p}$  in  $\vec{\alpha}$  so zu berechnen, dass der Fehlervektor  $\vec{\epsilon}_a$  im Sinne der kleinsten Fehlerquadrate minimal wird:

$$\min_p \left( \sum \epsilon_a^2 \right) = \min_p (\vec{\epsilon}_a^T \vec{\epsilon}_a) \quad (4.9.6)$$

Dazu wird dieser Ausdruck nach dem Parametervektor  $\vec{p}$  abgeleitet. Nach Umformen des Terms ergibt sich folgendes Gleichungssystem:

$$\frac{\partial(\vec{\epsilon}^T \vec{\epsilon})}{\partial \vec{p}} = 2 \frac{\partial \vec{\alpha}^T}{\partial \vec{p}} (\vec{H}^T \vec{H} \vec{\alpha} - \vec{H}^T \vec{y}) = 0 \quad (4.9.7)$$

Für den Spezialfall  $\vec{\alpha} = \vec{p}$  bzw. das direkt nach dem Vektor  $\vec{p}$  aufgelöst werden kann, sind verschiedene Schätzer für die Lösung dieses Gleichungssystems möglich. Dafür

eignen sich je nach Struktur Algorithmen zum Lösen von nichtlinearen Gleichungssystemen. Sonst können Schätzer zum Berechnen der unbekannt Parameter verwendet werden. Kapitel 2.3.1.6 zeigt eine Auswahl von Schätzern.

Je nach Rauschanteil können verschiedene Arten von Schätzern verwendet werden. Der Bekannteste und der Effizienteste hinsichtlich des Rechenaufwands ist der Least-Squares-Schätzer. Jedoch ist dieser Schätzer relativ anfällig bei Ausreißern in den Messdaten. Besser sind die Varianten weighted Least-Squares-Schätzer bzw. der iterated Least-Squares-Schätzer bzw. der Ransac-Schätzer. Diese Schätzer sind jedoch deutlich rechenaufwendiger (vgl. Kapitel 2.3.1.6 ).

#### 4.9.4.2 Identifikation offene Modellstrukturen - mehrere (gewichtete) Zustandsgrößen

Sollen bei der Identifikation mehrere Schätzgrößen (Reibkennwerte, Momentenkennlinie) und Zustände (z.B. Gänge oder Kennfeldbereiche) berücksichtigt werden, sieht die Gleichung folgendermaßen aus (Beispiel Kapitel 4.9.4.5):

$$\begin{pmatrix} \vec{y}_{1,1} \\ \vec{b}_1 \\ \vec{y}_{1,2} \\ \vec{b}_1 \\ \vdots \\ \vec{y}_{2,1} \\ \vec{b}_2 \\ \vec{y}_{2,1} \\ \vec{b}_2 \\ \vdots \end{pmatrix} + \vec{\epsilon} = \begin{pmatrix} H_{zust1} & \begin{pmatrix} 0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 0 \end{pmatrix} \\ \begin{pmatrix} 0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 0 \end{pmatrix} & H_{zust2} \end{pmatrix} \begin{pmatrix} \overrightarrow{\alpha_{zust1}} \\ b_1 \\ \overrightarrow{\alpha_{zust1}} \\ b_2 \end{pmatrix} \tag{4.9.8}$$

Wobei folgende Definitionen gelten:

- $\vec{\epsilon} = \epsilon_k = (:) \quad$  gewichteter relativer Fehler
- $\vec{b} = b_k = (:) = o * w \quad$  gewichteter Normierungsfaktor
- $o_j \quad$  Normierungsfaktor
- $w_k \quad$  Gewichtungsfaktor
- $j \quad$  Schätzgrößen

Die verschiedenen Schätzgrößen  $j$  sind dabei mit im Vektor  $\vec{y}_{j,k}$  untereinander enthalten. Wenn Schätzgrößen in verschiedenen Größenordnungen vorliegen, müssen diese normiert werden, da sonst die Schätzgrößen mit höheren absoluten Abweichungen bei der Identifikation höher gewichtet werden. Das wird durch den von der Schätzgröße abhängigen Normierungsfaktor  $o_j$  erreicht. Zusätzlich können einzelne Messpunkte durch den Gewichtungsfaktor  $w_k$  höher bzw. niedriger gewichtet werden. Dadurch können wichtige Punkte oder aber auch verschiedene Vertrauensmaße der einzelnen Messpunkte berücksichtigt werden. Verschiedene Vertrauensmaße können dadurch entstehen, dass Sensoren in unterschiedlichen Bereichen unterschiedliche Genauigkeiten aufweisen. Der Normierungsfaktor  $\vec{b}_z$  kombiniert den Normierungsfaktor  $o_j$  und den Gewichtungsfaktor  $w_k$  in der Gleichung (4.9.8). Diese Gleichung wird für die Schätzung der Parameter im Sinne der kleinsten Fehlerquadrate verwendet:

$$\min_p \left( \sum \epsilon_k^2 \right) = \min_p \left( \sum^{\text{Zustände}} \left( \sum (H * \alpha - y) / b \right) \right) \quad (4.9.9)$$

Um den Rechenaufwand zu senken, sollte der Vektor der unbekannt Parameter  $\vec{\alpha}$  so zusammengefasst werden, dass kein Parameter bzw. keine Parameterkombination mehrfach vorkommt. Ist beispielsweise unter anderem die Differenzialübersetzung unbekannt kommt diese in mehreren Zuständen (In diesem Beispiel in den Zuständen der einzelnen Gänge) vor.

#### 4.9.4.3 Identifikation offene Modellstrukturen - stückweise definierte Funktionen

Häufig werden stückweise definierte Funktionen für die Modellierung technischer Systeme benötigt (Beispiel Wandlormomentenübersetzung siehe Bild 38). Dazu wird die Gleichung (4.9.8) ohne Normierungsfaktoren verwendet. Die einzelnen Zustände sind hier die einzelnen Intervalle der Funktion. Dazu lässt sich dann wieder die Schätzergleichung definieren. Für den Fall, dass Geraden wie in Bild 37 verwendet werden, ergeben sich zwei Teilgleichungen:

$$y_{b1} = y_{a1} + m_1 * x \quad (4.9.10)$$

$$y_{b2} = b_2 + m_2 * x \tag{4.9.11}$$

Dabei lässt sich  $b_2$  so bestimmen, dass sich kein Sprung zwischen den Geradenstücken ergibt.

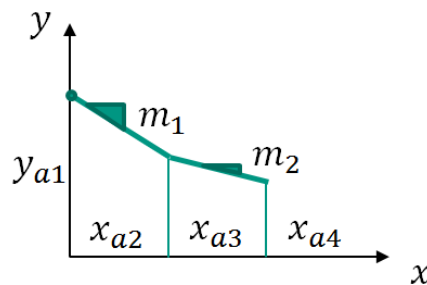


Bild 37: stückweise definierte Funktionen.

In diesem Fall wird der Schnittpunkt  $(x_{a2} | y_{b1}(x_{a2}))$  in beide Gleichungen eingesetzt und miteinander gleichgesetzt:

$$b_2 + m_2 * x_{a2} = y_{a1} + m_1 * x_{a2} \tag{4.9.12}$$

Daraus ergibt sich  $b_2$  zu:

$$b_2 = y_{a1} + m_1 * x_{a2} - m_2 * x_{a2} \tag{4.9.13}$$

Anschließend kann die Schätzergleichung für beide Intervalle aufgestellt werden (1. Zeile Intervall 1; 2. Zeile Intervall 2):

$$y + \epsilon = \begin{pmatrix} 1 & x & 0 \\ 1 & x_{a2} & -x_{a2} + x \end{pmatrix} * \begin{pmatrix} y_{a1} \\ m_1 \\ m_2 \end{pmatrix} \tag{4.9.14}$$

Wenn die Intervallgrenzen bekannt sind, können die Messpunkte direkt in die Matrix einsortiert werden. Danach wird die Matrix direkt verwendet, um die Parameter zu schätzen. Wichtig ist, dass in jedem Bereich genügend Punkte

vorhanden sind. Das heißt, mindestens so viele Punkte wie Parameter in diesem Kurvenabschnitt enthalten sind.

Wenn die Intervallgrenzen (Beispiel siehe Wandlerkennfeld Bild 38) nicht bekannt sind, müssen diese iterativ bestimmt werden. Das heißt, dass in einer Iteration die Intervallgrenzen festgelegt werden müssen. Nach der Festlegung erfolgt die Schätzung. Anschließend wird die beste Schätzung anhand der berechneten Residuen ausgewählt. Für die Ermittlung der optimalen Intervallgrenzen eignet sich ein Optimierer. Hier sollte ein Optimierer gewählt werden, bei dem die Randbedingungen so gewählt werden können, dass die Intervallgrenzen auch tatsächlich nur im Bereich der Messdaten liegen und die Grenzen  $x_a$  nur steigend oder fallend sind. Deswegen werden die Intervallgrenzen kumulativ definiert. Die absolute Position der Intervallgrenze wird dabei wie folgt definiert:

$$x_{re} = \sum_{r=1}^{re} x_{a,re} \quad (4.9.15)$$

Die Grenzen für den Optimierer sind in diesem Fall zwischen 0 und dem Messpunkt mit dem höchsten Abszissenwert gewählt werden. Außerdem sollte die Nebenbedingung für den Optimierer so gesetzt werden, dass die Summe der  $x_a$  kleiner gleich dem Abszissenwert des Messpunkts mit dem höchsten Abszissenwert ist. Dabei können bei vielen Optimierungsverfahren entsprechende Bedingungen an den Designraum gestellt werden. Jedoch ist diese Möglichkeit teilweise nicht in den bestehenden Implementierungen programmiertechnisch umgesetzt.

#### 4.9.4.4 Identifikation offene Modellstrukturen - Berücksichtigung bekannter (konstanter) Parameter

Teilweise sind auch Parameter, wie zum Beispiel Naturkonstanten oder Designparameter, in der Modellstruktur bekannt. In der Antriebstechnik sind zum Beispiel häufig die sogenannten Auto-Motor-Sport-Daten wie die Zylinderzahl, Fahrzeugmasse usw. vorhanden. Diese sollten, dann um Versuchs- und Rechenzeit bei der Identifizierung zu sparen, nicht berechnet/geschätzt werden. Wenn ein oder mehrere Parameter (hier als Beispiel  $\hat{\alpha}_1$ ) bekannt sind, kann die Schätzergleichung so umformuliert werden:

$$\begin{aligned}
 \vec{y} + \epsilon &= \begin{pmatrix} H_{1,1} & \cdots & H_{1,l} \\ \vdots & \ddots & \vdots \\ H_{k,1} & \cdots & H_{k,l} \end{pmatrix} \begin{pmatrix} \alpha_1 = \widehat{\alpha}_1 \\ \alpha_2 \\ \alpha_3 \end{pmatrix} \\
 &= \begin{pmatrix} H_{2,1} & \cdots & H_{1,l} \\ \vdots & \ddots & \vdots \\ H_{k+1,1} & \cdots & H_{k,l} \end{pmatrix} \begin{pmatrix} \alpha_2 \\ \alpha_3 \end{pmatrix} + \begin{pmatrix} H_{1,1} * \widehat{\alpha}_1 \\ \vdots \\ H_{k+1,1} * \widehat{\alpha}_1 \end{pmatrix}
 \end{aligned} \tag{4.9.16}$$

Daraus lässt sich die neue Schätzergleichung formulieren:

$$\underbrace{\vec{y} - \begin{pmatrix} H_{1,1} * \widehat{\alpha}_1 \\ \vdots \\ H_{k+1,1} * \widehat{\alpha}_1 \end{pmatrix}}_{y^*} + \epsilon = \underbrace{\begin{pmatrix} H_{2,1} & \cdots & H_{1,l} \\ \vdots & \ddots & \vdots \\ H_{k+1,1} & \cdots & H_{k,l} \end{pmatrix}}_{H^*} \underbrace{\begin{pmatrix} \alpha_2 \\ \alpha_3 \end{pmatrix}}_{p^*} \tag{4.9.17}$$

#### 4.9.4.5 Identifikation offene Modellstrukturen - rechnergestützte Umsetzung

Im Rahmen dieser Arbeit werden diese Ansätze rechnergestützt am Beispiel eines Fahrzeugs mit Wandlergetriebe umgesetzt. Der erste Schritt für diese Aufgabe ist die Erstellung der Modellstruktur. In diesem Beispiel wurde ein relativ einfacher Aufbau des Antriebsstrangs ohne Elastizitäten gewählt (siehe Bild 38). Dabei wird in diesem Beispiel aus der Gaspedalstellung mit Hilfe des Gaspedalkennfelds die Drosselklappenstellung berechnet. Diese wird dann verwendet um das Motormoment zu bestimmen, welches dann am Wandler anliegt. Der Wandler ist dabei als ein Kennfeld der Momentenübersetzung und einer idealisierten Überbrückungskupplung ausgeführt. Dann werden die Momente im Getriebe sowie im Differential mit dem entsprechenden Wirkungsgrad und der dazugehörigen Übersetzung umgerechnet.

