

Optimierung von Betriebsstrategien für elektrifizierte Antriebskonzepte

Dissertation zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften (Dr.-Ing.)

Vorgelegt im Fachbereich Elektrotechnik
der Universität Kassel

von Dipl.-Ing. Kostyantyn Bass

Kassel, März 2015

Die Ergebnisse, Meinungen und Schlüsse dieser Dissertation sind nicht notwendigerweise die der Volkswagen AG.

1. Gutachter:	Prof. Dr. rer. nat. Ludwig Brabetz
2. Gutachter:	Prof. Dr.-Ing. Michael Fister
Dissertation eingereicht am:	03.03.2015
Tag der Disputation:	16.10.2015

Vorwort

Die vorliegende Arbeit entstand während meiner Tätigkeit als Doktorand in der technischen Entwicklung der Volkswagen AG in Wolfsburg in den Jahren 2011 bis 2014.

Mein besonderer Dank gilt Herrn Prof. Dr. rer. nat. L. Brabetz für das entgegengebrachte Vertrauen und die angenehme und konstruktive Zusammenarbeit. Die zahlreichen fachlichen Diskussionen haben entscheidend zum Erfolg dieser Arbeit beigetragen.

Herrn Prof. Dr.-Ing. M. Fister danke ich für sein Interesse an meiner Arbeit, seine hilfreichen Hinweise und die Übernahme des Zweitgutachtens.

Herrn Dr.-Ing. M. Ayeb möchte ich für seine hilfreiche Unterstützung und seine wertvollen Hinweise und Anregungen meinen Dank aussprechen. Mit seinem umfangreichem Fachwissen und mit einem großem Interesse hat er meine Arbeit unterstützt und gefördert.

Für die freundliche und angenehme Zusammenarbeit möchte ich mich bei meinen Kollegen bedanken. Besonders erwähnen möchte ich den Herrn Dipl.-Ing. L. Fredersdorff, der mich über die gesamte Dauer des Promotionsvorhabens mit zahlreichen fachlichen Diskussionen und Anregungen unterstützt hat. Dem Dr.-Ing. B. Klages und dem Dipl.-Ing. D. Gundel danke ich besonders für das entgegengebrachte Vertrauen, das die Anfertigung der Arbeit ermöglicht hat und für die kompetenten und hilfreichen Hinweise bei der Veröffentlichung der Arbeitsergebnisse. Den Herrn Dr.-Ing. H. Klar, Dr.-Ing. J. Schmidgal, Dr.-Ing. H. Haberland, Dipl.-Ing. N. Zimbalist und Dipl.-Ing. N.-O. Ulrich danke ich für die vielen wertvollen Ratschläge, Ihre großartige Unterstützung und eine hervorragende Arbeitsatmosphäre.

Mein größter Dank gilt meiner Frau Anush und meiner Tochter Maria, die viel Verständnis, Geduld und Unterstützung während der Anfertigung der Arbeit geleistet haben. Meinem Bruder Yuriy danke ich für die vielen fachlichen Diskussionen und die moralische Unterstützung während des gesamten Promotionsvorhabens. Einen ganz besonderen Dank möchte ich meinen Eltern Nadiya und Borys aussprechen. Ihre Erziehung und Ihre unermüdliche Unterstützung während meiner Kindheit, Jugend und Studium haben diese Arbeit erst möglich gemacht.

Eidesstattliche Erklärung

Hiermit versichere ich, dass ich die vorliegende Dissertation selbstständig, ohne unerlaubte Hilfe Dritter angefertigt und andere als die in der Dissertation angegebenen Hilfsmittel nicht benutzt habe.

Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder unveröffentlichten Schriften entnommen sind, habe ich als solche kenntlich gemacht.

Dritte waren an der inhaltlich-materiellen Erstellung der Dissertation nicht beteiligt; insbesondere habe ich hierfür nicht die Hilfe eines Promotionsberaters in Anspruch genommen.

Kein Teil dieser Arbeit ist in einem anderen Promotions- oder Habilitationsverfahren verwendet worden.

