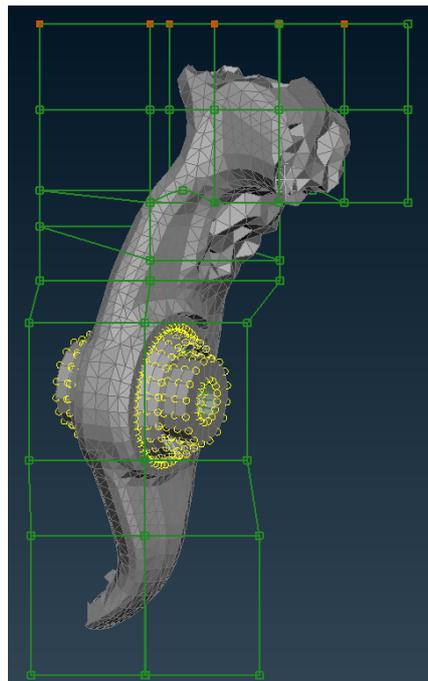


# Masterarbeit

## Automatisierte Gestaltoptimierung von Bauteilen bezüglich ihres NVH-Verhaltens durch Geometriemorphing

Michael Leupolz, B. Sc.



**Sperrvermerk:** Diese Arbeit enthält vertrauliche Daten der Daimler AG. Veröffentlichung oder Vervielfältigungen der Arbeit, auch nur auszugsweise, sind ohne ausdrückliche Genehmigung der Daimler AG nicht gestattet. Es dürfen keinerlei Kopien oder Abschriften - auch in digitaler Form - gefertigt werden. Die Arbeit ist nur Korrektoren sowie Mitgliedern des Prüfungsausschusses zugänglich zu machen. Im Rahmen des Notenfindungsprozesses kann die Arbeit weiteren Personen zugänglich gemacht werden. Dieser Sperrvermerk gilt bis zum 31.12.2021. Sollten Teile der Arbeit durch Dritte veröffentlicht werden oder bereits veröffentlicht worden sein, ist der Sperrvermerk hinfällig.

Betreuer: Prof. Dr. rer. nat. Frank Gauterin  
Projektleiter: Timo von Wysocki, M. Sc., Lehrstuhl für Fahrzeugtechnik

Nr.: 20-F-0036

Karlsruhe, Juli 2020

## **Erklärung zur Selbstständigkeit und wissenschaftlichen Praxis**

Hiermit versichere ich, die Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie, die wörtlich oder inhaltlich übernommenen Stellen als solche kenntlich gemacht zu haben.

Die Satzung des Karlsruher Instituts für Technologie (KIT) zur Sicherung guter wissenschaftlicher Praxis in der jeweils gültigen Fassung habe ich beachtet.

-----  
Karlsruhe, den 31.07.2020

# Kurzfassung

Optimierung bezüglich des NVH-Verhaltens ist oftmals nur eingeschränkt möglich. Dies liegt unter anderem daran, dass der NVH-Optimierungsprozess häufig erst spät in der Produktentwicklung startet und somit bereits viele Restriktionen bezüglich des Designs bestehen.

Gleichzeitig ist der Prozess, eine Gestalt mittels FE-Simulationsergebnissen zu optimieren, sehr zeitaufwendig. Eine Ursache dafür ist, dass zu betrachtende Designs mittels CAD erzeugt und neu vernetzt werden müssen. Dies macht umfangreiche Designstudien sehr zeit- und rechenaufwendig.

Neuere Ansätze vermeiden diese iterative Vorgehensweise in der Produktentwicklung und greifen auf Mesh-Morphing-Methoden zurück.

In dieser Arbeit wird das Box-Morphing Tool des Präprozessors ANSA verwendet. Anhand des Beispiels eines Radträgers wird eine Morphingstruktur entworfen, um dessen Lagersitze in drei Raumrichtungen zu verschieben. Der Morphingprozess läuft dabei vollständig automatisiert ab. Zur Erstellung der betrachteten Geometrievariationen wird ein Arbeitsablauf in der Optimierungsumgebung Optislang aufgebaut. Diese dient ebenfalls zur Identifikation der optimalen Geometrie sowie zur Ermittlung eines analytischen Metamodells. Die Evaluierung jeder Geometrie basiert auf einer Bewertungsgröße, welche aus Übertragungsfunktionen berechnet wird.

Diese Arbeit zeigt somit einen ganzheitlichen Prozess zur Gestaltoptimierung auf, der nahezu kein manuelles Eingreifen erfordert. Der Arbeitsablauf sowie Teilfunktionen des implementierten Morphing-Scriptes können dabei problemlos auf andere Geometrien übertragen werden.

# Abstract

## **Automated shape optimization of components regarding their NVH behavior by geometry morphing**

Optimization regarding NVH behavior is only possible to a limited extent. This is partly due to the fact that the NVH optimization process is often started late in the product development process and thus many design restrictions already exist.

At the same time, the design optimization using FE simulation results is very time consuming. The main reason is the need to generate structures in CAD and mesh them repeatedly. This makes extensive design studies very computational and time consuming.

More recent approaches use mesh morphing methods to abolish the iterative development process between CAD model and meshing.

In this thesis the box-morphing tool of the preprocessor ANSA is used. Using the example of a knuckle, a morphing structure is designed to move its bearing seats in three spatial directions. The morphing process is completely automated. A workflow is set up in the Optislang optimization environment to create the geometry variations under consideration. Optislang is also used to identify the optimal geometry and to determine an analytical meta model. The evaluation of each geometry is based on an evaluation quantity which is calculated from frequency-response functions.

This work thus shows a holistic process for shape optimization that requires almost no manual intervention. The workflow as well as sub-functions of the implemented morphing script can be easily transferred to other geometries.

# Abbildung

<b>Kurzfassung</b>	<b>v</b>
<b>Abstract</b>	<b>vi</b>
<b>Symbolverzeichnis</b>	<b>ix</b>
<b>Abkürzungsverzeichnis</b>	<b>xii</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Motivation der Arbeit . . . . .	1
1.2 Zielsetzung der Arbeit . . . . .	2
1.3 Struktur der Arbeit . . . . .	2
<b>2 Grundlagen</b>	<b>3</b>
2.1 Finite-Elemente-Methode . . . . .	3
2.2 Optimierungsalgorithmen . . . . .	4
2.3 Statistische Auswertung . . . . .	6
<b>3 Stand der Technik</b>	<b>10</b>
3.1 Morphingprinzipien . . . . .	10
3.2 Versuchsplanung . . . . .	20
<b>4 Geometriemorphing</b>	<b>27</b>
4.1 Aufbau der Morphingstruktur . . . . .	27
4.2 Gescrriptetes Morphing . . . . .	32
<b>5 Aufbau einer Designstudie</b>	<b>49</b>
5.1 Designvariation mittels Optislang . . . . .	49
5.2 Gesamtprozess der Designstudie . . . . .	50
5.3 Simulation - Frequency Response Function . . . . .	53
5.4 Optimierungsparameter . . . . .	54
5.5 Optimierung mittels Optislang . . . . .	55
5.6 Exemplarische Designstudie . . . . .	56
<b>6 Bewertung Designstudie</b>	<b>57</b>
<b>7 Zusammenfassung und Ausblick</b>	<b>65</b>
7.1 Zusammenfassung . . . . .	65
7.2 Ausblick . . . . .	65

<b>A Anhang</b>	<b>67</b>
<b>B Abbildungsverzeichnis</b>	<b>71</b>
<b>C Tabellenverzeichnis</b>	<b>73</b>
<b>Quellenverzeichnis</b>	<b>74</b>

# Symbolverzeichnis

Tabelle 1: Lateinische Symbole

Symbol	Bedeutung	Einheit
$c_{ijk}$	Koordinaten von Kontrollknotenpunkten vor Morphing	—
$c_j$	Funktionsursprung der j-ten Funktion	—
$c'_{ijk}$	Koordinaten von Kontrollknotenpunkten nach Morphing	—
$C^k$	Stetigkeit k-ten Grades	—
$d$	Verschiebungsvektor	—
$d$	Faktor	—
$E_b$	Koordinate Ebene b	—
$E_c$	Koordinate Ebene c	—
$f$	allgemeine Funktion	—
$h_i$	Ortsvektor handle point i vor Morphing	—
$h'_i$	Ortsvektor handle point i nach Morphing	—
$[Grid_{current}^{Control}]$	Matrix an Kontrollknotenpunkten vor Morphing	—
$[Grid_{new}^{Control}]$	Matrix an Kontrollknotenpunkten nach Morphing	—
$[Grid_{current}^{Deformable}]$	Matrix an verschiebbaren Knotenpunkten vor Morphing	—
$[Grid_{new}^{Deformable}]$	Matrix an verschiebbaren Knotenpunkten nach Morphing	—
$\Delta^{Control}$	Verschiebungsvektor Kontrollknotenpunkte	—
$M$	original Geometrie	—
$M'$	verformte Geometrie	—
$n$	Anzahl an Beobachtungen	—
$n_r$	Versuchsaufwand	—
$n_f$	Anzahl an Faktoren	—
$n_l$	Anzahl an Stufen	—
$N_i$	Basisfunktion	—
$N_j$	Basisfunktion	—
$N_k$	Basisfunktion	—
$P_3$	Koordinate Punkt 3	—
$\Delta P_3$	Verschiebung Punkt 3	m
$r$	Radius (Koordinatensystem)	—
$R1_X$	Verschiebung von R1 in X-Richtung	mm
$R1_Y$	Verschiebung von R1 in Y-Richtung	mm
	⋮	

Tabelle 1: Lateinische Symbole – Fortsetzung

Symbol	Bedeutung	Einheit
$R1_Z$	Verschiebung von R1 in Z-Richtung	mm
$R^2$	Bestimmtheitsmaß	—
$R^2_{Y,X}$	Bestimmtheitsmaß von Regression mit Beobachtungen Y und Variablen X	—
$R^2_{Y,X \sim i}$	Bestimmtheitsmaß von Regression mit Beobachtungen Y ohne Abhängigkeiten von $X_i$	—
$s$	Standardabweichung	variabel
$s_{Y_{Test}}$	Standardabweichung einer Testreihe	variabel
$s_{\hat{Y}_{Test}}$	Standardabweichung einer geschätzten Testreihe	variabel
$s$	Koordinatenachse eines lokalen Koordinatensystems	—
$s^2$	Stichprobenvarianz	variabel
$[T]$	Transformationsmatrix	—
$t$	Koordinatenachse eines lokalen Koordinatensystems	—
$u$	Koordinatenachse eines lokalen Koordinatensystems	—
$u_1$	Koordinatenachse eines lokalen Koordinatensystems	—
$u_2$	Koordinatenachse eines lokalen Koordinatensystems	—
$u_3$	Koordinatenachse eines lokalen Koordinatensystems	—
$\vec{v}$	Vektor	—
$\vec{u}$	Vektor	—
$w_j$	j-ter Gewichtungsvektor	—
$x$	Variable	variabel
$\mathbf{x}$	Ortsvektor eines Geometriepunktes	—
$x_{ij}$	Punkt ij im Versuchsplan	—
$x_j$	Punkt j im Versuchsplan	—
$x_j^*$	Spiegelung von Punkt j im Versuchsplan	—
$X_i$	i-te Variable	variabel
$\Delta X_1$	Verschiebung Ebene 1	m
$\Delta X_2$	Verschiebung Ebene 2	m
$X_{Ebene}$	Koordinate Morphingebene	—
$X_{ij}$	Koordinate i von Punkt j	—
$X_{Soll}$	Sollkoordinate	—
$X_{E1j}$	Koordinate j von Ebene 1	—
$X_{E2j}$	Koordinate j von Ebene 2	—
	⋮	

Tabelle 1: Lateinische Symbole – Fortsetzung

Symbol	Bedeutung	Einheit
$\bar{x}$	Mittlere Variable	variabel
$y$	Beobachtung	variabel
$y_i$	i-te Beobachtung	variabel
$y_{iTest}$	i-te Beobachtung einer Testreihe	variabel
$\hat{y}$	geschätzte Beobachtung	variabel
$\hat{y}_i$	i-te geschätzte Beobachtung	variabel
$\hat{y}_{iTest}$	i-te geschätzte Beobachtung einer Testreihe	variabel

Tabelle 2: Griechische Symbole

Symbol	Bedeutung	Einheit
$\alpha_1$	Parametervektor	—
$\alpha_2$	Parametervektor	—
$\alpha_3$	Parametervektor	—
$\alpha_4$	Parametervektor	—
$\gamma$	Rotationswinkel	°
$\epsilon$	Faktor	—
$\eta_i$	Zielfunktion für i	dB
$\bar{\eta}_i$	Gemittelte Zielfunktion für i	dB
$\mu_Y$	Mittelwert der Beobachtungen	variabel
$\mu_{\hat{Y}}$	Mittelwert der geschätzten Beobachtungen	variabel
$\mu_{Y_{Test}}$	Mittelwert der Beobachtungen einer Testreihe	variabel
$\mu_{\hat{Y}_{Test}}$	Mittelwert der geschätzten Beobachtungen einer Testreihe	variabel
$\phi$	Parameter	—
$\varphi$	Parameter	—
$\varphi$	Funktion	—

# Abkürzungsverzeichnis

- CAD** Computer-Aided Design
- CoD** Coefficient of Determination
- CoI** Coefficient of Importance
- CoP** Coefficient of Prognosis
- DM-FFD** Direct Manipulation Free-Form-Deformation
- DSG** Drehstabgestänge
- EFFD** Extended Free-Form Deformation
- FEM** Finite-Elemente-Methode
- FFD** Free-Form Deformation
- FRF** Frequency-Response-Function
- LHD** Latin-Hypercube Design
- LHS** Latin-Hypercube Sampling
- NLPQL** Non-Linear Programming by Quadratic Lagrangian
- NVH** Noice, Vibration, Harshness
- Rbfd** Radial Basis Function Deformation

# 1 Einleitung

Dieses Kapitel ist in drei Unterkapitel unterteilt. Hierbei wird zunächst die Motivation für die vorliegende Masterarbeit erörtert. Anschließend wird die Aufgabenstellung in Form der Zielsetzung konkretisiert und die Strukturierung der Arbeit dargelegt.

## 1.1 Motivation der Arbeit

Noise, Vibration, Harshness (NVH) ist ein Kriterium, welches der Kunde als maßgeblich für die Qualität eines Fahrzeugs ansieht.<sup>1</sup>

Schwingungen werden dabei bezüglich des Frequenzbereiches, in dem sie auftreten unterschieden. Der Teilbereich der Vibration ist von 0 bis 25 Hz definiert und bildet durch Fahrmanöver eingeleitete Schwingungen sowie die ersten Eigenfrequenzen von Teilsystemen ab. Der Harshness Bereich erstreckt sich von 25 bis 100 Hz und stellt alle Schwingungen dar, die sowohl als Vibration als auch akustisch wahrgenommen werden. Alle Schwingungen mit Frequenzen über 100 Hz werden hauptsächlich akustisch wahrgenommen, sodass diese als Noise zusammengefasst werden.<sup>2</sup>

Vor allem der Wandel in der Automobilindustrie, von einem Verbrennungsmotor zu einem E-Motor, führt zu einem deutlich geringeren Geräuschpegel den der Fahrer wahrnimmt.<sup>3</sup> Dies hat zur Folge, dass ein deutlich geringeres Masking<sup>4</sup> auftritt und dadurch Störgeräusche anderer Komponenten mehr in den Vordergrund rücken.<sup>5</sup>

Gerade um Geometrien bezüglich ihres NVH Verhaltens zu verbessern, können Morphing Prozesse enorme Unterstützung leisten. Das liegt daran, dass mittels Morphing die folgenden Problematiken verbessert werden können. Das NVH Verhalten wird meist erst spät im Entwicklungsprozess analysiert. Zu diesem Zeitpunkt ist jedoch das Computer-Aided Design (CAD)-Modell häufig bereits komplett fertiggestellt, wodurch sämtliche Änderungen sehr teuer werden. Außerdem muss eine Umgebung vorhanden sein, in der alle Teilkomponenten zusammengefügt werden können. Diese muss auch NVH Simulationen ermöglichen. Zuletzt müssen die zu untersuchenden Änderungen anhand der Umgebung schnell umgesetzt und analysiert werden können.<sup>6</sup>

Morphing kann somit als Schnittstelle zwischen Optimierungstool und CAD-Modell genutzt werden und bietet eine Möglichkeit, den Design Optimierungsprozess zu verändern und damit zu beschleunigen.<sup>7</sup>

---

<sup>1</sup> Siebenpfeiffer 2019, S. V.

<sup>2</sup> Winer, Bergles et al. 2011, S.239-240.

<sup>3</sup> Küppers 2015.

<sup>4</sup> Zwicker und Fastl 1999, S.61.

<sup>5</sup> Rick, Franck et al. 2015, Sekundärzitat Zwicker.1999.

<sup>6</sup> van der Auweraer, van Langenhove et al. 2007.

<sup>7</sup> Brune, Weber Martins et al. 2018.

## 1.2 Zielsetzung der Arbeit

Ziel dieser Arbeit ist es, den Gesamtprozess der Optimierung eines Bauteils mittels Box-Morphing darzulegen. Dies umfasst den Aufbau einer Morphingstruktur sowie eine Sensitivitätsanalyse und eine Optimierung bezüglich einer definierten Bewertungsgröße. Die Gestaltvariation soll dabei automatisiert mittels eines Python-Scriptes ausgeführt werden. Dabei soll gezeigt werden, dass Geometrieänderungen an einem Bauteil dessen Schwingungsverhalten wesentlich ändern können. Des Weiteren soll das Prinzip des Box-Morphings kritisch durchleuchtet und auf seine Eignung zur Bauteilvariation von komplexen Geometrien untersucht werden.

## 1.3 Struktur der Arbeit

Die vorliegende Arbeit wird inhaltlich in sechs Kapitel unterteilt.

Das diesem Kapitel folgende **Kapitel 2** beinhaltet theoretische Grundlagen, welche für das weitere Verständnis der Arbeit nötig sind und nicht als grundlegendes Wissen vorausgesetzt werden können. Dabei wird insbesondere auf Optimierungsalgorithmen, die Finite-Elemente-Methode (FEM) und die statistische Auswertung eingegangen.

Anschließend wird in **Kapitel 3** der Stand der Technik mit häufig verwendeten Morphingprinzipien sowie den sich dahinter befindenden mathematischen Beschreibungen dargestellt. Des Weiteren wird in diesem Kapitel ein Einblick in die Versuchsplanung gegeben. Dabei wird explizit auf die im weiteren Verlauf dieser Arbeit verwendeten Methoden eingegangen.

Folgend wird der vom Autor bearbeitete Arbeitsaufwand dargelegt. Dieser beginnt in **Kapitel 4** mit nötigen Begriffsdefinitionen der spezifischen Morphingumgebung. Anschließend erfolgt eine grundlegende Erörterung der angelegten Morphingstruktur und was beim Anlegen dieser zu beachten ist. Der zweite Teil dieses Kapitels behandelt das gescriptete Morphing. Hierbei werden insbesondere die verwendeten Methoden zur Verzerrung des Objektes erörtert.

In **Kapitel 5** wird der ganzheitliche Prozess der Gestaltoptimierung dargestellt. Dabei wird zunächst der generelle Aufbau einer Designstudie mittels Sensitivitätsanalyse dargestellt. Anschließend wird der automatisierte Prozess des Morphings aller Gestaltvariationen erklärt und auf die Bewertungsgröße eingegangen. Als finaler Schritt wird die Optimierung mittels Optislang erklärt und eine tatsächliche Designstudie exemplarisch aufgebaut.

**Kapitel 6** legt die Ergebnisse aus der betrachteten Designstudie dar. Diese werden interpretiert und es erfolgt eine Bewertung der Morphingmethode.

Abschließend erfolgt in **Kapitel 7** eine kritische Betrachtung der in dieser Arbeit gewonnenen Ergebnisse und Beobachtungen sowie ein Ausblick auf weitere Forschungsthematiken und Verbesserungsansätze.

## 2 Grundlagen

Dieses Kapitel umfasst alle technologischen sowie mathematischen Grundlagen, welche für das Verständnis der weiteren Kapitel nötig sind und nicht als grundlegendes Wissen vorausgesetzt werden. Dies umfasst die im weiteren Verlauf der Arbeit betrachteten Optimierungsalgorithmen sowie einen Einblick in die FEM und Methoden zur Versuchsauswertung.

### 2.1 Finite-Elemente-Methode

Eine der wichtigsten Methoden zur akustischen Simulation ist die FEM. Die Simulation erhält im Entwicklungsprozess eine steigende Bedeutung. Dies resultiert aus der wegfallenden Notwendigkeit, mehrere Prototypen herzustellen, wodurch der Entwicklungsprozess effizienter wird. Beschränkungen in der Rechenleistung und der aufzubringenden Zeit erzwingen einen möglichst effizienten Aufbau des Simulationsprozesses.<sup>1</sup>

Zur Anwendung der FEM wird das betrachtete Bauteil zunächst als CAD-Modell abgebildet. Aus diesem wird dann ein Finite-Elemente-Modell abgeleitet, indem die Bauteilgeometrie mit einem FE-Netz approximiert wird. Das Gitter besteht wiederum aus endlich vielen Knotenpunkten, welche zu Finiten-Elementen verknüpft werden. Eine höhere Auflösung des Gitters ermöglicht dabei die Berechnung höherfrequenter Schwingungsmoden. Mit höherer Auflösung steigt auch der Rechenaufwand exponentiell.<sup>2</sup>

Die Grundidee der FEM ist dabei, ein mathematisches Modell, welches aus physikalischen Beobachtungen abgeleitet wird, an diskreten Punkten zu lösen. Die FEM gehört somit zu den Diskretisierungsverfahren. Mit der FEM können generell alle durch partielle Differentialgleichungen beschriebenen Modelle berechnet werden. Die Lösungen stellen dabei jedoch nur Näherungslösungen und keine exakten Lösungen dar.<sup>3</sup>

Die Anwendungsbereiche erstrecken sich dabei über alle ingenieurtechnische Bereiche, unter anderem auch in die in dieser Arbeit betrachtete Strukturmechanik.<sup>4</sup>

Die zur Diskretisierung verwendeten Elemente können dabei theoretisch jegliche Formen annehmen. Als 2-D Elemente werden meist Drei- oder Vierecke verwendet, für 3-D Elemente meist Tetraeder oder Hexaeder.<sup>5</sup>

Folgend werden 2-D Elemente als „Shell“ und 3-D Elemente als „Solid“ bezeichnet. Auf der Basis eines Finite-Elemente-Modells wird das Schwingungsverhalten eines Bauteils simuliert.

---

<sup>1</sup>Martin 2010, S.60-61.

<sup>2</sup>Martin 2010, S.63-64.

<sup>3</sup>Wagner 2017, S.1-2.

<sup>4</sup>Wagner 2017, S.4.

<sup>5</sup>Ferziger und Peric 2008, S.44.

## 2.2 Optimierungsalgorithmen

Optimierungsalgorithmen werden in der Praxis sehr vielseitig eingesetzt und dienen hauptsächlich zur Analyse und Optimierung von vernetzten Systemen.<sup>6</sup> Gerds und Lempio unterteilen Optimierungsalgorithmen in acht Kategorien. Die lineare Optimierung, ganzzahlige und kombinatorische Optimierung, Netzwerkflussprobleme, konvexe Optimierung, differenzierbare Optimierung, diskrete dynamische Optimierung, nichtlineare Optimierung und evolutionäre Algorithmen.<sup>7</sup>

Nachfolgend wird das Prinzip des evolutionären Algorithmus sowie ein nichtlineares Optimierungsverfahren näher erörtert.

### 2.2.1 Evolutionäre Algorithmen

Evolutionäre Algorithmen beruhen auf dem Prinzip der Evolution. Die Natur schafft es, durch Mutation, Rekombination oder natürliche Selektion, die optimale Lösung für ein Problem zu finden. Dieses Grundprinzip kann auch auf Simulationsreihen angewendet werden.<sup>8</sup>

Abbildung 2.1 stellt den grundlegenden evolutionären Zyklus dar. Evolutionäre Algorithmen stellen eine Kombination

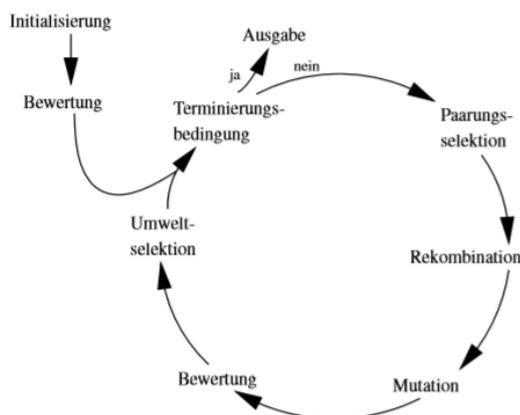


Abbildung 2.1: Schematischer evolutionärer Zyklus<sup>9</sup>

aus dem in der Natur beobachtbarem Problemlösungspotential und der Rechenleistung des Computers dar. Zur Lösung des Problems wird der Evolutionsprozess rechnerisch abgebildet. Dazu wird eine exemplarische Lösung als Objekt dargestellt und durchläuft den künstlich erzeugten Evolutionsprozess. Die Schritte, die das Objekt durchläuft, werden evolutionäre Operatoren genannt. Da diese Operatoren normalerweise auf Zufallszahlen basieren, werden

<sup>6</sup>Gerds und Lempio 2011, S.1.

<sup>7</sup>Gerds und Lempio 2011, S.1.

<sup>8</sup>Weicker 2015, S.19.

<sup>9</sup>Weicker 2015, S.25

evolutionäre Algorithmen den stochastischen Optimierungsverfahren zugeordnet.<sup>10</sup>

Nachfolgend werden die Teilprozesse des simulierten evolutionären Algorithmus kurz erläutert. Die Initialisierung stellt den Startpunkt des in Abbildung 2.1 abgebildeten Kreisprozesses dar. Diese kann zufällig oder aufgrund von Vorwissen als vermeintlich beste Lösung gewählt werden. Als Startpunkt können auch eine Teilmenge aus Individuen definiert werden. Diese werden anschließend erstmals bezüglich eines Gütekriteriums bewertet. Aufgrund des Gütekriteriums wird in der Paarselektion definiert, wieviele Kindindividuen aus einem Individuum entstehen sollen. Die Merkmale der Individuen werden gegenseitig kombiniert um die Kindindividuen zu bilden. Gemäß des Mutationsschrittes erfolgt eventuell noch eine Variation der den Kindindividuen vererbten Merkmalen. Diese ist im Regelfall jedoch klein. Die so erzeugten Kindindividuen werden ebenfalls mittels des selben Gütekriteriums bewertet und im Schritt der Umweltselektion in die Teilmenge der Eltern aufgenommen. Dabei werden entweder alle Elternindividuen durch die Kindindividuen oder zumindest ein Teil der Elternindividuen ersetzt. Dies ist allgemein nötig um die Menge der Individuen begrenzt zu halten. Jeder evolutionäre Zyklus stellt dabei eine neue Generation dar. Nach erfolgreicher Bildung einer neuen Generation wird immer das Terminierungskriterium überprüft.<sup>11</sup>

Als Terminierungskriterium werden drei generelle Szenarien unterschieden:<sup>12</sup>

1. Erfüllung eines Zielwertes des Gütekriteriums
2. Erreichen einer vorgegebenen Anzahl an Iterationen
3. Stagnation des Gütekriteriums über eine vorgegebene Anzahl an Iterationen

### 2.2.2 Non-Linear Programming by Quadratic Lagrangian

Non-Linear Programming by Quadratic Lagrangian (NLPQL) kann den Gradientenverfahren zugeordnet werden und ist das zuverlässigste Verfahren in dieser Kategorie.<sup>13</sup>

NLPQL ist eine Optimierungsmethode die ein großes Problem zunächst in kleine Teilprobleme aufteilt. Die Optimierungsrichtung wird in jeder Iteration neu berechnet. Das NLPQL beruht somit auf dem "sequential quadratic programming".<sup>14</sup>

Gradientenverfahren bestimmen anhand des Gradienten der jeweiligen Position die Suchrichtung, in welcher ein Optimum vermutet wird. Aufgrund dessen ist die mathematische Beschreibung der Problemstellung besonders relevant und höchste Restriktionen an Stetigkeit, Differenzierbarkeit, Glattheit, Skalierbarkeit und Genauigkeit der Gradientenbestimmung werden angenommen.<sup>15</sup>

Der Algorithmus identifiziert das nächstliegende lokale Optimum. Dazu wird die berechnete Steigung mit dem Wert null verglichen, wobei ein Toleranzwert festgelegt wird.<sup>16</sup>

---

<sup>10</sup>Weicker 2015, S.19-20.

<sup>11</sup>Weicker 2015, S.24-25.

<sup>12</sup>Weicker 2015, S.25.

<sup>13</sup>Norbert K. 2000, S.16, Sekundärzitat Schittkowski, K., 1985.

<sup>14</sup>Schittkowski 1986.

<sup>15</sup>Will 2005.

<sup>16</sup>Dynardo GmbH 2019, S.32-33.

In der vorliegenden Arbeit wird die Steigung mittels Zentraldifferenz berechnet.

Die Zentraldifferenz verwendet dabei die Werte der direkten Nachbarn vor und nach dem betrachteten Punkt um dessen Gradienten zu berechnen.<sup>17</sup>

## 2.3 Statistische Auswertung

Nachfolgend werden die im Rahmen dieser Arbeit verwendeten statistischen Bewertungsmethoden erörtert.

### 2.3.1 Stichprobenvarianz

Die Stichprobenvarianz kann anhand

$$s^2 = \frac{1}{n-1} \sum_{i=1}^n (y_i - \mu_Y)^2 \quad (2.1)$$

berechnet werden und beschreibt den „mittlere[n] quadratische[n] Abstand zum Mittelwert“. <sup>18</sup> In Gleichung 2.1 beschreibt  $y_i$  den Wert einer beliebigen Variable und  $\mu_Y$  den arithmetischen Mittelwert gemäß

$$\mu_Y = \frac{1}{n} \sum_{i=1}^n y_i \quad (2.2)$$

aller Variablenwerte  $y$ .<sup>19</sup>

Die Stichprobenvarianz dient als Streuungsmaß und hat den Wertebereich  $0 < s^2 \leq \infty$ . Anschaulich bedeutet eine Stichprobenvarianz von null, dass keine Streuung vorliegt. Gilt  $0 \leq s^2$  liegt eine Streuung vor. Häufig wird als Vorfaktor anstatt  $1/(n-1)$  die Abwandlung  $1/n$  verwendet. Die in Gleichung 2.1 dargestellte Ausführung ist jedoch aufgrund besserer statistischer Schätzeigenschaften zu bevorzugen.<sup>20</sup>

### 2.3.2 Standardabweichung

Die Standardabweichung wird mit

$$s = \sqrt{s^2} = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (y_i - \mu_Y)^2} \quad (2.3)$$

<sup>17</sup>Ferziger und Peric 2008, S.49.

<sup>18</sup>Kohn und Öztürk 2013, S.67-68.

<sup>19</sup>Kohn und Öztürk 2013, S.35.

<sup>20</sup>Kohn und Öztürk 2013, S.67-68.

berechnet. Sie ist somit die Wurzel der Stichprobenvarianz und hat dieselbe Einheit wie die Beobachtungen selbst. Auch die Standardabweichung liegt im Wertebereich  $0 \leq s^2 \leq \infty$  und die daraus folgenden Aussagen sind identisch zu denen der Stichprobenvarianz. Der Wert von  $s$  wird dabei sowohl von der Einheit als auch der Anzahl der Beobachtungen beeinflusst.<sup>21</sup>

### 2.3.3 Coefficient of Determination

Der Coefficient of Determination (CoD), zu deutsch „Bestimmtheitsmaß“<sup>22</sup>, wird verwendet, um die Güte der Approximation einer polynomialen Regression zu quantifizieren und wird mit

$$R^2 = \frac{SS_R}{SS_T} = 1 - \frac{SS_E}{SS_T} \quad (2.4)$$

berechnet. Als polynomiale Regression wird dabei die Bildung einer Beschreibungsfunktion anhand von Messwerten bezeichnet.<sup>23</sup> Die drei Anteile stehen dabei für Restvarianz ( $SS_E$ ), die Gesamtvarianz ( $SS_T$ ) und die Varianz der Erwartungswerte ( $SS_R$ )<sup>24, 25</sup>. Sie ergeben sich aus

$$SS_T = \sum_{i=1}^N (y_i - \mu_Y)^2 \quad (2.5)$$

$$SS_R = \sum_{i=1}^N (\hat{y}_i - \mu_{\hat{Y}})^2 \quad (2.6)$$

$$SS_E = \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (2.7)$$

Dabei ist  $\mu_{\hat{Y}}$  als Mittelwert der geschätzten Beobachtungen definiert. Der Wertebereich des Bestimmtheitsmaßes ist  $[0,1]$ , wobei für  $R = 1$  die Beobachtungen exakt von der Ansatzfunktion approximiert werden. Da sich automatisch eine exakte Approximationsfunktion ergibt, wenn die Anzahl der Stützpunkte um eins größer als der Grad des Polynoms ist, wird in der Literatur auf ein angepasstes Bestimmtheitsmaß verwiesen.<sup>26</sup>

<sup>21</sup>Kohn und Öztürk 2013, S.67-68.