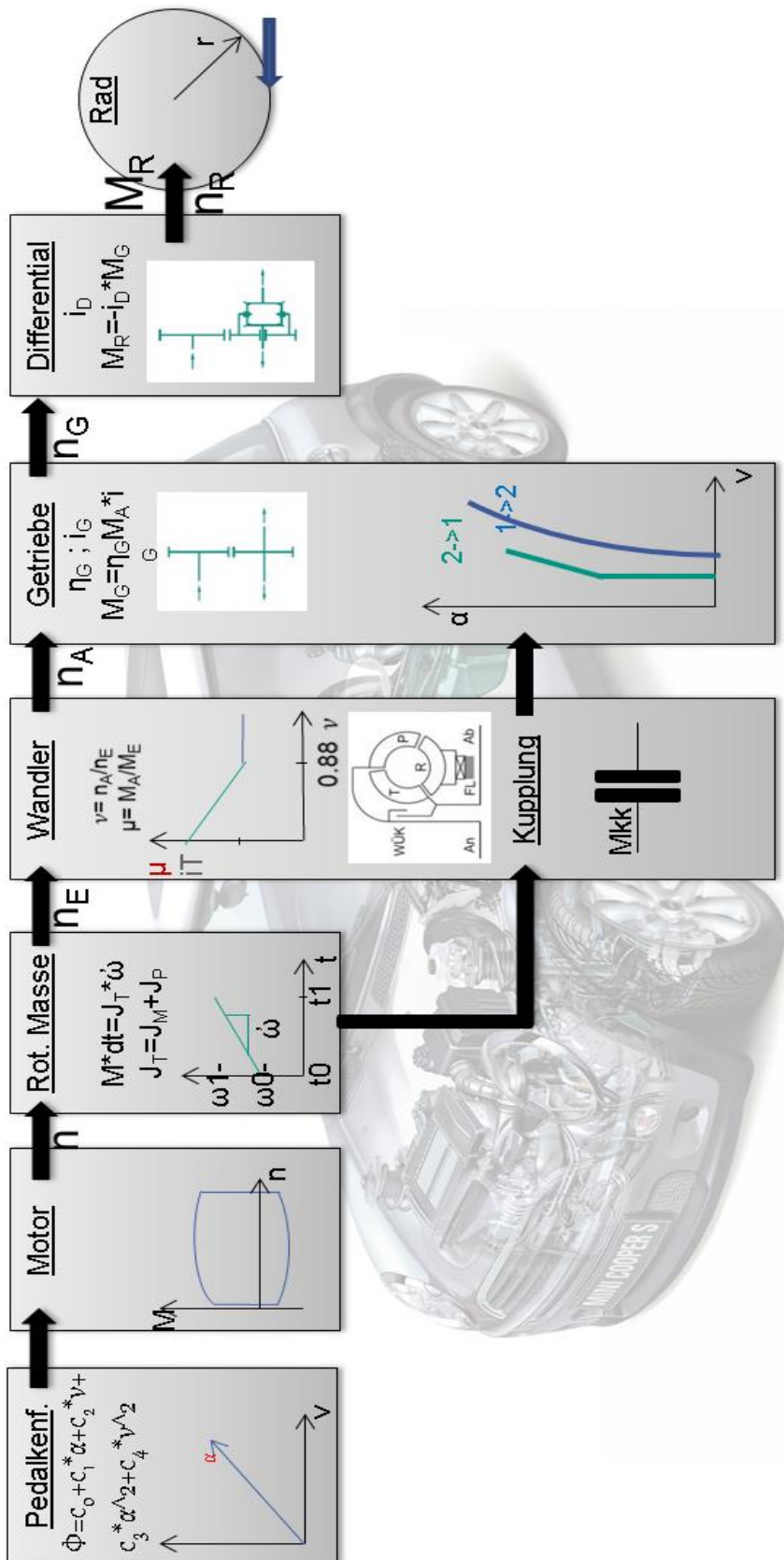


Bild 38: Modellstruktur Fahrzeug mit Wandlergetriebe

Der nächste Schritt ist die Zerteilung des Gesamtsystems in geschlossene Teilsysteme. Mit diesem Schritt kann erheblich Rechenzeit gespart werden, da nicht für jede Variation des Teilsystems das Restsystem berechnet werden muss. In diesem Beispiel werden vorab die Getriebeübersetzungen berechnet. Dazu wird aus der Eingangs- und Ausgangsdrehzahl des Getriebes die effektive Übersetzung für einen Versuchsdatensatz berechnet. Aus der Häufigkeitsverteilung der Messpunkte lassen sich die Übersetzungen der einzelnen Gänge ablesen (siehe Bild 39).

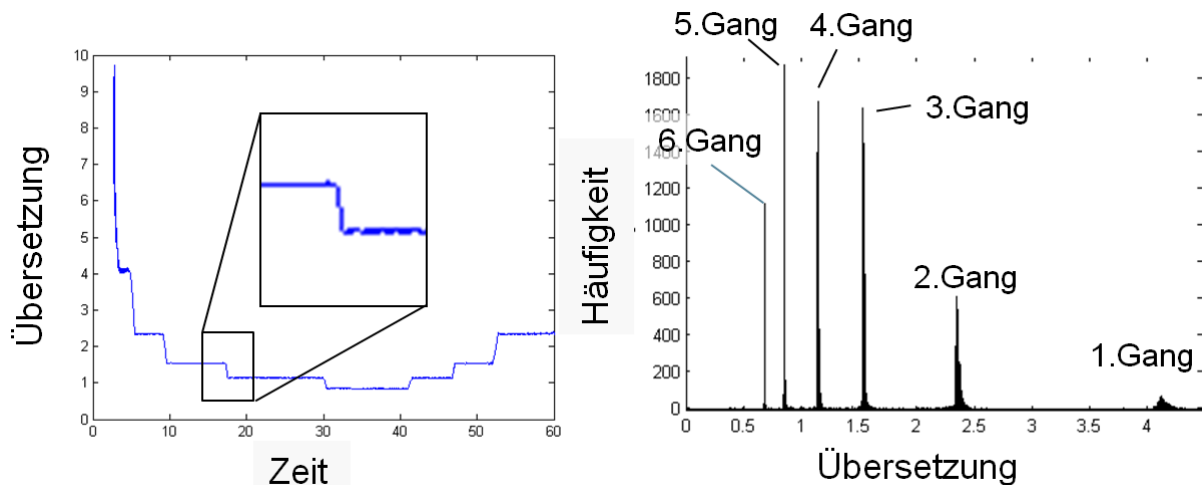


Bild 39: Häufigkeitsverteilung

Außerdem werden vorab die Vollastkurve und die Schubkurve des Motors mit Hilfe von Sensor- bzw. CAN-Daten berechnet. Voraussetzung ist, dass diese Punkte auch in den Versuchsdaten enthalten sind. Im nächsten Schritt müssen die Gleichungen der einzelnen Komponenten formuliert werden.

Durch einen entsprechenden Maple-code (siehe Anhang 9.1) wird dann automatisch eine Schätzergleichung formuliert. In diesem Fall ist es eine stückweise definierte Differenzialgleichung. Für das Einsetzen der Daten wird diese Gleichung in Form von zwei Excel-Tabellen in Matlab überführt. Dabei enthält die Tabelle *Bed* die verschiedenen Kombinationen an möglichen Zuständen. Die andere Tabelle enthält für jeden dieser Zustände die passende nach den Parametern geordnete Matrix. Außerdem wird die Liste der Parameter übergeben. In Matlab werden dann die Messdaten anhand der Bedingungen in die Matrix eingesetzt. So ergibt sich eine große Schätzmatrix, die dann für die Bestimmung der Parameter verwendet werden kann. Bild 40 zeigt den berechneten Verlauf der Längsbeschleunigung einer Messfahrt, der mit verschiedenen Identifikationsverfahren berechnet wurde, im Vergleich zur Messung. Dabei wurde bei jedem Verfahren die gleiche Modellstruktur zugrunde gelegt. Wichtig für die Beurteilung der Modellqualität ist, dass die hohen



Frequenzanteile nicht durch die Modellstruktur abgebildet werden können, da in der Modellstruktur keine Elastizitäten im Antriebsstrang vorgesehen sind. Dabei zeigt Bild 40 ebenfalls einen Vergleich zwischen einem einfachen Least-Squares Schätzer und dem iterated Least-Squares Schätzer. Der iterated Least-Squares Schätzer zeigt dabei ein besseres Ergebnis, benötigt aber auch mehr Rechenzeit. Hier wurden 10 Iterationen gewählt, die eine 10-mal längere Rechenzeit verursachen. Jedoch können auch andere Schätzverfahren, wie in Kapitel 2.3.1.6 beschrieben, verwendet werden.

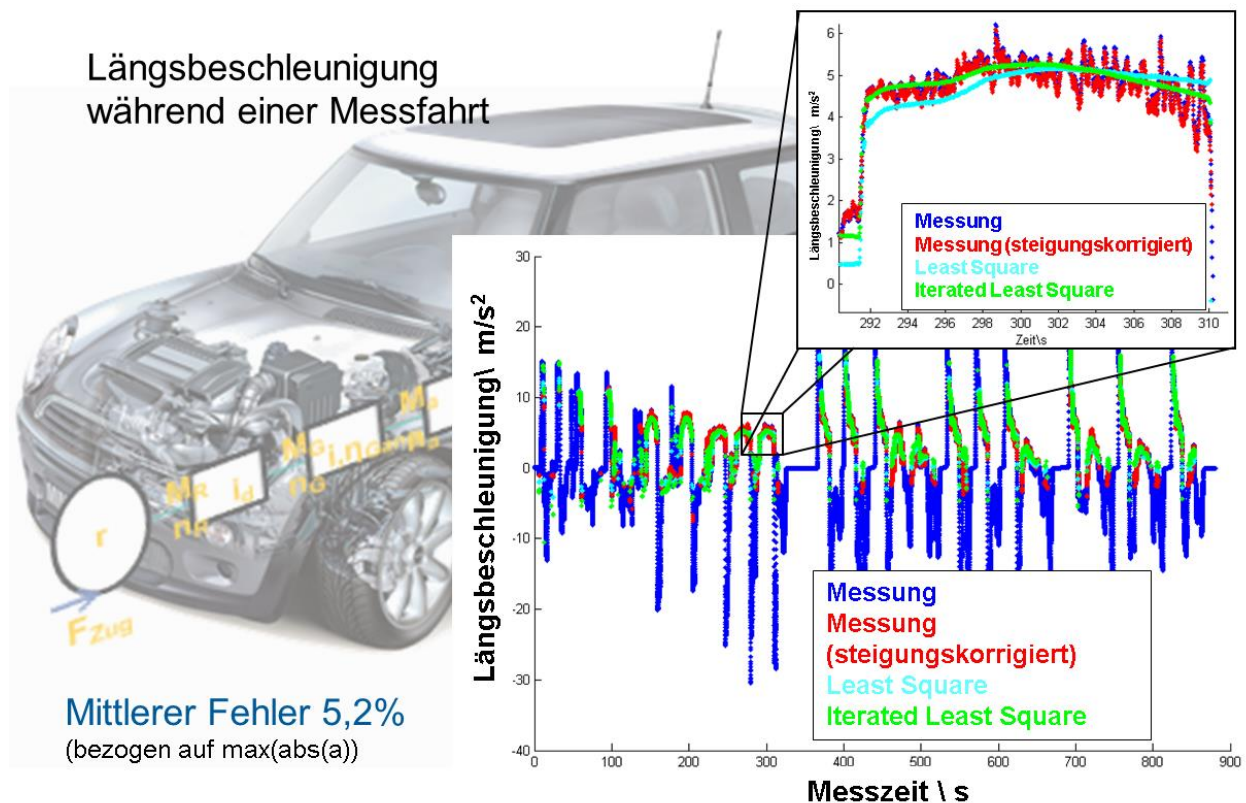


Bild 40: Ergebnis Identifikation Modellstruktur

#### 4.9.4.6 Identifikation geschlossene Modellstrukturen mittels Optimierer

Kann das System nicht in diesen Formen beschrieben werden oder liegt das Modell als geschlossenes Modell (siehe Kapitel 4.9.2) vor, können verschiedene Optimierungsalgorithmen (siehe Kapitel 2.3.1) wie z.B. genetische Optimierer verwendet werden. Damit können vorhandene Modelle bzw. Teilmodelle aus früheren Entwicklungsphasen bzw. von ähnlichen Produkten oder

Vorgängermodellen genutzt werden. So kann der Modellbildungsaufwand reduziert werden. Dies entspricht der von Albers<sup>84</sup> beschriebenen Produktgenerationsentwicklung, da im Allgemeinen davon ausgegangen werden kann, dass Informationen aus dem Zielsystem und dem Objektsystem früherer Produktgenerationen vorliegen.

Ein weiterer Vorteil ist, dass dem Optimierer oft (je nach Implementierung) noch Nebenbedingungen zum Parameterraum gegeben werden können. Sollen zum Beispiel die Anteile eines Gemisches optimiert werden, kann jeder Bestandteil von 0-100% variiert werden. Die Nebenbedingung hier ist, dass alle Anteile zusammen 100% ergeben müssen. Wird diese Nebenbedingung gesetzt, werden bei der Optimierung keine Variationen berechnet, die diese nicht erfüllen. Das heißt es werden keine Gemische berechnet deren Anteile zusammen mehr oder weniger als 100% ergeben. Zur Beurteilung der verschiedenen Verfahren wurde eine entsprechende rechnergestützte Analyseumgebung geschaffen. Die dazugehörigen Ergebnisse sind in Kapitel 4.9.8 dargestellt.

Ein komplexes Beispiel mit einem geschlossenen Modell liefert Albers<sup>85</sup>. Hier wird ein Modellstruktur eines Hybridfahrzeugs auf dem Rollenprüfstand modelliert. Ziel dieses Modelles ist die Abbildung der Dynamik sowie des Lade- bzw. Entladeverhalten des Testfahrzeugs. Mit Hilfe, dieses Modells sollen geplante Versuche mit definierten Startladezuständen schnell angefahren werden und so angeordnet werden können, dass möglichst wenig Konditionierungszeit zwischen den Versuchen benötigt wird. (Siehe Kapitel 4.8.2)

---

<sup>84</sup> Albers 2010a

<sup>85</sup> Albers et al 2013b

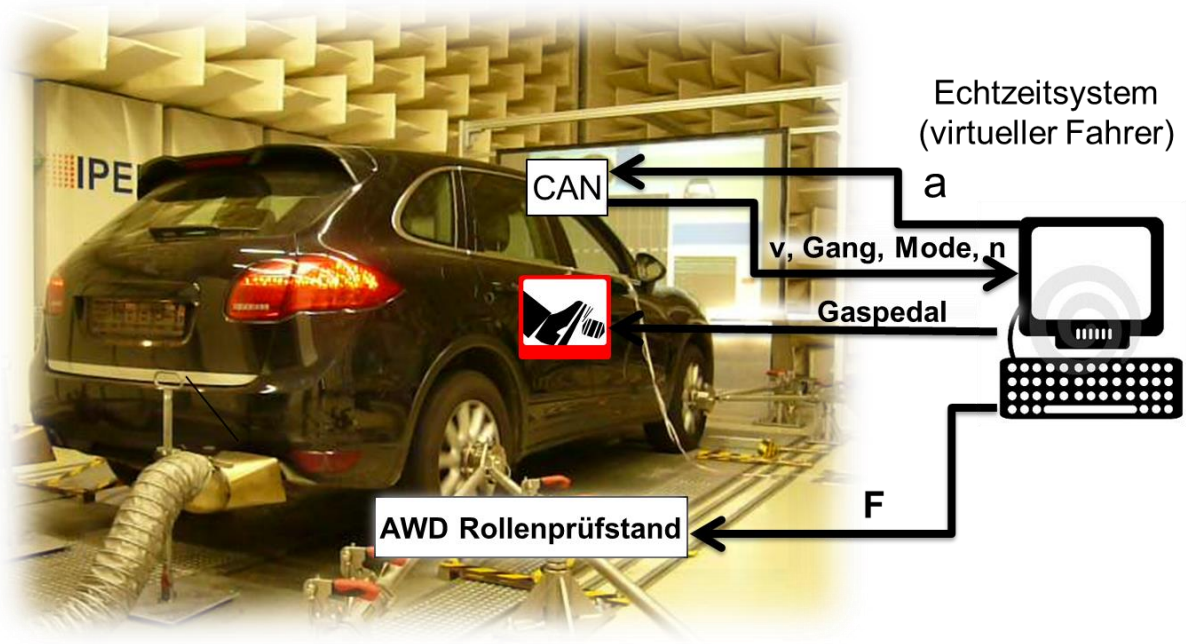


Bild 41: Hybridfahrzeug auf dem Rollenprüfstand

In diesem Fall liegt eine Matlab-Simulinkmodellstruktur (siehe Bild 42) vor.