Kassel, den 03.03.2015

Kostyantyn Bass

Inhaltsverzeichnis

Vorwort	III
Eidesstattliche Erklärung	V
Inhaltsverzeichnis	VIII
Symbolverzeichnis	IX
Abbildungsverzeichnis	XI
Tabellenverzeichnis	XIII
1 Einleitung	1
1.1 Motivation	1
1.2 Stand der Technik	2
1.3 Ziele der Arbeit	3
1.4 Aufbau der Arbeit	5
2 Erstellung eines Werkzeugs zur Simulation von Steuergerätefunktionen	7
2.1 Motivation	7
2.2 Optimierung der Betriebsstrategien von Hybrid-Fahrzeugen	9
2.2.1 Gesamtfahrzeugmodell	9
2.2.2 Optimierung der Betriebsstrategieparameter	10
2.3 Werkzeuge der Funktionsentwicklung und der Optimierung	12
2.4 Implementierung	12
2.4.1 Ziele und Anforderungen	12
2.4.2 Ansatz	13
2.4.3 Funktionale Ebene	14
2.5 Verifikation	15
2.5.1 Simulationsergebnisse	15
2.5.2 Eigenschaften und Einsatzmöglichkeiten	16
3 Modellierung des untersuchten Verbrennungsmotors	19
3.1 Motivation	19
3.2 Versuchsplanung	21
3.2.1 Bestimmung der Verstellgrenzen	23
3.2.2 Erzeugung der Kandidaten	24
3.2.3 Versuchsplanung	28
3.2.4 Vermessung	30
3.3 Modellierung des Verbrennungsmotorverhaltens	32
3.3.1 Modellierungsverfahren	32
3.3.2 Modellierungsansatz	40
3.3.3 Reallisierung	42
3.4 Modellierungsbeispiel	44
3.4.1 Versuchsträger	44
3.4.2 Modellierungsergebnisse	45

4	Optimierungswerkzeuge	49
4.1	Motivation	49
4.1.1	Basisoptimierung der Steuerungs- und Regelungsfunktionen	49
4.1.2	Änderungen im Simulationsmodell	50
4.2	Gütevektroptimierung	51
4.3	Entwicklung des Vermessungsverfahrens <i>DelaunaySearch</i>	54
4.3.1	Motivation und Ziele	54
4.3.2	Funktionsweise	55
4.3.3	Diskussion	58
5	Optimierung der Betriebsstrategie eines Hybridfahrzeugs	63
5.1	Motivation und Ziele	63
5.2	Implementierung	66
5.2.1	Ausgangszustand	66
5.2.2	Funktionsentwurf	68
5.3	Optimierung	71
5.3.1	Einflussparameter und Gütekriterien	71
5.3.2	Strategie	76
5.3.3	Ergebnisse	77
6	Zusammenfassung und Ausblick	89
6.1	Zusammenfassung	89
6.2	Ausblick	91
A	Anhang	93
A.1	Simulation von Steuergerätefunktionen, implementierte Funktionsumfänge	93
A.2	Relativer Fehler	97
A.3	Normierte Summe der Quadratabweichungen	98
A.4	Mittlere Summe der Quadratabweichungen	98
A.5	Bestimmtheitsmaß	98
A.6	Standardabweichung	98
A.7	Gaußverteilung	99
A.8	Delaunay-Triangulation	99
	Literaturverzeichnis	101

Symbolverzeichnis

Abkürzungen

M_{Mot}	Motordrehmoment
n_{Mot}	Motordrehzahl
HC	Kohlenwasserstoffe
NO_x	Stickstoffoxide
AGR	Abgasrückführrate
DoE	Design of Experiments
HD-AGR	Hochdruck-Abgasrückführrate
ND-AGR	Niederdruck-Abgasrückführrate
VTG	Variable Turbinen Geometrie

Formelzeichen

\mathbf{x}	$[x_1 \ x_2 \ \dots \ x_N]^T$
\mathbf{y}	$[y_1 \ y_2 \ \dots \ y_M]^T$
D	Datensatz $\{\mathbf{x}_i, y_i\}$ mit $i = 1 \dots M$
K	Kandidatensatz
L	Anzahl der Testpunkte im Kandidatensatz
M	Anzahl der Testpunkte im Versuchsplan
N	Dimension des Eingangsraums
T	Testplan, Versuchsplan
w	Modellkoeffiziente
X	Designmatrix, Versuchsmatrix
x	Einflussgröße
y	Zielgröße