<sup>22</sup>Hradetzky 1978.

<sup>23</sup>Brandt 2013.

<sup>24</sup>Hradetzky 1978.

<sup>25</sup>Montgomery und Runger 2003, S.387-397.

<sup>26</sup>Montgomery und Runger 2003, S.397-398.

### 2.3.4 Coefficient of Importance

Der Coefficient of Importance (CoI) dient zur Bewertung des Einflusses einzelner Variablen in einer Regression mit mehreren Variablen. Es handelt sich hierbei um einen von Dynardo eingeführten und von Most und Will beschriebenen Kennwert.<sup>27</sup> Der CoI kann gemäß

$$CoI(X_i, Y) = CoI_{Y,X} = R_{Y,X}^2 - R_{Y,X \sim i}^2 \quad (2.8)$$

berechnet werden.  $R_{Y,X}^2$  bezeichnet das Bestimmtheitsmaß der vollständigen Modellbeschreibung mit allen Abhängigkeiten von  $X$ . Für  $R_{Y,X \sim i}^2$  werden alle Abhängigkeiten von  $X_i$  nicht betrachtet. Wenn der CoI nahe an null liegt, bedeutet das, dass das reduzierte Modell etwa mit dem vollständigen Modell übereinstimmt. Die betrachtete Variable hat somit kaum einen Einfluss auf die Beobachtungen. Starke Wechselwirkungen zwischen einzelnen Variablen führen dazu, dass die Summe aller CoIs größer als das zugehörige Bestimmtheitsmaß ist. Der CoI ermöglicht somit eine Variable aus dem Regressionsmodell zu streichen, wenn für diese ein geringer Einfluss festgestellt wurde. Dadurch kann eine deutliche Reduktion der Variablenanzahl und somit der Komplexität des Modells erreicht werden.<sup>28</sup>

### 2.3.5 Coefficient of Prognosis

Der von Most und Will eingeführte Coefficient of Prognosis (CoP) dient dazu, ein aufgestelltes Metamodell anschließend auf Genauigkeit der Vorhersagen zu überprüfen. Dazu sind zusätzliche Werte einer Testreihe nötig. Wenn der zugehörige CoD anhand

$$R^2 = \left( \frac{\sum_{i=1}^n (y_i - \mu_Y)(\hat{y}_i - \mu_{\hat{Y}})}{(n-1)s_Y s_{\hat{Y}}} \right)^2 \quad (2.9)$$

berechnet wird, gilt

$$CoP = \left( \frac{\sum_{i=1}^n (y_{iTest} - \mu_{YTest})(\hat{y}_{iTest} - \mu_{\hat{Y}Test})}{(n-1)s_{YTest} s_{\hat{Y}Test}} \right)^2 \quad (2.10)$$

Der Wertebereich des CoP ist  $0 \leq CoP \leq 1$ , wobei für  $CoP = 1$  eine optimale Vorhersage aller getesteten Werte erreicht wird. Aus dem CoP kann eine direkte quantifizierbare Aussage getroffen werden. Konkret kann für  $CoP = 0,8$  gesagt werden, dass die Vorhersage eines Punktes anhand des Metamodells mit einer Vorhersagewahrscheinlichkeit von 80 % zutrifft. Da in konkreten Anwendungsfällen oft keine zusätzlichen Versuche für eine Testreihe durchgeführt werden, hält Optislang bei der Berechnung des Metamodells manche Versuche bewusst zurück, um mittels dieser eine Testreihe zu erzeugen.<sup>29</sup>

Ein Metamodell ist dabei ein analytisches Beschreibungsmodell, welches auf Basis bereits durchgeführter Versuche berechnet wird. Es ermöglicht bei ausreichender Genauigkeit die Vorhersage von Ergebnissen nicht untersuchter

<sup>27</sup>Most und Will 2011.

<sup>28</sup>Most und Will 2011.

<sup>29</sup>Most und Will 2008.

Parametereinstellungen. Die Berechnung des Modells erfolgt mit dafür entworfenen Algorithmen, die eine mathematische Beschreibung mit möglichst geringen Abweichungen der ermittelten Ergebniswerte ermitteln. Dabei werden auch komplexe Wechselwirkungen der betrachteten Faktoren analysiert.<sup>30</sup>

---

<sup>30</sup>Siebertz, van Bebber et al. 2017, S.231.

## 3 Stand der Technik

Dieses Kapitel befasst sich inhaltlich mit zwei unterschiedlichen Themengebieten. Zunächst werden gängige Morphingprinzipien sowie die nötigen mathematischen Operationen beschrieben. Anschließend wird ein Einblick in gängige Versuchsplanungsmodelle gegeben.

### 3.1 Morphingprinzipien

Der Begriff Morphing ist abgeleitet vom Wort Metamorphose und beschreibt die Überführung einer Ursprungsform in eine Zielform mittels mehrmaliger Transformationen. Gerade in Trickfilmen wurde dieses Verfahren bereits zu Anfang des 20. Jahrhunderts angewendet. Hierbei vor allem, um Charaktere altern zu lassen. Damals war der komplette Vorgang noch händisch durchzuführen, während dies heute meist von komplexen Algorithmen durchgeführt wird.<sup>1</sup> Das Mesh-Morphing findet mittlerweile auch in allen Ingenieursdisziplinen, wie der Automobilindustrie, Aerodynamik und Turbomaschinen, vor allem zur Formparametrisierung und Konstruktionsoptimierung Anwendung.<sup>2</sup> Nachfolgend werden drei verschiedene Formen des Mesh-Morphings näher beschrieben. Dabei erfolgt eine strukturelle Anlehnung an van der Auweraer, van Langenhove et al.<sup>3</sup> und Sieger, Menzel et al.<sup>4</sup>. In jedem der Prinzipien werden die Knotenpunkte in die drei folgenden Kategorien unterteilt:<sup>5</sup>

1. Kontrollknotenpunkte („Control nodes“) → Knotenpunkte, auf welche eine vorgegebene Transformation angewendet wird. Diese Punkte initiieren somit den Morphingprozess
2. Verschiebbare Knotenpunkte („Deformable nodes“) → Passen sich der Verschiebung der Kontrollknotenpunkte an
3. Fixierte Knotenpunkte („Fixed nodes“) → Werden vom Morphingprozess nicht beeinflusst und ändern somit ihre Position während des Morphingprozesses nicht

Folgend werden die drei populärsten Ansätze des Geometriemorphings näher erörtert. Die Nomenklatur ist dabei in der Literatur nicht einheitlich gehalten. So ist der Begriff „Free-Form Deformation“<sup>6</sup> mit „control-based approach“<sup>7</sup> und „Direct Manipulation FFD“<sup>8</sup> mit „free form morphing“<sup>9</sup> gleichzusetzen. Sieger, Menzel et al. beschreiben zudem

---

<sup>1</sup> Jackèl, Neunreither et al. 2006, S.89.

<sup>2</sup> Lupini und Epureanu 2019.

<sup>3</sup> van der Auweraer, van Langenhove et al. 2007.

<sup>4</sup> Sieger, Menzel et al. 2015.

<sup>5</sup> van der Auweraer, van Langenhove et al. 2007.

<sup>6</sup> Sieger, Menzel et al. 2015.

<sup>7</sup> van der Auweraer, van Langenhove et al. 2007.

<sup>8</sup> Sieger, Menzel et al. 2015.

<sup>9</sup> van der Auweraer, van Langenhove et al. 2007.

noch eine dritte Kategorie, das „radical basis function“<sup>10</sup>-Morphing.

Alle Morphingprinzipien stellen eine mathematische Transformation dar. Diese beschreibt Van der Auweraer wie in Gleichung 3.1 bis Gleichung 3.4 dargestellt. Die eckigen Klammern kennzeichnen dabei Matrizen.

$$[Grid_{new}^{Control}] = [T][Grid_{current}^{Control}][T]^T \quad (3.1)$$

$$[\Delta^{Control}] = [Grid_{new}^{Control}] - [Grid_{current}^{Control}] \quad (3.2)$$

$$[Grid_{new}^{Deformable}] = f([\Delta^{Control}])[Grid_{current}^{Deformable}] \quad (3.3)$$

$$f([\Delta^{Control}]) = f(\Delta, \phi, \varphi) \quad (3.4)$$

Dabei ist  $T$  eine Transformationsmatrix und  $\Delta^{Control}$  der vom Benutzer vorgegebene Verschiebungsvektor.  $f$  ist eine allgemeine Funktion mit Parametern  $\phi$  und  $\varphi$ , die maßgeblich die Gestalt der gemorphten Geometrie bestimmt. Durch Variieren der Parameter kann die Form dieser Funktion verändert werden. van der Auweraer, van Langenhove et al. bezeichnen sie deshalb als „morphing shape function“. Die Parameter  $\varphi$  und  $\phi$  nehmen dabei Einfluss auf die relative Lage der verschiebbaren Knotenpunkte. Ein kleines  $\varphi$  verschiebt diese näher an die Kontrollknotenpunkte, während die verschiebbaren Knotenpunkte für ein großes  $\phi$  näher an den fixierten Knotenpunkten liegen.<sup>11</sup>

### 3.1.1 Free-Form Deformation

Bei der Free-Form Deformation (FFD) wird zunächst ein Gitter aus Parallelepipeden um das zu verformende Objekt aufgebaut. Mit der Verformung dieses Gitters verformt sich auch das in ihm enthaltene Objekt.<sup>12</sup>

Ein Parallelepiped definiert ein Volumen, welches von sechs angrenzenden Parallelogrammen aufgespannt wird. Gegenüberliegende Parallelogramme sind dabei deckungsgleich.<sup>13</sup>

Die Verformung wird somit nicht direkt auf das Element, sondern auf Kontrollblöcke angewendet. Diese werden im weiteren Verlauf dieser Arbeit als Morphingboxen bezeichnet. Da die Verformung nicht direkt auf das Element angewendet wird, bezeichnen van der Auweraer, van Langenhove et al. diese Methode als indirektes Morphing. Sie definiert die folgenden drei Schritte:<sup>14</sup>

1. Anlegen von originalen Kontrollboxen
2. Anlegen verformter Kontrollboxen
3. Morphing der Geometrie anhand der originalen und verformten Kontrollboxen

Abbildung 3.1 stellt das zugrunde liegende Prinzip dar. Links ist eine originale Geometrie  $M$  zu sehen, welche in Kontrollboxen inkludiert wird. Diese Kontrollboxen werden vom Benutzer verschoben und mittels Gleichung 3.9 die

<sup>10</sup>Sieger, Menzel et al. 2015.

<sup>11</sup>van der Auweraer, van Langenhove et al. 2007.

<sup>12</sup>Sieger, Menzel et al. 2015.

<sup>13</sup>Manutchehr-Danai 2009.

<sup>14</sup>van der Auweraer, van Langenhove et al. 2007.

Verschiebungen der inkludierten Elemente berechnet. Die verformte Geometrie  $M'$  mit verformten Kontrollboxen ist rechts dargestellt.<sup>15</sup>

Die Funktion  $f$  ist im Falle der FFD ein lineares Polynom.<sup>16</sup> Nachfolgend wird die in Gleichung 3.1 bis Gleichung 3.4 dargestellte allgemeine Berechnung für Morphingprinzipien gemäß Sieger, Menzel et al. konkretisiert.

$$\mathbf{x} = \sum_{i=0}^l \sum_{j=0}^m \sum_{k=0}^n \mathbf{c}_{ijk} N_i(u_1) N_j(u_2) N_k(u_3) \quad . \quad (3.5)$$

Mit

$$\mathbf{u}(\mathbf{x}) := (u_1, u_2, u_3) \quad (3.6)$$

$$N_p(\mathbf{u}(\mathbf{x})) = N_i(u_1) N_j(u_2) N_k(u_3) \quad (3.7)$$

$$\delta \mathbf{c}_p := \delta \mathbf{c}_{ijk} = \mathbf{c}'_{ijk} - \mathbf{c}_{ijk} \quad (3.8)$$

ergibt sich die Verschiebung an jedem Punkt  $\mathbf{x}$  zu

$$\mathbf{d}(\mathbf{x}) = \sum_p \delta \mathbf{c}_p N_p(\mathbf{u}(\mathbf{x})) \quad . \quad (3.9)$$

In dieser Beschreibung steht  $\mathbf{x}$  für die Lage eines zu verschiebenden Punktes auf dem Objekt mit den lokalen Koordinaten  $u_1, u_2, u_3$ . Die Zähler  $i, j, k$  geben die Zahl der Kontrollknotenpunkte in jeder Raumrichtung an.  $\mathbf{c}_{ijk}$  steht für die Lage eines Kontrollknotenpunktes vor der Verschiebung und  $\mathbf{c}'_{ijk}$  für dessen Lage nach der Verschiebung.  $N_i, N_j, N_k$  beschreiben Basisfunktionen. Diese definieren die Verteilung der Kontrollknotenpunkte im Raum.<sup>17</sup>

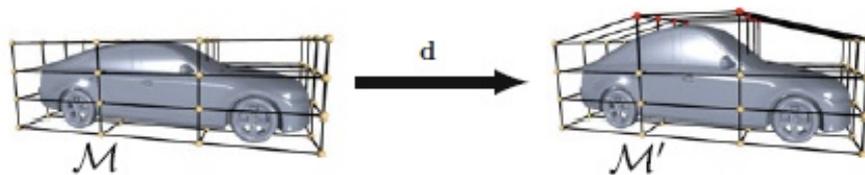


Abbildung 3.1: FFD (Beschriftung im Bild angepasst)<sup>18</sup>

In der von van der Auweraer, van Langenhove et al. verwendeten Darstellung werden diese Basisfunktionen durch die „morphing shape function“<sup>19</sup> definiert.

<sup>15</sup>Sieger, Menzel et al. 2015.

<sup>16</sup>van der Auweraer, van Langenhove et al. 2007.

<sup>17</sup>Sieger, Menzel et al. 2015.

<sup>18</sup>Sieger, Menzel et al. 2015

<sup>19</sup>van der Auweraer, van Langenhove et al. 2007.

FFD ist anwendbar auf sehr komplexe Geometrien und erlaubt dabei große wie kleine Verformungen. Da das Ausgangsnetz Einfluss auf die Qualität der Verformung hat, ist ein qualitativ hochwertiges Netz vor dem Morphingprozess nötig. Generell ist es möglich, den Verformungsprozess von verschiedenen Punkten bei einer FFD parallel auf verschiedenen Prozessoren laufen zu lassen, sodass so die Rechenzeit deutlich reduziert wird. Da die FFD ausschließlich die Knoten des Netzes verschiebt, bleiben alle Verbindungen über den Morphingprozess erhalten. Um qualitativ hochwertige Verformungen zu erhalten, werden am Übergang des nicht zu verformenden Bereiches und des zu verformenden Bereiches zusätzliche Kontrollpunkte, welche nicht verschoben werden, eingefügt. Zur Gewährleistung der „non-penetration condition“ muss ein Überschneiden der Kontrollpunkte verhindert werden. Dies wird durch eine Begrenzung der Verschiebung erreicht. Die Stetigkeit des verformten Teiles bleibt erhalten, wenn im unverformten Teil die Kontrollpunkte nicht verschoben werden.<sup>20</sup>

Durch Definition einzelner Kontrollboxen muss der Gewährleistung der Stetigkeit, gemäß Sederberg und Parry, besondere Aufmerksamkeit zugewendet werden. Dies muss an jeder Stelle betrachtet werden, an der zwei Kontrollboxen eine gemeinsame Ebene aufweisen.<sup>21</sup>

Sederberg und Parry leiten diese im Koordinatensystem  $s, t, u$  des Kontrollgitters her. Für Stetigkeit zwischen zwei benachbarten Kontrollboxen  $\mathbf{X}_1(s_1, t_1, u_1)$  und  $\mathbf{X}_2(s_2, t_2, u_2)$  mit einer gemeinsamen Ebene  $s_1 = s_2 = 0$  muss folglich

$$\frac{\partial \mathbf{X}_1(0, t, u)}{\partial s} = \frac{\partial \mathbf{X}_2(0, t, u)}{\partial s} \quad (3.10)$$

$$\frac{\partial \mathbf{X}_1(0, t, u)}{\partial t} = \frac{\partial \mathbf{X}_2(0, t, u)}{\partial t} \quad (3.11)$$

$$\frac{\partial \mathbf{X}_1(0, t, u)}{\partial u} = \frac{\partial \mathbf{X}_2(0, t, u)}{\partial u} \quad (3.12)$$

gelten. Dabei wird mit Gleichung 3.10 bis Gleichung 3.12 Stetigkeit ersten Grades ( $C^1$ ) erreicht. Um Stetigkeit k-ten Grades ( $C^k$ ) zu gewährleisten, werden die k-ten Ableitungen gleichgesetzt.<sup>22</sup>

Abbildung 3.2 zeigt eine Geometrie mit  $C^0$  und eine mit  $C^1$ , das originale Bauteil ist dabei jeweils rechts oben im Bild zu sehen.

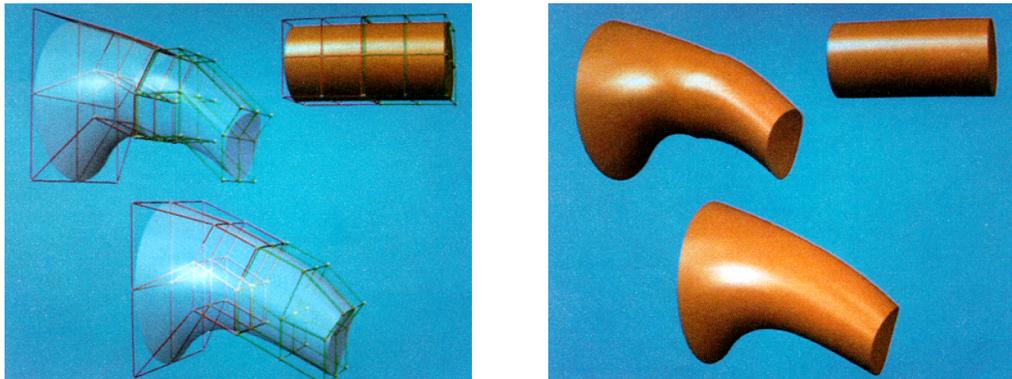
Bei lokal nötigen Verformungen der Geometrie muss ebenfalls auf die Stetigkeit des Bauteils geachtet werden. Der Übergang entsteht dabei zwischen dem zu verändernden und dem konstant gehaltenen Teil der Geometrie, sodass die Verschiebung bis zu diesem Übergang möglichst ausklingen soll. Um  $C^k$  zu gewährleisten, dürfen Kontrollknotenpunkte von k Ebenen nicht bewegt werden. Diese folgen auf die Ebene, welche den Übergang zwischen verformt und nicht verformtem Bereich darstellt.<sup>23</sup>

<sup>20</sup>Salmoiraghi, Scardigli et al. 2018.

<sup>21</sup>Sederberg und Parry 1986.

<sup>22</sup>Sederberg und Parry 1986.

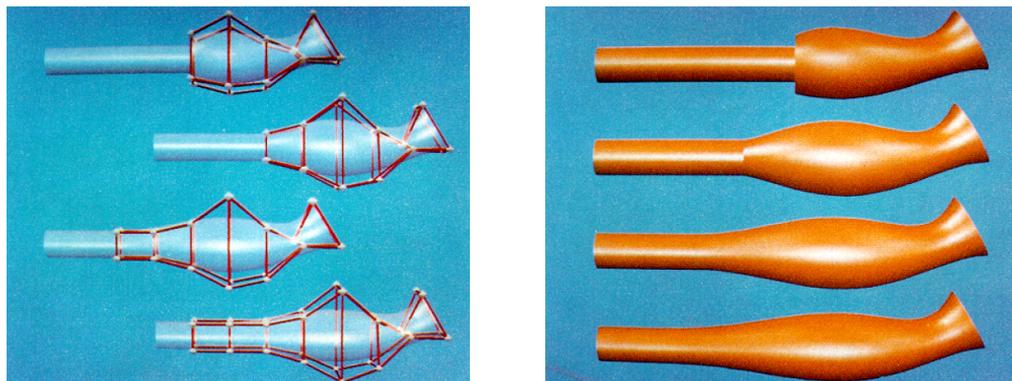
<sup>23</sup>Sederberg und Parry 1986.



(a) Kontrollknotenpunkte im Fall  $C^0$  (links oben) und  $C^1$  (mitte unten)  
 (b) Geometrie im Fall  $C^0$  (links oben) und  $C^1$  (mitte unten)

Abbildung 3.2: Veranschaulichung Bauteilstetigkeit in FFD<sup>24</sup>

In Abbildung 3.3 ist der Unterschied zwischen einem lokalen Morphing mit unterschiedlichen Stetigkeitsgraden dargestellt.



(a) Kontrollknotenpunkte lokales Morphing  $C^{-1}$ ,  $C^0$ ,  $C^1$  und  $C^2$  (von oben nach unten)  
 (b) Geometrie lokales Morphing  $C^{-1}$ ,  $C^0$ ,  $C^1$  und  $C^2$  (von oben nach unten)

Abbildung 3.3: Veranschaulichung Bauteilstetigkeit lokales Morphing in FFD<sup>25</sup>

Anwendung der FFD ermöglichen es ebenfalls, dass bei Definition entsprechender Randbedingungen das Volumen der Geometrie über den Morphingprozess konstant gehalten oder bewusst geändert wird.<sup>26</sup>

Als Erweiterung zu der FFD, in dem lediglich parallelepipedische Gitter verwendet werden, wurde die Extended Free-Form Deformation (EFFD) eingeführt. Das zugrundeliegende Prinzip ist äquivalent dem der FFD. Es können aber prismatische, zylindrische oder Kombinationen aus diesen und weiteren Formen wie Kugeln als Gitter angelegt

<sup>24</sup>Sederberg und Parry 1986

<sup>25</sup>Sederberg und Parry 1986

<sup>26</sup>Sederberg und Parry 1986.

werden. Dies ermöglicht eine deutlich bessere Darstellung der ursprünglichen Geometrie durch das Gitter. Außerdem können mittels EFFD variable Muster auf eine Oberfläche, zum Beispiel kreisförmiger Ausbuchtung, projiziert werden.<sup>27</sup>

### 3.1.2 Direct Manipulation FFD

Direct Manipulation Free-Form-Deformation (DM-FFD) ist ein Ansatz, bei dem die Verschiebung an einzelnen Punkten der Geometrie direkt vom Benutzer vorgegeben werden kann. Damit soll verhindert werden, dass wie bei Anwendung der FFD, die am Kontrollgerüst anzusetzende Verschiebung nicht immer intuitiv, mit der sich ergebenden Geometrie des Bauteiles, einhergeht. Dies ist exemplarisch in Abbildung 3.4 dargestellt. Hierbei soll eine ebene Fläche der Geometrie erreicht werden. Die Punkte stellen Kontrollknotenpunkte dar. Die durchgezogene Linie steht dabei für die Geometrie nach dem Morphing und die gestrichelte Linie für die Geometrie vor dem Morphing.<sup>28</sup>

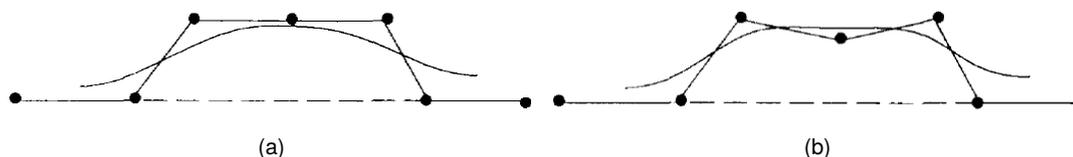


Abbildung 3.4: a) FFD mit gerader Linie in Morphingbox b) FFD mit ebener Fläche<sup>29</sup>

Im Gegensatz zur FFD muss vom Anwender in der DM-FFD keine Kontrollboxstruktur angelegt werden, welches die Geometrie des ursprünglichen Modells möglichst gut abbildet. Dennoch muss eine Struktur angelegt werden, welche aber meist in der Benutzeroberfläche ausgeblendet wird. Die Kontrollknotenpunkte müssen dabei nicht die Geometrie des Bauteils möglichst gut nachbilden. Die dabei verwendete Anzahl an Kontrollknotenpunkten ist entscheidend für die Variabilität der Verformung. Die Struktur ist nach wie vor nötig, um den eigentlichen Verformungsprozess abzubilden.<sup>30</sup>

Die Verschiebung wird vom Benutzer an mindestens einem Punkt der Geometrie vorgegeben. Diese werden in der Literatur unter anderem als „object points“<sup>31</sup> benannt. Anhand dieser werden die Verschiebungen für alle Kontrollknotenpunkte berechnet. Sieger, Menzel et al. definieren zur mathematischen Beschreibung „handle points“  $\mathbf{h}_m$ . Diese dienen als Randbedingungen und geben eine fixe Verschiebung  $\delta \mathbf{h}_i = \mathbf{h}'_i - \mathbf{h}_i$  vor. Die zur Erfüllung

<sup>27</sup> Coquillart 1990.

<sup>28</sup> Hsu, Hughes et al. 1992.

<sup>29</sup> Hsu, Hughes et al. 1992

<sup>30</sup> Menzel, Olhofer et al. 2006.

<sup>31</sup> Hsu, Hughes et al. 1992.

der Randbedingungen nötigen Verschiebungen der einzelnen Kontrollknotenpunkte können durch Lösung des folgenden Gleichungssystems berechnet werden.<sup>32</sup>

$$\begin{bmatrix} N_1(\mathbf{u}(\mathbf{h}_1)) & \cdots & N_n(\mathbf{u}(\mathbf{h}_1)) \\ \vdots & \ddots & \vdots \\ N_1(\mathbf{u}(\mathbf{h}_m)) & \cdots & N_n(\mathbf{u}(\mathbf{h}_m)) \end{bmatrix} \begin{pmatrix} \delta \mathbf{c}_1^T \\ \vdots \\ \delta \mathbf{c}_n^T \end{pmatrix} = \begin{pmatrix} \delta \mathbf{h}_1^T \\ \vdots \\ \delta \mathbf{h}_m^T \end{pmatrix} \quad (3.13)$$

Dieses Gleichungssystem ist dabei entweder unter- oder überbestimmt, je nachdem wie viele voneinander unabhängige Kontrollknotenpunkte und Randbedingungen definiert wurden. Die Verschiebung der Kontrollknotenpunkte hängt maßgeblich davon ab. Unterbestimmtheit führt zur „least-norm“-Lösung und Überbestimmtheit zur „least-square“-Lösung des Gleichungssystems. Ersteres führt zur geringsten Verschiebung der Kontrollknotenpunkte, zweiteres zu der geringsten Abweichung von den vorgegebenen Verschiebungen. Dabei gewährleistet die DM-FFD folglich weder, dass die vorgegebenen Verschiebungen exakt erreicht werden, noch, dass das erhaltene Modell physikalisch sinnvoll ist. Abbildung 3.5 stellt das Morphingprinzip dar.<sup>33</sup>

Im Falle der Überbestimmtheit der Matrix, also der Least-Square Lösung, ist die Abweichung zwischen Soll- und Ist-Verschiebung quantifizierbar. Treten hierbei große Abweichungen auf, sollten mehr Kontrollknotenpunkte verwendet werden.<sup>34</sup>

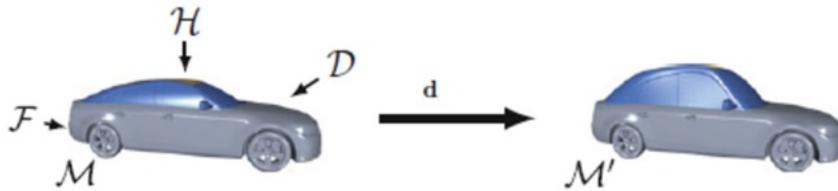


Abbildung 3.5: DM-FFD; H („handle vertices“), F („fixed vertices“), D („deformable vertices“)<sup>35</sup>

### 3.1.3 Radial Basis Function Deformation

Radial Basis Function Deformation (RBF) ermöglicht mehr Freiheitsgrade als FFD und DM-FFD und ist somit flexibler in der Gestaltgebung des Bauteils. Außerdem kann die Basisfunktion einen Energiewert beschreiben. Dessen Minimierung führt zu möglichst smoothen und physikalisch korrekten Geometrien.<sup>36</sup>

Der Begriff smooth wird im Folgenden für eine Geometrie verwendet, welche stetige Übergänge und somit keine sprunghaften Gradientenänderungen der Kontur aufweist und damit auch als glatt beschrieben würde.

<sup>32</sup>Sieger, Menzel et al. 2015.

<sup>33</sup>Sieger, Menzel et al. 2015.

<sup>34</sup>Hsu, Hughes et al. 1992.

<sup>35</sup>Sieger, Menzel et al. 2015

<sup>36</sup>Sieger, Menzel et al. 2015.

Mathematisch wird die Verschiebungsfunktion mittels mehrerer Teilfunktionen  $\varphi$  („kernel functions“) dargestellt. Als Teilfunktionen werden unter anderem die in Tabelle 3.1 dargestellten Funktionen verwendet.<sup>37</sup>

Name	f(x)
Gaussian	$\varphi(r) = e^{-(\epsilon r)^2}$
Multiquadric	$\varphi(r) = \sqrt{1 + (\epsilon r)^2}$
Inverse multiquadric	$\varphi(r) = \frac{1}{\sqrt{1^2 + (\epsilon r)^2}}$
Polyharmonic spline in $\mathbb{R}^d$	$\varphi_k(r) = \begin{cases} r^{2k-d} & d \text{ odd,} \\ r^{2k-d} \log r, & d \text{ even} \end{cases}$

Tabelle 3.1: Radial Basis Function Deformation Ansatzfunktionen<sup>38</sup>

Die Form der gemorphten Geometrie ist dabei stark abhängig von der verwendeten Basisfunktion. Bezüglich der Stetigkeit sind Gaußfunktion und multiquadratische Ansätze zu bevorzugen. Diese beinhalten jedoch einen zusätzlichen Formfaktor  $\epsilon$  der ausgewählt werden muss. Diese Aufgabe ist sehr komplex, da eine Abwägung zwischen der gewünschten Approximationsgenauigkeit und der damit einhergehenden numerischen Instabilität getroffen werden muss. Im Falle der polyharmonischen Funktionen ist keine Bestimmung eines Formfaktors nötig. Die Stetigkeit der Geometrie ist jedoch auch von geringerem Grad als die der Gaußfunktion und multiquadratischen Ansätze.<sup>39</sup>

Der Einfluss der ausgewählten Basisfunktion ist beispielhaft in Abbildung 3.6 dargestellt.

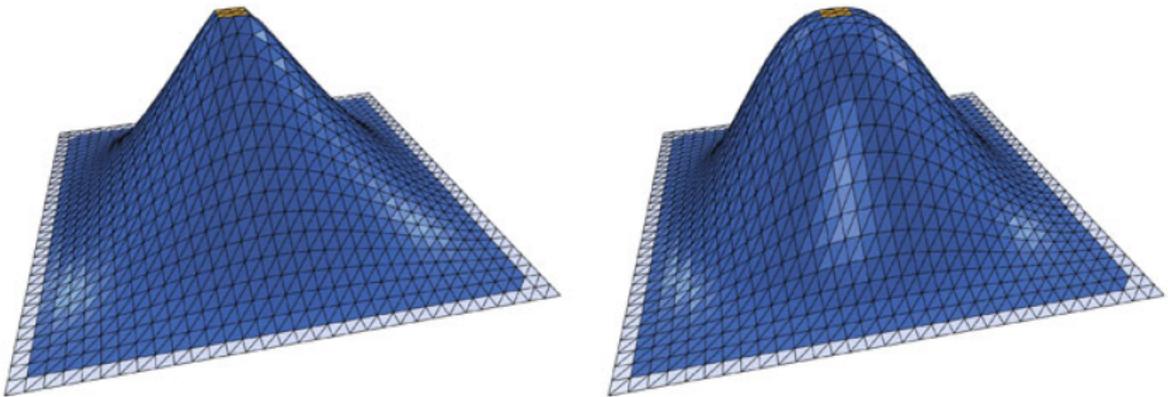


Abbildung 3.6: Einfluss Basisfunktion; links biharmonisch, rechts triharmonisch; goldene Teile werden verschoben, graue konstant gehalten und blaue deformiert<sup>40</sup>

<sup>37</sup> Sieger, Menzel et al. 2014.