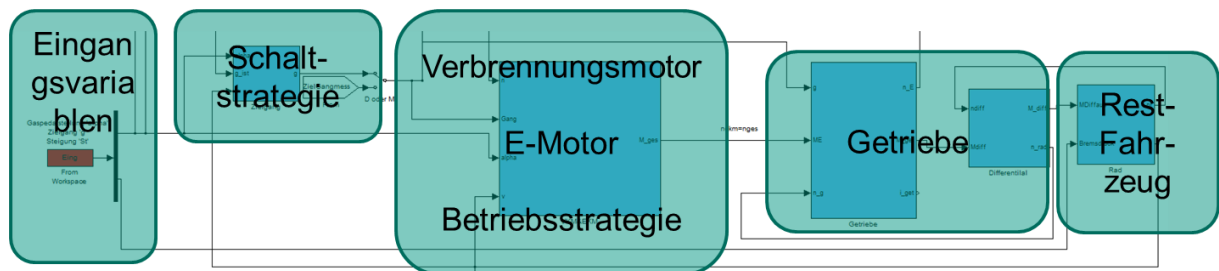


Bild 42: Modellstruktur

Mit Hilfe eines genetischen Optimierers und entsprechenden Versuchsdaten werden die fehlenden Parameter des Modelles wie die die Zusschalt-/Schaltlinien sowie die Motorenkennfelder identifiziert. Als Identifikationsmännöver wurden in diesem Fall Beschleunigungen mit konstanter Gaspedalstellung sowie verschiedene Konstantfahrten verwendet.

#### 4.9.5 Modellverifikation

Nach der Identifikation der unbekannt Parameter der Modellstruktur erfolgt die Aktivität der Modellverifikation. Hier müssen zwei Aspekte sichergestellt werden:

- Das Modell muss sich hinsichtlich der betrachteten Kriterien wie der Prüfling verhalten
- Das Modell muss sich physikalisch verhalten

Nach Stachowiak<sup>86</sup>, besitzt jedes Modell das Verkürzungsmerkmal. Das heißt, dass das Modell nicht alle Aspekte des Originals erfüllt bzw. erfüllen kann. Deswegen muss in dieser Teilaktivität überprüft werden, ob das Modell sich hinsichtlich der betrachteten Kriterien mit ausreichender Qualität wie der Prüfling verhält. Dazu werden die Versuche zunächst simuliert und anschließend mit den Versuchsdaten abgeglichen. Dabei kann die Modellqualität hinsichtlich der betrachteten Kriterien nur nach der Durchführung des Bewertungsmanövers bewertet werden. Mit den Versuchsdaten des Bewertungsmanövers kann der Einfluss der Modellabweichung auf die Kriterien bzw. auf den Zielfunktionswert, durch einen direkten Vergleich quantifiziert werden (siehe Bild 43).

Die zweite Anforderung ist, dass sich das Modell mit den relevanten Genauigkeiten physikalisch verhält. Nur so kann sichergestellt werden, dass sich das Modell auch mit einem anderen Designparametersatz so wie der reale Prüfling verhält. Dazu werden neben den Kriterien auch Ausgangs- sowie Zustandsgrößen abgeglichen. Außerdem müssen die Modellparameter in realistischen Grenzen liegen.

Für den Optimierungsalgorithmus wird ein Kriterium benötigt, das die Modellqualität sowie die Abweichung der Zustands- und Ausgangsgrößen repräsentiert. Für die Bewertung dieser Abweichung werden geeignete Gütekriterien benötigt.

Eine Möglichkeit ist die gewichtete Summe der relativen Abweichung der Bewertungskriterien sowie der Gütekriterien.

Zur Bewertung der Modellabweichung bei Least-Square-Schätzungen wird häufig der  $R^2$  Wert (siehe Formel (4.9.18)) bzw. abgewandelte Formen davon verwendet.

---

<sup>86</sup> Stachowiak 1973

$$R^2 = \frac{\sum H * p - \bar{y}}{\sum y - \bar{y}} \quad (4.9.18)$$

Mit steigender Ordnung des polynomialen Modells wird die Modellabweichung geringer. Ist die Anzahl der Parameter in der gleichen Größenordnung wie die Anzahl der Punkte, lernt das Modell die Punkte auswendig. Deshalb berücksichtigt der  $R_{adj}^2$  die Anzahl der Modellparameter  $k$  siehe (2.4.6):

$$R_{adj}^2 = 1 - \frac{(1 - R^2)(n - 1)}{(n - k - 1)} \quad (4.9.19)$$

Der Kennwert  $R_{pred}^2$  bewertet die Vorhersagekraft des Modells, indem betrachtet wird wie sich das Ergebnis ändert, wenn einzelne Messwerte zur Identifikation nicht zu Verfügung stehen. Diese Gütekriterien hängen jedoch von der Anzahl der Messpunkte ab. Deswegen können Modelle, die auf Basis unterschiedlich langer Versuche gebildet werden, nicht miteinander verglichen werden. Auf Grund dieses Zusammenhangs kann anhand des  $R^2$  nicht beurteilt werden, ob die Modellqualität durch zusätzliche Versuche gesteigert werden kann bzw. sich gesteigert hat. Außerdem ist das  $R^2$  Gütekriterium nicht anschaulich, was die Beurteilung der Modellqualität schwierig macht.

Besser ist die Erstellung der Abweichungsdichtefunktion mit relativen und absoluten Werten für jede Zustands- und Ausgangsgröße. Bild 43 zeigt ein Beispiel einer solchen Verteilung für eine der Flussgrößen z.B. der Fahrzeugbeschleunigung. Im Bild weicht diese im Mittel um 5% ab. Ob diese Modellqualität ausreichend ist hängt jedoch vom Anwendungsfall bzw. der Kriterien ab. Um entscheiden zu können ob die Modellqualität für den Anwendungsfall ausreicht, sind neben dem Schaubild weitere statistische Kennwerte der Abweichungen sowie der Mittelwert und die Varianz angegeben.

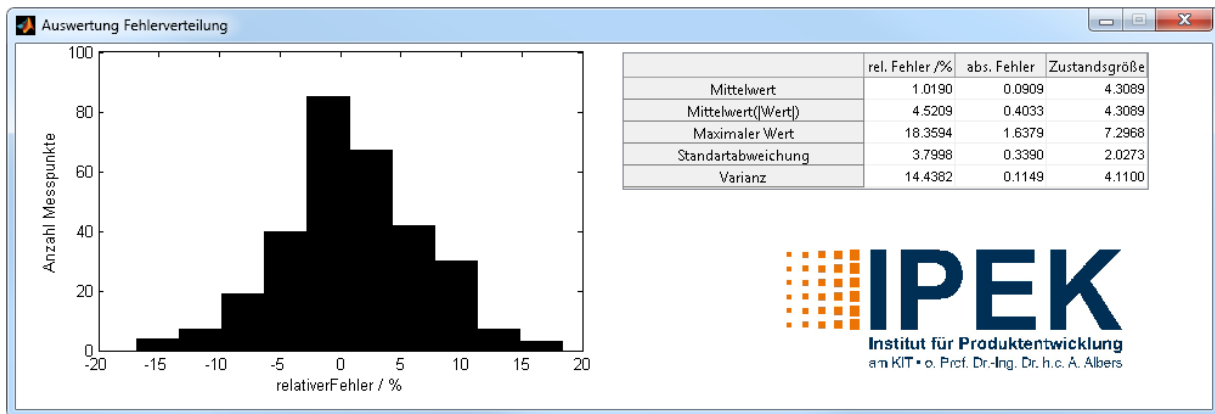


Bild 43: Abweichungsdichtefunktion

Zur Bewertung der Abweichung eignet sich das folgende Gütekriterium (mittlere relative Abweichung):

$$GK = \text{Mean}\left(\frac{\text{Abs}(\epsilon)}{\text{Mean}(\text{Abs}(y))}\right) \quad (4.9.20)$$

Dabei wird die Abweichung auf die mittlere absolute Zustandsgröße bzw. Ausgangsgröße bezogen. So ist gewährleistet, dass einzelne Ausreißer in den Messdaten das Gütekriterium nicht verfälschen. Außerdem werden die absoluten Größen verwendet, damit auch Schwingungsgrößen bewertet werden können. Ein Beispiel in der Fahrzeugentwicklung ist auch hier die Fahrzeuglängsbeschleunigung, deren Mittelwert eines Fahrversuchs im Mittel nahe Null ist, wenn das Fahrzeug bei Versuchsbeginn und –ende steht. Würde nicht der Betrag der Beschleunigung als Bezugsgröße verwendet werden, würden schon kleine Abweichungen für sehr große Kennwerte sorgen.

#### 4.9.6 Bestimmung der optimalen Designparameterwerte

Ergebnis der Identifikation ist ein virtueller Prüfling. Dieser wird für die Berechnung der optimalen Designparameter verwendet. Dabei wird für die Optimierung das Bewertungsmanöver verwendet. Der wesentliche Vorteil gegenüber dem physischen Versuch ist, dass virtuelle Versuche in kürzerer Zeit durchgeführt werden können, da die Manöver oft schneller als Echtzeit berechnet werden können. Dabei wird ein Optimierer (siehe Kapitel 2.3.1) verwendet, um die optimalen Designparameterwerte zu berechnen. Außerdem entfallen im virtuellen Versuch Umrüste- sowie Konditionierungszeiten. Bei der in dieser Arbeit betrachteten Schaltabstimmung

würde das Fahrzeug nur in einer Konfiguration auf dem Prüfstand vermessen. Danach würde die Fahrzeugmodellstruktur mit Hilfe der Versuchsdaten parametrisiert. Dieses Modell würde dann für die Berechnung der optimalen Designparameter verwendet werden. Anschließend würde das Bewertungsmanöver, das heißt in diesem Beispiel die Schaltungen, mit den zuvor berechneten (optimalen) Designparameter einmal durchgeführt werden. So werden weitere Designparametervariationen im Versuch eingespart.

#### 4.9.7 Verifikation Auslegung

Für die abschließende Bewertung der Modellqualität muss das Bewertungsmanöver mit dem physischen Prüfling mit den berechneten (optimalen) Designparametern durchgeführt werden. Nach dem Verifikationsversuch können folgende Kriterien (siehe Bild 44) zur Bewertung der Modellqualität bei dem optimalen Designparametersatz ausgewertet werden:

- Abweichung Zielfunktionswert
- Abweichung Einzelkriterien
- Abweichung Zustands- und Ausgangsgrößen

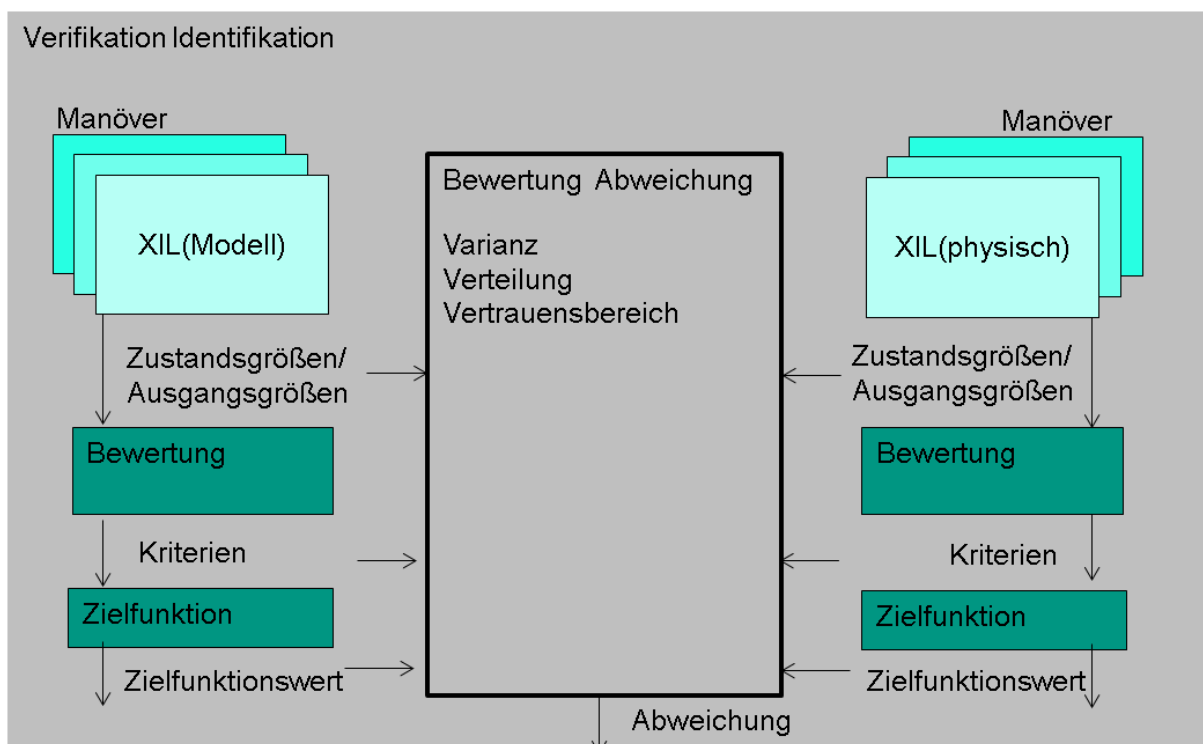


Bild 44: Modellbewertung

Reicht die Qualität nicht aus, werden zusätzlich die Messdaten des Verifikationsversuchs verwendet, um die Modellstruktur anzupassen. Bei ausreichender Qualität stehen die optimalen Designparameter fest. Ein Nebenprodukt ist ein verifiziertes Modell. Dieses Modell kann dann für weitere Optimierungsaufgaben verwendet werden (siehe Kapitel 6). Zum Beispiel können Teilmodelle im XiL-Kontext verwendet werden. Werden beispielsweise unbekannte Parameter wie die Differenzialreibung im Rahmen von Schaltabstimmungen identifiziert, kann das entsprechende Teilmodell auch beispielsweise für die Restsystemsimulation bei der Motorabstimmung verwendet werden. Darüber hinaus können diese verwendet werden um das Identifikationsmanöver für ähnliche Optimierungsaufgaben zu optimieren. Im obigen Beispiel könnte das Fahrzeugmodell aus der Schaltabstimmung als Grundlage verwendet werden um den Versuch der Motorabstimmung zu optimieren. Das heißt, die Eingangsgrößen des Prüflings werden so berechnet, dass die noch unbekannt Parameter möglichst gut identifiziert werden können.

#### **4.9.8 Analyse Einsatz frei definierbarer Modellstrukturen**

Für die Anwendung des Teilhandlungssystems als Prozess (siehe Kapitel 4.2) wurde im Rahmen dieser Arbeit ein rechnergestütztes Werkzeug (siehe Bild 45 bzw. Anhang 9.3) geschaffen. Zusätzlich wurde eine Analyseumgebung geschaffen, mit der es möglich ist, verschiedene Variationen sowohl der Optimierungsaufgabe (siehe Kapitel 4.9.8.3) als auch der Einstellung der Algorithmen (siehe Kapitel 4.9.8.4) innerhalb der Verfahren zu simulieren. Dafür greift diese Umgebung automatisiert auf dieses Werkzeug zu. Dabei wird beim Einsatz stochastischer Optimierungsverfahren das Verfahren automatisiert mehrfach durchgeführt, bis eine entsprechende statistische Aussage über das Optimierungsergebnis getroffen werden kann.