Abbildungsverzeichnis

2.1	Anpassung des Applikationsprozesses durch die Weiterentwicklung der Simulationswerkzeuge	8
2.2	Aufbau des Simulationswerkzeugs zur Simulation des längsdynamischen Fahrzeugverhaltens	10
2.3	Beispielhafter Aufbau des Simulationblocks des Verbrennungsmotors in der Simulationsumgebung der Fahrzeuglängsdynamik	11
2.4	Grundschrirte zum Transfer eines Steuergeräte–Funktionsmoduls in die Simulink Umgebung	14
2.5	Schematische Darstellung des Übertragungsprozesses von Ascet nach Simulink	15
2.6	Test des implementierten Simulationswerkzeugs. Direkter Vergleich einiger Größen des Luftpfads	17
2.7	Einsatzmöglichkeiten des implementierten Simulationswerkzeugs	18
3.1	Ansatz zur Implementierung des Motormodells in der Simulationsumgebung	20
3.2	Die wichtigsten Schritte der Versuchsplanung für die Vermessung eines modernen Verbrennungsmotors	22
3.3	Änderung der Verstellgrenzen der Zylinderfüllung in Abhängigkeit von der berechneten Abgabeleistung an der Kurbelwelle	24
3.4	Vergleich der orthogonalen und der radialen Rasterung am Beispiel eines 2 – D Versuchsplans	26
3.5	Reduktion des Einflusses des Messrauschens bei Anwendung der DoE-Methoden	31
3.6	Beispielhafter Aufbau eines MLP-Netzes	34
3.7	SSE-Fehler der Trainings- und Validierungsdaten in Abhängigkeit von der Neuronenzahl	36
3.8	Beispiel: <i>prior</i> und <i>posterior</i> Verteilung für den 2 – D Fall	39
3.9	Beispiel: <i>posterior</i> Verteilung für den 2 – D Fall	40
3.10	Beispiel: <i>posterior</i> Verteilung mit optimierten Hyperparametern	41
3.11	Prinzipiskizze eines Wiener- und eines Hammerstein-Modells	42
3.12	Überblick über die erstellte Modellstruktur des Verbrennungsmotors	44
3.13	Modellierungsergebnisse des Ladedruckmodells	46
3.14	Modellierungsergebnisse der Temperatur vor Turbine	47
3.15	Modellierungsergebnisse der Stickstoffoxide NO_x	47
4.1	Darstellung der Kostenfunktion innerhalb der Gütevektroptimierung	54
5.1	Simulation einer Fahrt des NEFZ-Zyklus mit verschiedenen Einstellungen der Betriebsstrategie	65
5.2	Darstellung der gemessenen NO_x und Ruß-Emissionen im gesamten Betriebsraum des Verbrennungsmotors	66
5.3	Darstellung der Softwarestruktur der Hybrid-Betriebsstrategie und der Dieselmotorsteuerung	67
5.4	Anforderungsprofile des Verbrennungsmotors in einem NEFZ-Zyklus	69
5.5	Einbindung des Funktionsentwurfes in die Softwarestruktur der Hybrid-Betriebsstrategie und der Dieselmotorsteuerung	70
5.6	Schwellwerte der Betriebsmodi der Verbrennungskraftmaschine aufgetragen über den Ladezustand der Traktionsbatterie	70

5.7	Einflussparameter der Verbrennungsmotorsteuerung	73
5.8	Variationsgrößen des Einspritzsystems	74
5.9	Graphische Darstellung der verschiedenen Betriebsmodi des Verbrennungsmotors . . .	81
5.10	Vergleich des Fahrzeugverhaltens bei verschiedenen Optimierungsansätzen	86
5.11	Unterschiede des Fahrzeugverhaltens für die Optimierungsansätze 1 und 2	87
A.1	Überblick über den aus Ascet erzeugten C-Code	94
A.2	Darstellung der Delaunay-Triangulation im 2 – <i>D</i> -Fall	100