<sup>38</sup> Sieger, Menzel et al. 2014

<sup>39</sup> Sieger, Menzel et al. 2014.

<sup>40</sup> Sieger, Menzel et al. 2015

Tabelle 3.1 stellt dabei keineswegs alle möglichen Basisfunktionen dar. Generell erfolgt auch eine Unterteilung in Funktionen mit „compact support“ und mit „global support“. „Compact support“ Funktionen sind nur in einem Teilbereich oder vorgegebenen Radius definiert. Außerhalb dieses Bereichs werden sie zu null gesetzt. Dies ist bei „global support“ Funktionen nicht der Fall, sie haben auf den gesamten Morphingbereich Einfluss. Im Falle der „compact support“ Funktionen ist die Lösung des sich ergebenden Gleichungssystems effizienter ausführbar, umso kleiner der Einflussradius ist, da somit eine schwach besetzte Matrix für Gleichung 3.16 resultiert. Gleichzeitig sinkt jedoch die Präzision mit der die vorgegebenen Verschiebungen erreicht werden. Die in Tabelle 3.1 beschriebenen Funktionen sind vom Typ „global support“. Bei Verwendung dieser Formen ergibt sich eine vollbesetzte Matrix.<sup>41</sup>

Damit an den Punkten vorgegebener Verschiebung die tatsächliche Verschiebung vorliegt, werden die einzelnen Funktionen so verschoben, dass sie an den handle points keinen Einfluss haben und  $c_j = \mathbf{h}_j$  gilt. Die verwendeten Teilfunktionen werden außerdem mit einer Gewichtung  $\mathbf{w}$  multipliziert.<sup>42</sup>

$$\varphi_j(\mathbf{x}) = \varphi(\|\mathbf{x} - \mathbf{c}_j\|) \quad (3.14)$$

Die tatsächliche Verschiebung jedes Knotenpunktes ergibt sich aus einer linearen Kombination der einzelnen Teilfunktionen und einem trivariaten Polynom.<sup>43</sup>

$$\mathbf{d}(\mathbf{x}) = \sum_{j=1}^m \mathbf{w}_j \varphi_j(\mathbf{x}) + \alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 z \quad (3.15)$$

Mittels Gleichung 3.16 werden die Gewichtungsfunktionen und die einzelnen Parameter  $\alpha_i$  berechnet. Anhand dieser können gemäß Gleichung 3.15 die einzelnen Koordinaten eines jeden Punktes berechnet werden.<sup>44</sup>

$$\begin{bmatrix} \varphi_1(\mathbf{h}_1) & \cdots & \varphi_m(\mathbf{h}_1) & x(\mathbf{h}_1) & y(\mathbf{h}_1) & z(\mathbf{h}_1) & 1 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \varphi_1(\mathbf{h}_m) & \cdots & \varphi_m(\mathbf{h}_m) & x(\mathbf{h}_m) & y(\mathbf{h}_m) & z(\mathbf{h}_m) & 1 \\ x(\mathbf{h}_1) & \cdots & x(\mathbf{h}_m) & 0 & \cdots & \cdots & 0 \\ y(\mathbf{h}_1) & \cdots & y(\mathbf{h}_m) & 0 & \cdots & \cdots & 0 \\ z(\mathbf{h}_1) & \cdots & z(\mathbf{h}_m) & 0 & \cdots & \cdots & 0 \\ 1 & \cdots & 1 & 0 & \cdots & \cdots & 0 \end{bmatrix} \begin{pmatrix} \mathbf{w}_1^T \\ \vdots \\ \mathbf{w}_m^T \\ \alpha_1^T \\ \alpha_2^T \\ \alpha_3^T \\ \alpha_4^T \end{pmatrix} = \begin{pmatrix} \delta \mathbf{h}_1^T \\ \vdots \\ \delta \mathbf{h}_m^T \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (3.16)$$

<sup>45,46,47</sup> Das Prinzip ist in Abbildung 3.7 dargestellt.

<sup>41</sup> Boer, van der Schoot et al. 2007.

<sup>42</sup> Sieger, Menzel et al. 2015.

<sup>43</sup> Sieger, Menzel et al. 2015.

<sup>44</sup> Sieger, Menzel et al. 2015.

<sup>45</sup> Michler 2011.

<sup>46</sup> Sieger, Menzel et al. 2015.

<sup>47</sup> Jakobsson und Amoignon 2007.

<sup>48</sup> Sieger, Menzel et al. 2015

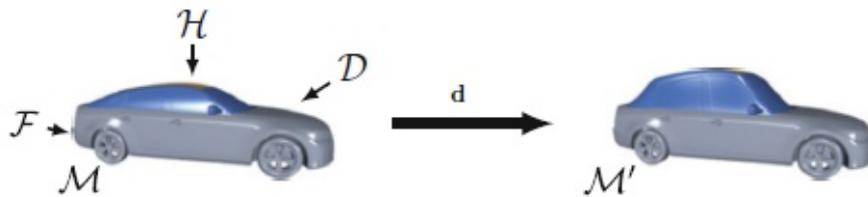


Abbildung 3.7: RBF-D; H („handle vertices“), F („fixed vertices“), D („deformable vertices“) (Beschriftung im Bild angepasst) <sup>48</sup>

### 3.1.4 Vergleich der Methoden

Sieger, Menzel et al. führen einen Vergleich der dargelegten Methoden durch. Dabei ist ein genereller Vergleich der Geschwindigkeit nicht möglich, da jede Methode eine leicht unterschiedliche Verzerrung des Objekts liefert und somit nicht das Erreichen eines einheitlichen Zielobjektes als Prozessende definiert werden kann. Dennoch ist es möglich, eine Zeit die ein Morphingprozess mit einer gegebenen Anzahl an Vorgaben benötigt, zu messen. Dabei zeigt sich, dass DM-FFD die längste Zeit benötigt und RBF-D die kürzeste. Eine parallele Berechnung der lokalen Koordinaten bei einer FFD führt zu gleicher Laufzeit wie bei RBF-D. Robust sind prinzipiell alle drei Verfahren, wobei bei DM-FFD numerische Instabilitäten beim Lösen des Gleichungssystems entstehen können. Robust bedeutet in diesem Fall, dass die Algorithmen stabil laufen, trotz Irregularitäten in den Eingangsdaten (zum Beispiel Dreiecke mit extremen Seitenverhältnissen). Die Qualität der Verformung wird generell damit bewertet, ob zum einen die Netzqualität über möglichst große Verformungen erhalten bleibt und zum anderen die Übergänge smooth dargestellt werden, sowie keine physikalischen Unstimmigkeiten auftreten. Bezogen auf die Smoothness liefert die RBF-D Methode deutlich bessere Ergebnisse, wobei zu beachten ist, dass bei der FFD Methode die Form und Güte des Gittergerüsts einen sehr starken Einfluss auf die Smoothness der Zielform hat. Falls nur ein Teil des Objekts in das Gittergerüst eingebettet ist, kann es zu sehr unsauberem (nicht smoothen) Übergängen an der Grenze des Gitters kommen. In diesen Fällen sollte das Gitter, wie in Abbildung 3.8 dargestellt, erweitert werden, damit eine Pufferzone entsteht. <sup>49</sup>

Die Differenz zwischen Soll- und Ist-Koordinate eines Referenzpunktes nach der Verschiebung ist ebenfalls ein Kriterium für die Güte der Morphingmethode. Dabei zeigt sich, dass mittels RBF-D die beste und mittels FFD die schlechteste Präzision erreicht wird. Die Soll-Verschiebung kann im Falle der RBF-D sogar exakt abgebildet werden. Alle Verfahren sind in der Lage, Volumennetze zu verzerren. Im Fall, dass eine global definierte trivariate Basisfunktion für die RBF-D verwendet wird, ist eine Verformung nur global umsetzbar, während FFD und DM-FFD lokal umsetzbar sind. Tabelle 3.2 fasst alle Vergleichskriterien tabellarisch zusammen. Die Flexibilität beschreibt dabei wie frei der Benutzer, bei vorgegebener Anzahl an Kontrollknoten, in der Auswahl der Soll-Form ist. Auch wenn RBF-D eine insgesamt bessere Bewertung erhält, ist diese nicht für jeden Anwendungsfall automatisch besser

<sup>49</sup>Sieger, Menzel et al. 2015.

<sup>50</sup>Sieger, Menzel et al. 2015

<sup>51</sup>Sieger, Menzel et al. 2015

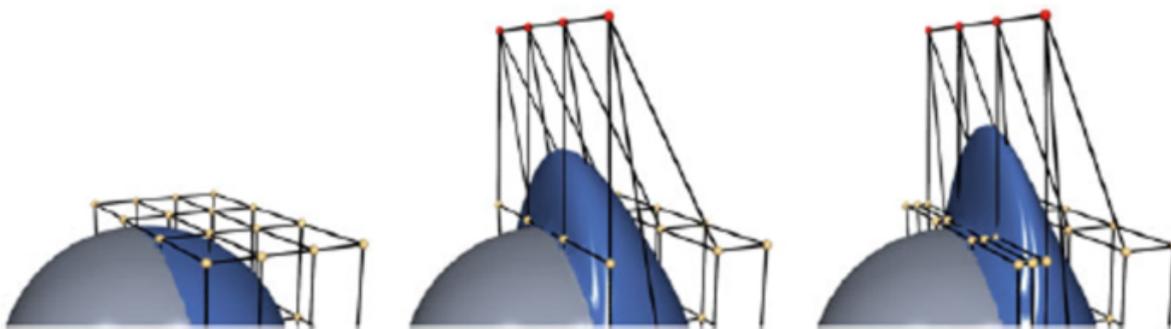


Abbildung 3.8: FFD mit Stetigkeitsproblem durch partielle Kontrollvolumen. Links: Original, Mitte: Nicht smoother Übergang, Rechts: Smoother Übergang<sup>50</sup>

	Laufzeit	Robustheit	Qualität	Flexibilität	Präzision
FFD	o	+	o	-	-
DM-FFD	-	o	o	-	o
Rbfd	o	+	+	+	+

Tabelle 3.2: Vergleich Morphingmethoden<sup>51</sup>

geeignet. Im Falle, dass eine Designanalyse durchgeführt werden soll ohne dass dabei exakte Verschiebungen erreicht werden müssen, ist dies mittels einer FFD problemlos durchführbar.<sup>52</sup>

### 3.2 Versuchsplanung

Versuchsplanung (engl. „Design of Experiments“<sup>53</sup>) bietet eine generelle Möglichkeit, aus einem minimalen Umfang an Versuchen, maximale Aussagen zu treffen.<sup>54</sup>

Zur Erstellung eines Versuchsplans werden zunächst die möglichen Einflussgrößen definiert. Diese werden als Parameter in einer Parameterliste zusammengefasst. In einer Versuchsreihe werden nicht alle möglichen Parameter simultan untersucht. Die tatsächlich in der Versuchsreihe berücksichtigten Parameter werden deshalb als Faktoren bezeichnet.<sup>55</sup>

Die Werte, welche die einzelnen Parameter annehmen können, werden als Stufen oder Level bezeichnet. Die Mindestanzahl an Stufen eines Faktors ist zwei. Um den Effekt eines Faktors entsprechend vollständig zu erfassen, sollten die Werte der Stufen gegenseitig abgestimmt werden. Dies ist wichtig, da zu kleine Abstände meist keine Beobachtungen zulassen. Die Abstände sollten außerdem so gewählt werden, dass der gesamte Faktorraum

<sup>52</sup>Sieger, Menzel et al. 2015.

<sup>53</sup>Siebertz, van Bebber et al. 2017, S. iii.

<sup>54</sup>Schiefer und Schiefer 2018, S.1.

<sup>55</sup>Siebertz, van Bebber et al. 2017, S.5.

betrachtet wird, da mögliche Extrapolationen aus dem Faktorraum heraus Risiken bergen.<sup>56</sup>

In den folgenden Kapiteln wird der Versuchsaufwand mit  $n_r$ , die Zahl der Faktoren mit  $n_f$  und die Zahl der Stufen mit  $n_l$  wiedergegeben<sup>57</sup>

Nachfolgend werden verschiedene Ansätze der Versuchsplanung dargelegt, wobei diese nur einen kleinen Ausschnitt der Versuchsplanung abbilden.

### 3.2.1 Vollfaktorplan

Im Vollfaktorplan erfolgt vor der Erstellung des Versuchsplans keine Analyse des Versuchsraums. Stattdessen werden alle möglichen Kombinationen betrachtet. Dabei wird immer nur ein Faktor variiert. Damit soll sichergestellt werden, dass die Veränderung der Bewertungsgröße sicher einer Variablen zugeordnet werden kann. Das ganze System setzt dabei voraus, dass alle Faktoren unabhängig sind und somit die Stufeneinstellungen anderer Faktoren auf die Wirkung des betrachteten Faktors keinen Einfluss haben. Gleichzeitig wird der gesamte Faktorraum von einem Bezugspunkt betrachtet, von welchem aus die Faktoren variiert werden. Es wird dabei nicht ersichtlich, ob auch die Wahl des Bezugspunktes eine Einfluss auf die erhaltenen Ergebnisse hat.<sup>58</sup>

$$n_r = n_l^{n_f} \quad (3.17)$$

Der Versuchsaufwand für etwas umfangreichere Versuchsreihen steigt somit gemäß Gleichung 3.17 exponentiell an. Tabelle 3.3 stellt den schematischen Aufbau eines Vollfaktorplans mit drei Faktoren und zwei Stufen dar.<sup>59</sup>

A	B	C	y
-	-	-	y <sub>1</sub>
+	-	-	y <sub>2</sub>
-	+	-	y <sub>3</sub>
+	+	-	y <sub>4</sub>
-	-	+	y <sub>5</sub>
+	-	+	y <sub>6</sub>
-	+	+	y <sub>7</sub>
+	+	+	y <sub>8</sub>

Tabelle 3.3: Vollfaktorplan mit drei Faktoren (A, B, C) und zwei Stufen gekennzeichnet mit – + (nach Siebertz, van Bebber et al.)<sup>60</sup>

<sup>56</sup>Siebertz, van Bebber et al. 2017, S.6.

<sup>57</sup>Siebertz, van Bebber et al. 2017, S.6.

<sup>58</sup>Siebertz, van Bebber et al. 2017, S.6, 7.

<sup>59</sup>Siebertz, van Bebber et al. 2017, S.6, 7.

<sup>60</sup>Siebertz, van Bebber et al. 2017, S.7

### 3.2.2 Matrixexperiment nach Taguchi

Unter der Annahme, dass zwischen den einzelnen Faktoren keine Wechselwirkungen bestehen, kann das standardisierte orthogonale Feld von Taguchi, gemäß Klein, verwendet werden. Für einen Anwendungsfall in dem vier Faktoren mit jeweils drei Stufen betrachtet werden, stellt Abbildung 3.9 den Versuchsplan dar. Die Spaltennummer steht hierbei für die einzelnen Faktoren, während die Werte eins bis drei in der Tabelle die Stufen kennzeichnen. Alle Stufen werden dabei für jeden Faktor gleich häufig betrachtet, was als Orthogonalität des Systems bezeichnet wird. Dass alle Stufen gleich häufig betrachtet werden ist wichtig, damit bei der Berechnung der Faktoreinflüsse, kein Faktor einen wahrscheinlichkeitstheoretischen Vorteil erlangt. Die Matrix wird dabei nach dem Hochvermengungsprinzip angelegt und es werden mehrere Faktorstufen simultan variiert.<sup>61</sup>

Exp. Nr.	Spalten-Nr.			
	1	2	3	4
1	1	1	1	1
2	1	2	2	2
3	1	3	3	3
4	2	1	2	3
5	2	2	3	1
6	2	3	1	2
7	3	1	3	2
8	3	2	1	3
9	3	3	2	1

Abbildung 3.9: Matrixexperiment  $L_9$ <sup>62</sup>

Zur Auswertung der Versuche wird ein vom Benutzer definiertes Qualitätskriterium verwendet. Mittels diesem wird eine Zielfunktion  $\eta_i$  definiert, welche schließlich zur Berechnung der einzelnen Einflüsse dient. Die Zielfunktion benennt Taguchi als Signal/Rausch-Verhältnis und gibt sie in dB an. Die Einflüsse einzelner Faktorstufen können mittels Gleichung 3.18 berechnet werden

$$\bar{\eta}_{i_m} = \frac{1}{n}(\eta_j + \eta_k + \eta_l) \quad . \quad (3.18)$$

$i$  steht dabei für einen Faktor und  $m$  kennzeichnet die betrachtete Stufe.  $j, k, l$  geben die Versuche an, in denen Stufe  $i$  von Faktor  $m$  verwendet wurde. Die dargestellte Formel gilt dabei für das dargestellte Matrixexperiment mit drei Stufen und neun Versuchen. Im Falle eines Versuchsplans mit anderer Häufigkeit einer Faktorstufe, verändert

<sup>61</sup> Klein 2014, S.35-36.

<sup>62</sup> Klein 2014, S.36

sich ebenfalls die Anzahl der zu addierenden Zielfunktionen. Die Variable  $n$  gibt an, wie oft Faktorstufe  $i$  in der gesamten Versuchsreihe verwendet wurde. Für Faktorstufe eins des Faktors zwei gilt somit

$$\bar{\eta}_{21} = \frac{1}{3}(\eta_1 + \eta_4 + \eta_7) \quad . \quad (3.19)$$

Wenn die dabei besten mittleren  $\bar{\eta}$  Werte kombiniert werden, wird normalerweise auch die insgesamt beste Faktoreinstellung erreicht.<sup>63</sup>

In Fällen, in denen ein Beschreibungsmodell der Versuchsergebnisse erstellt werden soll, muss zunächst die Annahme, dass die gewählten Faktoren unabhängig voneinander sind, überprüft werden. Eventuelle Abhängigkeiten müssen in die Ansatzfunktion miteinbezogen werden. Können keine Abhängigkeiten identifiziert werden, genügt die Verwendung eines rein additiven Ansatzes, welcher die einzelnen Wirkungen superponiert.<sup>64</sup>

Im oben dargestellten Fall wurde zunächst angenommen, dass zwischen den einzelnen Faktoren keine Wechselwirkungen bestehen. Dies ist natürlich nicht immer der Fall. Zur Überprüfung von Wechselwirkungen werden die Faktoren gemäß Abbildung 3.10 gegeneinander aufgetragen.

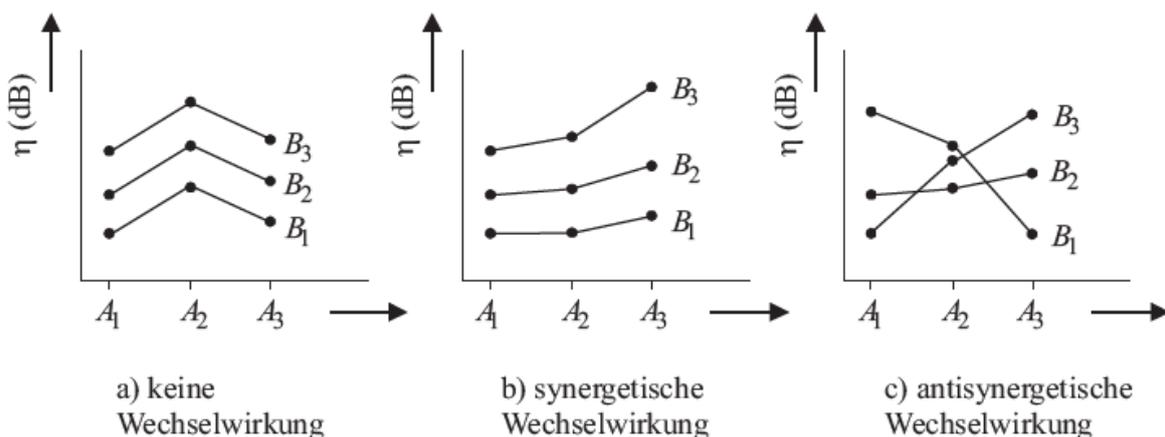


Abbildung 3.10: Beispiel für Faktor-Wechselwirkungen<sup>65</sup>

Dies ermöglicht deren Unterteilung in drei Kategorien:<sup>66</sup>

- Keine Wechselwirkung
- synergetische Wechselwirkung
- antiszynergetische Wechselwirkung

<sup>63</sup>Klein 2014, S.39-40.

<sup>64</sup>Klein 2014, S.52.

<sup>65</sup>Klein 2014, S.52

<sup>66</sup>Klein 2014, S.52-53.

Synergetisch bedeutet, dass die Entwicklungsrichtung der Faktoren dieselbe ist. Antisynergetisch, dass deren Entwicklungsrichtung gegenläufig ist. Hierbei zeigt sich, dass im Falle synergetischer Wechselwirkungen das verwendete additive Modell meist ausreichend gut ist. Für antisynergetische Wechselwirkungen gilt dies nicht. Generell sollte der Versuchsplan angepasst werden, sobald Wechselwirkungen zwischen Faktoren festgestellt werden.<sup>67</sup>

### 3.2.3 Latin-Hypercube

Eine  $n_r \times n_f$  Matrix  $X^{LHD}$  beschreibt nach Siebertz, van Bebber et al. ein Latin-Hypercube Design (LHD) vollständig. Die Spalten der Matrix stellen dabei zufällige Werte von 1 bis  $n_r$  dar. Auf Basis des LHDs wird ein Latin-Hypercube Sampling (LHS) erstellt. Dazu wird von den Werten des LHDs ein Zufallswert im Intervall  $[0, 1)$  abgezogen und der erhaltene Wert durch  $n_r$  geteilt. Abbildung 3.11 stellt dies exemplarisch dar.<sup>68</sup>

$$x_{ij} = \frac{x_{ij}^{LHD} - \text{rand}[0, 1)}{n_r} \quad \text{mit } x_{ij}^{LHD} \in 1, 2, \dots, n_r \quad (3.20)$$

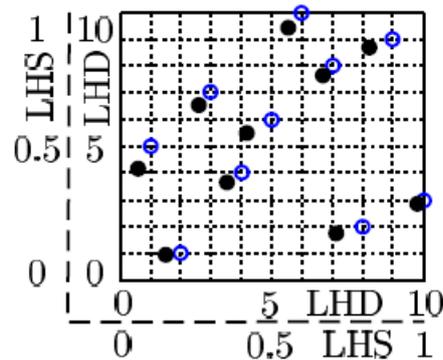


Abbildung 3.11: LHD (blau), LHS (schwarz) mit  $n_r = 10$ ,  $n_f = 2$  <sup>69</sup>

Wird statt eines Zufallswertes im Intervall  $[0, 1)$  von jedem Punkt  $x_{ij}$  des LHDs der konstante Wert 0,5 abgezogen, ergibt sich ein zentriertes LHD. Da beim LHS alle Werte zufällig angelegt werden, ist weder eine Gleichverteilung noch eine Korrelationsfreiheit des Versuchsplans gewährleistet. Dies muss somit gesondert betrachtet werden. Abbildung 3.12 zeigt zwei mögliche LHDs. Die Punkte des blauen LHDs bilden eine Gerade, was einer Korrelation entspricht. Durch Verteilung der Punkte auf einer Geraden wird ebenfalls keine Verteilung im Faktorraum erreicht. Im Falle des schwarzen LHDs ist keine Korrelation ersichtlich und eine Verteilung im Faktorraum liegt ebenfalls vor, sodass diese das zu präferierende LHD darstellt.<sup>70</sup>

Um eine geringere Varianz des LHSs zu erzielen, wird dieses in ein orthogonales Feld überführt. Dazu erfolgt

<sup>67</sup>Klein 2014, S.52-53.

<sup>68</sup>Siebertz, van Bebber et al. 2017, S.205.

<sup>69</sup>Siebertz, van Bebber et al. 2017, S.205

<sup>70</sup>Siebertz, van Bebber et al. 2017, S.205-206.

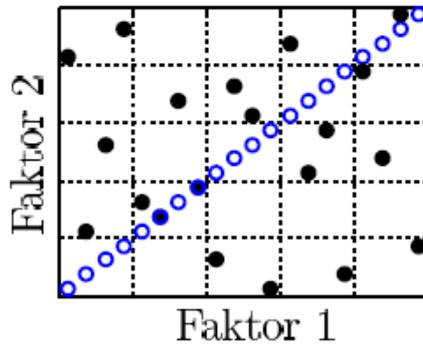


Abbildung 3.12: LHS mit 20 Testpunkten <sup>71</sup>

zunächst eine Zuordnung von Teilmengen der Stufen des LHSs zu jeder Stufe des orthogonalen Feldes. Anschließend werden die Stufen des orthogonalen Feldes durch eine zufällige Auswahl der zugewiesenen Teilmenge ausgetauscht. Das ist in Tabelle 3.4 dargestellt. Die linke Tabelle zeigt einen orthogonalen Versuchsplan für vier Faktoren mit jeweils drei Stufen und neun Versuchen. Aus diesem wird durch das eben beschriebene Prinzip ein LHD für vier Faktoren mit je neun Stufen generiert. Die Anzahl der Versuche bleibt dabei erhalten.<sup>72</sup>

$L_9 (3^4)$					$OA-LHD (9, 9^4)$			
1	1	1	1	→	1	3	1	2
1	2	2	2		2	4	6	5
1	3	3	3		3	9	7	8
2	1	2	3		5	2	5	9
2	2	3	1		6	5	9	1
2	3	1	2		4	8	3	6
3	1	3	2		8	1	8	4
3	2	1	3		7	6	2	7
3	3	2	1		9	7	4	3

Tabelle 3.4: Links: Orthogonaler Versuchsplan, Rechts: LHS auf Basis des orthogonalen Versuchsplans <sup>73</sup>

Durch die zufällige Anlegung der Versuche, sind  $n_r^{n_f}$  an unterschiedlichen Versuchsplänen für den selben Faktorraum möglich. Um die Auswahl des Versuchsplans zu erleichtern, werden vor allem in Computer-Experimenten symmetrische LHDs verwendet. Die Symmetrie besteht dabei zum Punkt  $(n_r + 1)/2$ . Dieser bildet das Zentrum des Versuchsplans. Die gespiegelten Punkte können anhand

$$x_j^* = n_r + 1 - x_j \tag{3.21}$$

<sup>71</sup>Siebertz, van Bebber et al. 2017, S.206

<sup>72</sup>Siebertz, van Bebber et al. 2017, S.206.

<sup>73</sup>Siebertz, van Bebber et al. 2017, S.207

berechnet werden. Die Laufvariable  $j$  wird dabei für jeden Faktor unabhängig betrachtet. Im Falle eines LHDs mit ungerader Stufenzahl, kann diese Rechnung problemlos angewendet werden. Hat das LHD eine ungerade Stufenzahl, muss Stufe Nummer  $(n_r + 1)/2$  in jedem Faktor den Wert  $(n_r + 1)/2$  haben, um das Zentrum der Symmetrie abzubilden. Zusätzlich zur Verwendung von symmetrischen LHDs ist auch darauf zu achten, dass möglichst geringe Korrelationen der einzelnen Faktoren auftreten. Dies führt zu genaueren Vorhersagen mittels des Metamodells.<sup>74</sup>

Ein weiterer Vorteil von LHDs ist, dass diese immer „noncollapsing“ sind. Das bedeutet, dass kein Wert eines Faktors mehr als einmal in der Versuchsreihe auftritt. Dadurch wird sichergestellt, dass alle Experimente zusätzliche Informationen in das Metamodell einbringen. Dies ist für „collapsing“ Versuchspläne, in denen sich herausstellt, dass eine Variable keinen Einfluss auf die Bewertungsgröße hat, nicht der Fall. Ein „noncollapsing“ sowie ein „collapsing“ Versuchsplan ist in Abbildung 3.13 dargestellt.<sup>75</sup>

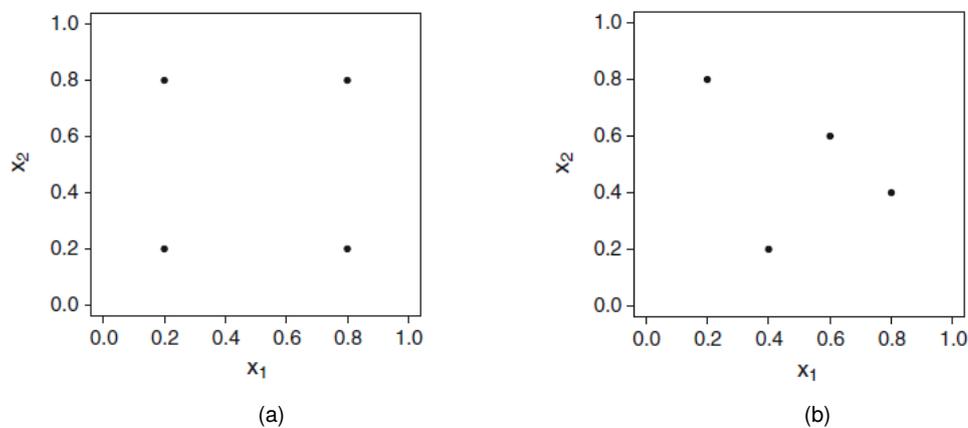


Abbildung 3.13: a) Collapsing Versuchsplan, b) Non-Collapsing Versuchsplan<sup>76</sup>

Eine optimierte Version des Latin-Hypercube ist das space-filling Latin-Hypercube. Dieses versucht alle Varianten möglichst gleichmäßig über den gesamten Versuchsraum aufzuteilen, um damit zum Beispiel eine minimale Korrelation zwischen den Faktoren oder maximalen Abstand einzelner Werte zu erreichen.<sup>77</sup>

<sup>74</sup>Siebertz, van Bebber et al. 2017, S.205-209.

<sup>75</sup>Dean, Voss et al. 2017, S.772-774.

<sup>76</sup>Dean, Voss et al. 2017, S.773

<sup>77</sup>Ranjan und Spencer 2014.

## 4 Geometriemorphing

In diesem Kapitel wird zunächst der Aufbau der Morphingstruktur erörtert. Als Umgebung wird hierzu der Präprozessor ANSA von BETA CAE Systems verwendet. Dieser bietet eine umfangreiche Plattform zum Anlegen einer Morphingstruktur. In dieser Arbeit wird das Prinzip des EFFD (siehe Abschnitt 3.1.1) verwendet. Nach Erörterung der erstellten Morphingstruktur wird auf den Morphingprozess eingegangen. Dieser wird in der vorliegenden Arbeit automatisiert, mittels eines Python-Scriptes, umgesetzt.

### 4.1 Aufbau der Morphingstruktur

Nachfolgend wird der Aufbau sowie die Entstehung der Morphingstruktur erläutert. Zum weiteren Verständnis werden zunächst nötige Begriffe erörtert. Diese wurden teilweise bereits in Abschnitt 3.1 definiert. Dennoch ist eine erneute Beschreibung einzelner Komponenten sinnvoll, da die Literatur unterschiedliche Begriffe verwendet und somit die Begriffe im Zusammenhang dieser Arbeit klar definiert sind.

**Nested-Elements** (engl. verschachtelt) beschreibt in ANSA ein manuell angelegtes Element. Dieses Element setzt sich aus beliebig vielen Unterelementen, zum Beispiel Knotenpunkten oder Tetraedern, sowie einem Bezugspunkt zusammen. Die Zusammenfassung der Unterelemente sorgt dafür, dass diese, bei korrekter Behandlung im Morphingprozess, sich alle relativ zu dem gewählten Bezugspunkt verschieben. Infolge dessen wird ein Nested-Element, das vor dem Morphingprozess einen Kreis dargestellt hat, auch nach dem Morphingprozess einen Kreis darstellen. Ohne die Zusammenfassung würde der Kreis bei entsprechendem Morphingprozess zu einem Oval verzerrt werden. Die Nested-Elements können dabei auch in ihren Freiheitsgraden begrenzt werden. Wenn alle Freiheitsgrade eliminiert werden, wird das Element beim Morphingprozess nicht verändert. Die Elemente werden als „Frozen“ bezeichnet und blau hinterlegt. Wenn mindestens ein Freiheitsgrad erlaubt ist, ist die Rede von „Rigid“ und die Elemente werden gelb hinterlegt. Die Freiheitsgrade werden dabei als Zahlen von eins bis sechs beschrieben. Eins stellt die X-Achse dar, vier eine Rotation um die X-Achse.<sup>1</sup>

Abbildung 4.1 stellt ein solches Nestes-Element exemplarisch dar.

**Morphpoint/Kontrollpunkte** Diese definieren im Falle von Morphingboxen auf Polyeder-Basis die Eckpunkte des Polyeders. Durch Bewegung der Kontrollpunkte werden die zugehörigen Morphingboxen verformt, womit die tatsächliche Verformung der eingebetteten Elemente erfolgt.