Das DOE-Verfahren stellt bisher hinsichtlich der Versuchszeit das effizienteste Verfahren dar. Deshalb wird in jedem Szenario das Verfahren mit frei definierbaren Modellstrukturen dem DOE-Verfahren zur Analyse gegenübergestellt. Anhand von verschiedenen Kennwerten (siehe Kapitel 4.9.8.2) wird die Leistungsfähigkeit des Verfahrens bewertet.



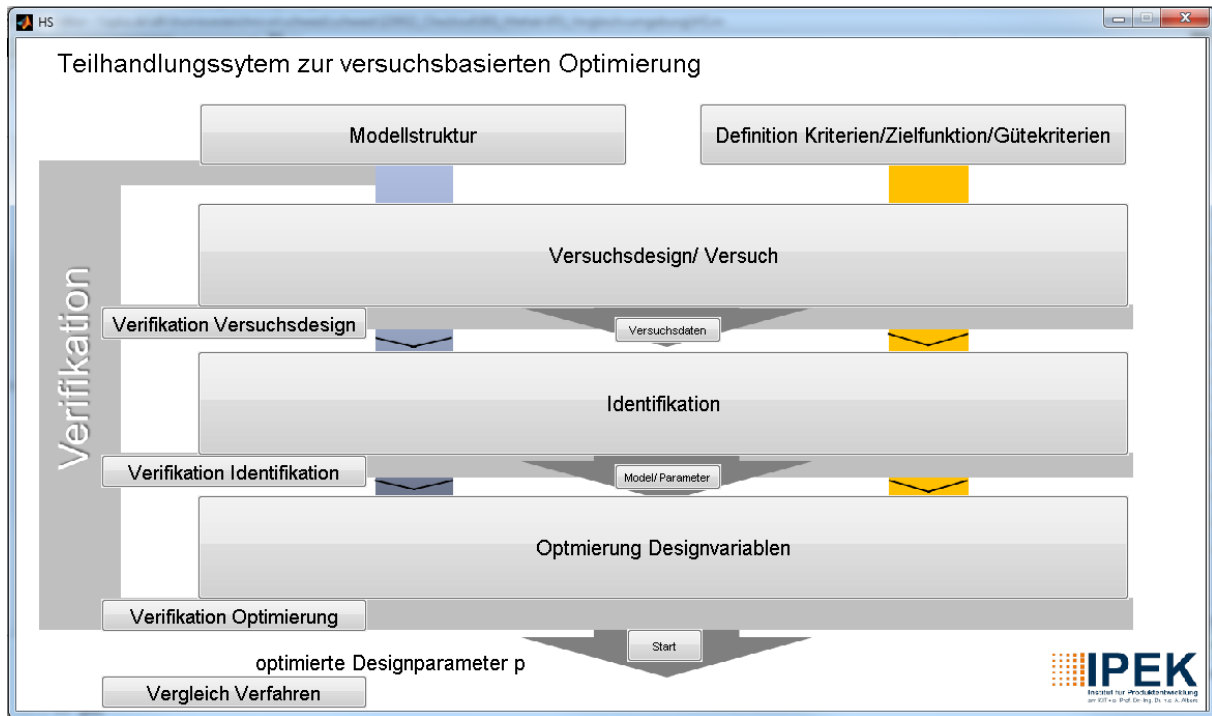


Bild 45: Rechnergestütztes Werkzeug

#### 4.9.8.1 Der Einmassenschwinger als Beispiel für die Vorgehensweise

Zur groben Quantifizierung der Einflüsse wird in den folgenden Kapiteln ein Einmassenschwinger als einfache Optimierungsaufgabe verwendet (siehe Bild 46). Ziel dieser Optimierungsaufgabe ist es die optimalen Werte für die Feder- und Dämpferkonstante, hinsichtlich zweier in einer Zielfunktion gewichteter Kriterien, versuchsbasiert zu berechnen. Als Bewertungsmanöver wird ein definierter Sprung verwendet. Als Kriterien werden die maximale Auslenkung und die Fläche unter der Auslenkung verwendet. Die Masse des Einmassenschwingers ist für die Algorithmen unbekannt.

System: Einmassenschwinger			
Bewertungsmanöver:			
Kriterien:			} Zielfunktion
Gewichtung:	frei wählbar (Standard	1 : 1 )	
Designparameter:	c, d		
unbekannte Parameter:	m		

Bild 46: Beispiel Optimierungsaufgabe

Anstatt einem physischen Prüfling wird ein Simulationsmodell (siehe Bild 47) eingesetzt, bei dem die Zustandsgrößen verrauscht werden. Damit lassen sich verschiedene Rauschanteile sowie Modellfehler simulieren, um so die Robustheit der Algorithmen zu bestimmen.

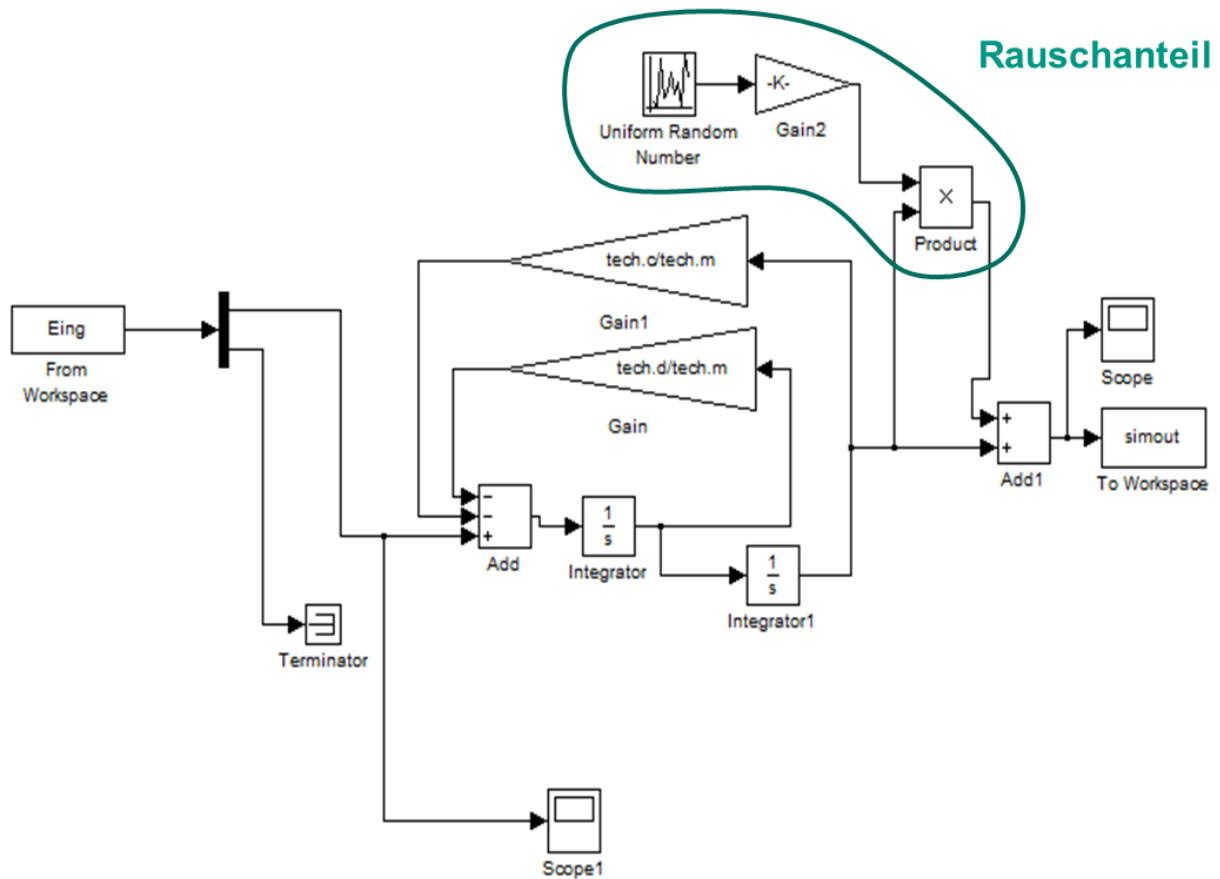
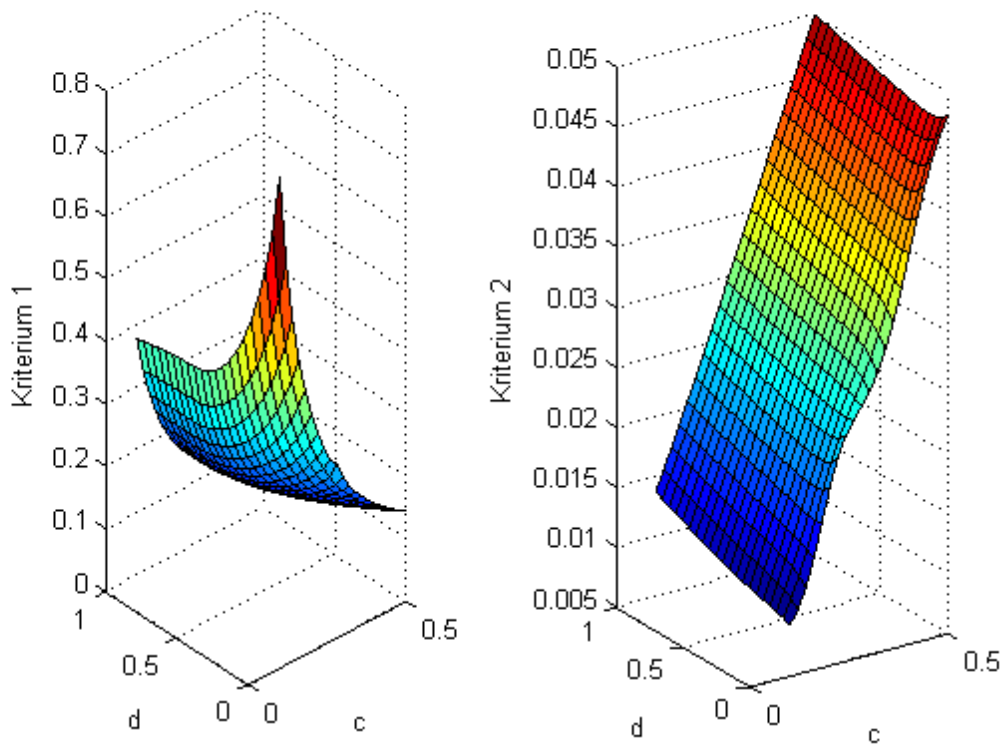


Bild 47: Virtueller Prüfling (Einmassenschwinger)

In Bild 48 sind die Einzelkriterien bzw. der Zielfunktionswert der Beispielaufgabe für den gesamten Designraum abgebildet.



Optimum bei  $c=0.16$   $d=0.65$

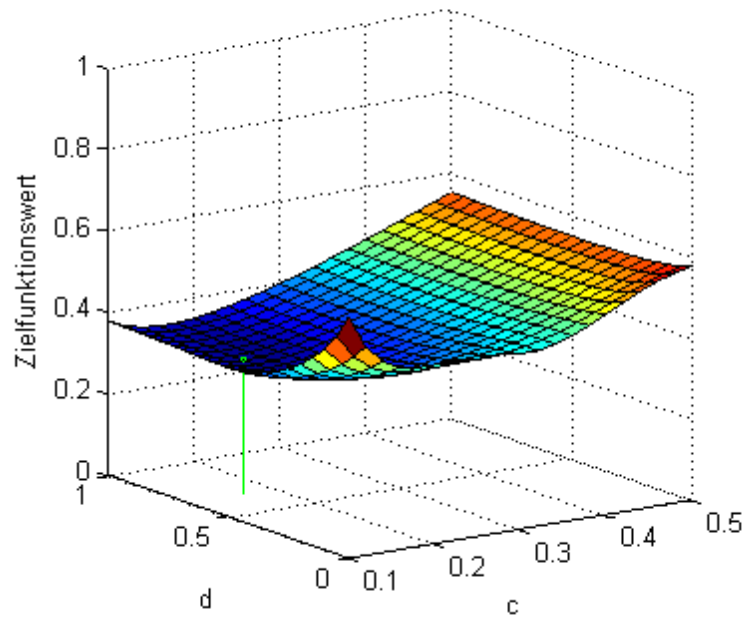


Bild 48: Einzelkriterien und Zielfunktionswert im Designraum

#### 4.9.8.2 Kennwerte für die Leistungsfähigkeit des Verfahrens

Die Kennwerte der Leistungsfähigkeit des Verfahrens lassen sich wieder auf die einzelnen Teilaktivitäten des Teilhandlungssystems aufteilen. In der Teilaktivität "Versuchsdesign/Versuch" gibt es folgende Kennwerte:

- **Versuchszeit**  
Einer der wichtigsten Kennwerte ist die Versuchszeit, die für alle Messpunkte inklusive aller Konditionier- und Umrüstzeiten benötigt werden, da hiervon ein wesentlicher Teil der Optimierungskosten verursacht wird.
- **Kosten für den Versuch**  
Werden die Kosten für den Prüfstand (Stundensätze), die Kosten für Umrüstung (Variation Designparameter) und der Sensoren etc. gegeben, kann das System direkt die Gesamtkosten berechnen und das Versuchsdesign danach optimieren.
- **Konditionierzeit**  
Die Konditionierzeit ist die Gesamtzeit, die zwischen den eigentlichen Manövern benötigt wird, um den Prüfling in den richtigen Zustand (z.B. Batterieladezustand, Geschwindigkeit) zu bringen.
- **Anzahl Verstellungen**  
Ein anderer Kennwert ist die Anzahl der notwendigen Variationen der Designparameter, da sich hieraus zusammen mit den Konditionierzeiten und den Umrüstzeiten die Versuchszeit ergibt.

In der Teilaktivität Identifikation sind folgende Kennwerte wichtig:

- **mittlere Abweichung des Modells**  
Ein Kennwert für die Beurteilung der Modellqualität ist die mittlere Abweichung des Zielfunktionswerts bzw. Einzelkriterien bezogen auf den Versuchswert.
- **Rechenzeit für Identifikation**  
Teilweise benötigt die Identifikation von frei definierbaren Modellstrukturen sehr viel Rechenzeit (im Bereich von Tagen). Deshalb muss diese bei der Analyse mit betrachtet werden. Außerdem ist das ein Kennwert, der benötigt wird um die verschiedenen Identifikationsalgorithmen bzw. deren Einstellungen zu vergleichen.
- **Abweichung Verläufe Zustands- und Ausgangsgrößen (nur für frei definierbare Modellstrukturen)**

Um die Modellgüte für die frei definierbaren Modellstrukturen bzw. für verschiedene Algorithmen zu vergleichen, ist dieser Kennwert relevant.