Tabellenverzeichnis

3.1	Technische Daten des modellierten Dieselmotors	45
4.2	Zusammenstellung der Einflussparameter und der Sollkennfelder des modellierten Dieselmotors	59
4.3	Zielgrößen und Randbedingungen der Optimierung	60
5.1	Einflussgrößen der Hybrid-Betriebsstrategie und der Steuerung des Verbrennungsmotors bei der Optimierung.	72
5.2	Zusammenstellung der verwendeten Gütekriterien	76
5.3	Ausgangszustand vor der Optimierung der Hybrid-Betriebsstrategie und des Verbrennungsmotorverhaltens	77
5.4	Optimierungsergebnisse Ansatz 1: Optimierung des Ausgangszustands	78
5.5	Optimierungsziele bei der Berechnung der Betriebsmodi des Verbrennungsmotors . . .	80
5.6	Optimierungsergebnisse der Betriebsmodi des Verbrennungsmotors	80
5.7	Optimierungsergebnisse Ansatz 2: Verifikation des Funktionsentwurfs	81
5.8	Optimierungsergebnisse Ansatz 3: gesamtheitliche Optimierung	82
5.9	Optimierungsergebnisse Ansatz 4: vertiefte gesamtheitliche Optimierung	83
5.10	Aufteilung der für die Optimierungsrechnungen benötigten Simulationen	84

1 Einleitung

1.1 Motivation

Der Wettbewerbsdruck in der Automobilindustrie führt zusammen mit den Anforderungen des Gesetzgebers zu stetig steigender Komplexität und Vielfältigkeit der automobilen Antriebsstränge. Die Fahrzeughersteller sind seit einigen Jahren bestrebt auf Basis von gleichen Antriebskomponenten immer mehr unterschiedliche Fahrzeugkonzepte darzustellen. Hierbei spielt nicht nur die Existenz der verschiedenen Antriebstopologien eine entscheidende Rolle. Die verschiedenen Aggregate erfahren aufgrund von ständiger Weiterentwicklung eine starke Zunahme in der Komplexität.

Auf der Seite der Steuergerätefunktionen, die für die Steuerung und Regelung der Antriebstechnik zuständig sind, kann ein entsprechender Umfangsanstieg beobachtet werden. Mit der steigenden Komplexität des Antriebsstranges wächst somit auch der Applikationsaufwand der Steuergerätefunktionen überproportional an. Die Applikation der einzelnen Funktionsmodule und deren Zusammenspiel innerhalb eines Antriebsstranges spielt hierbei eine zentrale Rolle. Besonders bei der Betrachtung eines hybriden Antriebsstrangs, mit einem Dieselmotor als primärer Antriebseinheit, eröffnen sich neue Freiheitsgrade bei der Optimierung der Zielgrößen wie Verbrauch, Emissionen und Fahrleistung. Ein Weg den steigenden Zeitaufwand zur Applikation zu verringern, ist die Verlagerung von Applikationsarbeiten in die Simulation.

Für die Applikation der Steuergeräteparameter von konventionellen Antrieben arbeitet Volkswagen mit einer etablierten und effizienten Werkzeugkette. Die Weiterentwicklung dieser Werkzeugkette soll von der Funktionsentwicklung bis zur Steuergeräteapplikation eine simulationsgestützte Parameteroptimierung eines diesel-hybriden Antriebsstrangs ermöglichen. Ein Ziel ist es, das Verhalten des Antriebsstrangs in den transienten Betriebsphasen nachzubilden und wesentliche Teile des Steuergerätescodes mit in die Simulation zu übertragen. Die Optimierungskriterien beziehen sich hierbei sowohl auf die gesetzlichen, als auch auf die kundenspezifischen Anforderungen. Die Einhaltung der gesetzlich vorgegebenen Emissionsgrenzen, die Minimierung der Verbrauchswerte und die Maximierung der kundenrelevanten elektrischen Fahrphasen sind Kriterien, die in die Optimierung einfließen.

Ein weiteres Ziel ist es, durch die Übertragung wesentlicher Teile des Steuergerätescodes in die Simulation eine effiziente Testumgebung für Steuergerätefunktionen zu implementieren. Hierbei ist die Verifikation des Funktionsverhaltens im Kontext der längsdynamischen Fahrzeugsimulation im Fokus der Betrachtungen.