**Morphingbox** Eine Morphingbox ist im folgenden definiert durch alle direkt angrenzenden Morphpoints oder auch Kontrollpunkte. Die Form der Morphingbox kann dabei nahezu frei gewählt werden. Die Basisformen sind

---

<sup>1</sup>BETA CAE Systems 2017, S.2162.

Quader und Zylinder. Im Falle eines Quaders sind somit acht Kontrollpunkte, sechs Morphingebenen und zwölf Morphingkanten einer Morphingbox zugewiesen. Aneinander angrenzende Morphingboxen teilen sich entsprechende Elemente. Auf Basis des Quaders können jegliche andere Polyeder, auch mit Rundungen in Kanten, dargestellt werden. Dazu werden beliebig viele zusätzliche Kontrollpunkte auf einer Kante hinzugefügt. Die zugehörigen Ebenen werden dabei entsprechend mit verformt. In die jeweiligen Boxen werden die entsprechenden Elemente geladen, sodass diese Elemente auf die Verformung ihrer zugehörigen Morphingbox reagieren.

**Negative Elemente** Durch Verschiebung eines Knotens innerhalb eines Elements kann es zu Elementen mit negativen Volumen kommen. Dies ist vor allem der Fall, wenn ein Knoten durch eine ihm gegenüber liegende Seite seines Elements hindurch geschoben wird. Das Auftreten von negativen Elementen definiert maßgeblich den Bereich, in welchem ein Morphen sinnvoll ist, da bei Auftreten negativer Elemente keine Simulationen mehr möglich sind.

**CBAR** Balkenelement das zwei Knotenpunkte verbindet.<sup>2</sup>

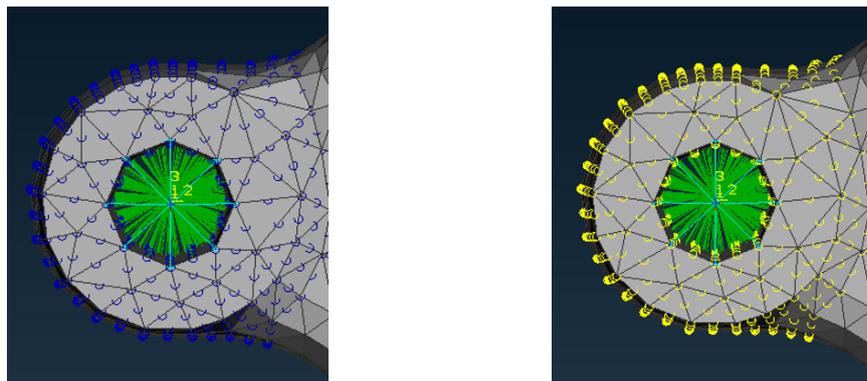


Abbildung 4.1: Veranschaulichung Nested-Element (Links: Rigid, Rechts: Frozen)

Aus den in Abschnitt 3.1.1 erörterten Grundlagen geht hervor, dass umso größer die Verschiebung eines Anbindungspunktes sein soll, desto mehr Elemente müssen in den Morphingprozess einbezogen werden, um beste Bedingungen für eine akzeptable Netzqualität zu schaffen. Ist dies nicht der Fall, werden einzelne Elemente relativ zu ihren Dimensionen sehr stark verzerrt, was zu ungenügender Netzqualität oder gar negativen Elementen führen kann. Deshalb ist es erstrebenswert, möglichst viele Elemente mit in den Morphingprozess einzubeziehen. Als Netzqualität ist dabei zunächst allgemein die Anzahl und Form der Elemente zu verstehen. Ein Netz von guter Qualität weist dabei möglichst identische Elemente mit nicht extremen Seitenverhältnissen auf. Gleichzeitig sollte die Anzahl der Elemente in Zusammenhang mit den aufzulösenden Schwingungsphänomenen einhergehen.

Wie das Übersichtsbild Abbildung 4.2 des betrachteten Bauteils zeigt, sind Verschiebungen an sechs Anbindungspunkten zu betrachten. Die Beschriftungen sind dabei so zu interpretieren, dass beispielsweise der Punkt R1 den Anbindungspunkt des Radträgers an die Schubstrebe darstellt. Aufgrund der Nähe zwischen dem Drehstabgestänge

<sup>2</sup>Siemens 2017, Kap. 11 S.20-24.

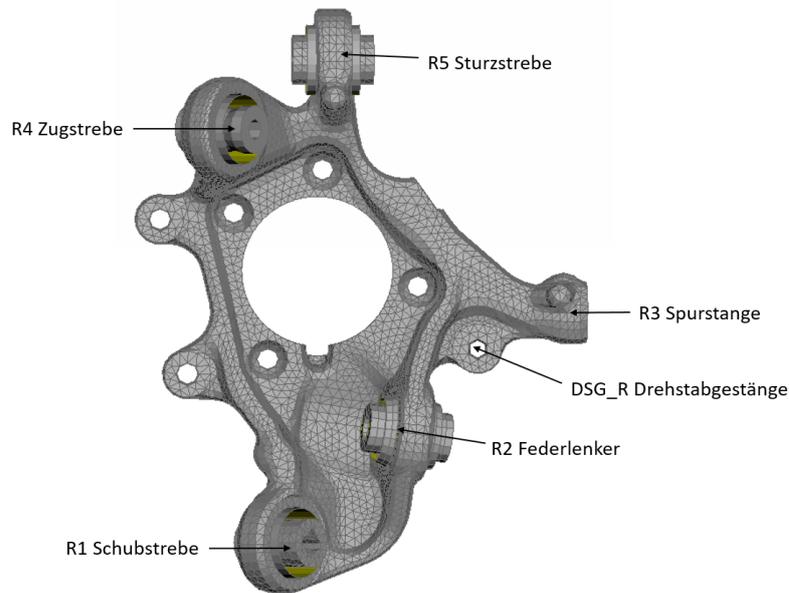


Abbildung 4.2: Übersichtsbild Radträger mit Benennungen

(DSG) und R3 ist eine unabhängige Verschiebung der beiden Punkte nicht ohne Weiteres unbeschränkt möglich. Aus diesem Grund erfolgt die Verschiebung des DSGs, im Umfang dieser Arbeit, in Abhängigkeit der Verschiebung von Anbindungspunkt R3. Der DSG wird dabei für eine Verschiebung in X-Richtung festgehalten. Bei einer Verschiebung in Y- und Z-Richtung von R3 vollzieht er eine Ausgleichsbewegung. Der Kompromiss wird an dieser Stelle getroffen, da sonst eine unabhängige Bewegung der beiden Punkte mit lediglich ca. 3 mm möglich ist, bevor die Netzqualität unzureichend wird. Die Morphingstruktur soll jedoch eine Verschiebung der Anbindungspunkte mit Verschiebungsvektor der Länge 10 mm ermöglichen.

Um die Anbindungspunkte R1, R2, R3, R4 und R5 unabhängig voneinander verschieben zu können, werden diese Punkte zunächst in Hexa-Block Systeme (siehe Abbildung 4.3) eingebunden. Dadurch ist sichergestellt, dass keine der Morphingboxen, welche einen Anbindungspunkt beinhaltet, sich eine Ebene mit einer der anderen Morphingboxen mit Anbindungspunkt teilt. Nur so kann eine wirkliche Entkopplung realisiert werden. Anschließend werden Ebenen nahe aneinander liegender Morphingboxen aufeinander gelegt. Dies führt dazu, dass die Kontinuität der Geometrie während des Morphingprozesses weitestgehend gewährleistet wird. Um möglichst große Verschiebungen zu ermöglichen, sollten die an Morphingboxen mit Anbindungspunkt angrenzenden Morphingboxen ebenfalls möglichst groß sein. In diesen Geometriebereichen wird später ein Hauptteil der Verschiebung realisiert. Ebenen, die während des Morphingprozesses gesondert verschoben werden sollen, werden parallel zu den Ebenen des globalen Koordinatensystems angeordnet. Somit führen die Verschiebungen dieser Ebenen lediglich zu einer Verzerrung der Geometrie in einer Koordinatenachse, wodurch die exakte Vorhersage des sich ergebenden Zustandes vereinfacht wird und nicht für eine Verschiebung in einer Achse mehrere Ebenen mit individuell zu bestimmenden Verhältnissen verschoben werden müssen.

Teile des Radträgers, welche unverändert bleiben sollen, müssen nicht mit in die Morphingstruktur miteinbezogen

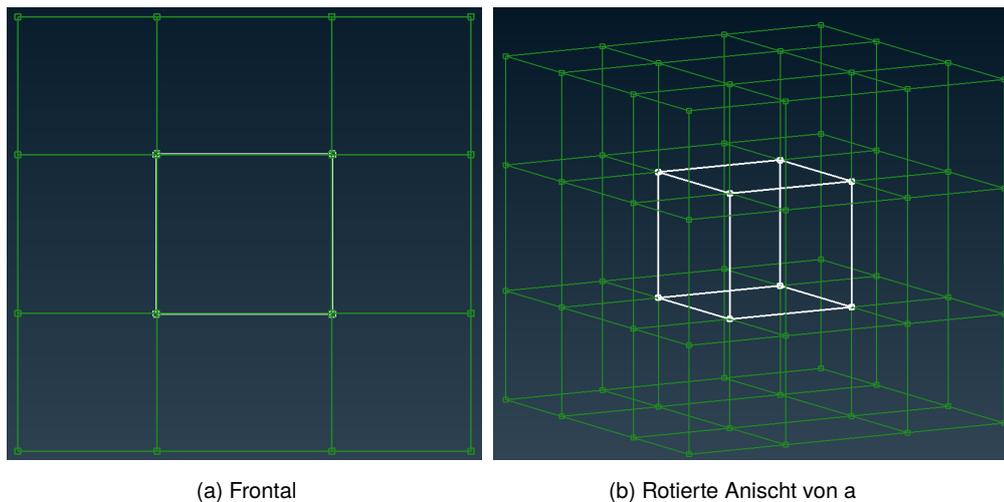


Abbildung 4.3: Hexablock Morphing

werden. Generell ist es möglich, diese auch mit in den Prozess einzubeziehen, um noch flexibler in der Gesamtgeometrie zu sein. Im konkreten Anwendungsfall ergibt sich daraus jedoch kein Mehrwert, da die nicht miteinbezogenen Elemente räumlich deutlich von den zu verschiebenden Elementen getrennt sind. Diese Trennung stellt sich unter anderem durch Bohrungen dar.

Im Falle des Radträgers sind alle Bohrungen, welche später der Verbindung des Radträgers und des Radlagers dienen, während des Morphingprozesses festzuhalten. Diese Geometriebereiche werden deshalb als Frozen-Nested-Elements definiert. Hierbei sind nur die Bohrungen betrachtet, welche zumindest zum Teil innerhalb einer der Morphingboxen liegen und somit vom Morphingprozess beeinflusst werden. Bohrungen, welche nicht der Fixierung des Radträgers dienen, werden als Rigid-Nested-Elements definiert, um keine unnötigen zusätzlichen Restriktionen des Netzes zu erstellen. Die Anbindungspunkte, deren Verschiebung vorgegeben wird, werden ebenfalls als Rigid-Nested-Elements definiert. Dies ist nötig, damit weder eine ungewollte Verformung der Lagerkerne und Hülsen sowie Bohrungen, noch eine Änderung des Durchmessers erfolgt. Da Anbindungspunkt R3 eine Bohrung darstellt, werden für R3 mehrere Nested-Elements definiert. Diese liegen auf verschiedenen Tiefen der Bohrung, spannen jeweils eine 'Spinne' im Querschnitt auf und sorgen somit dafür, dass diese über den Morphingprozess erhalten bleiben. Das am Grund der Bohrung liegende Nested-Element beinhaltet, zuzüglich zu den auf dem Umfang liegenden Knotenpunkten, ebenfalls die auf der Grundfläche liegenden Knotenpunkte. Die Verwendung von mehreren Nested-Elements ist dabei nötig, um die Tiefe der Bohrung zu variieren. Dies ist notwendig, da die variable Tiefe als Ausgleich der am Referenzpunkt vorgegebenen Verschiebung dient. Das Inkludieren aller Elemente der Bohrung in ein Nested-Element würde dazu führen, dass in der Bohrung selbst keine Verzerrung erfolgt und somit diese komplett in anderen Bereichen kompensiert werden muss. Dadurch wären geringere Verschiebungen von R3 in X-Richtung möglich.

In den Grundlagen wurde bereits erwähnt, dass möglichst smooth Geometrien erreicht werden, wenn die Morphingboxen der Geometrie möglichst genau folgen. Dieses Prinzip ist im Falle des Radträgers nur in gewissem Maße realisierbar. Dies liegt zum einen daran, dass nicht nur mehrere Anbindungsstellen unabhängig voneinander

verformt werden sollen, sondern diese auch nahe aneinander liegen. Zum anderen handelt es sich bei dem Radträger um eine Geometrie mit großen Gradienten und Krümmungen in drei Raumrichtungen, sodass eine manuelle Nachbildung aller Strukturen mittels Morphingboxen nicht zielführend ist. Dies ist durch das Fehlen einer Zielkontur begründet. Es sollen einzelne Anbindungspunkte verschoben und dabei die restliche Geometrie möglichst smooth mitverformt werden. Dennoch müssen auch im vorliegenden Fall einzelne Strukturen, wie das DSG, nachgebildet werden. Dies liegt jedoch eher an der räumlichen Nähe der zu bewegenden Anbindungspunkte. Die Anpassung der Form entsprechender Morphingboxen wird durch Einfügen zusätzlicher Morphingpunkte auf den einzelnen Kanten der Morphingboxen erreicht. Damit die Netzqualität ohne jegliche Korrektur möglichst gut ist, werden keine Morphingboxen mit spitzen Innenwinkeln und relativ zu ihrer Größe, extremen Seitenverhältnissen, angelegt.

Das grundlegende Prinzip, dass der zu verschiebende Anbindungspunkt mit allen dazugehörigen Elementen in einer Morphingbox inkludiert wird, kann in den Fällen R1, R2 und R5 umgesetzt werden.

Dass dies bei R3, R4 und dem DSG nicht direkt möglich ist, wird im folgenden erörtert. Im Falle von R3 handelt es sich um CBARs. Diese dürfen zwar eine relative Längenverzerrung erfahren, jedoch müssen alle Knotenpunkte der CBARs auf einer Geraden liegen. Da somit alle diese Elemente beim Morphingprozess betrachtet werden müssen, muss diese Gerade möglichst parallel verschoben werden, um die Anbindungsachse nicht zu verändern. Durch die Betrachtung der CBARs können einzelne Elemente kaum Distanz zueinander aufweisen, weshalb in diesem Fall keine vollständige Morphingbox als Puffer möglich ist.

Im Fall von R4 ist die Problematik eine ähnliche. Die räumliche Nähe zu R5 lässt kaum eine Pufferzone in X- und Y-Richtung zu. Das Hauptproblem besteht hierbei jedoch aufgrund von R5. In ANSA ist es nicht möglich lediglich eine zweier verbundenen Morphingboxen zu unterteilen. Dies hat zur Folge, dass die Begrenzung der Pufferbox von R5 gleichzeitig auch R4 etwa im Lagermittelpunkt teilt. Infolge dessen muss R4 in mehrere Morphingboxen gelegt werden. Gleichzeitig ist eine Vergrößerung des Puffergebietes von R5, um eine Zerteilung von R4 im Lagerkern zu vermeiden, ebenfalls nicht zielführend. Dies liegt daran, dass der blaue Bereich in Abbildung 4.4 die Verschiebungen von R4 und R5 ausgleicht. Da durch eine Vergrößerung des Puffergebietes von R5 eben diese Zone verkleinert wird, wäre hier eine geringe Netzqualität zu erwarten. Die gesamte Morphingstruktur umfasst 227 Morphingboxen, sowie 17 Nested-Elements von denen sechs als Frozen definiert sind. Die Referenzpunkte werden so gewählt, dass sie im Falle von R1, R2, R4 und R5 im Lagermittelpunkt liegen. Für das DSG und R3 werden die äußersten Punkte der Anbindungsachsen als Referenzpunkte definiert. Obwohl das betrachtete Bauteil ein zweigeteilter, linker und rechter Radträger ist, wird in dieser Arbeit der Prozess nur an dem rechten Radträger abgebildet. Die Entscheidung dazu ist darauf zurückzuführen, dass auch eine Verknüpfung der angelegten Morphingboxen des rechten Radträgers mit denen des linken Radträgers, wie sich in einer Versuchsreihe zeigte, nicht zu der exakt gleichen Geometrie führt. Des Weiteren kann aufgrund der Symmetrie des Bauteils die nicht gemorphte Seite durch eine Spiegelung der gemorphten Seite generiert werden. Da dies ein trivialer Schritt ist, genügt an dieser Stelle die einseitige Betrachtung des Bauteils

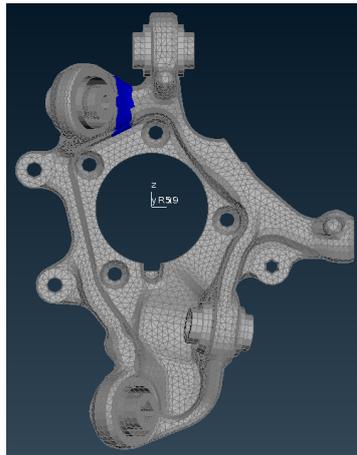


Abbildung 4.4: Puffergebiet R4 und R5

## 4.2 Gescriptetes Morphing

ANSA beinhaltet eine integrierte Scripting Umgebung. Diese kann mit einer ANSA/BETA spezifischen Programmiersprache angesprochen werden, es kann aber auch Python verwendet werden. Um lediglich eine Programmiersprache während des Gesamtprozesses zu verwenden, wird in dieser Arbeit Python als Programmiersprache genutzt. Generell ermöglicht ANSA auch ein manuelles Morphing, wobei der Benutzer ausgewählte Kontrollpunkte verschiebt und somit eine intuitive Verzerrung der Geometrie darstellt. Im Falle, dass lediglich einzelne Geometrievariationen erzeugt werden sollen, wäre dies zielführend. Da in dieser Arbeit jedoch eine Geometrieoptimierung stattfinden soll, wird ein automatisierter Morphingprozess implementiert. Dieses Unterkapitel beinhaltet dabei lediglich den reinen Morphingprozess. Dieser wird in eine gewünschte Translation, eine Rotationskorrektur, diverse Smoothingprozesse und eine Kontrolle der Netzqualität unterteilt.

Der Code ist dabei so aufgebaut, dass ein Anbindungspunkt nach dem anderen gemorphet wird. Die Reihenfolge ist dabei definiert als R3, R1, R2, R4, R5. Der Code ist so implementiert, dass durch einfache Zuweisungsänderungen die Morphingreihenfolge verändert werden kann. Die Reihenfolge ist dabei bezüglich des reinen Morphings von keiner Bedeutung. Im vorliegenden Fall werden jedoch, wie in Abschnitt 4.2.4 erörtert, Ausgleichsbewegungen durchgeführt. Hierzu sind Variablen nötig, welche erst während des Morphingprozesses der entsprechenden Anbindungspunkte definiert werden, sodass eine schlichte Veränderung der Morphingreihenfolge zu Fehlern führt. Der in diesem Kapitel erörterte Ablauf wird in Kapitel 5 in eine Optimierungsumgebung eingebunden, sodass das erstellte Morphing-Script von einem übergeordneten Script aufgerufen wird. Den gesamten schematischen Aufbau der Main-Funktion des Morphing-Scriptes stellt das Flussdiagramm in Abbildung 4.5 dar. Folgend werden alle weiteren Funktionen erörtert:

**Global** Vom Autor implementierte Funktion. Innerhalb dieser werden alle Variablen und Datenstrukturen, die während des gesamten Prozesses in einer Funktion benötigt werden, als global definiert. Eine Definition globaler Speicherelemente außerhalb der Main-Funktion ist hier nicht möglich, da bei Aufruf des

Morphing-Scriptes über ein übergeordnetes Script das Morphing-Script in der Main-Funktion gestartet wird. In dieser Funktion werden auch die zu implementierenden Verschiebungen der einzelnen Anbindungspunkte eingelesen. Dies geschieht über Regular-Expressions aus einer Input-Datei.

**ReadQualityCriteria** Mit der ANSA-spezifischen Funktion `ReadQualityCriteria` werden Qualitätskriterien aus einer dafür speziellen ANSA-Datei eingelesen und direkt im geöffneten ANSA-Projekt aktiviert.<sup>3</sup>

**GetReferenceCoords** Vom Autor implementierte Funktion. Gibt Koordinaten der aktuell zu morphenden Anbindungsstelle zurück.

**GetNormalPlane** Vom Autor implementierte Funktion. Für jeden der Anbindungspunkte R1, R2, R4 und R5 sind drei Referenzpunkte im Lagersitz definiert. Mittels dieser wird eine Ebene aufgespannt. Im Falle von R3 und dem DSG sind zwei Punkte definiert, welche eine Achse abbilden. Je nach Fall, wird ein Normalenvektor einer Ebene oder ein Vektor für eine Achse zurückgegeben.

**TranslMorphing** Vom Autor implementierte Funktion. Tatsächlicher Morphingprozess wird, wie in Abschnitt 4.2.1 näher erläutert, ausgeführt.

**RotationalCorrection** Vom Autor implementierte Funktion. Korrigiert Rotationen die durch das translatorische Morphing entstehen, sodass die Anbindungsachsen immer erhalten bleiben. Das Funktionsprinzip wird in Abschnitt 4.2.2 näher erläutert.

**ImproveMorphing** Prüft, ob die Abweichung zwischen Soll- und Ist-Wert der Anbindungskoordinaten nach dem translatorischen und rotatorischen Morphing geringer ist als eine Toleranz. Wenn das der Fall ist, erfolgt kein weiteres Morphing. Wenn die Abweichung größer als die Toleranz ist, erfolgt ein erneuter translatorischer Morphingschritt.

Die Funktionen `GetNormalPlane` und `GetReferenceCoords` werden dabei einmal vor und einmal nach dem translatorischen Morphingschritt aufgerufen. Die Rückgabewerte werden hierbei in unterschiedliche Verzeichnisse gespeichert. Dies ist nötig, um die Ist-Abweichung zu berechnen und anhand dieser, die Rotation sowie die Morphingverbesserung durchzuführen.

---

<sup>3</sup>BETA CAE Systems 2019.

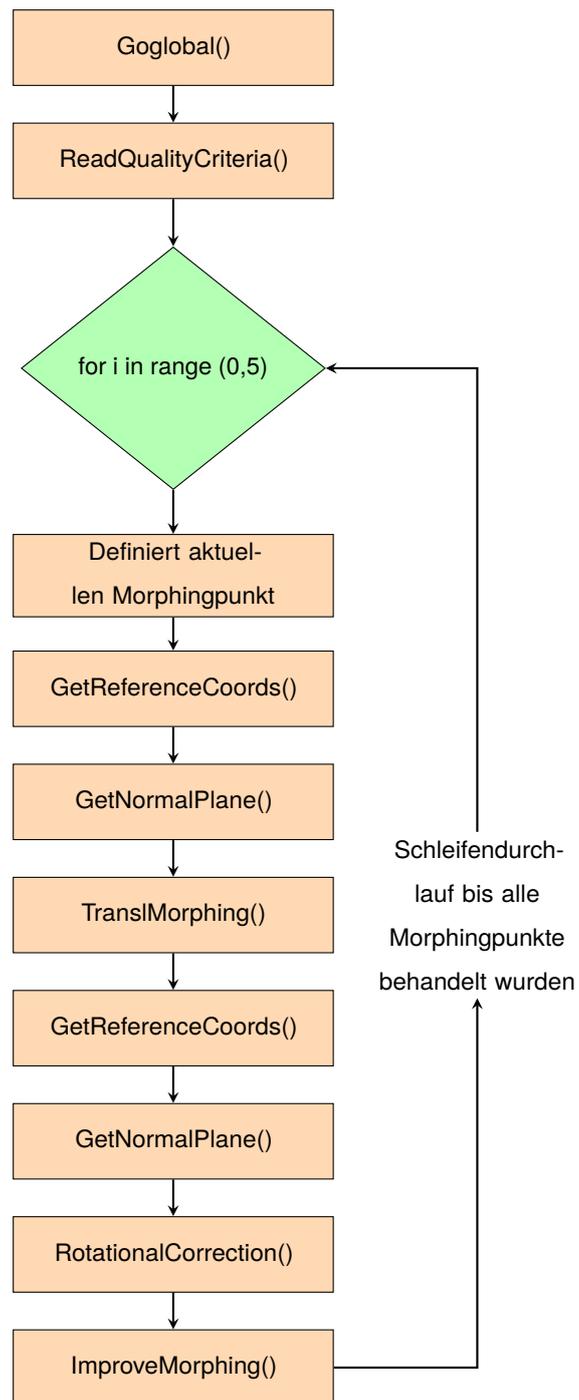


Abbildung 4.5: Mainfunktion Morphing

### 4.2.1 Translation

Zu Beginn des Morphingprozesses werden alle sichtbaren Elemente in die Morphingboxen geladen. Das ist essenziell, um trotz Verschiebung der Morphingboxen und –ebenen sicherzustellen, dass immer nur die tatsächlich von einer Box eingeschlossenen Elemente in diese Box geladen sind. Dadurch wird sicher gestellt, dass jedes Element bezüglich der richtig zugewiesenen Morphingbox verschoben wird. Um den Morphingprozess übersichtlicher und damit anschaulicher zu gestalten, wird jede Soll-Verschiebung in ihre Anteile gemäß den Koordinatenachsen aufgeteilt. Der Morphingprozess wird somit schrittweise in X-, Y- und Z-Richtung ausgeführt.

Die Soll-Verschiebung wird außerdem in eine Verschiebung anhand der Morphingbox und einer Morphingebene aufgeteilt. Dieses Prinzip führt allgemein zu smootheren Übergängen zwischen nicht gemorphten und gemorphten Bereichen. Die prozentuale Aufteilung ist dabei für jeden Anbindungspunkt unterschiedlich und wurde heuristisch ermittelt sowie getestet. Das Morphing durch Verschiebung von nur einer Ebene ist lediglich möglich, da die einzelnen Lager als Nested-Elements definiert sind. Abbildung 4.6 stellt die Morphingstruktur um R1 dar. Im Sinne der Anschaulichkeit sind dabei alle, nicht nahtlos an die R1 beinhaltende Morphingbox, angrenzenden Morphingboxen ausgeblendet. Die Bilder zeigen, dass um die eigentliche Morphingstruktur noch weitere Morphingboxen angelegt sind. Dies ist in Abbildung 4.6a am unteren Bildrand gut ersichtlich. Diese beinhaltet initial keinerlei Elemente. Sie dient jedoch als Puffer, sodass falls durch eine ungewollte Rotation von einzelnen Elementen die Ebene der ursprünglichen Morphingbox überschritten wird, eben diese Elemente einer neuen Morphingbox zugeordnet werden und somit bei weiteren Verformungen nicht unmittelbar zu negativen Volumen führen.

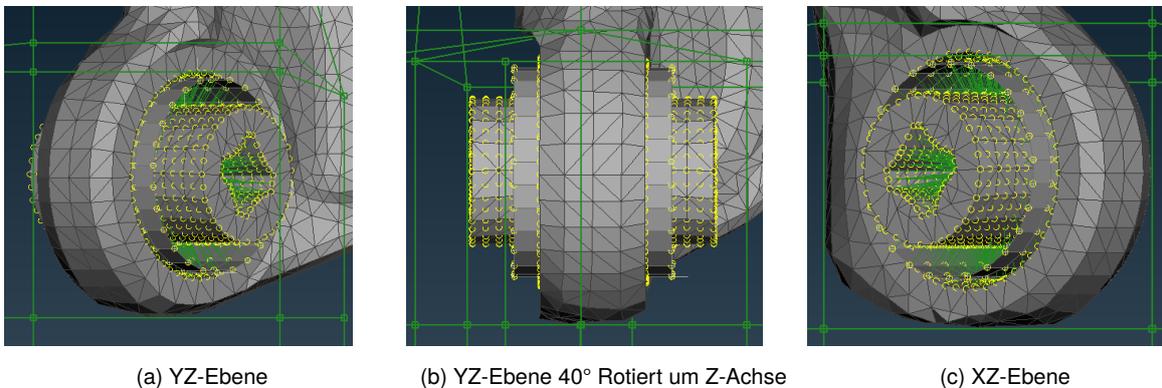


Abbildung 4.6: R1 Morphingstruktur dargestellt an der original Geometrie

Durch eine Verschiebung der unter R1 liegenden Ebene in negativer Z-Richtung wird das Material, innerhalb der das Lager umschließenden Morphingbox, verformt. Somit würde dieses zu einem Oval werden. Da es aber als Nested-Element definiert ist, wird lediglich der äußere Teil verzerrt und der innere Teil verschoben. Durch die Verschiebung der gesamten Morphingbox werden die anliegenden Morphingboxen verzerrt. Somit werden möglichst viele Elemente in die Verzerrung miteinbezogen, sodass bestmögliche Bedingungen für hohe Netzqualität nach dem Morphing bestehen. Des Weiteren führt eine reine Ebenenverschiebung am Außenrand der Geometrie auch von Natur aus zu smootheren Geometrien. Dies ist dadurch bedingt, dass somit lediglich die Elemente einer

Morphingbox verzerrt werden und dies linear über eine Morphingbox passiert. Es entsteht dadurch keine harte Kante, ab der kein Morphing erfolgt. Die Verschiebung wird somit mittels einer mathematisch stetigen Funktion abgebildet und beinhaltet keinerlei Sprünge.

Im Gegensatz zu einer konkreten Gestaltänderung der Außenfläche einer Bauteilgeometrie, wie es in Designanwendungen oft der Fall ist, liegen die Referenzpunkte bei dem betrachteten Bauteil nicht auf der Außenfläche. Die Referenzpunkte liegen im Falle von R1, R2, R4 und R5 im Bauteil. Im Falle des DSGs und R3 liegen die Referenzpunkte außerhalb des Bauteils. Wie bereits beschrieben, ist der Verlauf der Verschiebung innerhalb einer Morphingbox eine lineare Funktion, sodass mit der Formel

$$X_{Ebene} = \frac{1}{\frac{X_{ij} - X_{E_1j}}{X_{E_2j} - X_{E_1j}}} X_{Soll} \quad (4.1)$$

die benötigte Verschiebung einer Ebene berechnet werden kann, welche zur gewünschten Soll-Verschiebung am Referenzpunkt führt. Dabei steht  $X_{ij}$  für die Koordinate  $j$  des Referenzpunktes  $i$  und der Index  $E$  kennzeichnet die Ebenen der Morphingbox, welche durch den Normalenvektor  $j$  beschrieben werden. Dabei gilt  $X_{E_2j} > X_{E_1j}$ . Diese Formel ist nötig, sobald parallele Ebenen der Morphingbox nicht um den gleichen Wert verschoben werden, um die tatsächliche Verschiebung am Referenzpunkt zu berechnen. Wird die gesamte Morphingbox um einen vorgegebenen Wert verschoben, verschiebt sich auch der Referenzpunkt  $i$  um eben diesen Wert. Im Falle, dass nur eine Ebene der Morphingbox verschoben wird, muss der betrachtete Referenzpunkt gleichzeitig der Bezugspunkt des definierten Nested-Elements sein.

Der Morphingprozess kann in jede der drei Raumrichtungen unabhängig voneinander durchgeführt werden, da die verschobenen Ebenen parallel zu den Ebenen des Koordinatensystems sind, sodass die Verschiebungen gesondert in eine Raumrichtung erfolgen können. Die verwendeten Ebenen variieren dabei bezüglich der Anbindungspunkte und der Raumrichtungen, nicht jedoch bezüglich der Orientierung der Verschiebung. Lediglich bei einer Verschiebung von R3 in Y-Richtung wird die verschobene Morphingebene auch diesbezüglich variiert. Der Grund dafür wird im weiteren Verlauf dieses Kapitels erörtert.

Der gesamte translatorische Morphingprozess eines Referenzpunktes durchläuft dabei die in Abbildung 4.7 dargestellten sechs Schritte. Der Anteil der Verschiebung mittels Box-Morphing wird dabei, nach eben diesen Schritten, von den jeweils vorgegebenen Verschiebungen abgezogen und an das Ebenenmorphing übergeben. Der Morphingprozess am Punkt R3 in X-Richtung erfordert nähere Betrachtung. Zwar wird auch in diesem Fall lediglich ein Punkt als Referenzpunkt verwendet, dennoch ist beim Morphingprozess die komplette Achse, welche im Bauteil eine Bohrung darstellt, zu betrachten. Bei der Gestaltung der Morphingstruktur wurde die Achse nicht in eine, sondern in zwei Morphingboxen eingebettet. Dies führt dazu, dass die einzelnen Boxen und Ebenen in einem besonderen Verhältnis verschoben werden müssen, da sich sonst keine Gerade bei Verbindung der einzelnen Achspunkte ergibt. Die folgende Graphik Abbildung 4.8 stellt eben dieses Problem dar. Als Referenzpunkt wird dazu der Punkt, welcher am nächsten an der oberen rechten Ecke des Bildes liegt, gewählt. Dieser wird in beiden Fällen um gewünschte 20 mm exakt verschoben.