- Abweichung identifizierte Parameter (nur für frei definierbare Modellstrukturen)  
Bei Vorabsimulationen können ebenfalls die identifizierten Parameter mit denen der Simulation verglichen werden.

Für die Teilaktivität Optimierung sind folgende Kennwerte wichtig

- Modellwert im Optimum  
Der Modellwert im Optimum ist der Zielfunktionswert, der aus allen gewichteten Kriterien in der Simulation errechnet wird.
- Zielfunktionswert im Optimum (Verifikationsversuch)  
Dieser Kennwert ergibt sich im Verifikationsversuch mit den berechneten optimalen Designparameterwerten. Außerdem ist dieser Kennwert ein absolutes Maß, für die Qualität des Systems am Ende des Optimierungsprozesses.
- Rechenzeit für Optimierung  
Ist die Zeit, die benötigt wird, um die optimalen Designparameter mit Hilfe des parametrisierten Modells zu berechnen.

#### 4.9.8.3 Einflussfaktoren – Optimierungsaufgabe

Im Wesentlichen hängen die Leistungsfähigkeit des Verfahrens bzw. der Aufwand sowie das Ergebnis der versuchsbasierten Optimierung von der Optimierungsaufgabe ab. Einen wesentlichen Einfluss dabei haben die folgenden Eigenschaften der Optimierungsaufgabe:

- Komplexität  
Einen wesentlichen Einfluss haben die Art sowie die Anzahl der un- bzw. bekannten Parameter, Designparameter, Zustandsgrößen, Ein- und Ausgangsgrößen, Störgrößen sowie Informationen zu den Grenzen in denen die Parameter liegen können bzw. liegen dürfen.
- Modellstruktur  
In Wechselwirkung zum physischen System steht die angenommene Modellstruktur. Je detaillierter diese ist, desto mehr Parameter müssen identifiziert werden. Das führt zu höherem Berechnungs- und Versuchsaufwand. Dabei sollte die einfachste Modellstruktur gewählt werden, die den physischen Prüfling hinsichtlich der betrachteten Kriterien in ausreichender Qualität abbildet.

- Kriterien

Je nachdem, welche Kriterien betrachtet werden, muss die Modellstruktur angepasst werden. Außerdem erfordert jedes Kriterium ein Bewertungsmanöver. Die Länge des Bewertungsmanövers beeinflusst sowohl den Versuchsaufwand, als auch den Rechenaufwand, da zumindest ein Teil des Manövers bei jeder Variation durchgeführt werden muss.

#### 4.9.8.4 Einflussfaktoren –Verfahren

Neben der Optimierungsaufgabe haben die Verfahren bzw. die eingesetzten Algorithmen Einstellungsparameter, die die Leistungsfähigkeit des Verfahrens beeinflussen:

- Rauschanteil

Je nachdem welche Sensoren verwendet bzw. wie die Daten nachbearbeitet werden, ergeben sich verschiedene Rauschanteile der Messdaten. Diese beeinflussen die Identifizierung sowie die Auswertung der Kriterien. Bild 49 (links) zeigt den Verlauf der Auslenkung des Einmassenschwingers mit verschiedenen Rauschanteilen. Das rechte Bild zeigt die Werte der Einzelkriterien bzw. der Zielfunktion in Abhängigkeit des Rauschanteils. In diesem Fall wird das gleiche System mit der gleichen realen Systemantwort aufgrund des unterschiedlichen Rauschanteils unterschiedlich bewertet. Durch den Einsatz eines entsprechenden Filters kann dies jedoch wesentlich verbessert werden. Eine andere Möglichkeit ist es, diesen Einfluss mit zu berücksichtigen, indem der Rauschanteil mit identifiziert wird und auf die simulierten Größen addiert wird. So entsteht keine Verzerrung der Bewertung in der Simulation, relativ zum Versuch, da der gleiche Rauschanteil vorliegt.

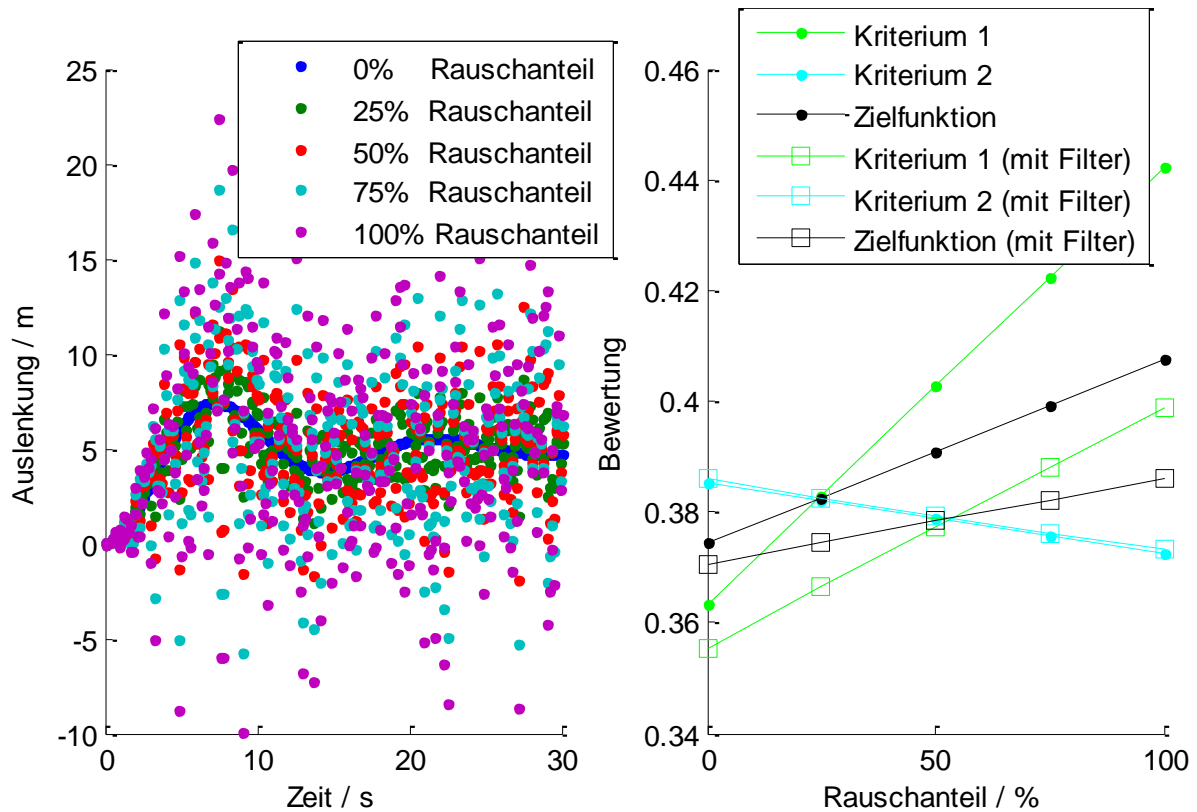


Bild 49: Auswertung der Kriterien für verschiedene Rauschanteile

- Einstellung der Algorithmen

Jeder verwendete Algorithmus hat üblicherweise unterschiedliche Einstellparameter. Dazu zählen Designraumgrenzen, Anzahl von Iterationen Generationen, Startwerte etc.

- Streuung

Nicht immer verhalten sich Systeme bei jedem Versuch gleich und weisen daher eine gewisse Streuung auf. Das gilt ebenso für Algorithmen, die wie die genetischen Algorithmen, stochastisch arbeiten. Bild 50 zeigt die Streuung des Verfahrens anhand von jeweils zehn Wiederholungen für verschiedene Rauschanteile des Systems mit sonst gleichen Einstellungen des genetischen Algorithmus. Das führt dazu, dass je nach Streuung die Versuche bzw. Berechnungen mehrfach durchgeführt werden müssen, um die Streuung beurteilen zu können.



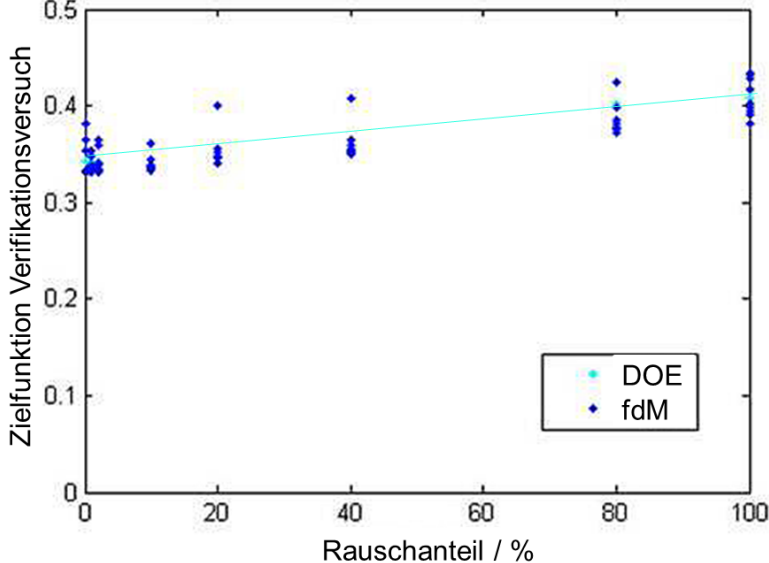


Bild 50: Streuung genetischer Algorithmus bei verschiedenen Rauschanteilen

## 5 Bewertung der Verfahren

Neben der versuchsbasierten Optimierung sind die Verfahren bzw. Teilmethoden für andere Entwicklungsaufgaben einsetzbar. Diese werden in Kapitel 6 betrachtet. Für die versuchsbasierte Optimierung steht der Aufwand der Verfahren für die betrachtete Optimierungsaufgabe im Vordergrund. Ein wesentlicher Anteil des Aufwands (Entwicklungszeit und -kosten) ist die Versuchszeit, da hier neben den Personal- und Lizenzkosten noch die Stundensätze für Prüfstand, Sensoren und vor allem des Prototypen hinzukommen. Für die Entscheidung, welches Verfahren für welche Optimierungsaufgabe eingesetzt wird, müssen jedoch weitere Aufwände miteinbezogen werden:

- Modellbildungsaufwand

### DOE:

Gegenüber einfachen Verfahren wie OFAOT (one factor at one time) ist der Modellbildungsaufwand höher. Dafür wird eine Reihe von Vorteilen, wie eine erheblich kürzere Versuchsdauer erzielt. Das DOE-Verfahren nutzt nichtphysikalische Strukturen (meistens polynomiale Modelle oder künstliche neuronale Netze), um den Zusammenhang zwischen Designparameter und den Kriterien bzw. der Zielfunktion zu modellieren. Deshalb muss bei der Modellbildung nur eine Modellstruktur gefunden werden, die diesen Zusammenhang mit ausreichender Qualität abdeckt. Im Falle der polynomialen Modelle muss für jeden Designparameter die richtige Ordnung bestimmt werden. Ist das System unbekannt, muss ein sog. Screening-Versuch, das heißt, viele Variationen der Designparameter, durchgeführt werden. Die eigentliche Modellbildung und der damit verbundene Aufwand sind mit entsprechender Schulung und Softwarewerkzeugen relativ gering. Verbreitet sind DOE-Schulungen über drei Tage. Ein grober Anhaltswert für die Modellbildung ist ca. drei Stunden.

### Zeiteffizientes DOE:

Beim zeiteffizienten DOE werden die "klassischen" DOE-Messpunkte so angeordnet, dass die Gesamtkonditionierungszeit minimal wird. Für diese Ordnung ist ein zusätzliches Modell notwendig, das die Konditionierungszeiten zwischen den Anfangs- und Endzuständen abschätzen kann. Der Aufwand sowie der Nutzen hängen dabei sehr stark vom System bzw. der Optimierungsaufgabe ab. Bei der Schaltqualitätserfassung werden inklusive Vorversuch (ca. 20 min) maximal 30 min für die Modellbildung benötigt.

Frei definierbare Modellstrukturen:

Gegenüber den DOE-Verfahren ist der Modellbildungsaufwand höher, wenn keine Modelle bzw. Modellvorstellungen vorhanden sind. Jedoch können vorhandene Modelle aus vorhergehenden Entwicklungsphasen genutzt werden. Wie groß der Modellbildungsaufwand ist, hängt jedoch auch stark von der Optimierungsaufgabe sowie der Erfahrung der beteiligten Personen ab.

- **Versuchsaufwand**

Der Versuchsaufwand beinhaltet folgende Teile:

- **Verstellzeiten**

- DOE/zeiteffizientes DOE-Methode:

- In den DOE-Verfahren muss das Bewertungsmanöver für verschiedene Designparametervariationen mit dem physischen System durchgeführt werden. Je nach System und Parameter kann das sehr schnell gehen, wie zum Beispiel bei Regelparametern im Steuergerät oder viel Zeit in Anspruch nehmen, wie zum Beispiel Hardwareänderungen.

- Frei definierbare Modellstrukturen:

- Im Idealfall werden bei diesem Verfahren nur zwei Designparametereinstellungen des physischen Systems benötigt - das initiale System und das Optimierte zur Verifikation. In diesem Fall ist nur ein Umbau notwendig.

- **Konditionierungszeiten**

- DOE:

- Zwischen den einzelnen Manövern bzw. den Messpunkten wird je nach Optimierungsaufgabe Zeit benötigt, um die erforderlichen Anfangszustände einzustellen.

- zeiteffiziente DOE-Methode:

- Bei der zeiteffizienten DOE-Methode werden die einzelnen Messpunkte bzw. die einzelnen Manöver so angeordnet, dass die Konditionierungszeit minimal wird. Werden immer die gleichen Anfangszustände benötigt, kann hier keine Zeitersparnis erzielt werden.

- Frei definierbare Modellstrukturen:

- Im Identifizierungsmanöver sind je nachdem keine Konditionierungszeiten enthalten. Im Verifikationsversuch werden Konditionierungszeiten benötigt, wenn diese im Bewertungsmanöver enthalten sind.

○ **Zeit für Bewertungsmanöver**

DOE/zeiteffiziente DOE-Methode

Das Bewertungsmanöver wird mehrfach, abhängig von der Anzahl der Designparameter und der Anzahl der Faktorstufen in diese im Modell vorliegen (müssen) (siehe Formel (2.4.6)), durchgeführt. Dazu kommt noch der Verifikationsversuch, sodass sich folgende Formel ergibt:

$$\text{Anzahl Bewertungsman}_{DOE} \sim = 1,5 * \left( \sum_{l=1}^{p_q} \text{Faktorstufen}_l \right) + 1 \quad (4.9.1)$$

Frei definierbare Modellstrukturen:

Bei den frei definierbaren Modellstrukturen muss im Idealfall nur für die Verifikation einmal das Bewertungsmanöver durchgeführt werden. In dem Beispiel mit dem Einmassenschwinger aus Kapitel 4.9.8.1 kann so 94,1% Versuchszeit gegenüber dem klassischen DOE Versuch eingespart werden (siehe Bild 51). Dabei wird hier angenommen, dass keine Zeit für das Verstellen der Parameter und für die Konditionierung zwischen den Messpunkten benötigt wird. In diesem Fall würde der Aufwand für das DOE-Verfahren noch weiter steigen.