1.2 Stand der Technik

Nach einer konstruktiven Auslegung der Antriebskomponenten und der Erprobung dieser, müssen die gesetzten Ziele bezüglich des Verbrauchs, der Abgasemissionen und der kundenrelevanten Anforderungen umgesetzt werden. Im Vergleich zum konventionellen Antriebsstrang verfügt ein Dieselhybrid-Fahrzeug über zusätzliche Antriebskomponenten. Diese bewirken zusammen mit der Zunahme der Stellgrößen eines Dieselmotors einen Anstieg der zur Verfügung stehender Freiheitsgrade. Die Optimierung des Verhaltens des Antriebsstrangs bezüglich der beschriebenen Anforderungen wird somit immer schwieriger.

Das Verhalten des Antriebsstrangs wird maßgeblich durch die Hybrid-Betriebsstrategie und die Dieselmotorsteuerung beeinflusst. Bei der Betriebsstrategie sind die wichtigsten Einflussgrößen durch verschiedene Schwellwerte zum Ein- bzw. Ausschalten des Verbrennungsmotors und die Nach- bzw. Entladekennfelder gegeben. Ein aufgeladener Dieselmotor mit Direkteinspritzung verfügt über weitere Freiheitsgrade. Im Luftpfad sind diese durch die Zylinderfüllung, die variable Ventilstellung, die Variation der Abgasrückfrührate usw. gegeben. Die Einspritzung kann durch die Konfiguration der Vor-, Haupt- und Nacheinspritzung, sowie den Raildruck beeinflusst werden. Zusätzliche Einflussgrößen sind durch die Abstimmung der Schaltkennfelder des Getriebes gegeben.

Jede der aufgeführten Einflussgrößen hat Auswirkungen auf die definierten Kriterien aus dem Bereich der Abgasemissionen, des Verbrauchs und des Fahrverhaltens. Die Bestimmung einer optimalen Einstellung der vielen Schwellwerte und Kennfelder, die für die Verstellung der Einflussgrößen entscheidend sind, wird mit steigender Anzahl der Einflussgrößen und den vielfältigen Wechselwirkungen immer komplexer.

Bei der Suche nach dem Optimum scheidet eine Rastervermessung am Prüfstand sowohl aus zeitlichen, als auch aus wirtschaftlichen Gründen aus. Auch eine Online-Optimierung am Prüfstand unter Berücksichtigung aller Einflussgrößen ist sehr zeitaufwendig und kaum durchführbar. Die Verlagerung der Optimierungsrechnungen in die Simulation gewinnt somit immer mehr an Bedeutung. Sie bietet nicht nur die Möglichkeit eine optimale Einstellung der verschiedenen Parameter zu bestimmen, sondern erlaubt gleichzeitig die Wechselwirkungen der einzelnen Einflussgrößen zu untersuchen.

Im Umfeld der Parameteroptimierung der Betriebsstrategie eines Hybridfahrzeugs werden unter anderem Simulationsmodelle der Fahrzeuglängsdynamik eingesetzt [FFSZ13], [Von09, S. 16 ff.]. Durch die Modellierung der Komponenten des Antriebsstrangs und die Abbildung der Fahrwiderstände können die Antriebs- und die Fahrwiderstandskräfte simuliert werden. Die Simulation einer Fahrt entlang eines vorgegebenen Geschwindigkeitsprofils wird hierdurch möglich. Die einzelnen Komponenten des Antriebsstrangs sind oft durch Wirkungsgradkennfelder dargestellt. Vor allem die Implementierung des Verbrennungsmotors ist in der Regel durch messtechnisch erfasste Kennfelder für Verbrauchs- und Abgasemissionswerte gegeben.

Bei der Optimierung der Betriebsstrategie konzentrieren sich die Betrachtungen auf die Einstellung relevanter Kennlinien und Kennfelder. Die Steuergerätefunktionen, die mit Hilfe der Kennfeldvorgaben eine Änderung der Stellgrößen bewirken, werden in einer vereinfachten Form betrachtet. Somit entsprechen die optimierten Parameter nicht zwangsläufig den Kennlinien und Kennfeldern des Steuergeräts.