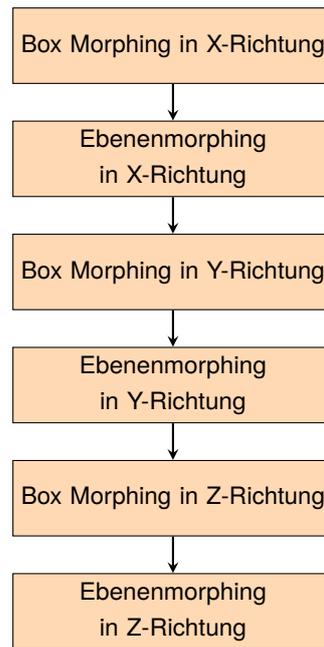
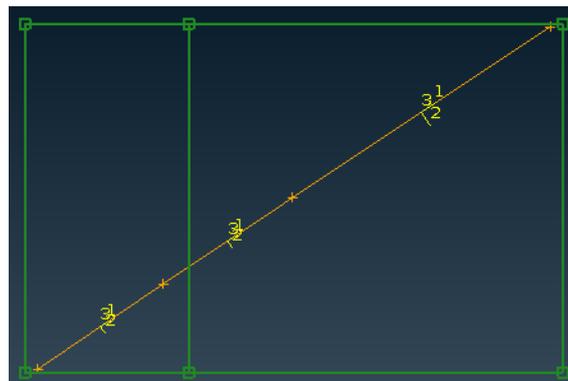
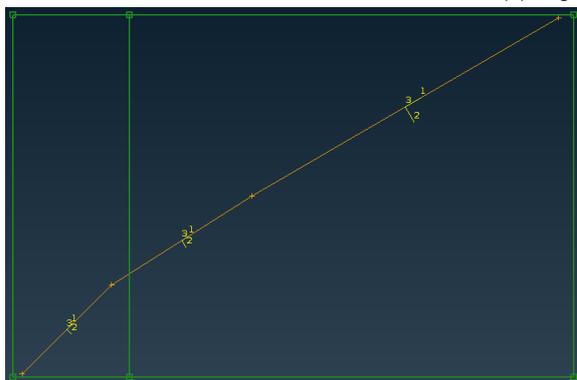


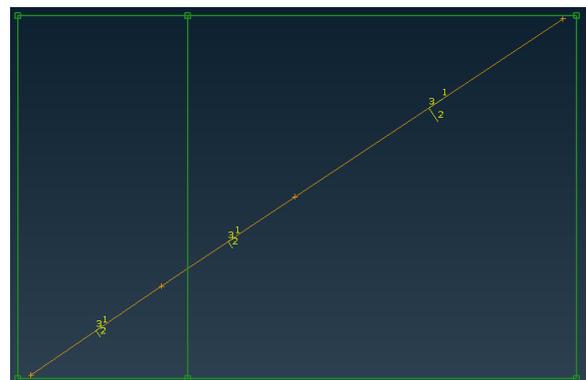
Abbildung 4.7: Ablauf Morphing



(a) Original Zustand



(b) Reine Ebenenverschiebung



(c) Kombination Ebenen-, Boxverschiebung

Abbildung 4.8: Vergleich reine Ebenenverschiebung und Kombination aus Ebenen-, Boxverschiebung bei Elementen in mehreren Boxen

Die Aufteilung der Verschiebungen wird dabei anhand folgendem Prinzip berechnet.

Abbildung 4.9 stellt das Problem schematisch dar. Ebenen der Morphingboxen sind hierbei mit  $E$  gekennzeichnet und mit Buchstaben durchnummeriert. Die drei Punkte sind gekennzeichnet mit  $P$ . Die Entfernungen der Morphingebenen, bezogen auf Ebene  $E_a$ , werden durch die Platzhalter  $X$  beschrieben.  $E_a$  bleibt während des Morphings fixiert und die an  $P_3$  vorgegebene Verschiebung wird mittels Verschiebung von  $E_b$  und  $E_c$  so dargestellt, dass gemäß Abbildung 4.8c eine Gerade durch alle Punkte entsteht. Da die Verzerrung der Elemente in einer Morphingbox linear ist, gilt

$$\Delta P_3 = \Delta X_1 \frac{P_3 - E_c}{E_b - E_c} + \Delta X_2 \frac{P_3 - E_b}{E_c - E_b} \quad (4.2)$$

Um eine lineare Verschiebung boxübergreifend zu realisieren, wird

$$\Delta X_2 = \Delta X_1 \cdot \frac{X_2}{X_1} \quad (4.3)$$

angesetzt. Einsetzen von Gleichung 4.3 in Gleichung 4.2 ergibt

$$\Delta P_3 = \Delta X_1 \frac{P_3 - E_c}{E_b - E_c} + \Delta X_1 \frac{X_2}{X_1} \frac{P_3 - E_b}{E_c - E_b} \quad (4.4)$$

$$\Leftrightarrow \Delta X_1 = \frac{\Delta P_3}{\frac{P_3 - E_c}{E_b - E_c} + \frac{X_2}{X_1} \frac{P_3 - E_b}{E_c - E_b}} \quad (4.5)$$

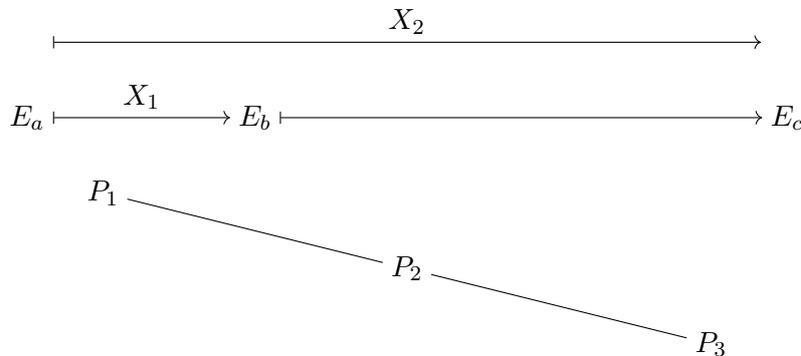


Abbildung 4.9: Skizze zur Anteilberechnung von Verschiebungen für eine Gerade über mehrere Morphingboxen hinweg, um die Gerade zu erhalten

Anhand Gleichung 4.5 und Gleichung 4.3 kann somit, bei vorgegebener Verschiebung  $\Delta P_3$ , die an  $E_b$  und  $E_c$  nötige Verschiebung berechnet werden.

Wie bereits erwähnt, wird im Falle einer Verschiebung von R3 in Y-Richtung, bei positiver und negativer Verschiebung, jeweils eine andere Ebene für deren Realisierung herangezogen. Abbildung 4.10 zeigt R3 als Ansicht von oben. Der größere Teil ( $2/3$ ) der Verschiebung in Y-Richtung wird dabei anhand der beiden, die Achse von R3 beinhaltenden, Morphingboxen realisiert.  $1/3$  wird durch Verschiebung der Ebenen der Morphingboxen abgebildet, sodass die

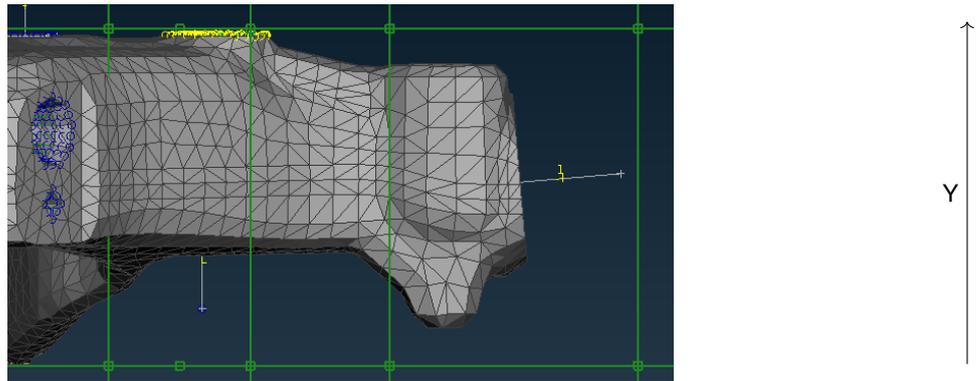


Abbildung 4.10: R3 Y-Verschiebung

Morphingboxen selbst veränderte Volumen aufspannen. Durch diese Veränderung der Morphingboxen wird auch das Volumen des Bauteils angepasst. Wenn sowohl für eine negative Verschiebung als auch eine positive Verschiebung in Y-Richtung dieselben Ebenen (Ebenen da es sich um zwei Morphingboxen handelt) verwendet werden, führt dies in einem Fall zu einer Verbreiterung und im anderen Fall zu einer Verschmälerung der Geometrie. Da dies maßgeblich die Steifigkeit der Geometrie beeinflusst, wäre in diesem Fall bei einem Vergleich der Geometrien nicht mehr beurteilbar, ob die Veränderung des Bewertungskriteriums aus dem Versatz des Anbindungspunktes oder aus der Querschnittsänderung resultiert. Um dies zu vermeiden, werden die Ebenen für die reine Ebenenverschiebung von R3 jeweils so gewählt, dass der Querschnitt für beide Verschiebungsrichtungen größer wird.

#### 4.2.2 Rotation

Im Allgemeinen ist eine Rotation der Elemente um eine festgelegte Achse möglich, aber nur manchmal gewollt. In der, dieser Arbeit zugrundeliegenden Anwendung, soll jedoch eine Rotation der Anbindungspunkte verhindert werden. Die Achsen der Lagerkerne, der Spurstange und des DSGs sollen über den Morphingprozess unverändert bleiben. Dies führt zu einer Reduktion der Freiheitsgrade von sechs zu drei an jeder Anbindungsstelle und somit zu einer Reduktion von 30 zu 15 im gesamten System.

Durch die Verzerrung des Lagerkerns oder der Bohrung kommt es zu einer ungewollten Rotation deren Achse, solange diese nicht parallel zur verschobenen Ebene liegt. Dies wird deutlich durch die Betrachtung eines einfachen Beispiels. Abbildung 4.11 zeigt ein einfaches Balkenelement. Bevor das Element gemorphet wird, ist der Anstellungswinkel  $45^\circ$ . Dies ist nach einer Verzerrung in X-Richtung um 15 mm nicht mehr der Fall. Es kann also eine Rotation der Elementenachse beobachtet werden.

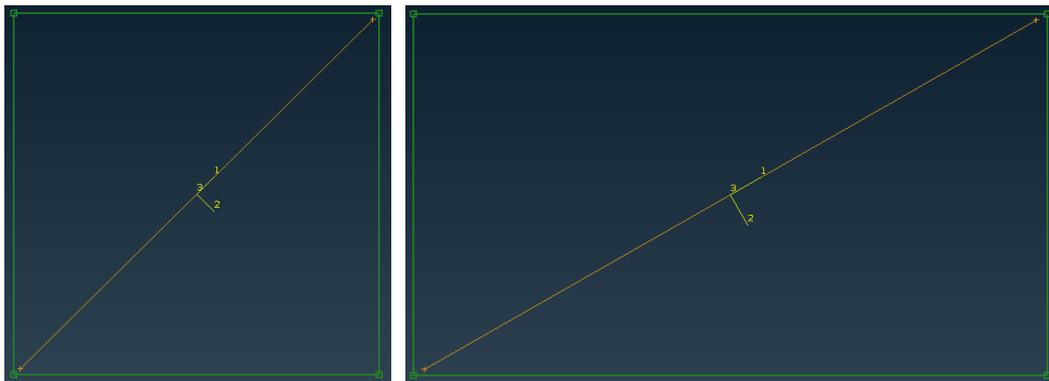


Abbildung 4.11: Rotation durch Ebenenverschiebung; Links vor Morphing, Rechts nach Morphing

Aufgrund der soeben erörterten ungewollten Rotation, muss diese nachträglich rückgängig gemacht werden. Zu diesem Zweck werden zunächst Stützpunkte definiert. Im Falle der Lagerkerne werden hierzu jeweils drei Knotenpunkte im Lagersitz des Radträgers verwendet. Diese liegen jeweils auf halber Dicke des Lagersitzes und spannen ein möglichst gleichseitiges Dreieck auf. Im Falle des DSGs und der Spurstange, können die ersten und letzten Knotenpunkte der Balkenelemente verwendet werden. Da diese Elemente eine Gerade darstellen, führt die Verbindung des ersten und letzten Knotens automatisch zur gewünschten Achse. Im Falle der Lagerkerne ergibt sich die Orientierung der Achse als Normalenvektor der aufgespannten Ebene. Die Wahl von Ausgangs-, Endpunkt des Vektors spielt dabei, ebenso wie die Ausrichtung des Normalenvektors, keine Rolle. Mit der in Abbildung 4.5 aufgezeigten Funktion `GetNormalPlane` werden diese Referenzebenen bzw. Referenzachsen aufgespannt und der Normalenvektor der Ebene, bzw. der die Achse beschreibende Vektor, zurückgegeben. Diese werden in ein Verzeichnis für den jeweiligen Anbindungspunkt abgespeichert. Dies wird sowohl vor, als auch nach dem translatorischen Morphing durchgeführt, sodass die Achsen des Ausgangszustandes sowie des gemorphten Zustandes für die rotatorische Korrektur erhältlich sind. Der ungewollte Rotationswinkel wird gemäß

$$\cos \gamma = \frac{\mathbf{u} \cdot \mathbf{v}}{|\mathbf{u}| |\mathbf{v}|} \quad (4.6)$$

nach Furlan<sup>4</sup> berechnet. Um die zugehörige Rotationsachse zu berechnen, wird mittels der Vektoren  $\mathbf{u}$  und  $\mathbf{v}$  eine neue Ebene aufgespannt.  $\mathbf{u}$  beschreibt dabei den Vektor vor dem Morphing und  $\mathbf{v}$  den nach dem Morphing. Der Normalenvektor, der neu aufgespannten Ebene, stellt die nötige Rotationsachse dar. Der Ursprung der Rotationsachse wird dabei in den gemorphten Anbindungspunkt gelegt, sodass dieser bei der Rotation unverändert bleibt. Der mittels Gleichung 4.6 berechnete Winkel wird in jedem Fall positiv um die Rotationsachse gedreht. Dies ist gültig, da die Rotationsachse mittels des Kreuzproduktes der anderen beiden Vektoren aufgestellt wird. Dadurch wird immer ein Rechtssystem gebildet. Der Vektor nach dem Morphing wird dazu als erster Vektor (Rechtssystem X-Achse) und der Vektor vor dem Morphing als zweiter Vektor (Rechtssystem Y-Achse) verwendet.

Durch Definition der Lagerkerne als ein zusammenhängendes Nested-Element wäre es ebenso möglich, deren

<sup>4</sup>Furlan 2012, S.24.

Rotation über die Freiheitsgrade des Elements von vornherein zu limitieren. Dieses Prinzip kann jedoch nicht auf R3 angewendet werden, da es sich hierbei um mehrere Nested-Elements handelt und deren einzelne Rotationsbeschränkung somit keine Beschränkung der gesamten Elementenachse nach sich zieht. Des Weiteren bietet das hier angewendete Prinzip die Möglichkeit, die Rotation und Translation eines Elementes zu entkoppeln und somit eine gewünschte Rotation unabhängig von einer Translation durchzuführen. Eine Untersuchung der rotatorischen Freiheitsgrade ist also ebenso möglich, wie die Untersuchung der translatorischen Freiheitsgrade. Um einen gewünschten Rotationswinkel abzubilden, muss der hier dargestellte Prozess rückwärts durchgeführt werden.

An dieser Stelle sei angemerkt, dass eine rotatorische Korrektur lediglich nötig ist, wenn die Verschiebung des Referenzpunktes durch Verschiebung einer Morphingebene realisiert wird. Im Falle, dass diese durch Verschiebung der gesamten Morphingbox abgebildet wird, ist das Auftreten einer ungewollten Rotation nicht möglich. Die tatsächliche Rotation des Elements erfolgt durch eine Rotation der umschließenden Morphingbox.

### 4.2.3 Morphing Verbesserung

Die ImproveMorphing-Funktion in Abbildung 4.5 verbessert die Morphingqualität. Diese ist die Präzision, mit welcher die Anbindungspunkte an ihren Soll-Zustand verschoben werden. Das Vorgehen erfolgt dabei iterativ. In jedem Schritt wird zunächst der Soll-Zustand mit dem Ist-Zustand des zu morphenden Punktes verglichen. Die Abweichung wird als neuer Morphingvektor festgelegt und auf die das Element umgebende Morphingbox angewendet. Es werden somit alle Morphingpunkte um den berechneten Morphingvektor verschoben. Dieser Prozess wird maximal fünf mal ausgeführt.

Während Versuchsdurchläufen zeigte sich, dass eine häufige, wenn auch kleine Verschiebung mit jeweils erneutem Laden der Elemente in die Morphingboxen, zu einer Verschlechterung der Elementequalität führt, sodass eine Abstandsüberprüfung eingeführt wurde. Diese prüft vor jeder Iteration, ob zwischen Soll- und Ist-Zustand ein Mindestabstand von größer als 25 µm besteht. Ist dies nicht der Fall, wird keine Iteration durchgeführt. Das beschleunigt auch den Gesamtprozess, da das Einladen der Elemente in die Morphingbox rechenleistungsintensiv ist. Das liegt primär an der verwendeten Konstellation des Ladebefehls.

ANSA bietet hierzu den „MorphLoad“-Befehl an. In diesem wird eine Liste an zu betrachteten Morphingboxen und eine Liste der einzuladenden Elemente abgefragt. Statt der Elementeliste kann auch der String „db\_or\_visib“ mit „DB“ oder „Visib“ gleichgesetzt werden. Ersteres verwendet dabei alle in der Datenbank der geöffneten Datei verfügbaren Elemente als Elementeliste. Zweiteres lediglich die in der Benutzeroberfläche sichtbaren Elemente.<sup>5</sup> In dieser Arbeit werden zur Ausführung dieses Befehls immer alle Morphingboxen und alle sichtbaren Elemente verwendet. Das gewährleistet, dass immer alle für den folgenden Morphingprozess nötigen Elemente in den korrekten Boxen geladen sind. Es fordert jedoch mehr Rechnerleistung als das Laden von nur einer Teilmenge der Morphingboxen. Eine Begrenzung der Iterationen ist somit auch aus Sicht der Rechnerleistung sinnvoll. Diese spiegelt sich direkt in der Zeit wieder, die benötigt wird, um eine gemorphte Geometrie zu erzeugen.

---

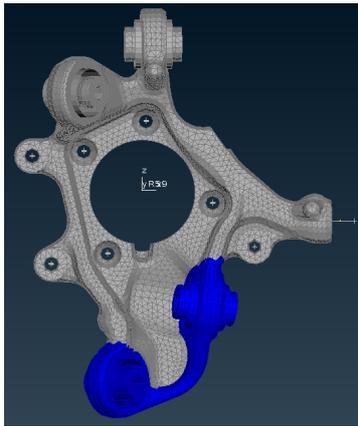
<sup>5</sup>BETA CAE Systems 2019.

#### 4.2.4 Smoothing

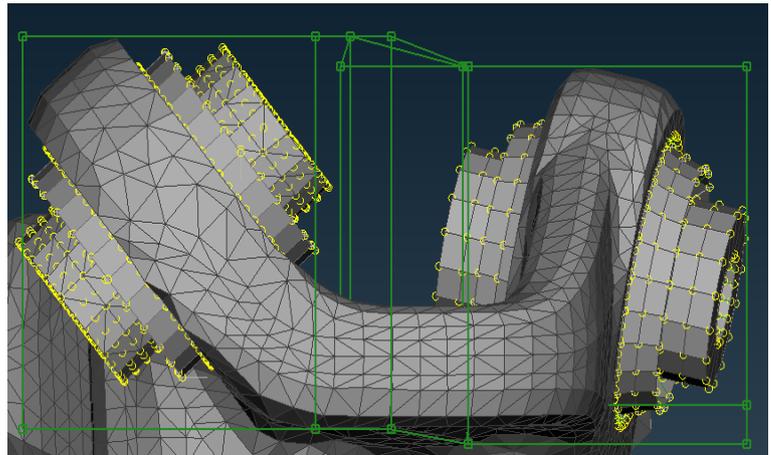
Wie bereits erwähnt, beschreibt der Begriff smooth in dieser Arbeit eine Geometrie, welche stetige Übergänge und somit keine sprunghaften Gradientenänderungen der Kontur aufweist, sowie als weich empfunden wird.

In Abschnitt 3.1.1 wurde erörtert, dass smooth Geometrien erreicht werden, indem Kontrollpunkte der Morphingboxen, welche in einen unveränderten Bereich übergehen, nicht verschoben werden. Wenn dies jedoch nicht möglich ist, da alle Elemente der Morphingbox eine gewisse Verschiebung erfahren sollen, muss ein Ausgleich in anliegenden Morphingboxen erfolgen, um eine smooth Geometrie zu erhalten. Die folgende Abbildung verdeutlicht dies für den Fall einer Verschiebung in Y-Richtung der Anbindungspunkte R1 ( $\Delta y = 10mm$ ) und R2 ( $\Delta y = -10mm$ ). Abbildung 4.12 stellt den Ausgangszustand mit der Morphingstruktur dar. Dabei sind alle angrenzenden Morphingboxen, die keinen unmittelbaren Einfluss auf diesen Teil der Geometrie haben oder als äußere Pufferboxen dienen, ausgeblendet. Abbildung 4.13a zeigt R1 und R2 als Ansicht von unten in XY-Ebene im Ausgangszustand, im gemorphten Zustand ohne Smoothing und im gemorphten Zustand mit Smoothing. In Abbildung 4.13b wird dabei deutlich, wie sich starke lokale Änderungen der Gradienten darstellen, wenn zwischen den Morphingboxen mit vorgegebener Verschiebung ein, bezogen auf den Versatz, relativ geringer Abstand besteht. Ein Übergang mittels lediglich geringer Gradientenänderung zu den angrenzenden Morphingboxen ist in diesen Fällen nicht möglich.

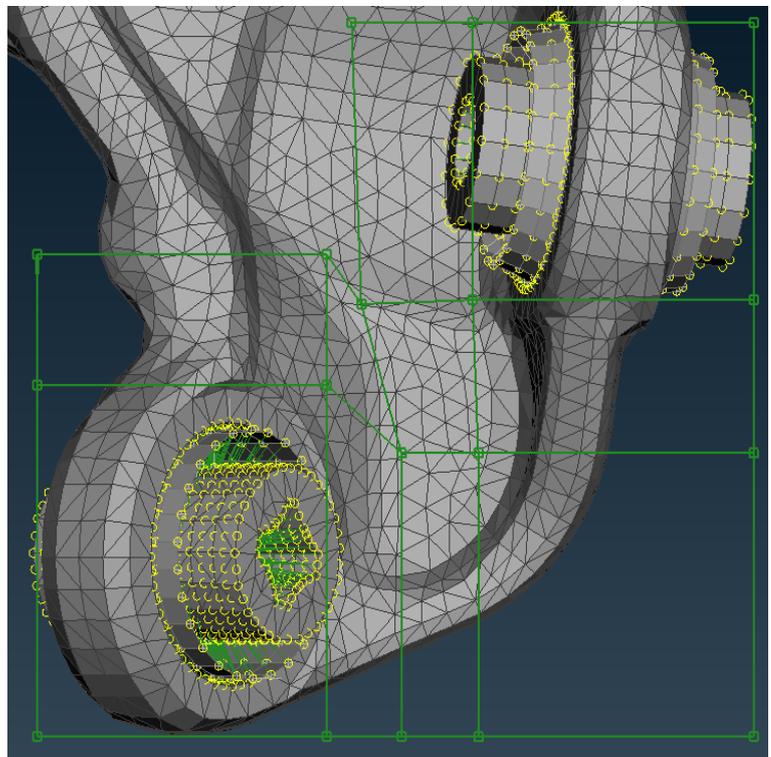
Im vorliegenden Beispiel ist dies zweistufig implementiert. Zunächst wird der Anbindungspunkt R2 gemorphet. Die in Z-Richtung, unter R2 liegende Morphingbox führt diese Bewegung ebenfalls vollständig aus, um ein smoothes Ausklingen der vorgegebenen Verschiebung zu ermöglichen. Dies ist die erste Stufe. In der zweiten Stufe wird der Mittelwert der vorgegebenen Verschiebungen an den Morphingboxpunkten von R1 und R2 berechnet und der Verbindungsbox der beiden Bereiche vorgegeben. Hierdurch entsteht der kontinuierliche Übergang am Steg zwischen R1 und R2, wie er in Abbildung 4.13c dargestellt ist. Dabei ist zu beachten, dass die gesamte Verschiebung am Anbindungspunkt R1, in Y-Richtung mittels Verschiebung von lediglich einer Morphingebene realisiert wird. Daraus resultiert, dass die Kontrollpunkte um mehr als 10 mm verschoben werden, weshalb trotz einer mittleren Verschiebung von R1 und R2 von null, dennoch eine Verschiebung an der Verbindungsbox angesetzt wird.



(a) Übersicht Smoothing  
zwischen R1 und R2

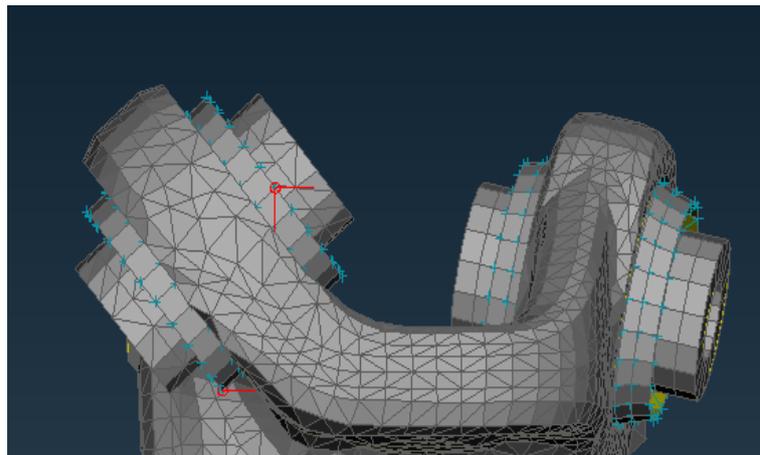


(b) Morphingboxen R1, R2 XY-Ebene

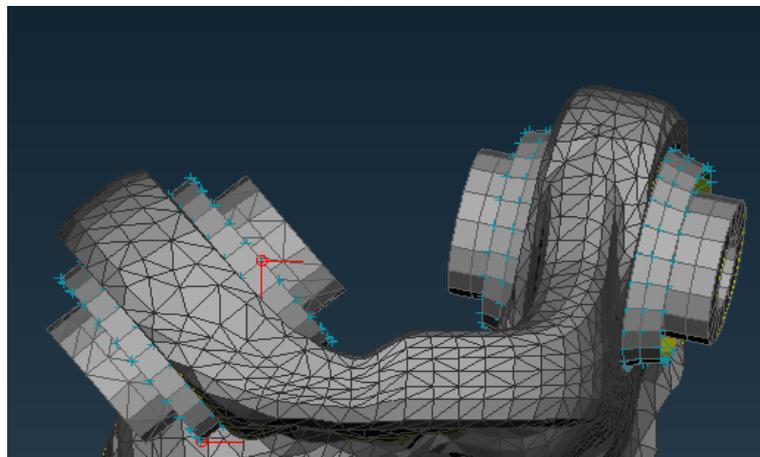


(c) Morphingboxen R1, R2 XZ-Ebene

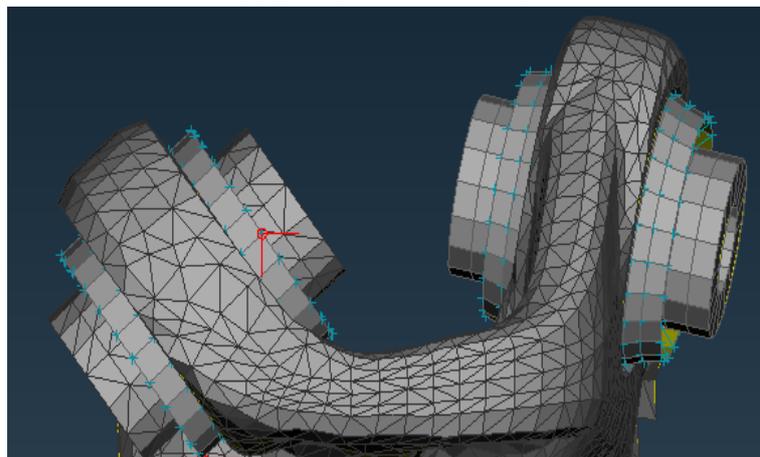
Abbildung 4.12: Ausgangszustand Morphing R1, R2



(a) Ausgangszustand Morphing



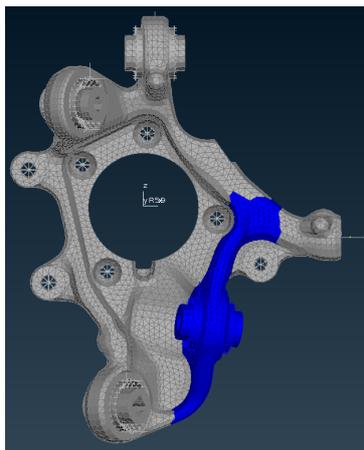
(b) gemorpt ohne Smoothing



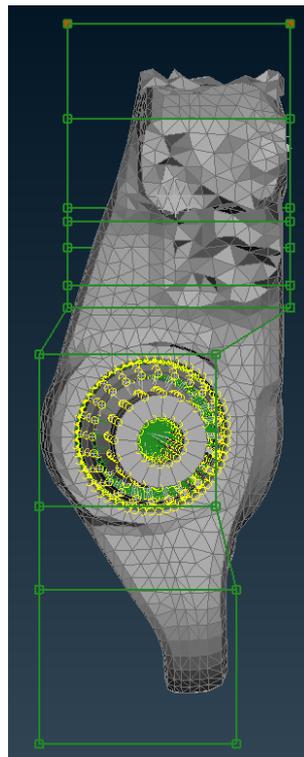
(c) gemorpt mit Smoothing

Abbildung 4.13: Morphing von R1 (links) , R2 (rechts) um  $y = 10mm$  und  $y = -10mm$  verglichen mit und ohne Smoothing als Ansicht von unten in XY-Ebene

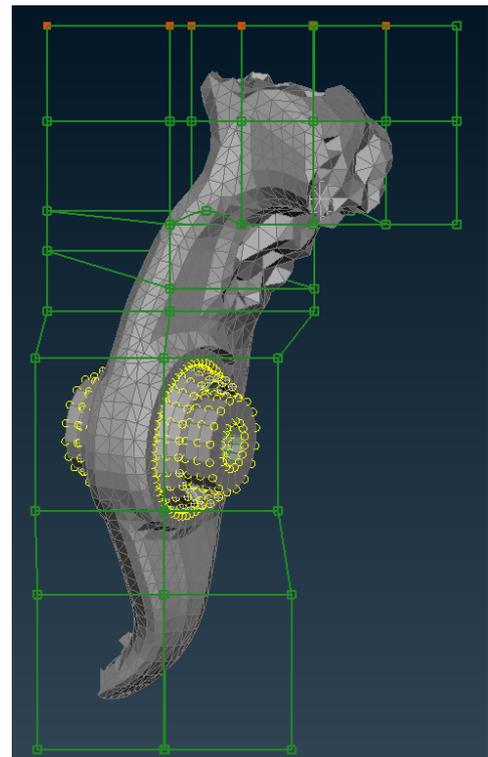
Das mehrstufige Ausblenden der Verschiebung wird in Abbildung 4.16 deutlich. Hierbei wurde zur Erzeugung der Bilder R2 um 10 mm in Y-Richtung verschoben. Die zur gesamtheitlichen Verschiebung verwendeten Morphingpunkte sind in Abbildung 4.14 dargestellt, werden aber unterschiedlich stark verschoben. Punkte eins bis acht werden um die vollen 10 mm verschoben, um die Verschiebung von R2 abzubilden. Die erste Smoothingstufe mit Punkt neun bis 12 wird ebenfalls um die vollen 10 mm verschoben. Die übrigen Punkte 13, 14 werden lediglich um  $\frac{2}{3}$  der 10 mm verschoben. Die Nummerierung der Kontrollpunkte ist in Abbildung 4.15 dargestellt. Die Faktoren sind hierbei auf Grundlage von Gleichung 3.12 heuristisch ermittelt. Generell werden die Faktoren immer kleiner, umso weiter die zu verschiebenden Kontrollpunkte vom eigentlichen Morphingobjekt entfernt sind. Wie groß der Faktor in den jeweiligen Smoothingstufen ist, hängt somit von der Ausgangsgeometrie, der Form der umschließenden Morphingboxen sowie der Distanz zum betrachteten Anbindungspunkt ab. Die Abhängigkeit von der Ausgangsgeometrie wird ersichtlich bei Betrachtung von Abbildung 4.14b. Die zu verschiebende Geometrie ist ein Lagersitz, der auch auf der Außenseite abgerundet ist. Da die Morphingboxen das Bauteil nicht exakt am Übergang der Rundung in eine Gerade unterteilt, wird ein Teil dieser Gerade mit verschoben. Durch eben diesen Teil wird der Gradient definiert, welcher in der nächsten Morphingbox nötig ist, um eine stetige Steigung und somit einen kontinuierlichen Übergang der Morphingboxen zu ermöglichen. Auf diese Art und Weise werden auch an den anderen Anbindungspunkten Smoothingebenen mit zugehörigen Faktoren ermittelt und überprüft.