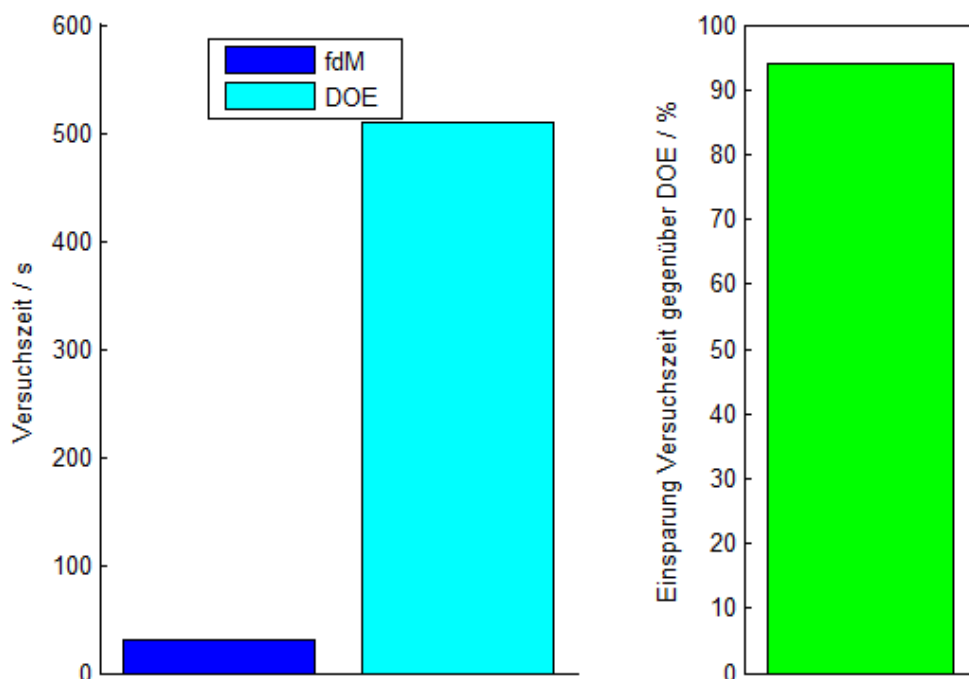


Bild 51: Vergleich Versuchszeit ohne Verstell- oder Konditionierzeiten

- Modellidentifikation/Verifikation/Optimierung

DOE/zeiteffizientes DOE:

Hier stehen entsprechende rechnergestützte Werkzeuge und Standardalgorithmen zu Verfügung. Deshalb sind hier sowohl die Rechenzeit als auch der Bearbeitungsaufwand gering.

frei definierbare Modellstrukturen:

Diese Arbeit schafft Möglichkeiten für die Identifikation frei definierbarer Strukturen sowie ein rechnergestütztes Werkzeug (siehe Anhang 9.3), mit dem diese Aktivität unterstützt werden kann. Jedoch müssen hier je nach Optimierungsaufgabe die passenden Algorithmen bzw. deren Einstellungen ausgewählt werden. Das führt zu einem höheren Aufwand als bei den DOE-Verfahren. Die Rechenzeit hängt ebenfalls stark von der Optimierungsaufgabe ab, ist aber wesentlich höher als bei den DOE-Verfahren.

- Qualität

Eine andere Form des Aufwands, d.h. die damit verbundenen Kosten, ist die Qualität der Ergebnisse, da ein schlechteres Endprodukt auch einen geringeren Gegenwert besitzt. Nach der Modellbildung folgt beim klassischen DOE-Verfahren wie auch beim Einsatz frei definierbarer Modellstrukturen der Verifikationsversuch. Hier werden die berechneten optimalen Designparameter im System eingestellt und das Bewertungsmanöver durchgeführt. Dieser Versuch wird anschließend hinsichtlich der Kriterien ausgewertet und der Zielfunktionswert, z.B. gewichtete Einzelkriterien, gebildet. Will man die Verfahren oder zwei Verfahrenseinstellungen vergleichen, eignet sich dieser Zielfunktionswert als ein Kennwert für die Qualität des Verfahrens. Dieser gibt an wie gut das System nach der Optimierung ist.

Der Zielfunktionswert hängt im Wesentlichen von den eingesetzten Modellen bzw. der Optimierungsaufgabe ab. Im Beispiel wurden vergleichbare Ergebnisse erzielt, wobei das stark von den Einstellungen des Verfahrens sowie der Streuung abhängt. Im Mittel zeigt jedoch das Verfahren mit frei definierbaren Parametern einen Vorteil (siehe Bild 52). Hier ist das Ergebnis von 10 Durchläufen des Verfahrens mit freidefinierbaren Strukturen bei verschiedenen Rauschanteilen und gegebener Rechenzeit zur Identifikation dem DOE-Verfahren gegenübergestellt. Im rechten Teil des Bildes wird das Ergebnis für einen Rauschanteil von 10% herausgestellt. Diese Darstellung zeigt die stochastische Eigenschaft des genetischen Algorithmus.

## Einfluss Rechenzeit/ Rauschanteil :

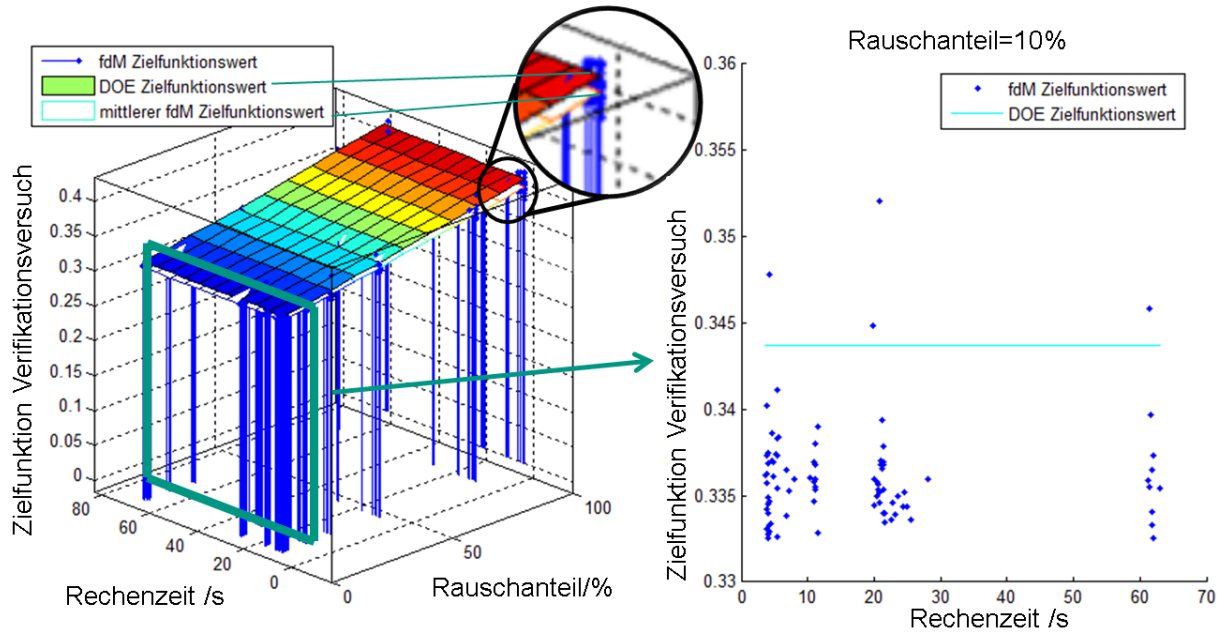


Bild 52: Vergleich der Qualität

Zusammenfassend lässt sich sagen, dass sich durch den Einsatz von frei definierbaren Modellstrukturen Versuchszeit/aufwand bei gleicher oder besserer Qualität einsparen/reduzieren lässt. In der Beispielaufgabe mit dem Einmassenschwinger aus Kapitel 4.9.8.1 über 90%. Die Ersparnis steigt weiter, wenn eine oder mehrere der folgenden Kriterien erfüllt sind:

- große Anzahl von wechselwirkenden Designparametern
- Lange Bewertungsmanöver
- Lange Konditionierungszeiten/Umrüstzeiten

Dieses Verfahren erfordert aber einen höheren Modellbildungsaufwand, wenn keine entsprechenden Modelle/Modellvorstellungen vorhanden sind. Ein weiterer Vorteil der frei definierbaren Modellstrukturen ist, dass der Anwender die modellierten Effekte genau kennt und somit sicher sein kann, welche davon das Modell abbildet und wo die Grenzen des Modells sind. Bei der DOE-Vorgehensweise muss diese Erkenntnis durch einen sogenannten Screeningversuch, d.h. einen Versuchsplan mit vielen Versuchspunkten für ein vergleichbares System gewonnen werden. Streng mathematisch ist auch dann nicht gewährleistet, dass sich das System zwischen den Versuchspunkten nicht anders verhält.

Der Aufwand für die Verfahren lässt sich vorab nur qualitativ abschätzen. Ist schon ein grobes Modell bzw. eine grobe Modellvorstellung vorhanden, kann jedoch mit Hilfe des hier erarbeiteten Werkzeugs (siehe Bild 45) der Versuchsaufwand abgeschätzt werden und dem geschätzten Modellierungsaufwand gegenübergestellt werden. Hier können auch verschiedene Sensoren/Sensorpositionen sowie das optimale Identifikationsmanöver berechnet werden.

## 6 Weitere Anwendungsmöglichkeiten

Neben dem versuchsbasierten Optimieren kann das erarbeitete Teilhandlungssystem für weitere Anwendungsmöglichkeiten eingesetzt werden:

- **Modellvereinfachungen**  
Häufig werden Modelle benötigt, die echtzeitfähig sind. Ein Beispiel für einen Anwendungsfall ist die Restfahrzeugsimulation im XiL-Framework<sup>87</sup>. Mit Hilfe der frei definierbaren Modellstrukturen können verschiedene Modellvarianten identifiziert werden und hinsichtlich der Modellqualität getestet werden. So kann die einfachste Modellstruktur gefunden werden, die das Systemverhalten hinsichtlich der betrachteten Kriterien genau genug abbildet. Damit kann wiederum Rechenzeit eingespart werden.
- **Testen von Modellvorstellungen**  
In experimentellen Untersuchungen können Flussgrößen, wie den Reibkoeffizient, häufig nicht direkt als Messgröße erfasst werden. Mit einem physikalischen Modell können solche Parameter identifiziert werden.
- **Nutzen der Modelle für die Restsystemsimulation**  
Das Ergebnis des fdM-Verfahrens ist ein physikalisches Modell, das sich zumindest hinsichtlich der betrachteten Kriterien wie der Prüfling verhält. Dieses Modell kann dann für den Optimierungsprozess anderer Subsysteme verwendet werden. Wird zum Beispiel das Getriebe optimiert entsteht als ein Nebenprodukt ein Getriebemodell. Dieses kann dann im Sinne des XiL-Framework als Restsystemsimulation z.B. auf dem Motorenprüfstand verwendet werden bzw. die Grundlage bilden.
- **Vorhersage des Systemverhaltens**  
Das physikalische Modell kann außerdem verwendet werden, um das Systemverhalten bei einer gegebenen Eingangsgrößen zu berechnen. Damit können zum Beispiel Zustandsgrößen, wie lokalen Drücke und Temperaturen auf einer Kupplungsscheibe, abgeschätzt werden, die schwer oder gar nicht mit Sensoren erfassbar sind.
- **Verkürzen konventioneller Prüfläufe**  
Oft bestehen Prüfläufe aus mehreren Einzelmanövern wie z.B. verschiedene Schaltungsarten, die jedoch bei unterschiedlichen Startzuständen (z.B.

---

<sup>87</sup> Albers et al 2008



definierte Batterieladezustände oder Geschwindigkeiten durchgeführt werden sollen. Deshalb muss das Fahrzeug zwischen den eigentlichen Manövern in den richtigen Zustand gebracht werden. Mit der Methode aus Kapitel 4.8 kann das fertige Modell verwendet werden um die Manöver so anzuordnen, dass möglichst wenig Zeit für diese Konditionierung benötigt wird.

## 7 Zusammenfassung und Ausblick

In diesem Kapitel werden die Ergebnisse dieser Arbeit zusammengefasst. Außerdem wird ein Ausblick für anknüpfende Forschungsarbeiten gegeben.

### 7.1 Zusammenfassung

In dieser Arbeit wird ein Teilhandlungssystem zur versuchsbasierten Optimierung erarbeitet. Mit diesem Teilhandlungssystem ist es möglich, die optimalen Designparameterwerte für multikriterielle Optimierungsaufgaben zeit- und kosteneffizient zu berechnen.

Dabei integriert dieses Teilhandlungssystem die bestehende DOE-Methode nach Taguchi und erweitert diese um zwei Ansätze:

- Zeiteffizientes DOE  
Mit Hilfe dieser Erweiterung ist es möglich, Messpunkte bzw. die Manöver einer Designparametervariationsstufe so anzuordnen, dass die insgesamt benötigte Zeit zwischen den Versuchspunkten minimal wird. So kann Versuchszeit/-aufwand bei gleichem Ergebnis eingespart/reduziert werden. Diese Methode wird anhand der Schaltqualitätsbeurteilung eines Fahrzeugs mit einem 7-Gang Doppelkupplungsgetriebe analysiert. In diesem Beispiel können auf dem Rollenprüfstand im Vergleich zu einem erfahrenen Testfahrer 80% Versuchszeit eingespart werden.
- Frei definierbare Modellstrukturen  
Bei dieser Erweiterung werden Versuchsdaten verwendet, um physikalische Modellstrukturen zu parametrieren. Anschließend werden diese physikalischen Modelle für die Optimierung der Designparameter verwendet. Der Vorteil dieses Ansatzes ist, dass auch Parameter, die nur langsam im Versuch (z.B. Getriebeübersetzung der einzelnen Gänge) verstellt werden können, mit Hilfe des Simulationsmodells für die Optimierung zu Verfügung stehen. Außerdem ist es möglich auch Kriterien, die lange Versuchszeiten erfordern, schnell mit Hilfe des Modells zu berechnen. Ein Beispiel für ein solches Kriterium ist der Verbrauch im NEFZ-Zyklus, welcher 20 Minuten für einen Durchlauf benötigt. Im Versuch müsste dieser Zyklus für jeden Parametersatz einzeln abgefahren werden. Mit diesem Ansatz wird zunächst anhand von Versuchsdaten ein physikalisches Fahrzeugmodell gebildet, welches sich hinsichtlich der betrachteten Kriterien hinreichend genau verhält.

Anschließend wird dieses Modell für die Optimierung der Parameter verwendet. Ein weiterer Vorteil dieses Ansatzes ist, dass im Gegensatz zu nichtphysikalischen Modellen bekannt ist, welche Effekte das Modell abbilden kann. Neben der Optimierung können diese Modelle verwendet werden, um bei gegebenen Eingangsgrößen und dem Ausgangszustand das Systemverhalten vorherzusagen.