Im Gegensatz zum kennfeldbasierten Verbrennungsmotormodell innerhalb der Betriebsstrategieoptimierung, wird dieser bei der Optimierung der Verbrauchs- und Emissionswerte konventioneller Fahrzeuge sehr aufwendig und mit hohen Genauigkeitsanforderungen modelliert [ALWT98], [PST⁺08], [Sch10a], [Kla08]. Die Modelle basieren auf einer physikalischen Grundstruktur und beschreiben mittels empirischer Teilmodelle die charakteristischen innermotorischen Zusammenhänge. Dabei ergibt sich eine geschlossene mathematische Beschreibung der Strecke Einflussgröße-Zielgröße. Exemplarisch kann die Auswirkung einer Änderung der Abgasrückführrate auf die NO_X -Emissionen beschrieben werden. Bei der Optimierung selbst wird oft die Betrachtung einzelner gewichteter Betriebspunkte vorgenommen, so wie dies in [Ise10, S. 155 f.] beispielhaft beschrieben ist. Falls eine Bewertung anhand eines Geschwindigkeitsprofils angestrebt wird, so können die einzelnen Betriebspunkte gegen eine vorgegebene Drehzahl/- Drehmomentfolge getauscht werden.

Bei der Betrachtung des Verbrennungsmotorverhaltens sind, durch die geschlossene mathematische Darstellung der Modelle, die relevanten Funktionsmodule ein Teil der beschriebenen Modellstrecke. Somit führt eine Änderung des Datenstandes oder ein Update der Steuergerätesoftware zu einer Anpassung der modellierten Strecke und können im ungünstigsten Fall eine Neuvermessung des Prozesses notwendig machen.

1.3 Ziele der Arbeit

Das primäre Ziel der vorliegenden Arbeit ist die gemeinsame Optimierung der Betriebsstrategie und des Verbrennungsmotorverhaltens eines Hybridfahrzeugs. Durch die Zunahme der Komplexität der Antriebsstränge und der Antriebsaggregate, sowie den steigenden Umfang der Steuergerätefunktionen, ist eine Online-Optimierung der Applikationsparameter am Prüfstand sehr aufwendig. Am Beispiel eines modernen Dieselmotors kann gezeigt werden, dass die Ableitung der optimalen Applikationseinstellungen ohne den Einsatz von Modellierungs- und Optimierungswerkzeugen mit einem vertretbaren Aufwand kaum durchführbar ist, vor allem wenn alle relevanten Einflüsse und Wechselwirkungen des Verbrennungsmotors und der Betriebsstrategie in Betracht gezogen werden. Ein effizienter Weg die optimalen Einstellungen der Applikationsparameter zu berechnen, ist die simulationsbasierte Optimierung. Ausgehend von den gesetzlichen und kundenspezifischen Anforderungen kann ein Teil der Applikationsarbeiten in die Simulationsumgebung verlagert werden. Die hierbei berechneten Ergebnisse unterliegen einer Validierung am Prüfstand bzw. im Fahrzeug. Eine modellbasierte Untersuchung der komplexen Zusammenhänge ist für das Verständnis des Gesamtsystems und die Abschätzung der Stärke der einzelnen Wechselwirkungen von besonderer Bedeutung. Eine Bewertung der Robustheit der berechneten Optimierungsergebnisse bildet einen weiteren wichtigen Aspekt dieser Vorgehensweise.

Bei der Umsetzung der simulationsbasierten Optimierung der Applikationsparameter werden Modelle der zu optimierenden Strecke benötigt. Die Modellierung umfasst hierbei sowohl die Nachbildung physikalischer Zusammenhänge, als auch die Implementierung von Modellen der Steuergerätefunktionen. Die gemeinsame Optimierung des Verbrennungsmotorverhaltens und der Betriebsstrategie erfordert zunächst eine detaillierte Darstellung der Verbrennungskraftmaschine in der Simulationsumgebung. Bei der Nachbildung von physikalischen Strecken liegt somit der Fo-

kus der vorliegenden Arbeit auf der Modellierung dieser Antriebseinheit. Aufgrund des Umfangs und des komplexen Aufbaus der Steuergerätesoftware wird im Rahmen dieser Arbeit auf die Modellierung der Funktionsmodule verzichtet. Stattdessen soll die Emulation der im Steuergerät verwendeten Steuerungs- und Regelungsfunktionen durchgeführt werden. Die hierdurch vereinfachte Übertragbarkeit der optimierten Applikationsparameter ins Steuergerät, die Robustheit gegenüber Softwareänderungen und eine effiziente Testumgebung für neue Funktionsentwürfe sind nur einige Vorteile dieser Methode.