(a) Übersicht R2



(b) R2 Morphingboxen YZ-Ebene



(c) Bild b Rotiert um Z-Achse

Abbildung 4.14: Morphingstruktur und Übersichtsbild R2

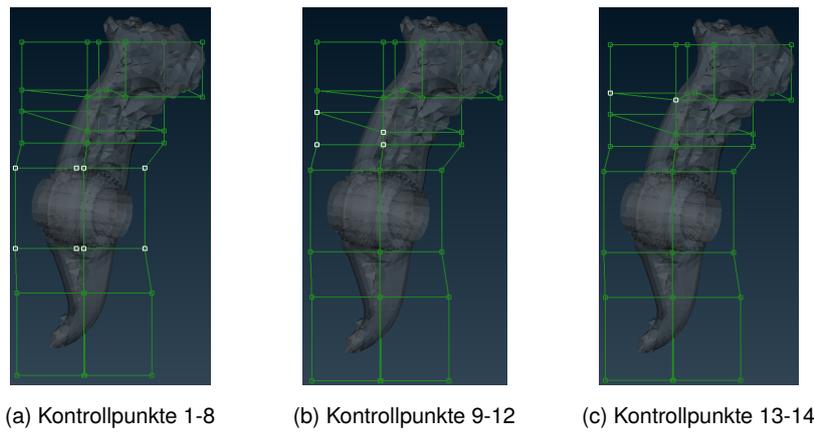
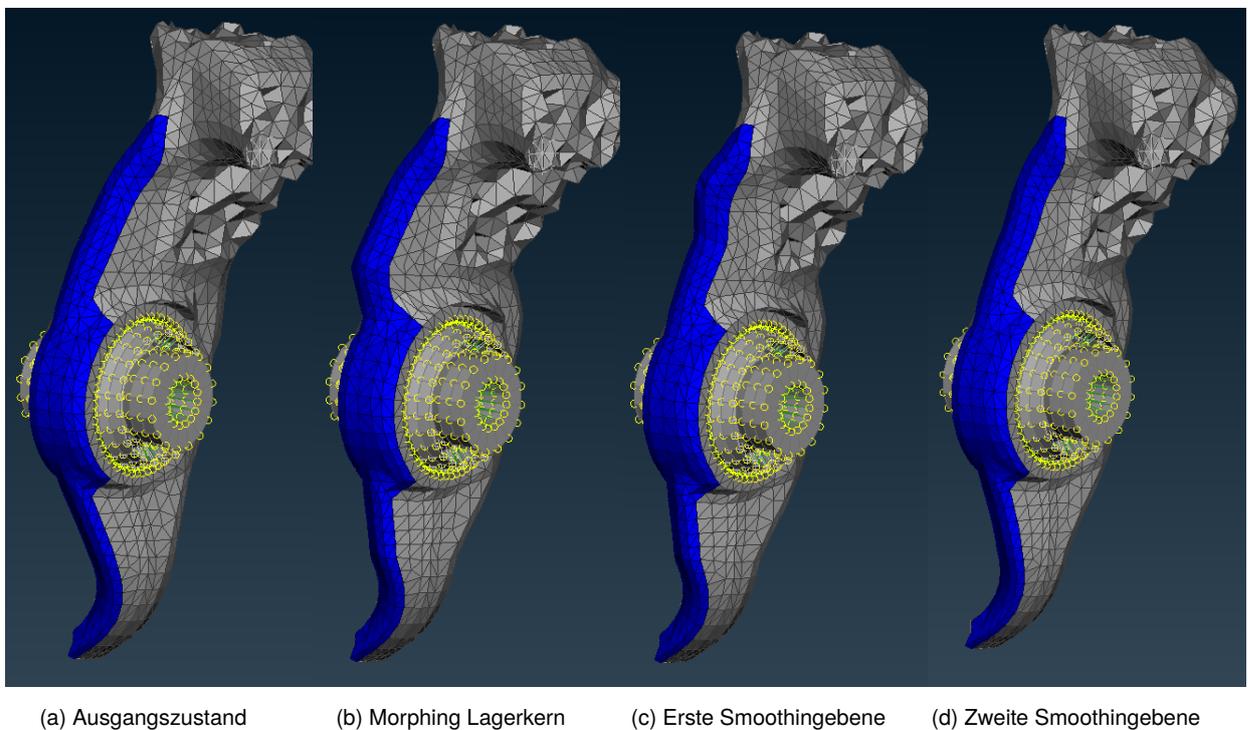


Abbildung 4.15: Veranschaulichung Nummerierung der Kontrollpunkte R2

Abbildung 4.16: Morphing R2 um  $y=10\text{mm}$ 

#### 4.2.5 Netzgüte

Die Netzgüte bestimmt maßgeblich die Qualität und Brauchbarkeit von Simulationsergebnissen. Deshalb ist es notwendig, Qualitätsmerkmale zu definieren, welche von allen betrachteten Geometrievariationen erfüllt werden. Die Verwendung und Definition dieser Kriterien erfolgt, wie in Abschnitt 4.2 erörtert, in einer extern abgespeicherten

Datei, welche zu Beginn des Morphingprozesses geladen wird. Innerhalb dieser Abschlussarbeit werden für Solids zwei Qualitätskriterien verwendet:

1. Größenverhältnisse
2. Negatives Volumen

Da Shells keine Volumen aufspannen, wird für diese das zweite Kriterium nicht betrachtet.

Die in dieser Arbeit verwendete Funktion zur nachträglichen Anpassung des Netzes ist die „Fix Quality“-Funktion. Diese verändert standardmäßig lediglich Elemente, welche selbst nicht die Kriterien erfüllen oder in deren direkten Umgebung liegen. Standardmäßig werden die nächsten drei Nachbarlemente mit verändert. Außerdem wird durch die Verwendung dieser Funktion gewährleistet, dass keine neuen Knotenpunkte oder Elemente wie z.B. bei einer „Reconstruct“-Funktion hinzugefügt werden, sodass jede Geometrievariation die gleiche Anzahl an Knotenpunkten und Elementen aufweist. Die Funktion verschiebt somit lediglich die Knotenpunkte der einzelnen Elemente um möglichst alle Probleme zu beheben.<sup>6</sup>

Die Kriterien werden dabei wie in Tabelle 4.1 festgelegt:

Elementtype	Größenverhältnisse	Negatives Volumen
Solid	8	Partiell
Shell	3	—

Tabelle 4.1: Netzqualitätskriterien

Der Wert partiell bedeutet dabei, dass auch kein Anteil des Volumens eines Solids negativ sein darf, obwohl sein Gesamtvolumen positiv ist. Für die betrachteten Tetraeder ist ein negatives Volumen nur möglich, wenn das gesamte Volumen negativ ist. Ein partielles negatives Volumen kann somit nicht auftreten. Das Qualitätskriterium versagt aber sowohl im Falle eines partiellen als auch bei einem gesamten negativen Volumen eines Elements.

Die „Fix Quality“<sup>7</sup>-Funktion wird im Script nach Abschluss eines Morphingprozesses in eine Richtung aufgerufen, um mögliche fehlerhafte Elemente zu korrigieren.

Neben den Qualitätskriterien sind die in Abschnitt 4.2.4 beschriebenen Smoothingoperationen ebenfalls ausschlaggebend für die Qualität des gemorphten Netzes. Dies ist damit zu erklären, dass im vorliegenden Fall keine harten Kanten der Geometrie erwünscht sind. Eben diese Kanten werden durch eine lokale extreme Verzerrung von einzelnen Elementen hervorgerufen, sodass es dabei häufiger zu einer Missachtung der Qualitätskriterien kommt. Trotz diverser Smoothingoperationen und der „Fix Quality“<sup>8</sup>-Funktion können Probleme mit der Netzqualität auftreten. Dies liegt primär daran, dass die „Fix Quality“<sup>9</sup>-Funktion durch ein reines Verschieben der Knotenpunkte sehr

<sup>6</sup>BETA CAE Systems 2017, S.1081.

<sup>7</sup>BETA CAE Systems 2019, s.1081.

<sup>8</sup>BETA CAE Systems 2019, s.1081.

<sup>9</sup>BETA CAE Systems 2019, s.1081.

eingeschränkt ist und aufgrund dessen nicht jegliche Verwicklungen des Netzes beheben kann. Die Grenzen des in dieser Arbeit dargestellten Morphingprozesses werden somit maßgeblich durch die Größe der Verschiebungen definiert, ab denen Smoothing-Verschiebungen oder die „Fix Quality“<sup>10</sup>-Funktion eine Erfüllung der Qualitätskriterien nicht mehr gewährleisten können.

---

<sup>10</sup>BETA CAE Systems 2019, s.1081.

## 5 Aufbau einer Designstudie

Der Sinn der in dieser Arbeit dargelegten Designstudie besteht darin, Tendenzen bezüglich ausgewählter Bewertungskriterien durch die Verschiebung der Anbindungspunkte aufzudecken. Im weiteren Schritt soll aus diesen Tendenzen die optimale Lage der Anbindungspunkte berechnet, sowie ein analytisches Metamodell aufgestellt werden.

Zur Erstellung der Designs wird die Optimierungssoftware Optislang verwendet. Die automatisierte Durchführung der einzelnen Geometrievariationen wird mittels eines Python-Scriptes in Spyder durchgeführt. Der dazu verwendete Arbeitsablauf wird im Folgenden dargestellt, sodass dieser auf weiteren Designstudien angewendet werden kann.

### 5.1 Designvariation mittels Optislang

Optislang bietet die Möglichkeit, Parameter zur Durchführung einer Designstudie zu variieren. Der dazu erstellte Arbeitsablauf ist in Abbildung 5.1 dargestellt. Der grau hinterlegte Block stellt den generellen Ablauf für ein Design

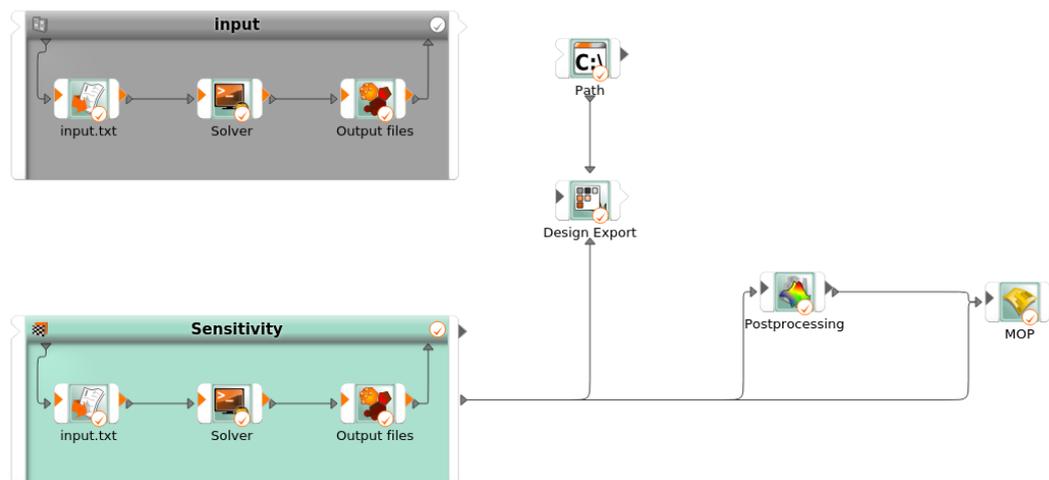


Abbildung 5.1: Ablaufplan Designvariation Optislang <sup>1</sup>

dar. Dieser besteht aus den folgenden drei Schritten:

1. Input-Datei einlesen
2. Solver ausführen
3. Output-Datei speichern

<sup>1</sup>Wysocki 2020b

Die Input-Datei enthält dabei lediglich die Bezeichnungen der Anbindungspunkte, kombiniert mit der Richtung der Verschiebung und den Wert, um den der Punkt verschoben werden soll. Es ergibt sich somit für eine positive Verschiebung von R3 um 10 mm in X-Richtung die Zeile  $R3_X = 10$  in der Input-Datei. Nach diesem Prinzip werden alle Verschiebungen der anderen Anbindungspunkte und Raumrichtungen in der Input-Datei gespeichert, sodass sich in Summe 18 Variablen ergeben. Als Solver wird ein Python-Script aufgerufen, welches eine Dummy-Output-Datei in den Ordner des jeweiligen Designs kopiert. Als Output-Datei wird ebenfalls eine Dummy-Datei verwendet, da der in Abbildung 5.1 dargestellte Ablauf der Erzeugung unterschiedlicher Designs dient und somit noch keine Bewertungsgröße vorliegt. Optislang fordert an dieser Stelle die Verwendung eines Solvers sowie einer Output-Datei. Hier dient Optislang jedoch lediglich zur Input-Variation des Morphing-Scriptes, sodass der den Output betreffende Teil vernachlässigt werden kann.

Der grün hinterlegte Block beruht auf dem soeben erörterten grauen Block. Dieser führt eine Sensitivitätsanalyse durch, um die gewünschten Designvariationen zu erzeugen. Alle der 18 Variablen, welche nicht betrachtet werden, werden als Konstanten mit dem Wert null definiert. Für die betrachteten Variablen werden Wertebereiche sowie mögliche andere Restriktionen festgelegt. Im vorliegenden Fall eignet sich vor allem die betragsmäßige Begrenzung der Verschiebungen eines Anbindungspunktes in alle Raumrichtungen, sodass alle Grenzdesigns auf einer Kugel um den ursprünglichen Anbindungspunkt liegen. Anschließend werden der Versuchsplan space-filling Latin-Hypercube und die Versuchsstufen definiert und die Anzahl der Designs berechnet. Mit dieser Kombination wird von Optislang für jedes Design ein Ordner angelegt, in dem jeweils die Input-Datei mit den tatsächlichen Verschiebungen dieses Designs, die Dummy-Output-Datei und das Solver-Python-Script abgelegt werden. Der nachgeschaltete 'Design-Export'-Block, erstellt eine 'Output.csv', welche in jeder Zeile ein Design mit allen Verschiebungen und dem Dummy-Output-Wert beschreibt. Der Speicherpfad der CSV-Datei wird vom 'Path'-Block vorgegeben.

Die Blöcke 'Postprocessing' und 'MOP'(Metamodel of Optimal Prognosis) werden von Optislang automatisch angelegt und dienen der graphischen Darstellung des Versuchsraumes. Daher können auch diese an dieser Stelle noch vernachlässigt werden.

## 5.2 Gesamtprozess der Designstudie

Da die Anzahl der unterschiedlichen Designs einer Designstudie theoretisch unbegrenzt ist, ist es nötig, dass nicht nur das Morphing gscriptet erfolgt, sondern auch die Initialisierung des Morphingprozesses für jedes Design automatisch gestartet wird.

Dazu wird ein übergeordnetes Script angelegt. Dieses behandelt die einzelnen Geometrievariationen, Simulationen der Frequency-Response-Functions (FRFs) sowie die Speicherung der Ergebnisse und deren Auswertung für alle zu betrachtenden Designs.

Der Programm-Ablauf ist in Abbildung 5.2 als Flussdiagramm visualisiert und nachfolgend erklärt.

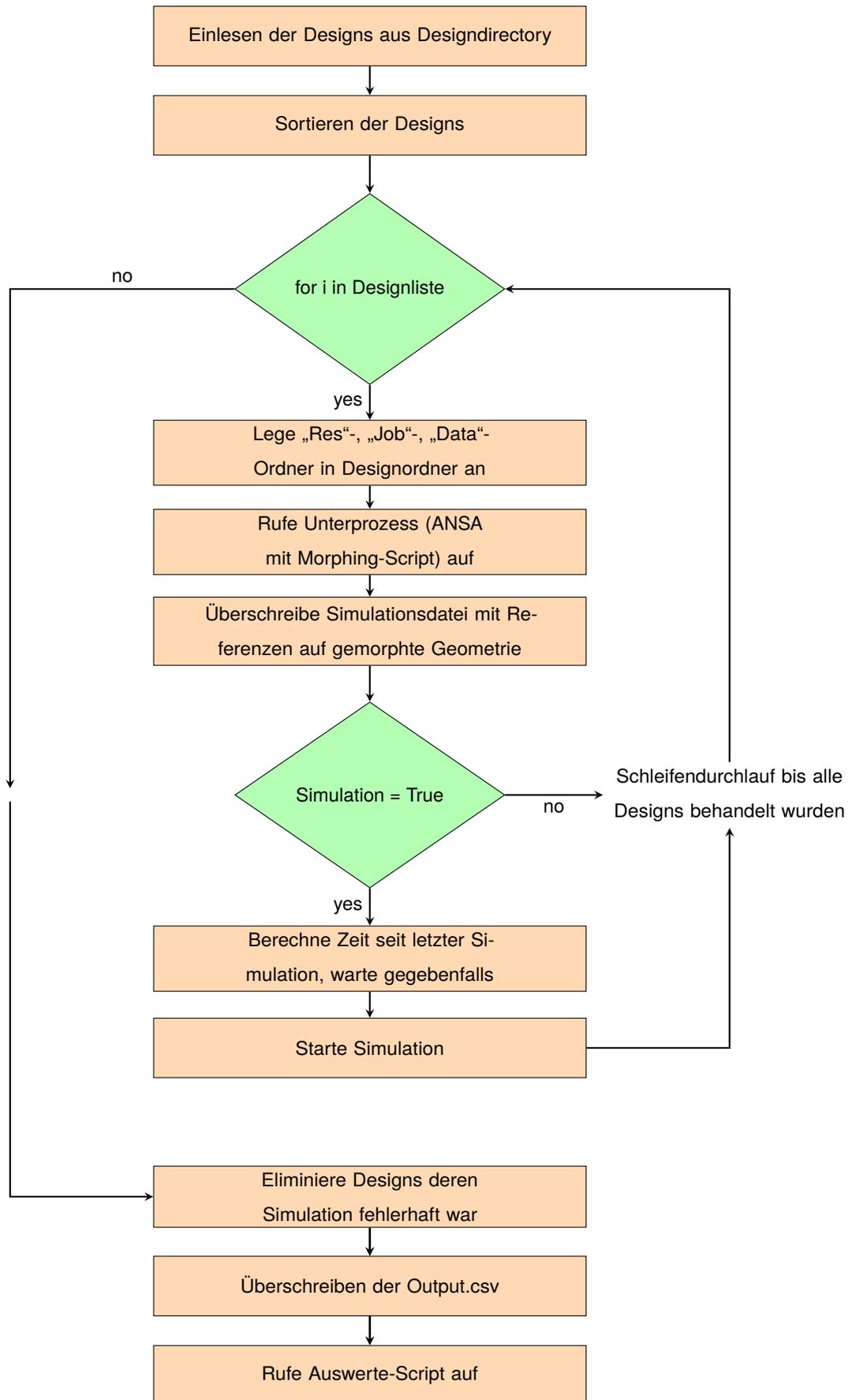


Abbildung 5.2: Flussdiagramm Gesamtprozess

Zu Beginn des Scriptes werden zunächst aus einem in Abschnitt 5.1 erstellten Designstudienordner alle Unterordner eingelesen. Da jeder der Unterordner ein Design darstellt, kann somit eine Liste, welche alle Designs beinhaltet, erstellt werden. In Python wird diese Liste zunächst unsortiert angelegt. Deshalb wird als nächster Schritt eine sortierte Liste angelegt, in welcher der erste String den Wert 'Design0001' hat. Dies ist nötig um im weiteren Verlauf des Scriptes, aufgrund der Zeilennummer der Liste, auf das Design zurückschließen zu können.

Anschließend wird für jedes Element der angelegten Design-Liste eine Schleife ausgeführt. Als erstes wird die Ordnerstruktur so vorbereitet, dass im folgenden auch eine Simulation direkt durchgeführt werden kann. Dazu wird in jedem der Design Ordner ein 'Job'-, 'Res'- und 'Data'-Ordner erstellt. Um ANSA das aktuell zu morphende Design zu übergeben, wird eine Text-Datei angelegt, welche den Ordnerpfad zu dem aktuell behandelten Designordner enthält. ANSA wird nun in einem Unterprozess aufgerufen. Die zu morphende Radträger-Datei wird geöffnet und das Morphing-Script ausgeführt. Das ausführende Python-Script wartet dabei, bis der Unterprozess wieder geschlossen wird und somit der Morphingprozess abgeschlossen ist. Zur Vorbereitung der Simulation der FRFs wird die Simulationsdatei in eine neue Datei geschrieben, wobei jede Zeile mittels Regular-Expressions nach Schlüsselwörtern abgesucht wird. Diese Schlüsselwörter sind in diesem Fall 'include'-Befehle. Diese müssen ersetzt werden, damit nicht mehr auf die originale Geometrie, sondern auf die gemorphte Geometrie referenziert wird. Die neue Datei wird in dem angelegten 'Job'-Ordner abgelegt. Der letzte Schritt der Schleife wird nur durchgeführt, wenn er zuvor vom Benutzer, mit durch das auf 'True' setzen einer Variablen, aktiviert wird. In diesem Schritt wird die Simulationsdatei an den Rechencluster geschickt. Die hieraus resultierenden Simulationsergebnisse werden automatisch im 'Res'-Ordner des jeweiligen Designs abgelegt. Diese Schleife wird für jede zu morphende Geometrie durchgeführt. Im Falle, dass die Simulation durchgeführt wird, wird die Zeit zwischen den Schleifendurchläufen überwacht. Dies ist nötig, da ein eindimensionaler Morphingprozess einige Sekunden benötigt, die zugehörige Simulation der FRFs aber etwa drei Minuten dauert. Es wird deshalb zu Anfang des Scriptes eine Pufferzeit eingeführt, damit nicht zu viele Simulationen gleichzeitig auf dem Cluster laufen und das Script dadurch abgebrochen wird. Am Ende jedes Schleifendurchlaufs wird die vorgegebene Pufferzeit von der Zeit, welche seit dem letzten Starten einer Simulation vergangen ist, abgezogen. Ist der erhaltene Wert negativ, wird der Prozess angehalten und nach Ablauf des betragsmäßigen Wertes in Sekunden fortgesetzt. Ist der Wert positiv, wird das Script weiter ausgeführt.

Nachdem alle Simulationen durchgeführt wurden, werden die Simulationsergebnisse zur Auswertung vorbereitet. Zunächst werden alle Geometrievariationen mit ungenügender Netzqualität aussortiert. Trotz der in Abschnitt 4.2.5 dargestellten Ansätze zur Erhaltung der Netzqualität, kann nicht für jede Geometrie garantiert werden, dass die Kriterien an die Netzqualität nicht verletzt werden. Da im Rechencluster vor Durchführung der Simulation ebenfalls ein Qualitätscheck durchgeführt wird und bei Nichterfüllung dieser Kriterien die Simulation nicht durchgeführt wird, kann durch das Vorhandensein von Simulationsergebnissen ein Rückschluss auf die Netzqualität geschlossen werden. Speziell das Vorhandensein von Elementen mit negativen Volumen führt instantan zum Versagen der Simulation. Da die während dieser Arbeit gemachten Beobachtungen zeigten, dass sobald die „Fix Quality“<sup>2</sup>-Funktion das Kriterium der Größenverhältnisse der Elemente nicht mehr einhalten kann, auch negative Volumen auftreten, kann

---

<sup>2</sup>BETA CAE Systems 2019, S.1081.

generell davon ausgegangen werden, dass eine erfolgreich durchgeführte Simulation beide Qualitätskriterien erfüllt. Da während des Simulationsprozesses immer eine log-Datei angelegt wird, bestätigt diese, dass die Simulation erfolgreich an den Rechencluster übergeben wurde. Diese Datei enthält dabei lediglich Informationen über den Prozessablauf. Das Nichtvorhandensein der Ergebnisdatei zeigt wiederum, dass die angestoßene Simulation nicht erfolgreich abgeschlossen wurde. Die somit als fehlerhaft identifizierten Designs werden aus der Designliste gelöscht und im weiteren Verlauf nicht betrachtet. Dieser Schritt ist wichtig, da nicht aussortierte Designs ohne Bewertung in den Optimierungsprozess mit einbezogen würden und somit jegliche Aussagen verfälschen.

Der vorletzte Teil des Ausführungs-Scriptes überschreibt die in Abschnitt 5.1 erstellte 'Output.csv'. Hierbei werden die als nicht fehlerhaft identifizierten Designs betrachtet. Außerdem werden die tatsächlichen Verschiebungen der Anbindungspunkte für jedes Design eingelesen und abgespeichert. Als letzter Schritt wird jede Variable, die in der Designstudie nicht variiert wurde, gelöscht. Auch die Dummy-Output Größe wird eliminiert und die Ergebnisdateien der Simulation mittels eines extern aufgerufenen Python-Scriptes für die Bewertung vorbereitet.

Das Einlesen der tatsächlichen Verschiebungen erfolgt dabei aus einer Numpy-Datei, welche im gescripteten Morphing für jedes Design erstellt und in den jeweiligen 'Data'-Ordner abgelegt wird. Anschließend wird ein externes Auswerte-Script aufgerufen, welches die tatsächlichen Bewertungsgrößen für die Designstudie berechnet und diese in die 'Output.csv' schreibt.

### 5.3 Simulation - Frequency Response Function

Als Grundlage für die Bewertung einer Geometrievariation werden FRFs simulativ für jedes Design berechnet und verglichen.

Die einzelnen Funktionen werden ermittelt, indem eine Einheitskraft an den Anbindungspunkten eindimensional angreift. Die Reaktion auf diese Kraft wird am Radanbindungspunkt ebenfalls eindimensional gemessen. Somit ergeben sich für jede Kraft am Anbindungspunkt drei Übertragungsfunktionen. In Summe ergeben sich somit aus fünf Anbindungspunkten, ohne DSG, drei Kraftdimensionen und jeweils drei Messungen,  $5 \cdot 3 \cdot 3 = 45$  Übertragungsfunktionen. Die 15 verschiedenen Anregungsfälle werden dabei als Subcases bezeichnet. Zur Berechnung der FRFs wird der Radträger an den in Abbildung 5.3 blau hinterlegten Punkten fixiert. Dies sind auch die Verbindungsstellen zwischen Radträger und Radlager.

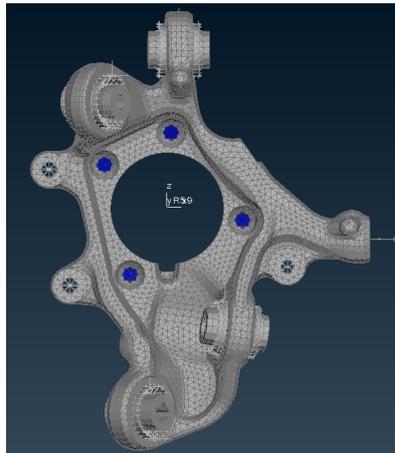


Abbildung 5.3: Fixierungen Radträger

Der zur Simulation verwendete Frequenzbereich weist eine konstante Schrittweite von 2 Hz auf und erstreckt sich von 0 bis 2000 Hz, sodass somit alle relevanten Eigenfrequenzen des Radträgers in der Simulation ersichtlich sind.

## 5.4 Optimierungsparameter

Die Vielfalt der Optimierungsparameter, welche bei einer FRF verwendet werden können, ist groß. Unter anderem werden Ansätze verfolgt, einzelne Peaks zu verringern oder zu verschieben. Das Verschieben einer Frequenz wird vor allem dann angewendet, wenn eine Resonanz mehrerer Bauteile bei gleicher Frequenz unbedingt vermieden werden soll. Da der Hauptaspekt dieser Arbeit vor allem das Aufzeigen der Funktion des Arbeitsablaufs, sowie die Betrachtung des Box-Morphings ist, wird der Auswahl des Optimierungsparameters geringere Bedeutung zugewiesen. Im Folgenden wird somit zur Optimierung der FRF ein Summenwert der diskreten Amplituden-Frequenz-Funktion gebildet, indem die einzelnen Werte addiert werden. Da in der betrachteten Simulation die Schrittweite der Frequenz konstant ist, ist der so berechnete Wert proportional zur Schwingungsleistung des Systems und lässt Aussagen bezüglich dieser zu. Der eingeführte Summenwert soll minimiert werden. Um dabei die Anzahl der Bewertungsparameter gering zu halten, wird die Annahme getroffen, dass eine Verschiebung eines Anbindungspunktes lediglich einen Einfluss auf die Übertragungsfunktionen dieses Punktes hat. Im Falle, dass eine Designstudie durchgeführt wird, in der nur R3 verschoben wird, werden nur die neun sich daraus ergebenden Übertragungsfunktionen in der Auswertung betrachtet. Zur Vervollständigung des Arbeitsablaufs, wird das am Ende des Ausführungs-Scriptes auszuführende Auswerte-Script ausgeführt. Dies liest zunächst die im Ausführungs-Script verarbeiteten Simulationsergebnisse aus Numpy-Dateien ein. Anschließend wird, wie bereits beschrieben, der Summenwert der diskreten Funktionen durch Aufaddieren der einzelnen Werte gebildet. Dies erfolgt für jedes Design und jeden gemorphten Anbindungspunkt. Die berechneten Werte werden abschließend

in die 'Output.csv' angehängt, sodass in dieser jeweils eine Zeile ein Design mit den Verschiebungen an den entsprechenden Anbindungspunkten (siehe Spaltennamen) sowie allen berechneten Bewertungsgrößen erhält. Die Bewertungsgrößen werden dazu nach den in der Simulation verwendeten Subcases, kombiniert mit der Dimension der Kraftmessung, benannt.

## 5.5 Optimierung mittels Optislang

Über ein Excel-Add-In werden die Variablen, sowie Optimierungsparameter für Optislang nutzbar gemacht. In Optislang wird ein Arbeitsablauf erstellt, welcher auf Basis der eingelesenen Daten, die optimale Gestalt des Bauteils berechnet. Hierzu werden die in Abschnitt 2.2 erörterten Optimierungsalgorithmen verwendet. Da die optimale Gestalt nicht notwendigerweise eine der bereits berechneten Gestalten darstellt, wird die als optimal ermittelte Gestalt manuell erstellt. Diese dient somit auch der Validierung des analytisch erstellten Metamodells. Der dazu verwendete Systemaufbau ist in Abbildung 5.4 dargestellt. Der 'MOP'-Block beinhaltet das analytisch

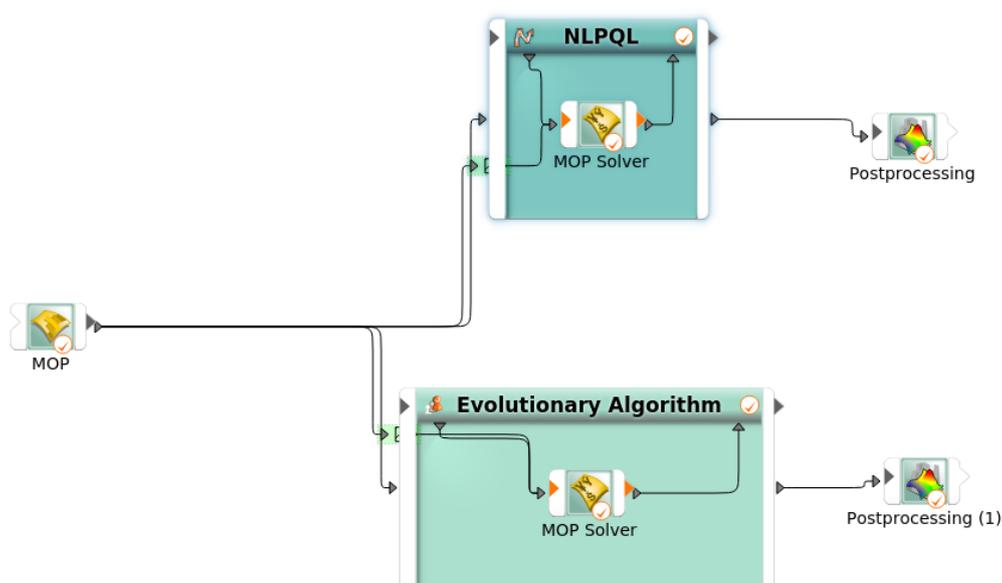


Abbildung 5.4: Systemaufbau Optimierung Optislang<sup>3</sup>

ermittelte Metamodell, in Abhängigkeit der Variablen, für jede der einzelnen Bewertungsgrößen. Die Optimierung kann bezüglich einzelner FRFs oder einer Kombination dieser erfolgen.