Mit Hilfe des Teilhandlungssystems werden die, hinsichtlich der betrachteten Kriterien, optimalen Eigenschaften der Wirkflächenpaare und Leitstützstrukturen für das Objektsystem gefunden und verifiziert. Da die Verifikation Teil der Validierung ist, wird das Teilhandlungssystem in die Aktivitäten "Modellierung von Prinzip und Gestalt" und "Validierung" des integrierten Produktentstehungsmodell (iPeM) eingeordnet.

Neben der Umsetzung des Teilhandlungssystems für verschiedene Anwendungen wird in dieser Arbeit ein rechnergestütztes Werkzeug (siehe Anhang 9.3) entwickelt. Mit Hilfe dieses Werkzeugs werden die Einflussgrößen auf die Optimierung analysiert und die Limitierungen aufgezeigt.

Das Teilhandlungssystem nutzt die XiL-Validierungsumgebung für die Versuchsphasen. Damit können auch einzelne Komponenten, Subsysteme im Gesamtsystemkontext, d.h. auf allen Layer des XiL-Framework optimiert werden.

## **7.2 Ausblick**

Diese Arbeit schafft die Grundlage für die versuchsbasierte Optimierung von technischen Systemen. Dabei werden in dieser Arbeit verschiedene Beispieloptimierungsaufgaben verwendet. Ein Ziel für die Zukunft ist es dieses Portfolio zu erweitern und den Optimierungsaufwand für die verschiedenen Verfahren zu analysieren. Für diese Optimierungsaufgaben müssen entsprechende Kriteriendefinitionen bzw. geeignete Modellstrukturen für das betrachtete System gefunden werden. Dies liefert Erfahrungswerte für den Aufwand und die Ersparnis des Verfahrens für ähnliche Optimierungsaufgaben.

Im Fokus dieser Arbeit stehen genetische Optimierungsalgorithmen sowie Verfahren, die auf Least-Squares-Schätzern beruhen. Hier können auch andere Algorithmen bzw. Kombinationen verschiedener Algorithmen untersucht werden. Eine interessante Kombination ist ein eher "grob" eingestellter genetischer Optimierer und ein gradientenbasiertes Verfahren. Die Idee ist, dass der genetische Optimierer die Position des globalen Extremum grob findet. Anschließend wird das

gradientenbasierte Verfahren mit diesem Startwert verwendet, um die genaue Position des globalen Optimums zu bestimmen. Damit könnte es möglich sein, die Optima in kürzerer Rechenzeit, genauer zu berechnen.

Eine wesentliche Einflussgröße beim DOE-Verfahren ist die Modellqualität. Diese hängt vor allem von der Ordnung der Parameter ab. Je höher die Ordnung desto genauer können Extrema abgebildet werden. Jedoch steigt damit auch die "Welligkeit" des Modells, welches das tatsächliche Systemverhalten nicht widerspiegelt. Ein Ansatz ist die niedrige Ordnung des Modells beizubehalten, jedoch die Punkte in der Nähe des Minimums bei der Modellbildung höher zu gewichten. Damit nimmt zwar die Qualität des Modells in den anderen Bereichen ab, jedoch wird das benötigte Optimum wesentlich besser abgebildet und so ein besseres Gesamtergebnis erzielt.

## 8 Literaturverzeichnis

### 8.1 Quellen

#### **Albers et al 2005**

Albers, A., Burkardt, N., Meboldt, M., & Saak, M. (2005). SPALTEN Problem Solving Methodology in the Product Development. In W. A. and L. Samuel (Ed.), 15th International Conference on Engineering Design. 15th International Conference on Engineering Design -- ICED'05. The Design Society.

#### **Albers et al 2008**

Albers, A., Düser, T., & Ott, S. (2008). X-in-the-loop als integrierte Entwicklungsumgebung von komplexen Antriebsystemen. In 8. Tagung Hardware-in-the-loop-Simulation. Haus der Technik.

#### **Albers et al 2009**

Albers, A., Düser, T. Integration von Simulation und Test am Beispiel Vehicle-in-the-loop auf dem Rollenprüfstand und im Fahrversuch (2009).

#### **Albers 2010a**

Albers, A.: Five Hypotheses about Engineering Processes and their Consequences. In: International Conference on Tools and Methods of Copetitive Engineering (TMCE). Ancona, 2010

#### **Albers et al 2010b**

Albers, A., Schwarz A., Bogner, E., Düser, T. (2010). Methode zur kombinierten Bewertung der Energieeffizienz und der Fahrbarkeit von Fahrzeugen.

VDI Erprobung und Simulation 2010

#### **Albers et al. 2010c**

Albers, A., Behrendt, & Ott, S. (2010). Validation - Central Activity to Ensure Individual Mobility

#### **Albers/Braun 2011**

Der Prozess der Produktentstehung. In F. Henning, & E. Moeller (Eds.), Handbuch Leichtbau (pp. 5–30). HanserVerlag.

#### **Albers et al. 2011b**

Albers, A., Sadowski, E., & Marxen, L. (2011). A new Perspective on Product Engineering – Overcoming Sequential Process Models. In The Future of Design Methodology. Birkhofer, Herbert.

#### **Albers et al. 2011c**

Albers, A., Brezger, F., Geier, M., Freudenmann, T., & Stier, C. (2011). Phenomena-Based Methods in Powertrain Validation. In Innovative Automotive Transmissions and Hybrid & Electric Drives (10th ed., Vol. 10).

#### **Albers et al 2012**

Albers, A., Alexander Schwarz, Matthias Behrendt, & Rolf Hettel. (2012). Method for the time-saving capturing of characteristic Gear shifting Diagrams of vehicles on the Rollertest Bench carried out with dual clutch transmissions. In 6th International CTI Symposium and Exhibition. CTI.

#### **Albers et al 2012b**

Albers, A., Alexander Schwarz, Matthias Behrendt, & Rolf Hettel. (2012). TIME-EFFICIENT METHOD FOR TEST-BASED OPTIMIZATION OF TECHNICAL SYSTEMS USING PHYSICAL MODELS. In ASME 2012 International Mechanical Engineering Congress & Exposition. SAE.

**Albers et al 2012c**

Albers, A., Schröter, J., & Behrendt, M. (2012). Entwicklungsumgebung zur automatisierten optimalen Konfiguration und Erstellung von optimierten echtzeitfähigen Gesamtfahrzeugverbrauchsmodellen. In IAV (Ed.)

**Albers/Sadowski 2013**

Albers, A., & Sadowski, E. (2013). The Contact and Channel Approach (C&C<sup>2</sup>-A) – relating a system's physical structure to its functionality. In A. Chakrabarti, & L. Blessing (Eds.), *An Anthology of Theories and Models of Design: Philosophy, Approaches and Empirical Explorations*. Springer.

**Albers et al 2013**

Albers, A., Stier, C., & Geier, M. (2013). Herausforderungen von Kupplungssystemen in modernen Antriebssystemen. In VDI-Berichte 2206. Düsseldorf: VDI-Verlag.

**Albers et al 2013b**

Albers Albert, Schwarz, A., Behrendt, M., & Hettel, R. (2013). Time efficient testing of Hybrid electric vehicles using automated identified physical model structures. In 2013 the 2nd International Conference on Advances in Mechanics Engineering..

**Auto-Moto-Sport 2011**

Auto-Moto-Sport.: So testet auto motor und sport.URL

<http://www.auto-motor-und-sport.de/so-testet-auto-motor-und-sport-den-kraftstoffverbrauch-22696942.html>

**Bub 1992**

Bub, W., & Lugner, P. (1992). Systematik der Modellbildung Teil 1: Konzeptionelle Modellbildung. (pp. 1–43). VDI Berichte, 925. Düsseldorf: VDI Verlag.

**Cooper 1990**

Cooper, R. G. (1990). Stage-Gate Systems: A New Tool for Managing New Products. *Business Horizons*, 33(3), 44–54.

**Domschke 2006**

Domschke, Wolfgang und Scholl, Armin: Heuristische Verfahren, Jenaer Schriften zur Wirtschaftswissenschaft Friedrich-Schiller-Universität Jena, 2006

**Dresig 2006**

Dresig, H. (2006). Schwingungen und mechanische Antriebssysteme : Modellbildung, Berechnung, Analyse, Synthese (2. Aufl. ed.). Berlin: Springer.

**Duddeck 2005**

Duddeck, F. (2005). Multidisziplinäre Optimierung im Produktentwicklungsprozess der Automobilindustrie.

**Ehrlenspiel 2007**

Ehrlenspiel, K. (2007). Integrierte Produktentwicklung Denkabläufe, Methodeneinsatz, Zusammenarbeit (3rd ed.). München, Wien: Carl Hanser.

**Furch 2002**

Furch, D. (2002). Modellgestützte Kennfeldoptimierung mit evolutionären Algorithmen.257–267

**Grzesik 2009**

Grzesik, A. (2009). Physiologiebasierte Simulation des Bremsverhaltens vonFahrzeugführern Universitätsverlag Ilmenau.

**Heißing 2002**

Heißing, B.: Die Simulation als Tool im Produktentstehungsprozess von Kraftfahrzeugen. In: Virtual ProductCreation. Berlin, 2002.

**Isermann 1992**

Isermann, R. (1992). Identifikation dynamischer Systeme 1: Grundlegende Methoden. Berlin: Springer.

**Isermann 2006**

Isermann, R. (2006). Fahrdynamik – Regelung Modellbildung, Fahrerassistenzsysteme, Mechatronik. Wiesbaden: vieweg.

**Klein 2004**

Klein, B. (2004). Versuchsplanung DoE Einführung in die Taguchi/Shainin-Methodik. München: Oldenburg Wissenschaftsverlag GmbH..

**Lerspalungsanti 2010**

Lerspalungsanti, S. (2010). Ein Beitrag zur Modellierung des menschlichen Komfortempfindens und Beurteilung der NVH-Eigenschaften in der Antriebsstrangentwicklung auf Basis von Künstlichen Neuronalen Netzen (Vol. 43). IPEK Dissertation, IPEK, Karlsruhe.

**Meboldt 2008**

Meboldt, M. (2008). Mentale und formale Modellbildung in der Produktentstehung-als Beitrag zum integrierten Produktenstehung-Modell (iPeM). Forschungsberichte/IPEK, 29. IPEK Dissertation, Universität Karlsruhe, Institut für Produktentwicklung, Karlsruhe.

**Münz 2009**

Münz, E. (2009). Identifikation und Diagnose hybrider dynamischer Systeme. Karlsruhe: Universitätsverlag.

**Naumann 2002**

Naumann, T. (2002). Wissensbasierte Optimierungsstrategien für elektronische Steuergeräte an Common-Rail-Dieselmotoren. Technische Universität Berlin, 2002.

**Ortlieb 2009**

Ortlieb C., Dresky c., Gasser I., Günzel S.(2009):Mathematische Modellierung, Vieweg+Teubner Verlag

**Pahl 2005**

Pahl, G., Beitz, W., Feldhusen, J., & Grote, K. - H. (2005). Pahl/Beitz Konstruktionslehre – Grundlagen erfolgreicher Produktentwicklung Methoden und Anwendung (6th ed.). Berlin; Heidelberg; New York: Springer.

**Rohpol 2009**

Ropohl, G. (2009). Allgemeine Technologie – Eine Systemtheorie der Technik. Allgemeine Technologie – Eine Systemtheorie der Technik, 3., Überarbeitete Auflage. Karlsruhe: Universitätsverlag Karlsruhe 2009

**Roy/Ranjit 2001**

Roy, Ranjit K. (2001). Design of Experiments Using the Taguchi Approach

**Sauter 2000**

Sauter, J. und Fricke, H. und Güngör, Z. und Himmler, G. und Hougardy, P. und Lauber, B. und Müller, O. und Neithardt, W. und Schirrmacher, R.: Integrierte Topologie- und Gestaltoptimierung im virtuellen Produktentstehungsprozess – Einbindung in die iViP Architektur und industrielle Anwendung, VDI Berichte Berechnung und Simulation im Fahrzeugbau Würzburg, 2000

**Schlecht 2006**

Schlecht, B. (2006). Skript zur Vorlesung Modellbildung und Simulation, TU Dresden

**Schneider 2001**

Schneider, Stefan: Rechnergestützte, kooperativ arbeitende Optimierungsverfahren am Beispiel der

Fabriksimulation, Dissertation Universität Kassel, 2001

**Silver 2004**

Silver, E.A.: An overview of heuristic solution methods, Working Paper University of Calgary, 2004

**Siebertz 2010**

Siebertz, K., van Bebber, D., & Hochkirchen, T. (2010). Statistische Versuchsplanung – Design of Experiments (DoE). Berlin, Heidelberg, New York: Springer - Verlag.

**Stachowiak 1973**

Stachowiak, H. (1973). Allgemeine Modelltheorie. Wien: Springer Verlag.

**Sutor 2003**

Sutor, Ariane: Ein stochastisches Verfahren zur globalen Optimierung bei diskreten und kontinuierlichen Variablen, Dissertation Universität der Bundeswehr München, 2003

**VDI Richtlinie 2206**

VDI-Fachbereich Produktentwicklung und Mechatronik. (2004). Entwicklungsmethodik für mechatronische Systeme. VDI-Richtlinien 2206, (06), 118.

**VDI Richtlinie 2223**

(2004). VDI 2223: Methodisches Entwerfen technischer Produkte; Systematic embodiment design of technical products

## 8.2 Co-betreute Studien-/Diplom-/Bachelor-/Masterarbeiten

**Römer 2011**

Jürgen Römer; Co Betreuer: Schwarz, A.; Betreuer: Albers, A.: Methode zur zeitoptimierten Erfassung eines Schaltkennfelds auf dem Rollenprüfstand. IPEK, Karlsruher Institut für Technologie, Bachelorarbeit



## 9 Anhang

### 9.1 Maple Code für Generierung der Schätzermatrix

```

> restart, clear,
with(LinearAlgebra) :
with(ExcelTools) :
with(Matlab) :
with(Matlab) :
with(ExcelTools) :
with(linalg) :
tol := 10 ; # % Eingangsdrehz Toleranzrelativdrehzahl
#Parameter := [c(1), c(2), c(3), c(4), c(5), e0, JM, JP, JT, w0, w1, w2];
#Parameter := [c(1), c(2), c(3), c(4), c(5), η, J, w0, w1, w2]; # m ergänzen
Parameter := [c[1], c[2], c[3], c[4], c[5], JT, JG, w0, w1, w2, iT0, iT1, deltansoll, e0, e1, dp,
etasch];
#konsParameter :=

                                clear
                                tol := 10

Parameter := [c1, c2, c3, c4, c5, JT, JG, w0, w1, w2, iT0, iT1, deltansoll, e0, e1, dp, etasch]      (1)
> EXPORT [ ] := proc(A, PF)
    description "Export von Tabellen";
    global ALPH, MA, a, b, la1, la2;
    ALPH := ["A", "B", "C", "D", "E", "F", "G", "H", "I", "J", "K", "L",
"M", "N", "O", "P", "Q", "R", "S", "T", "U", "V", "W", "X", "Y", "Z", "AA", "AB", "AC",
"AD", "AE", "AF", "AG", "AH", "AI", "AJ", "AK", "AL", "AM", "AN", "AO", "AP", "AQ",
"AR", "AS", "AT", "AU", "AV", "AW", "AX", "AY", "AZ"];
    MA := Matrix(convert(table(A), array));
    a := rowdim(MA);
    b := coldim(MA);
    for la1 from 1 to a do
        for la2 from 1 to b do
            Export(Array([convert(MA[la1, la2], string)]), PF, "1",
cat(ALPH[la2], convert(la1, string)));
        end do;
    end do;
end proc:
> φ := c[1] + c[2]*v + c[3]*v^2 + c[4]*α + c[5]*α^2;
M := Mminn + (Mmaxn - Mminn) * φ;
ME := M - 2 * π * JG *  $\frac{np}{60}$ ;
nE := n;

                                φ := c1 + c2v + c3v2 + c4α + c5α2

                                M := Mminn + (Mmaxn - Mminn) (c1 + c2v + c3v2 + c4α + c5α2)

                                ME := Mminn + (Mmaxn - Mminn) (c1 + c2v + c3v2 + c4α + c5α2) -  $\frac{1}{30}$  π JG np
                                nE := n

(2)
> #mue :=  $\frac{nA}{nE}$ ;
Bed[1, 1] := mue > nuekrit

```

```

> # Generierung Gleichung Bedigungsvektor
  q := 0;
  printlevel := 4;
  for la1 from 1 to Anz[1] do
  for la2 from 1 to Anz[2] do
  for la3 from 1 to Anz[3] do
    assume(Bed[1, la1], Bed[2, la2], Bed[3, la3]);
    q := q + 1;
    Bedingung[q, 1] := Bed[1, la1];
    Bedingung[q, 2] := Bed[2, la2];
    Bedingung[q, 3] := Bed[3, la3];
    x := ([coeffs( (collect(xpp, Parameter, distributed) ), Parameter, 'f' )]);
    num := Matlab[dimensions](vector( [x[ ] ]));
    for ll from 1 to num[1] do
      Gx[q, ll] := x[ll];
      Gf[q, ll] := f[ll];
    end do;
    #x[q] := ([coeffs( (collect(xpp, Parameter, distributed) ), Parameter, 'hilf' )]);
    #f[q] := hilf;
    #Export(Array( [convert(Bedingung[q, 1], string) ]), "C:\Bed.xls", convert(la1, string) , cat("A",
      convert(q, string) ) );
    #Export(Array( [convert(Bedingung[q, 2], string) ]), "C:\Bed.xls", convert(la1, string) , cat("A",
      convert(q, string) ) );
    #Export(Array( [convert(Gleichung[q, 2], string) ]), "C:\Gleichung.xls", convert(la1, string) ,
      cat("A", convert(q, string) ) );
  end do;
end do;
end do;

-
> EXPORT[ ](Gf, "C:\Gf.xls");

> EXPORT[ ](Gx, "C:\Gx.xls");

> EXPORT[ ](Bedingung, "C:\Bedingung.xls");

> EXPORT[ ](Parameter, "C:\Parameter.xls");

```

Maple von Maplesoft <http://www.maplesoft.com>

## 9.2 Leistungsdaten Rechner

Klassifikation:	 <a href="#">Windows-Leistungsindeks</a>
Prozessor:	Intel(R) Core(TM) i5-2520M CPU @ 2.50GHz 2.50 GHz
Installierter Arbeitsspeicher (RAM):	4,00 GB (3,89 GB verwendbar)
Systemtyp:	64 Bit-Betriebssystem

### 9.3 Rechnergestütztes Werkzeug

Für folgende Aufgabenstellungen wurde, im Rahmen dieser Arbeit ein rechnergestütztes Werkzeug erarbeitet:

- Vergleich des Versuchsaufwands/Kosten des fdM Verfahrens mit dem DOE-Verfahren
- Identifikation von Parameter einer Modellstruktur mit gegeben Versuchsdaten.
- Analyse, Auswahl von Optimierungsalgorithmen/-einstellungen für eine gegebene Aufgabenstellung

Bevor das Hauptprogramm *Vergleichsumgebung.m* gestartet wird muss die Optimierungsaufgabe definiert werden. Hierzu gibt es je nach Anwendungsfall für den Benutzer folgende Möglichkeiten.

- Laden einer bestehenden Konfiguration (mat-File). Üblicherweise wird diese Möglichkeit gewählt, wenn das Projekt schon vorhanden ist.
- Modifizieren und Ausführen einer Standartdefinition (m-File) einer Optimierungsaufgabe. Als Ausgangsdatei kann dazu die *StadtEinst\_OP\_CD\_sys.m* verwendet werden. Diese Möglichkeit eignet sich vor allem für neue Projekte.
- Sollen verschieden Einstellungen der Algorithmen oder Modellvorstellungen d.h. Variationen der Optimierungsaufgabe durchgerechnet werden, eignet sich die Ausführung der *Variationsumg.m*. Hier können für jede Variable die gewünschten Variationen bzw. der Bereich in dem die Variationen liegen sollen definiert werden. Dabei können sowohl Abhängigkeiten als auch klassische Sensitivitätsstudien durchgeführt werden.
- Verwenden der grafischen Benutzer Oberfläche (GUI) *HS.fig*. Diese bietet aktuell nicht alle Funktionen der vorhergenannten Möglichkeiten. Dafür ist die GUI in Bild 53 benutzerfreundlicherer und für einfache Optimierungsaufgaben geeignet.

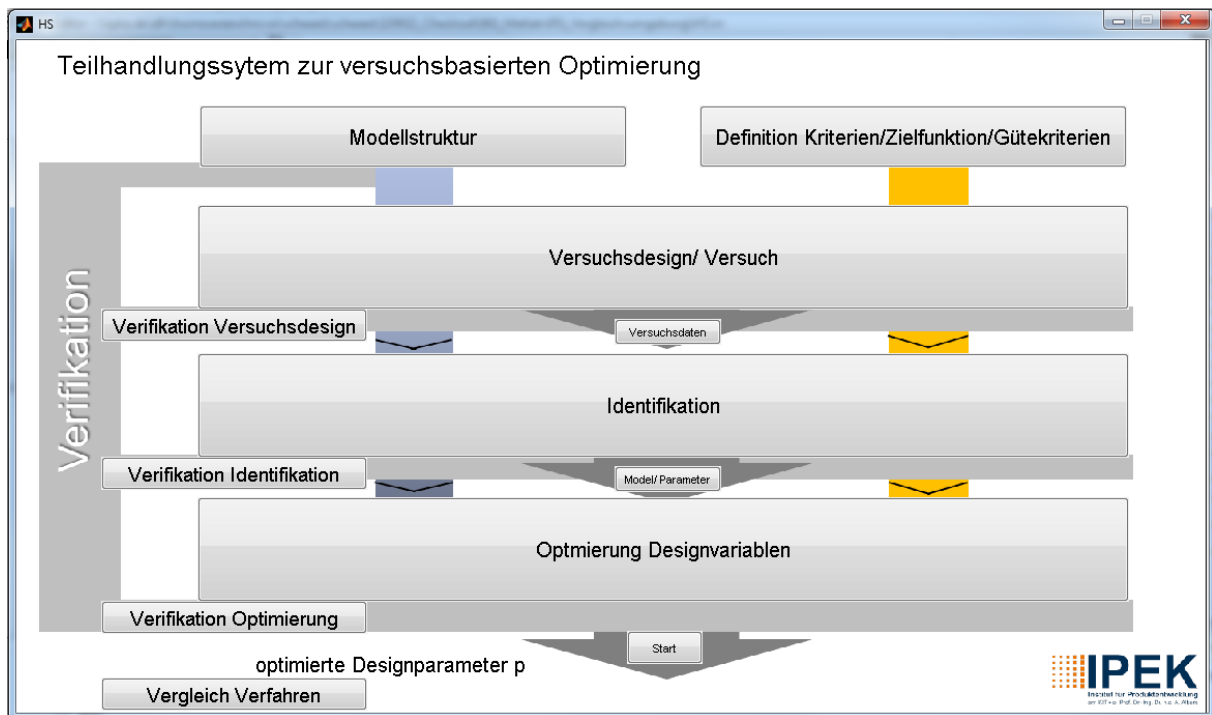


Bild 53: GUI

Nach der Durchführung einer der oben genannten Möglichkeiten sind folgende Variablen (sog. Strukturen) (siehe Bild 54) mindestens im Workspace geladen.

Name	Value
DOE	<1x1 struct>
Eing	<1x1 struct>
Gen	<1x1 struct>
OP	<1x1 struct>
Var	<1x1 struct>
fdM	<1x1 struct>
waitbarspeicher	<1x1 struct>
wasRechnen	<1x1 struct>

Bild 54: Variablen für die Definition der Optimierungsaufgabe

Dabei enthalten diese Strukturen alle relevanten Informationen zu der definierten Optimierungsaufgabe sowie aller Verfahrenseinstellungen:

- *OP*

In *OP* sind alle Parameter für die eigentliche Definition der Optimierungsaufgabe enthalten. Bild 55 zeigt exemplarisch den Inhalt von *OP*.

- Kriterien ist ein Cellarray das die Namen aller Bewertungskriterien enthält.
- Kriteriengewichtung enthält die Gewichtung der Einzelkriterien

- *Zustandsgrößen* ist ein Cellarray das die Namen aller Zustandsgrößen enthält.(Definition siehe Kapitel 4.4)
- *Konditionierungszeiten* Zeit in Sekunden (Definition siehe Kapitel 4.4)
- *Verstellzeit* Zeit in Sekunden (Definition siehe Kapitel 4.4)
- *Rauschen* ist der Rauschanteil in % mit dem die simulierten Messdaten (Zustandsgrößen) verrauscht sind.
- *P0* aktuell eingestellte Designparameterwerte vor Beginn der versuchsbasierten Optimierung
- *Designvar* ist ein Cellarray das die Namen aller Designvariablen enthält (Definition siehe Kapitel 4.4)
- *UGrenze* und *OGrenze* beschreiben den Designraum. Das heißt, die Ober- und die Untergrenzen in der die Designvariablen gewählt werden dürfen.
- *Parameter* ist ein Cellarray, das die Namen aller unbekannt Parameter enthält. (Definition siehe Kapitel 4.4)
- *ParameterUGrenze* bzw. *ParameterOGrenze* enthält die Grenzen in der sich die unbekannt Parameter befinden dürfen.
- *TechnSys* ist der Dateiname des Simulinkmodells das für den Versuch verwendet wird. (simuliertes physisches System/Prüfstandsmodell) exemplarisches Beispiel (siehe Bild 47)

Field ▲	Value
Kriterien	<1x2 cell>
Kriteriengewichtung	[1 1]
Zustandsgrößen	'x'
Konditionierungszeiten	0
Verstellzeit	0
Rauschen	10
P0	[0.2000 0.2000]
Designvar	<1x2 cell>
UGrenze	[0.1000 0]
OGrenze	[0.5000 1]
Parameter	<1x1 cell>
ParameterUGrenze	0
ParameterOGrenze	100
TechnSys	'techSys1'

Bild 55: Beispiel Inhalt OP

- *DOE* enthält alle Einstellungen für das DOE- Verfahren (siehe Bild 56)
  - *AnzahlVariationenproDesignvar* enthält die Anzahl der Variationsstufen für jede Designvariable.
  - *Grad* enthält die Ordnung der einzelnen Designvariablen (für das Polynomialmodell)

- *OptAnzahlSchritte* gibt die Anzahl der Schritte an mit der nach dem Optimum des gebildeten Modells gesucht wird.

Field ▲	Value
AnzahlVariationenproDesignvar	4
Grad	[3 3]
OptAnzahlSchritte	30

Bild 56: exemplarischer Inhalt DOE

- *fdM* enthält zunächst nur den Namen *Modell* des Simulinkmodells das als Modellvorstellung verwendet werden soll
- *Gen* enthält die Einstellung der verwendeten genetischen Optimierer (siehe Bild 57)
  - *options* ist eine Struktur in der Optionen eingestellt werden können wann die Optimierung beendet wird oder wie das Ergebnis dargestellt wird (siehe Matlabhilfe)
  - *nvars* ist die Anzahl der Variablen, die optimiert werden sollen.
  - *LB, UB* enthält die Grenzen in denen die Variablen optimiert werden
  - Mit *A,b,Aeq,beq* können zusätzliche Bedingungen (Grenzen) festgelegt werden in denen sich die Designparameter (hier *x*) optimiert werden. Die dazugehörigen Gleichungen bzw. Ungleichungen sind:

$$A x \leq b \quad (9.3.1)$$

$$Aeq x = beq \quad (9.3.2)$$

Field ▲	Value
options	<1x1 struct>
nvars	1
A	[]
b	[]
Aeq	[]
beq	[]
nonlcon	[]
LB	0
UB	100

Bild 57: exemplarischer Inhalt Gen

- *Eing* enthält die Eingangssignale für das Bewertungsmanöver in der in Bild 58 illustrierten Form.

time	<301x1 double>
signals	<1x1 struct>

Bild 58: exemplarischer Inhalt Eing

- *Var* enthält die Gewichtungen der Kriterien im DOE Verfahren
- *waitbarspeicher* enthält die Namen der einzelnen Phasen, die während des Programmdurchlaufs ausgeführt werden.
- *wasRechnen* enthält die Variablen von Bild 59. Diese Variablen geben, an welche Teile des Programms durchgeführt werden sollen. Dabei bedeutet 1 durchführen bzw. 0 nicht durchführen des entsprechenden Programmteils.

Field ▲	Value
SimrealesSys	1
ladenVersuchsdaten	0
DOE	1
GEN	1
fdMoptimierung	1
fdMVerifikation	1
Vergleich	1
DOEAnzVerVDesgnDOE	0

Bild 59: exemplarischer Inhalt wasRechnen

Abhängig wie diese Variablen (Bild 54) eingestellt sind werden nur die gewünschten Teile des Hauptprogramms *Vergleichsumgebung.m* ausgeführt. So hat der Benutzer die Möglichkeit gezielt die Auswertungen anzeigen zu lassen, die für ihn relevant sind. Außerdem wird Rechenzeit eingespart.

Das Hauptprogramm besteht aus folgenden Teilen:

- Berechnen bzw. Durchführen des Versuchs mit den Standardparametereinstellungen
- DOE Verfahren
  - Versuchsdesign
  - Versuch
  - Optimierung
  - Verifikationsversuch
- fdM Verfahren
  - Identifikationsmaöver
  - Identifikation
  - Verifikation(Modell)

- Optimierung
  - Verifikationsversuch
- Vergleich der Verfahren
- Ergebnis in der Datenbank ablegen