Die Blöcke 'Evolutionary Algorithm' und 'NLPQL' bilden die Optimierungsalgorithmen ab. Die Verwendung von zwei Algorithmen ermöglicht einen gegenseitigen Vergleich. Im Falle, dass beide Algorithmen das selbe Design als optimal identifizieren, ist dies mit höherer Wahrscheinlichkeit das tatsächlich optimale Design, als wenn nur ein Algorithmus

<sup>3</sup>Wysocki 2020a

es identifiziert. Beim NLPQL nimmt vor allem das Startdesign Einfluss auf das als optimal gefundene Design, sodass dieses auch maßgeblich beeinflusst, ob beide Algorithmen das selbe Design als optimal identifizieren. Als Startdesign wird nachfolgend das originale Design verwendet. Das Postprocessing dient zur graphischen Darstellung der Optimierungsalgorithmen.

## 5.6 Exemplarische Designstudie

Um die Funktion des in dieser Arbeit aufgestellten Arbeitsablaufs zu überprüfen, wird in diesem Kapitel eine Designstudie und alle damit einhergehenden Schritte dargelegt.

Um die Gestaltvariation dabei nicht zu umfangreich abzubilden, werden lediglich Verschiebungen an Punkt R1 betrachtet. Diese erfolgen in alle drei Raumrichtungen, wobei die betragsmäßige Verschiebung auf 10 mm begrenzt ist. Zur Erstellung der Designvariationen wird das, in Abschnitt 3.2.3 vorgestellte, space-filling Latin-Hypercube im in Abbildung 5.1 beschriebenen Prozess, verwendet.

Optislang ermöglicht dabei intuitiv keine Betrachtung eines kugelförmigen Versuchsraumes. Es können zwar Restriktionen an die Faktoren, in diesem Fall

$$10 \geq \sqrt{R1_X^2 + R1_Y^2 + R1_Z^2} \quad (5.1)$$

gestellt werden, diese werden jedoch lediglich nach Durchführung der Sensitivitätsanalyse auf ihre Erfüllung überprüft. Um dennoch einen kreisförmigen Versuchsraum abzubilden, wird ein Zwischenschritt durchgeführt. Zunächst wird gemäß Abschnitt 5.1 ein würfelförmiger Versuchsraum mit Verschiebungen  $-10 \leq d \leq 10$  nach dem space-filling Latin-Hypercube Verfahren angelegt. Dazu wird die Versuchsanzahl auf 300 festgelegt. Anschließend wird mittels eines Python-Scriptes jede der Varianten, welche Gleichung 5.1 verletzen, aus der Versuchsreihe entfernt. Um sicher zu stellen, dass Randpunkte des Versuchsraumes betrachtet werden, werden manuell sechs Designs erzeugt, welche jeweils einen der maximalen Verschiebungswerte beinhalten. Die anderen Werte werden um Gleichung 5.1 nicht zu verletzen als null definiert. Zur Anlegung der nötigen Ordnerstruktur wird in Optislang eine Platzhalterdesignstudie angelegt, welche die exakte Anzahl an Designs der zu betrachtenden Designstudie enthält. Die tatsächlichen Verschiebungen werden dann in die Platzhalterstudie übertragen, sodass hieraus die gewünschte Designstudie entsteht. Aus dem dargelegten Prozess ergibt sich ein Versuchsplan mit 164 Designs. Diese sind im Anhang (A) vollständig dargestellt.

## 6 Bewertung Designstudie

Nachfolgend werden die Ergebnisse der in Abschnitt 5.6 implementierten Designstudie dargestellt und bewertet. Zunächst soll jedoch die Güte des Morphingmodells validiert werden. Hierzu werden die Soll-Verschiebungen mit den Ist-Verschiebungen der gemorphten Geometrien verglichen. Die Verschiebung der gemorphten Geometrien werden, vor deren Übergabe, auf die vierte Nachkommastelle gerundet. Die Verschiebung der Geometrievariation wird somit auf ein tausendstel Millimeter genau angegeben.

Gemäß Abschnitt 2.3.2 und Abschnitt 2.3.1 wird die Varianz, die Standardabweichung der Abweichung von Soll- und Ist-Verschiebung der Designstudie und die dabei auftretende mittlere prozentuale Abweichung berechnet. Diese sind in Tabelle 6.1 dargestellt. Die Standardabweichung sowie die Standardvarianz sind dabei dimensionslos, da sie auf Grundlage der relativen Abweichung der Soll- und Ist-Verschiebung berechnet wurden. Dabei kann

Verschiebung	mittlere prozentuale Abweichung [%]	Standardvarianz	Standardabweichung
R1 in X	1,1394e-03	8,7184e-10	2,9527e-05
R1 in Y	1,0836e-03	7,9741e-10	2,8238e-05
R1 in Z	6,9000e-04	7,9304e-10	2,8161e-05

Tabelle 6.1: Bewertungskriterien Abweichungen Soll-, Ist-Verschiebung, gerundet auf vierte Nachkommastelle, bezogen auf die Potenz

lediglich eine verschwindende Abweichung beobachtet werden, sodass die Soll-Verschiebung annähernd exakt erreicht wird. Hierbei sei angemerkt, dass die betrachtete Genauigkeit der Referenzpunkte lediglich theoretische Relevanz zur Validierung der Morphingmethode aufweist. Eine praktische Anwendung dieser Auflösung ist nicht vorhanden, da eine solche Präzision im Lagersitz des Radträgers keinerlei Mehrnutzen bringt. Die gefundenen Werte für die Standardabweichung sind auch deutlich geringer als gängige Bauteiltoleranzen. Des Weiteren sei darauf hingewiesen, dass die zur Berechnung der Werte in Tabelle 6.1 betrachteten Soll-Verschiebungen nicht auf die vierte Nachkommastelle gerundet wurden. Es ist somit fraglich, ob die minimalen Diskrepanzen tatsächlich aus dem Morphingprozess oder lediglich aus der Rundung der ausgegebenen Verschiebungswerten resultieren. Da die mittleren Abweichungen der einzelnen Koordinatenrichtungen von der Größenordnung  $\mathcal{O}(-5, -6)$  sind, ist anzunehmen, dass die Verschiebungen exakt abgebildet werden. Die Box-Morphing Methode ist somit definitiv geeignet um eine exakte Verschiebung einzelner Referenzpunkte abzubilden.

Abbildung 6.1 stellt die CoP Werte der Variablen  $R1_X$ ,  $R1_Y$  und  $R1_Z$  über die unterschiedlichen Bewertungsgrößen dar. Die Nummerierung der Modelle ist dabei wie folgt zu lesen. Out steht für Output und signalisiert, dass es sich um die Bewertungsgröße handelt. 11091 bis 11093 bezeichnen den Anregungspunkt der simulierten FRF. 1109 beschreibt R1, den Anbindungspunkt an die Schubstrebe. Die Werte 1 bis 3 geben die Achse der eindimensionalen Anregung gemäß karthesischem Koordinatensystem an. Die folgenden Nummern 77706 sind die Kennung des Radmittelpunktes, an welchem die in den Radträger eingeleitete Kraft gemessen wird. Die daran angefügte Ziffer stellt wiederum die Achse der Kraftmessung dar. Der Term 'Out11093\_77062' bezeichnet somit

den Summenwert der FRF mit Anregung am Punkt R1 in Z-Richtung und Messung am Radmittelpunkt in Y-Richtung. Aus der vorliegenden Matrix wird ersichtlich, dass die Optimierung eines komplexen Bauteils, wie dem in dieser

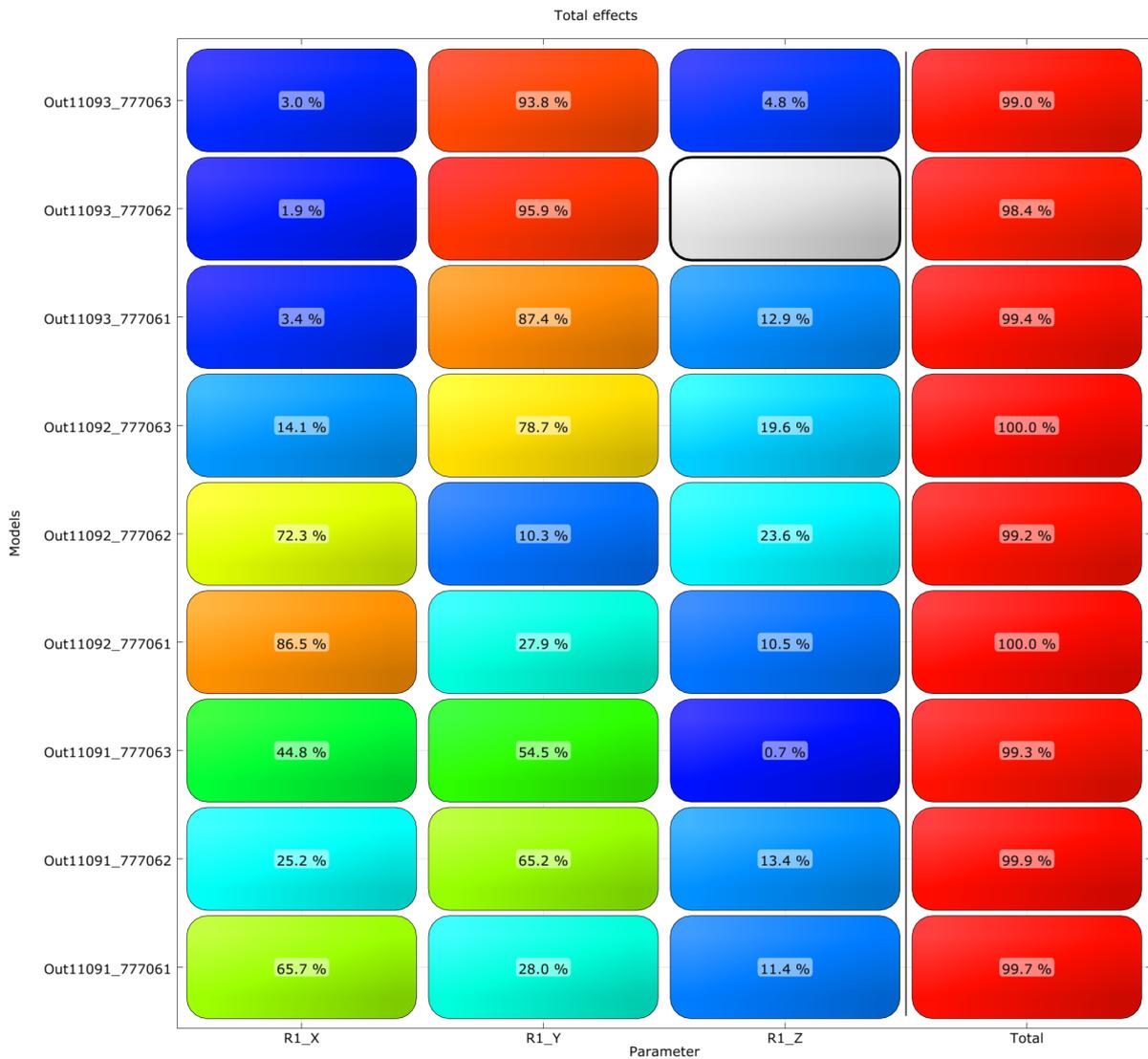


Abbildung 6.1: CoP Matrix für Metamodell mit neun betrachteten Subcases

Arbeit betrachteten Radträger, nicht durch reine Betrachtung von einfachen Zusammenhängen, wie der Analyse von Hebelarmen, durchgeführt werden kann.

Eine Optimierung des Bauteils ist somit keineswegs trivial. Der Begriff Optimierung ist dabei kritisch zu sehen, da in schwingungstechnischen Fragestellungen oft mit einer Verbesserung des einen Kriteriums, eine Verschlechterung an einer anderen Stelle einhergeht.

Da in dieser Arbeit die generelle Fragestellung nach der Optimierung betrachtet wird, werden im folgenden zwei Ansätze untersucht.

1. Minimierung der Gesamtheit aller neun FRFs
2. Minimierung einer spezifischen FRF

Zur Berechnung des ersten Falls werden die neun betrachteten FRFs durch simples Addieren zusammengefasst. Das so gefundene Bewertungskriterium minimiert folglich die gesamte übertragene Schwingungsleistung.

Zur Optimierung wird der in Abbildung 5.4 dargestellte Arbeitsablauf verwendet und die als optimal ermittelten Geometrien der unterschiedlichen Algorithmen verglichen. Sowohl der Evolutionäre Algorithmus, als auch der NLPQL werden mit den von Optislang standardmäßig vorgeschlagenen Parametern betrieben.

Die resultierenden Graphen der Algorithmen sind in Abbildung 6.2 dargestellt.

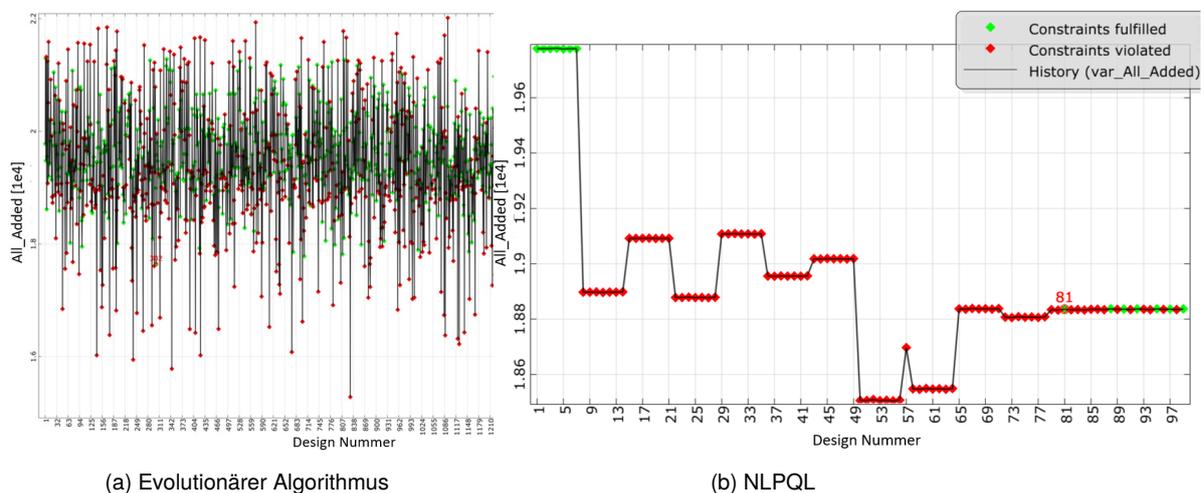


Abbildung 6.2: Algorithmen für zusammengefasste FRFs mit Verschiebungsrestriktion ( $Betrag \leq 10$ )

Durch Legen einer gedanklichen Ausgleichsgerade ist in Abbildung 6.2a kaum ein Abwärtstrend zu beobachten, wohingegen Abbildung 6.2b eine klare Verbesserung zum Ausgangszustand findet. Eine zusätzliche Betrachtung der als optimal identifizierten Designs (Tabelle 6.2) zeigt, dass für den betrachteten Fall beide Algorithmen vollständig unterschiedliche Optimierungsrichtungen finden. Dies liegt daran, dass der NLPQL instantan die Suchrichtung ändert, falls in der zunächst abgetasteten Richtung keine Verbesserung vorliegt. Es ist somit möglich, dass lediglich ein lokales Minimum gefunden wird. Der Evolutionäre Algorithmus hingegen findet mit größerer Wahrscheinlichkeit eine Lösung die sehr nahe am globalen Minimum liegt. Dies wird auch von den Bewertungsgrößen unterstützt. Der Evolutionäre Algorithmus findet ein weitaus besseres Design als der NLPQL. Die letzte Spalte in Tabelle 6.2 dient zur Validierung des Modells. Hierzu wird das als optimal identifizierte Design nachträglich gemorph und auf der gleichen Grundlage bewertet. Die erhaltenen Abweichungen von  $-0,50\%$  (evolutionär) und  $-0,58\%$  deuten darauf hin, dass die analytisch berechneten Metamodelle definitiv belastbar sind.

Algorithmus	Restriktion	R1_X	R1_Y	R1_Z	Bewertung Metamodell	Bewertung Gemorhpt
Evolutionär	10 mm	3,35278	8,63747	3,2652	17644,1	17733,0
NLPQL	10 mm	-7,67304	-2,23366	-6,0102	18835,9	18945,0

Tabelle 6.2: Optimale Designs bei Betrachtung der addierten FRFs mit Restriktion

Das vom Evolutionären Algorithmus als optimal identifizierte Design wird auch von einem Surface Plot des Metamodells gestützt. Hierbei sind die Verschiebungen in X- und Y-Richtung aufgetragen, da der Matrixplot in Abbildung 6.4

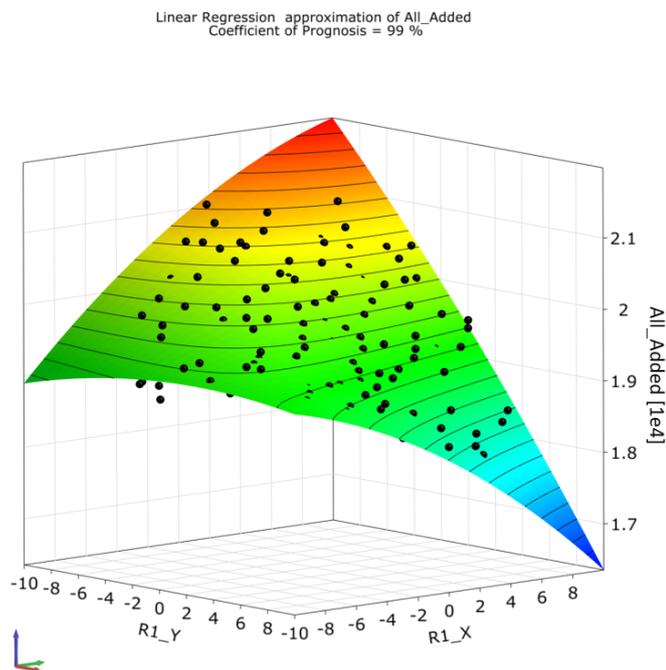


Abbildung 6.3: Plot des Metamodells für die Summe der betrachteten Übertragungsfunktionen über der Verschiebung von R1 in X- und Y-Richtung aufgetragen

zeigt, dass deren Anteile an der Bewertungsgröße akkurater vorhersagbar sind. Abbildung 6.3 impliziert, dass eine betragsmäßige Verschiebung mit weitaus größerem Betrag, als den betrachteten 10 mm, die Geometrie des Bauteils weiter optimieren würde. Um diese Prämisse zu überprüfen, wird an dieser Stelle noch einmal der Optimierungsalgorithmus für den ersten betrachteten Fall bemüht, wobei nun die Restriktion der betragsmäßigen Verschiebung auf 10 mm aufgehoben wird. Die sich ergebenden Graphen der Optimierungsalgorithmen sind in Abbildung 6.5 dargestellt. Dabei ist auch im Falle des Evolutionären Algorithmus sofort ersichtlich, dass dieser während der Laufzeit eine kontinuierliche Verbesserung findet. Interessanterweise sind auch für diesen Fall die gefundenen Optima der unterschiedlichen Algorithmen gänzlich verschieden. Es ist in beiden Fällen eine deutliche Verbesserung bezüglich des Anfangswertes ersichtlich. Fraglich ist jedoch, ob die als optimal identifizierten Verschiebungen noch mittels der verwendeten Morphingstruktur abbildbar sind, ohne negative Elemente zu erzeugen. Des Weiteren wurden beide

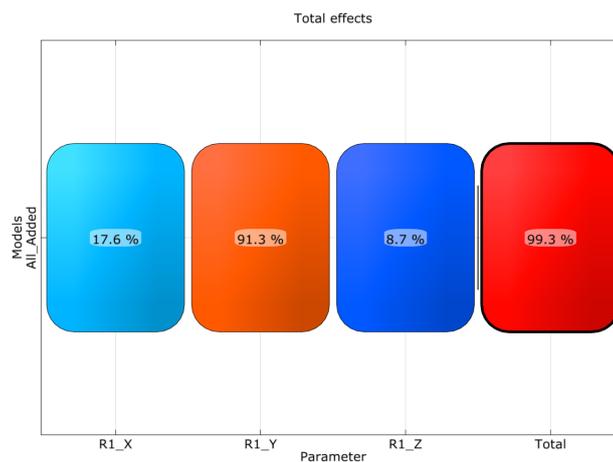


Abbildung 6.4: CoP Matrix für die Summe der betrachteten Übertragungsfunktionen

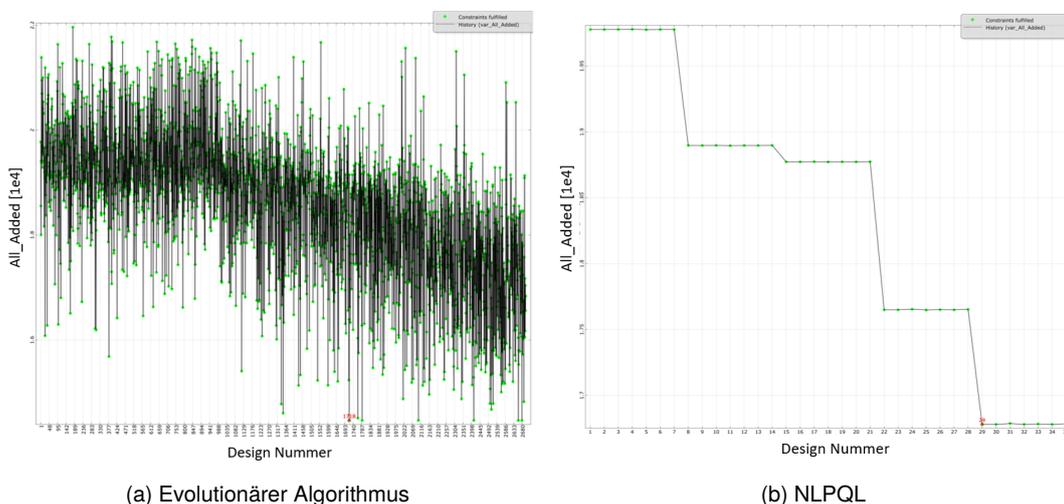


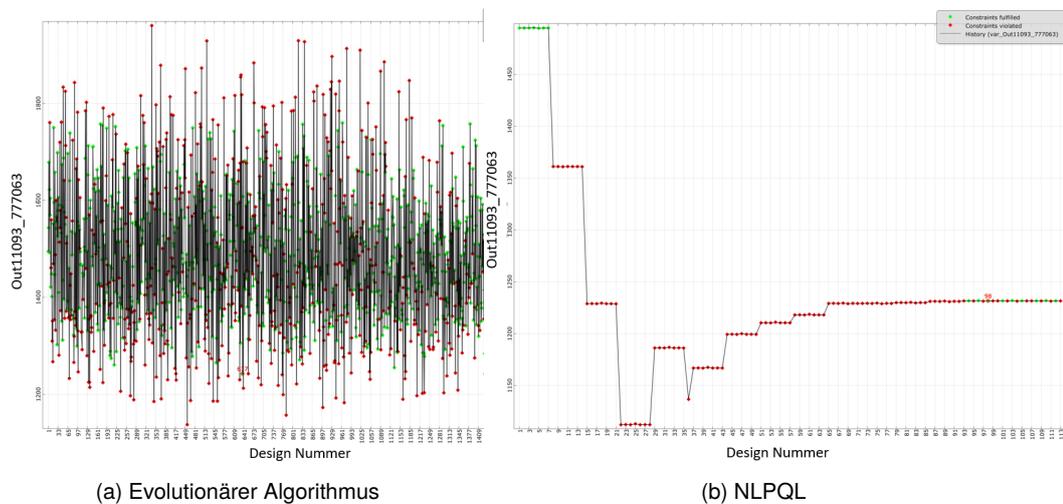
Abbildung 6.5: Algorithmen für zusammengefasste FRFs ohne Verschiebungsrestriktion

als optimal identifizierten Geometrien nicht in der ursprünglichen Designstudie betrachtet, sodass es sich hierbei um eine Extrapolation des Metamodells in Gebiete außerhalb des ursprünglich definierten Versuchsraumes handelt. Die Gültigkeit muss somit hinterfragt werden. Tabelle 6.3 stellt die als optimal identifizierten Designs nach dem selben Prinzip wie Tabelle 6.2 dar. Äquivalent wird auch die Validierung der Aussagen ausgeführt. Das optimale Design des Evolutionären Algorithmus ist anhand der aufgebauten Morphingstruktur nicht ohne Generierung von negativen Elementen erstellbar, sodass eine Simulation und Bewertung des Designs nicht möglich ist. Das vom NLPQL vorgeschlagene Design kann zwar erfolgreich abgebildet und simuliert werden, jedoch fällt die Bewertungsgröße deutlich anders aus, als vom Metamodell vorhergesagt. Der Unterschied liegt bei 6,90%. Das angesprochene Problem der Extrapolation außerhalb des ursprünglichen Versuchsraumes ist somit nicht zu vernachlässigen. Gleichzeitig fällt die zur Validierung berechnete Bewertungsgröße schlechter aus, als die des vom Evolutionären Algorithmus mit Restriktionen vorgeschlagenen Designs, sodass dieses präferiert wird.

Algorithmus	Restriktion	R1_X	R1_Y	R1_Z	Bewertung Metamodell	Bewertung Gemorht
Evolutionär	keine	10	10	10	14464,6	—
NLPQL	keine	-10	-10	-10	16780,1	17942,27

Tabelle 6.3: Optimale Designs bei Betrachtung der addierten FRFs ohne Restriktionen

Anschließend soll eine Optimierung einer der ausgewählten FRFs betrachtet werden. Dies wird am Beispiel 'Out11093\_777063' durchgeführt. Abbildung 6.6 zeigt die dabei entstehenden Graphen.

Abbildung 6.6: Algorithmen für eine FRF mit Verschiebungsrestriktion ( $Betrag \leq 10$ )

Auch hier ist bei Verwendung des Evolutionären Algorithmus auf den ersten Blick lediglich eine geringe Verbesserung ersichtlich, während der NLPQL wieder nach deutlich weniger erstellten Designs ein Optimum findet.

Die Tendenzen sind hierbei in beiden Optimierungsalgorithmen ähnlich und auch die als optimal identifizierten Werte liegen für die Y- und Z-Koordinate sehr nahe beieinander und lediglich in Bezug auf die X-Koordinate ist eine erhebliche Diskrepanz zu beobachten, wie Tabelle 6.4 zeigt. Auch die Güte der Modelle ist mit  $-3,50\%$  (Evolutionär) und  $-3,30\%$  (NLPQL) akzeptabel. Die exemplarische Optimierung des Designs für eine Bewertungsgröße am Anbindungspunkt R1 zeigt somit deutlich, wie viel Potential in der Variation der Kinematikpunkte steckt. Die gemittelte Bewertungsgröße aller 164 simulierten Designs liegt bei  $1493,5$ , sodass das optimale Design, NLPQL gemorht, eine Verbesserung von  $14,76\%$  liefert. Bezüglich des verwendeten Inputdesigns (Verschiebung von null) mit  $1479,97$  wird eine Verbesserung von  $13,98\%$  realisiert.

Algorithmus	Restriktion	R1_X	R1_Y	R1_Z	Bewertung Metamodell	Bewertung Gemorphpht
Evolutionär	10 mm	0,340387	9,78894	1,26209	1242,37	1286,16
NLPQL	10 mm	2,40623	9,57803	1,56234	1231,66	1273,07

Tabelle 6.4: Optimale Designs bei Betrachtung der FRF für *Out11093\_777063*

Abbildung 6.7 stellt die FRF für den Fall *Out11093\_777063*, des optimierten Designs, verglichen mit der des originalen Designs, graphisch dar. Dabei ist klar ersichtlich, dass die optimierte Kurve ein etwas niedrigeres Gesamtniveau aufweist. Gleichzeitig wird das erste, dritte, vierte und letzte Peak auch offensichtlich bezüglich der Frequenz verschoben. In diesem Fall erfolgt dies zu höheren Frequenzen. Im Falle des zweiten Peaks bleibt die Frequenz annähernd die selbe. Interessanterweise weist genau dieser Peak im optimierten Design eine höhere Amplitude auf. Die Darstellung verdeutlicht dabei, dass durch die gezielte Veränderung der Geometrie einzelne Schwingungsmoden in ihrer Ausprägung und Lage im Frequenzbereich beeinflusst und bewusst ausgewählt werden können.

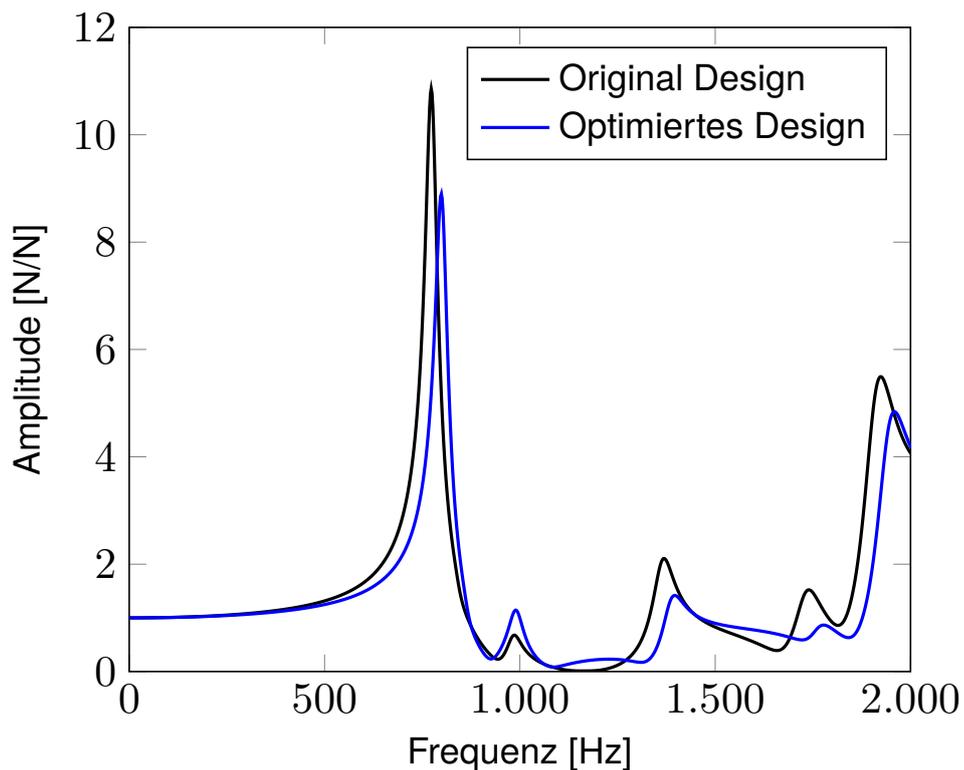


Abbildung 6.7: FRF Vergleich original und optimiertes Design (NLPQL)

Der Vollständigkeit halber seien an dieser Stelle auch die von Optislang berechneten Werte für CoD und CoP für die einzelnen Metamodelle in Tabelle 6.5 dargestellt. Hierbei ist ersichtlich, dass die gefundenen Metamodelle alle

eine sehr gute Beschreibung der Simulationsergebnisse darstellen. Optislang berechnet dabei eine angepasste Version des CoD (siehe<sup>1</sup>)

FRF	CoD angepasst	CoP
Out11091_777061	0,997307	0,997207
Out11091_777062	0,998565	0,998506
Out11091_777063	0,992834	0,992784
Out11092_777061	0,999998	0,99999
Out11092_777062	0,992501	0,992499
Out11092_777063	1	0,999994
Out11093_777061	0,993952	0,993917
Out11093_777062	0,984381	0,984477
Out11093_777063	0,990327	0,990347
Addition aller FRFs	0,992932	0,992647

Tabelle 6.5: CoD und CoP der verwendeten Metamodelle

<sup>1</sup>Roos 2008.

## 7 Zusammenfassung und Ausblick

Zum Abschluss dieser Arbeit wird der Inhalt zusammengefasst, sowie ein Ausblick für weitere Anwendungsbereiche und Entwicklungen der verwendeten Methodik dargelegt.

### 7.1 Zusammenfassung

Um die Gestalt eines Bauteils exemplarisch bezüglich seines Schwingungsverhaltens zu optimieren, wurde mittels der Box-Morphing Methode in ANSA eine Morphingstruktur für den betrachteten Radträger angelegt. Diese ermöglicht eine unabhängige Verschiebung der fünf Anbindungspunkte (Lagersitze). Eine Verschiebung der Anbindungspunkte um einen Vektor der Länge 10 mm ist dabei sichergestellt.

Zur angelegten Morphingstruktur wurde ein Python-Script implementiert, das den Prozess des Morphings komplett automatisiert. Dies beinhaltet sowohl die reine Verschiebung der einzelnen Punkte, als auch eine Behebung von auftretenden Rotationen sowie Verzerrung angrenzender Geometrien, um eine insgesamt möglichst smooth Geometrie zu erhalten. Ein Erhalt der nötigen Netzqualität wird dabei durch ein Mesh-Improving gewährleistet.

Um eine Optimierung des Bauteils durchzuführen, wird ein vollständiger Arbeitsablauf dargelegt.

Dabei wird zunächst mittels eines space-filling Latin-Hypercube Verfahrens eine vorgegebene Anzahl an möglichen Designvariationen erstellt. Diese werden von einem übergeordneten Python-Script eingelesen und einzeln an den Morphingprozess in ANSA übergeben. Das übergeordnete Script startet nach erfolgreichem Morphing eines Designs automatisch die Simulation der FRFs, aufgrund welcher die Optimierung durchgeführt werden soll. Abschließend bündelt das Script die Daten und sortiert die Simulationsergebnisse vor, um im Optimierungsprozess nur noch gewünschte Werte zu betrachten und somit unnötige Inputvariablen zu eliminieren. Die so erzeugten Daten werden in Optislang zur Generierung eines Metamodells eingelesen. Auf Basis dieses Metamodells wird eine Optimierung mittels einem Evolutionären Algorithmus sowie dem NLPQL durchgeführt.

Abschließend wurde der beschriebene Prozess exemplarisch anhand einer Designstudie am Anbindungspunkt R1 mit 164 Designs durchgeführt. Dabei wurden zwei unterschiedliche Optimierungsparameter gewählt und die Ergebnisse der unterschiedlichen Optimierungsalgorithmen verglichen. Die auf Grund des Metamodells als optimal identifizierten Designs wurden einzeln mittels des Morphing-Scriptes erstellt und deren FRFs simuliert. Diese wurden verwendet, um die Aussagen des Metamodells zu validieren.

### 7.2 Ausblick

In der Arbeit wurde gezeigt, dass die Lage der Kinematikpunkte das Schwingungsverhalten gravierend beeinflusst. Box-Morphing bietet eine ideale Möglichkeit dieses Potential an nahezu beliebig komplexen Geometrien

aufzudecken. Wenngleich das dargestellte Tool eine Möglichkeit bietet, einzelne Frequenzbereiche oder ganze Übertragungsfunktionen gezielt zu beeinflussen, sollte jeweils das ganzheitliche System betrachtet werden. Da häufig eine Senkung an einer Stelle mit einer Verstärkung der Amplitude an einer anderen Frequenz einhergeht, ist vor allem diesem Wechselspiel bei zukünftigen Optimierungen besondere Aufmerksamkeit zu widmen.

Dennoch besteht in dem dargelegten Arbeitsablauf ein immenses Potential, gerade wenn es sich um ein Bauteil handelt, welches in ein größeres schwingungsfähiges System eingebunden wird. In diesen Fällen können die Resonanzbereiche gekoppelter Systeme durch eine Verschiebung der Kinematikpunkte bewusst in unterschiedliche Frequenzbereiche verschoben werden.

Im Rahmen dieser Abschlussarbeit wurde ein großer zeitlicher Aufwand in das Anlegen der Morphingstruktur gesteckt. Hierbei sind jedoch immer noch weitere Verbesserungen möglich, sodass nicht nur die Kinematikpunkte verschoben werden und die umgebende Geometrie Ausgleichbewegungen vollführt, sondern bewusste Veränderungen von Geometrieparametern wie zum Beispiel den Krümmungsradien vollzogen werden. Des Weiteren ist eine Untersuchung, ob Morphingstrukturen an komplexen Geometrien sinnvoll automatisiert angelegt werden können, sinnvoll.

Weiterhin bietet das Verfahren des Box-Morphing auch andere Anwendungsmöglichkeiten als die Optimierung von NVH-Phänomenen und ist für Geometrieänderung anwendbar.

## A Anhang

Design Nummer	$R1\_X$ [mm]	$R1\_Y$ [mm]	$R1\_Z$ [mm]
1	-6,033333333	-5,566666667	-1,433333333
2	7,166666667	-1,433333333	-6,5
3	1,766666667	-5,7	4,766666667
4	-7,9	-2,233333333	2,766666667
5	-6,7	3,766666667	5,633333333
6	3,233333333	-7,1	1,7
7	5,966666667	-1,566666667	-3,233333333
8	-2,5	7,833333333	-4,766666667
9	9,5	0,566666667	-1,166666667
10	-3,633333333	5,3	0,166666667
11	6,166666667	0,033333333	-7,833333333
12	-1,9	3,1	7,966666667
13	4,7	3,633333333	2,833333333
14	4,366666667	7,033333333	-1,366666667
15	3,9	2,966666667	-3,9
16	-7,566666667	-4,033333333	4,166666667
17	-1,166666667	1,566666667	9,5
18	-6,366666667	6,766666667	3,633333333
19	1,3	2,166666667	-7,1
20	-2,633333333	4,9	3,033333333
21	-7,1	-1,166666667	-4,5
22	6,566666667	-6,433333333	-3,633333333
23	0,5	-3,033333333	5,366666667
24	-2,833333333	-7,3	-0,233333333
25	2,433333333	3,5	-1,5
26	-4,033333333	7,3	1,633333333
27	-8,9	-2,5	-2,5
28	-5,7	2,1	-0,1
29	-2,233333333	1,9	3,766666667

Tabelle A.1: Designstudie Input Werte Design 1-29

Design Nummer	R1_X [mm]	R1_Y [mm]	R1_Z [mm]
30	-0,233333333	3,166666667	-5,766666667
31	-4,433333333	-4,433333333	0,433333333
32	-1,966666667	-3,633333333	-6,966666667
33	7,1	5,833333333	-2,833333333
34	4,233333333	2,7	-6,633333333
35	2,766666667	-1,633333333	3,233333333
36	1,433333333	-3,5	-1,766666667
37	6,366666667	5,5	2,633333333
38	-1,7	7,233333333	-3,366666667
39	-2,1	6,233333333	-1,7
40	-2,166666667	-5,9	-2,3
41	-2,566666667	-2,1	-9,366666667
42	9,166666667	-3,233333333	-0,033333333
43	-4,9	5,166666667	-0,833333333
44	4,1	-2,166666667	7,166666667
45	0,233333333	-7,566666667	3,966666667
46	0,566666667	6,7	0,5
47	1,833333333	1,366666667	-5,633333333
48	-1,233333333	0,366666667	-7,9
49	7,966666667	-3,7	3,833333333
50	0,766666667	-4,233333333	-4,566666667
51	3,766666667	-2,3	-7,566666667
52	7,5	-0,366666667	2,5
53	-4,633333333	7,7	3,3
54	5,233333333	-2,433333333	4,433333333
55	3,033333333	4,1	0,7
56	-8,766666667	0,433333333	1,3
57	-1,633333333	-6,3	5,566666667
58	6,833333333	-2,833333333	6,033333333
59	2,233333333	-1,3	0,1
60	-3,833333333	3,7	-3,566666667
61	3,366666667	-0,1	-1,833333333
62	-5,033333333	1,7	-0,433333333
63	-0,633333333	-0,033333333	8,5
64	5,833333333	2,633333333	4,566666667
65	0,3	-9,366666667	-2,966666667
66	0,166666667	9,233333333	-3,033333333
67	4,833333333	-5,033333333	-2,766666667
68	4,566666667	-0,9	-8,233333333
69	1,033333333	-2,566666667	8,033333333
70	5,166666667	1,633333333	-4,833333333
71	-6,566666667	4,766666667	-1,9
72	0,9	4,833333333	-4,966666667
73	-3,033333333	-5,1	2,7
74	0,033333333	-5,5	-2,366666667
75	7,566666667	2,233333333	-4,633333333
76	-0,5	3,9	-0,766666667
77	-1,766666667	-1,033333333	-5,433333333
78	-1,566666667	0,966666667	-1,1
79	1,233333333	-9,3	2,9

Tabelle A.2: Designstudie Input Werte Design 30-79

Design Nummer	R1_X [mm]	R1_Y [mm]	R1_Z [mm]
80	-1,1	-8,166666667	1,966666667
81	-6,766666667	-3,433333333	1,9
82	-3,9	-0,5	5,166666667
83	-7,633333333	2,566666667	-2,9
84	-5,566666667	6,033333333	1,766666667
85	-1,033333333	-2,633333333	-4,233333333
86	6,233333333	3,433333333	6,833333333
87	5,566666667	3,3 -2,433333333	
88	-3,766666667	-2,366666667	-0,7
89	-9,7	-1,766666667	-1,233333333
90	-6,433333333	-0,233333333	-2,7
91	8,433333333	-2,033333333	1,1
92	2,366666667	5,9	5,433333333
93	5,5	-0,633333333	1,833333333
94	0,7	8,633333333	2,966666667
95	-1,433333333	-4,166666667	6,766666667
96	8,5	4,633333333	-2,233333333
97	6,7	1,3	0,233333333
98	-2,3	-0,766666667	1,033333333
99	-7,166666667	3,233333333	4,5
100	3,5	-3,366666667	3,7
101	1,9	-3,966666667	1,566666667
102	-0,766666667	5,566666667	-3,833333333
103	1,5	0,633333333	-4,166666667
104	-8,633333333	1,433333333	3,433333333
105	0,366666667	2,833333333	9,366666667
106	3,633333333	0,9	5,5
107	-5,766666667	-2,766666667	-6,233333333
108	-2,7	-1,833333333	4,833333333
109	-4,833333333	-8,033333333	-3,3
110	8,033333333	0,3	-3,433333333
111	9,1	1,5	1,5
112	0,433333333	4,366666667	-6,1
113	-0,033333333	-1,7	-8,766666667
114	-1,3	8,1	4,7
115	-9,3	-0,7	-3,5
116	-0,166666667	-0,966666667	3,5
117	5,9	-4,1	-5,166666667
118	-4,366666667	0,766666667	-6,433333333
119	-5,366666667	0,1	-5,366666667
120	1,566666667	5,366666667	2,3
121	4,633333333	-4,5	-0,3
122	7,033333333	-6,233333333	-0,633333333
123	-0,433333333	-4,9	8,566666667
124	1,166666667	-1,233333333	-0,5
125	3,1	-9,166666667	-0,9
126	-0,3	-6,566666667	1,366666667
127	4,5	6,1	4,033333333
128	-3,233333333	-4,7	-4,1
129	0,833333333	-0,166666667	6,433333333

Tabelle A.3: Designstudie Input Werte Design 80-129

Design Nummer	$R1\_X$ [mm]	$R1\_Y$ [mm]	$R1\_Z$ [mm]
130	3,433333333	-6,7	-4,433333333
131	-4,3	-3,9	-3,166666667
132	-0,566666667	-2,9	0,766666667
133	-0,833333333	6,633333333	5,966666667
134	4,166666667	7,766666667	0,366666667
135	-0,966666667	4,5	1,433333333
136	-0,9	-4,766666667	-1,3
137	6,3	-4,833333333	2,366666667
138	-3,566666667	2,9	-8,3
139	2,9	6,833333333	-3,7
140	0,1	1,166666667	4,966666667
141	2,033333333		4,166666667 6,166666667
142	0,966666667	-6,1	0,3
143	2,166666667	9,5	-2,1
144	1,7	2,033333333	7,1
145	6,966666667	1,766666667	5,233333333
146	7,9	-2,7	-1,966666667
147	-9,1	2,766666667	2,166666667
148	2,7	-5,233333333	-6,7
149	-2,966666667	0,833333333	-2,166666667
150	3,966666667	0,166666667	2,1
151	4,966666667	-3,1	-5,033333333
152	-3,166666667	4,433333333	6,633333333
153	-5,433333333	2,3	5,833333333
154	2,833333333	1,966666667	-3,1
155	-4,1	0,233333333	4,1
156	-5,3	-5,366666667	2,233333333
157	-4,966666667	-1,5	6,366666667
158	-3,3	2,5	0,966666667
159	10	0	0
160	-10	0	0
161	0	10	0
162	0	-10	0
163	0	0	10
164	0	0	-10

Tabelle A.4: Designstudie Input Werte Design 130-164

## B Abbildungsverzeichnis

1	Titelbild . . . . .	i
2.1	Schematischer evolutionärer Zyklus <sup>1</sup> . . . . .	4
3.1	FFD (Beschriftung im Bild angepasst) <sup>2</sup> . . . . .	12
3.2	Veranschaulichung Bauteilstetigkeit in FFD <sup>3</sup> . . . . .	14
3.3	Veranschaulichung Bauteilstetigkeit lokales Morphing in FFD <sup>4</sup> . . . . .	14
3.4	a) FFD mit gerader Linie in Morphingbox b) FFD mit ebener Fläche <sup>5</sup> . . . . .	15
3.5	DM-FFD; H („handle vertices“), F („fixed vertices“), D („deformable vertices“) <sup>6</sup> . . . . .	16
3.6	Einfluss Basisfunktion; links biharmonisch, rechts triharmonisch; goldene Teile werden verschoben, graue konstant gehalten und blaue deformiert <sup>7</sup> . . . . .	17
3.7	Rbfd; H („handle vertices“), F („fixed vertices“), D („deformable vertices“) (Beschriftung im Bild angepasst) <sup>8</sup> . . . . .	19
3.8	FFD mit Stetigkeitsproblem durch partielle Kontrollvolumen. Links: Original, Mitte: Nicht smoother Übergang, Rechts: Smoother Übergang <sup>9</sup> . . . . .	20
3.9	Matrixexperiment $L_9$ <sup>10</sup> . . . . .	22
3.10	Beispiel für Faktor-Wechselwirkungen <sup>11</sup> . . . . .	23
3.11	LHD (blau), LHS (schwarz) mit $n_r = 10, n_f = 2$ <sup>12</sup> . . . . .	24
3.12	LHS mit 20 Testpunkten <sup>13</sup> . . . . .	25
3.13	a) Collapsing Versuchsplan, b) Non-Collapsing Versuchsplan <sup>14</sup> . . . . .	26
4.1	Veranschaulichung Nested-Element (Links: Rigid, Rechts: Frozen) . . . . .	28
4.2	Übersichtsbild Radträger mit Benennungen . . . . .	29
4.3	Hexablock Morphing . . . . .	30
4.4	Puffergebiet R4 und R5 . . . . .	32
4.5	Mainfunktion Morphing . . . . .	34
4.6	R1 Morphingstruktur dargestellt an der original Geometrie . . . . .	35
4.7	Ablauf Morphing . . . . .	37
4.8	Vergleich reine Ebenenverschiebung und Kombination aus Ebenen-, Boxverschiebung bei Elementen in mehreren Boxen . . . . .	37
4.9	Skizze zur Anteilberechnung von Verschiebungen für eine Gerade über mehrere Morphingboxen hinweg, um die Gerade zu erhalten . . . . .	38
4.10	R3 Y-Verschiebung . . . . .	39
4.11	Rotation durch Ebenenverschiebung; Links vor Morphing, Rechts nach Morphing . . . . .	40
4.12	Ausgangszustand Morphing R1, R2 . . . . .	43

---

4.13 Morphing von R1 (links) , R2 (rechts) um $y = 10mm$ und $y = -10mm$ verglichen mit und ohne Smoothing als Ansicht von unten in XY-Ebene . . . . .	44
4.14 Morphingstruktur und Übersichtsbild R2 . . . . .	45
4.15 Veranschaulichung Nummerierung der Kontrollpunkte R2 . . . . .	46
4.16 Morphing R2 um $y=10mm$ . . . . .	46
5.1 Ablaufplan Designvariation Optislang <sup>15</sup> . . . . .	49
5.2 Flussdiagramm Gesamtprozess . . . . .	51
5.3 Fixierungen Radträger . . . . .	54
5.4 Systemaufbau Optimierung Optislang <sup>16</sup> . . . . .	55
6.1 CoP Matrix für Metamodell mit neun betrachteten Subcases . . . . .	58
6.2 Algorithmen für zusammengefasste FRFs mit Verschiebungsrestriktion ( $Betrag \leq 10$ ) . . . . .	59
6.3 Plot des Metamodels für die Summe der betrachteten Übertragungsfunktionen über der Verschiebung von R1 in X- und Y-Richtung aufgetragen . . . . .	60
6.4 CoP Matrix für die Summe der betrachteten Übertragungsfunktionen . . . . .	61
6.5 Algorithmen für zusammengefasste FRFs ohne Verschiebungsrestriktion . . . . .	61
6.6 Algorithmen für eine FRF mit Verschiebungsrestriktion ( $Betrag \leq 10$ ) . . . . .	62
6.7 FRF Vergleich original und optimiertes Design (NLPQL) . . . . .	63

## C Tabellenverzeichnis

1	Lateinische Symbole . . . . .	ix
1	Lateinische Symbole – Fortsetzung . . . . .	x
1	Lateinische Symbole – Fortsetzung . . . . .	xi
2	Griechische Symbole . . . . .	xi
3.1	Radial Basis Function Deformation Ansatzfunktionen <sup>1</sup> . . . . .	17
3.2	Vergleich Morphingmethoden <sup>2</sup> . . . . .	20
3.3	Vollfaktorplan mit drei Faktoren (A, B, C) und zwei Stufen gekennzeichnet mit – + (nach Siebertz, van Bebber et al.) <sup>3</sup> . . . . .	21
3.4	Links: Orthogonaler Versuchsplan, Rechts: LHS auf Basis des orthogonalen Versuchsplans <sup>4</sup> . . . . .	25
4.1	Netzqualitätskriterien . . . . .	47
6.1	Bewertungskriterien Abweichungen Soll-, Ist-Verschiebung, gerundet auf vierte Nachkommastelle, bezogen auf die Potenz . . . . .	57
6.2	Optimale Designs bei Betrachtung der addierten FRFs mit Restriktion . . . . .	60
6.3	Optimale Designs bei Betrachtung der addierten FRFs ohne Restriktionen . . . . .	62
6.4	Optimale Designs bei Betrachtung der FRF für <i>Out11093_777063</i> . . . . .	63
6.5	CoD und CoP der verwendeten Metamodelle . . . . .	64
A.1	Designstudie Input Werte Design 1-29 . . . . .	67
A.2	Designstudie Input Werte Design 30-79 . . . . .	68
A.3	Designstudie Input Werte Design 80-129 . . . . .	69
A.4	Designstudie Input Werte Design 130-164 . . . . .	70

# Quellenverzeichnis

## Literaturverzeichnis

**BETA CAE Systems 2017** BETA CAE Systems (2017). *ANSA version 18.0x User's Guide*.

**BETA CAE Systems 2019** – (2019). *Documentation Index*.

**Boer, van der Schoot et al. 2007** A. de Boer, M. S. van der Schoot und H. Bijl (2007). „Mesh deformation based on radial basis function interpolation“. In: *Computers & Structures* 85.11-14, S. 784–795. ISSN: 00457949. DOI: 10.1016/j.compstruc.2007.01.013.

**Brandt 2013** Siegmund Brandt (2013). *Datenanalyse für Naturwissenschaftler und Ingenieure*. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN: 978-3-642-37663-4. DOI: 10.1007/978-3-642-37664-1.

**Brune, Weber Martins et al. 2018** Alexander Brune, Thiago Weber Martins und Reiner Anderl (2018). „Morphing boxes for the integration of shape optimization in the product design process“. In: *Computer-Aided Design and Applications* 15.2, S. 219–226. DOI: 10.1080/16864360.2017.1375672.

**Coquillart 1990** Sabine Coquillart (1990). „Extended free-form deformation: a sculpturing tool for 3D geometric modeling“. In: *Proceedings of the 17th annual conference on Computer graphics and interactive techniques - SIGGRAPH '90*. Hrsg. Forest Baskett. New York, New York, USA: ACM Press, S. 187–196. ISBN: 0201509334. DOI: 10.1145/97879.97900.

**Dean, Voss et al. 2017** Angela Dean, Daniel Voss und Danel Draguljić (2017). *Design and Analysis of Experiments*. Second edition. Springer Texts in Statistics. Cham: Springer. ISBN: 978-3-319-52248-7. DOI: 10.1007/978-3-319-52250-0.

**Dynardo GmbH 2019** Dynardo GmbH (2019). *Methods for multi-disciplinary optimization and robustness analysis*.

**Ferziger und Peric 2008** Joel H. Ferziger und Milovan Peric (2008). *Numerische Strömungsmechanik*. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN: 978-3-540-67586-0. DOI: 10.1007/978-3-540-68228-8.

**Furlan 2012** Peter Furlan (2012). *Lineare Algebra, Differentialrechnung*. [7.Nachdr.] Bd. Rechenverfahren der Höheren Mathematik in Einzelschritten erklärt; mit vielen ausführlich gerechneten Beispielen / Peter Furlan ; 1. Das gelbe Rechenbuch für Ingenieure, Naturwissenschaftler und Mathematiker. Dortmund: Furlan. ISBN: 978-3-931645-00-7.

**Gerds und Lempio 2011** Matthias Gerds und Frank Lempio (2011). *Mathematische Optimierungsverfahren des Operations Research*. De Gruyter Studium. Berlin: De Gruyter. ISBN: 978-3-11-024998-9. DOI: 10.1515/9783110249989.

**Hradetzky 1978** J. Hradetzky (1978). „Das Bestimmtheitsmaß“. In: *Forstwissenschaftliches Centralblatt* 97.1, S. 168–181. ISSN: 0015-8003. DOI: 10.1007/BF02741104.

- Hsu, Hughes et al. 1992** William M. Hsu, John F. Hughes und Henry Kaufman (1992). „Direct manipulation of free-form deformations“. In: *Proceedings of the 19th annual conference on Computer graphics and interactive techniques - SIGGRAPH '92*. Hrsg. James J. Thomas. New York, New York, USA: ACM Press, S. 177–184. ISBN: 0897914791. doi: 10.1145/133994.134036.
- Jackèl, Neunreither et al. 2006** Dietmar Jackèl, Stephan Neunreither und Friedrich Wagner (2006). *Methoden der Computeranimation*. eXamen.press. Berlin: Springer. ISBN: 3-540-26114-1. doi: 10.1007/3-540-33407-6.
- Jakobsson und Amoignon 2007** Stefan Jakobsson und Olivier Amoignon (2007). „Mesh deformation using radial basis functions for gradient-based aerodynamic shape optimization“. In: *Computers & Fluids* 36.6, S. 1119–1136. ISSN: 0045-7930. doi: 10.1016/j.compuid.2006.11.002. (Abgerufen am 30.07.2020).
- Klein 2014** Bernd Klein (2014). *Versuchsplanung - DoE: Einführung in die Taguchi/Shainin-Methodik*. Berlin. doi: 10.1524/9783110343847.
- Kohn und Öztürk 2013** Wolfgang Kohn und Riza Öztürk (2013). *Statistik für Ökonomen*. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN: 978-3-642-37351-0. doi: 10.1007/978-3-642-37352-7.
- Küppers 2015** Thomas Küppers (2015). *Results of a structured development process for electric vehicle target sounds*. doi: 10.13140/RG.2.1.5038.4728.
- Lupini und Epureanu 2019** Andrea Lupini und Bogdan I. Epureanu (2019). „On the use of mesh morphing techniques in reduced order models for the structural dynamics of geometrically mistuned blisks“. In: *Mechanical Systems and Signal Processing* 127, S. 262–275. ISSN: 08883270. doi: 10.1016/j.ymsp.2019.03.004.
- Manutchehr-Danai 2009** Mohsen Manutchehr-Danai (2009). *Dictionary of Gems and Gemology*. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN: 978-3-540-72816-0. doi: 10.1007/978-3-540-72816-0.
- Martin 2010** Pflüger Martin (2010). *Fahrzeugakustik*. Der Fahrzeugantrieb. Vienna: Springer-Verlag Vienna. ISBN: 978-3-211-76741-2. doi: 10.1007/978-3-211-76741-2.
- Menzel, Olhofer et al. 2006** Stefan Menzel, Markus Olhofer und Bernhard Sendhoff (2006). „Direct Manipulation of Free Form Deformation in Evolutionary Design Optimisation“. In: *Parallel Problem Solving from Nature - PPSN IX*. Hrsg. David Hutchison, Takeo Kanade et al. Bd. 4193. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, S. 352–361. ISBN: 978-3-540-38990-3. doi: 10.1007/11844297\_36.
- Michler 2011** Andreas K. Michler (2011). „Aircraft control surface deflection using RBF-based mesh deformation“. In: *International Journal for Numerical Methods in Engineering* 88.10, S. 986–1007. ISSN: 0029-5981. doi: 10.1002/nme.3208.
- Montgomery und Runger 2003** Douglas C. Montgomery und George C. Runger (2003). *Applied statistics and probability for engineers*. 3rd ed. New York und Chichester: Wiley. ISBN: 0-471-20454-4.
- Most und Will 2008** Thomas Most und Johannes Will (2008). „Metamodel of Optimal Prognosis-an automatic approach for variable reduction and optimal metamodel selection“. In: *Proc. Weimarer Optimierungs-und Stochastiktag* 5, S. 20–21. URL: [https://www.dynardo.de/fileadmin/Material\\_Dynardo/WOST/Paper/wost5.0/Paper\\_Most.pdf](https://www.dynardo.de/fileadmin/Material_Dynardo/WOST/Paper/wost5.0/Paper_Most.pdf) (abgerufen am 30.07.2020).
- Most und Will 2011** – (2011). „Sensitivity analysis using the Metamodel of Optimal Prognosis“. In: *Weimar Optimization and Stochastic Days* 8, S. 24–40. URL: [https://www.dynardo.de/fileadmin/Material\\_Dynardo/bibliothek/WOST\\_8.0/Paper\\_Most.pdf](https://www.dynardo.de/fileadmin/Material_Dynardo/bibliothek/WOST_8.0/Paper_Most.pdf) (abgerufen am 30.07.2020).

- Norbert K. 2000** Norbert K. (2000). *Neue Ansätze für die zuverlässigkeitsorientierte Strukturoptimierung unter Beachtung zeitvarianter Aspekte*. URL: <https://www.osti.gov/etdeweb/servlets/purl/20160562> (abgerufen am 30.07.2020).
- Ranjan und Spencer 2014** Pritam Ranjan und Neil Spencer (2014). „Space-filling Latin hypercube designs based on randomization restrictions in factorial experiments“. In: *Statistics & Probability Letters* 94, S. 239–247. ISSN: 01677152. DOI: 10.1016/j.spl.2014.07.032.
- Rick, Franck et al. 2015** Sebastian Rick, David Franck und Kay Hameyer (2015). „Hybrid NVH Simulation for Electrical Vehicles I-Force Excitation Model for Electrical Machines“. In: *Fortschritte der Akustik: DAGA*. URL: [http://pub.dega-akustik.de/DAGA\\_2015/data/articles/000346.pdf](http://pub.dega-akustik.de/DAGA_2015/data/articles/000346.pdf) (abgerufen am 30.07.2020).
- Roos 2008** Dirk Roos (2008). „Advanced methods of stochastic and optimization in industrial applications“. URL: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.411.4199&rep=rep1&type=pdf> (abgerufen am 30.07.2020).
- Salmoiraghi, Scardigli et al. 2018** F. Salmoiraghi, A. Scardigli, H. Telib und G. Rozza (2018). *Free Form Deformation, mesh morphing and reduced order methods: enablers for efficient aerodynamic shape optimization*. URL: <http://arxiv.org/pdf/1803.04688v2> (abgerufen am 30.07.2020).
- Schiefer und Schiefer 2018** Hartmut Schiefer und Felix Schiefer (2018). *Statistik für Ingenieure*. Wiesbaden: Springer Fachmedien Wiesbaden. ISBN: 978-3-658-20639-0. DOI: 10.1007/978-3-658-20640-6.
- Schittkowski 1986** K. Schittkowski (1986). „NLPQL: A fortran subroutine solving constrained nonlinear programming problems“. In: *Annals of Operations Research*. Bd. 5, S. 485–500. DOI: 10.1080/02331938308842847.
- Sederberg und Parry 1986** Thomas W. Sederberg und Scott R. Parry (1986). „Free-form deformation of solid geometric models“. In: *Proceedings of the 13th annual conference on Computer graphics and interactive techniques - SIGGRAPH '86*. Hrsg. David C. Evans und Russell J. Athay. New York, New York, USA: ACM Press, S. 151–160. ISBN: 0897911962. DOI: 10.1145/15922.15903.
- Siebenpfeiffer 2019** Wolfgang Siebenpfeiffer (2019). *Automotive Acoustics Conference 2015*. Wiesbaden: Springer Fachmedien Wiesbaden. ISBN: 978-3-658-27647-8. DOI: 10.1007/978-3-658-27648-5.
- Siebertz, van Bebber et al. 2017** Karl Siebertz, David van Bebber und Thomas Hochkirchen (2017). *Statistische Versuchsplanung: Design of Experiments (DoE)*. 2nd ed. VDI-Buch Ser. Berlin, Heidelberg: Vieweg. ISBN: 978-3-662-55742-6. DOI: 10.1007/978-3-662-55743-3.
- Sieger, Menzel et al. 2014** Daniel Sieger, Stefan Menzel und Mario Botsch (2014). „RBF morphing techniques for simulation-based design optimization“. In: *Engineering with Computers* 30.2, S. 161–174. ISSN: 0177-0667. DOI: 10.1007/s00366-013-0330-1.
- Sieger, Menzel et al. 2015** – (2015). „On Shape Deformation Techniques for Simulation-Based Design Optimization“. In: *New Challenges in Grid Generation and Adaptivity for Scientific Computing*. Hrsg. Simona Perotto und Luca Formaggia. Bd. 5. SEMA SIMAI Springer Series. Cham: Springer International Publishing, S. 281–303. ISBN: 978-3-319-06052-1. DOI: 10.1007/978-3-319-06053-8\_14.
- Siemens 2017** Siemens (2017). *NX Nastran 12 Quick Reference Guide*.

- van der Auweraer, van Langenhove et al. 2007** H. van der Auweraer, T. van Langenhove et al. (2007). „Application of mesh morphing technology in the concept phase of vehicle development“. In: *International Journal of Vehicle Design* 43.1/2/3/4, S. 281. ISSN: 0143-3369. DOI: 10.1504/IJVD.2007.012308.
- Wagner 2017** Marcus Wagner (2017). *Lineare und nichtlineare FEM: Eine Einführung mit Anwendungen in der Umformsimulation mit LS-DYNA®*. Lehrbuch. Wiesbaden: Springer Vieweg. ISBN: 978-3-658-17865-9. DOI: 10.1007/978-3-658-17866-6.
- Weicker 2015** Karsten Weicker (2015). *Evolutionäre Algorithmen*. Wiesbaden: Springer Fachmedien Wiesbaden. ISBN: 978-3-658-09957-2. DOI: 10.1007/978-3-658-09958-9.
- Will 2005** Johannes Will (2005). *Kombination von Optimierung und Robustheitsbewertung aus praktischer Sicht*. URL: [https://www.dynardo.de/fileadmin/Material\\_Dynardo/WOST/Library/09\\_NAFEMS\\_will\\_2005\\_deutsch.pdf](https://www.dynardo.de/fileadmin/Material_Dynardo/WOST/Library/09_NAFEMS_will_2005_deutsch.pdf) (abgerufen am 30.07.2020).
- Winer, Bergles et al. 2011** Ward O. Winer, Arthur E. Bergles et al. (2011). *The Automotive Body*. Bd. 0. Dordrecht: Springer Netherlands. ISBN: 978-94-007-0515-9. DOI: 10.1007/978-94-007-0516-6.
- Wysocki 2020a** Timo von Wysocki (2020a). *Optimierung Ablaufplan*. Unter Mitarb. von Michael Leupolz. *Wissenschaftliches Gespräch*.
- Wysocki 2020b** – (2020b). *Senitivitätsanalyse*. Unter Mitarb. von Michael Leupolz. *Wissenschaftliches Gespräch*.
- Zwicker und Fastl 1999** Eberhard Zwicker und Hugo Fastl (1999). *Psychoacoustics: Facts and Models*. Second Updated Edition. Bd. 22. Springer Series in Information Sciences. Berlin und Heidelberg: Springer. ISBN: 978-3-540-65063-8. DOI: 10.1007/978-3-662-09562-1.