Die Weiterentwicklung der eingesetzten Werkzeuge zur Realisierung des beschriebenen Vorhabens stellt einen wesentlichen Teil dieser Arbeit dar. Das Untersuchungsobjekt für die Parameteroptimierung der im Steuergerät implementierten Steuerungs- und Regelungsfunktionen ist durch ein Dieselhybridfahrzeugmodell abgebildet.

Für die Auswahl der Realisierungsstrategie werden die im dargestellten Zusammenhang eingesetzten Werkzeuge und Prozesse betrachtet. Folgende Ziele werden im Rahmen dieser Arbeit verfolgt:

Simulationsumgebung Erweiterung der eingesetzten Simulationsumgebung um ein Werkzeug zur Simulation von Steuergerätefunktionen.

Modellbildung Für den Betrieb der Steuergerätemodule werden Steller-Sensor-Streckenmodelle benötigt. Ein Ziel ist es, durch die Anwendung geeigneter Modellierungsverfahren, dynamische Modelle der zu untersuchenden Größen zu erstellen. Besonders die Modellierung eines Verbrennungsmotors soll sowohl in der Komplexität, als auch im zeitlichen Verhalten abgebildet werden können. Eine Optimierung des Verhaltens dieser Antriebseinheit innerhalb des hybriden Antriebsstranges soll die Verbrauchs- und Emissionswerte im stationären und dynamischen Betriebsbereich abdecken.

Optimierungswerkzeuge Die Simulation der Steuergerätesoftware und die detaillierte Darstellung der Verbrennungskraftmaschine in der Simulationsumgebung führen zum Anstieg der Simulationszeit, der Qualität und der Quantität der Anforderungen bei den Optimierungsrechnungen. Das Ziel ist es, Werkzeuge zur Verfügung zu stellen, die den Änderungen entgegenwirken und eine effiziente Optimierung der modellierten Zusammenhänge erlauben. Sowohl eine Reduktion der Iterationsanzahl zur Einsparung der Simulationszeit, als auch eine systematische Betrachtung der verschiedenen Entwurfsspezifikationen stehen hierbei in Mittelpunkt der Betrachtungen.

Optimierung Die beschriebenen Werkzeuge sollen zur gemeinsamen Optimierung der Betriebsstrategie eines Hybridfahrzeuges und des Verbrennungsmotorverhaltens eingesetzt werden. Eine neue Funktion, die die Verbindung zwischen der Betriebsstrategie und der Motorsteuerung erweitert, soll innerhalb der Simulationsumgebung implementiert und optimiert werden. Um die Optimierungspotentiale aufzudecken, ist ein direkter Vergleich der Optimierungsergebnisse mit und ohne den zusätzlichen Funktionalitäten durchzuführen. Insgesamt ist eine Verbesserung des Fahrzeugverhaltens bezüglich des Verbrauchs und der kundenrelevanten Entwurfsspezifikationen unter Einhaltung der Emissionenvorgaben zu gewährleisten.

1.4 Aufbau der Arbeit

Für die Realisierung der vorgestellten Ziele ist die Abarbeitung mehrerer Arbeitspakete erforderlich. Daran ist auch der Aufbau der vorliegenden Arbeit geknüpft.

Im Kapitel 2 ist ein Werkzeug vorgestellt, welches die Simulation von Steuergerätefunktionen in der Simulationsumgebung ermöglicht. Dieses bildet das Bindeglied zwischen den Werkzeugen der Funktionsentwicklung und den Werkzeugen der Hybrid-Betriebsstrategie-Optimierung. Das Ziel ist es, eine Optimierung der Parametereinstellungen der Steuergerätefunktionen innerhalb der Simulationsumgebung für Fahrzeuglängsdynamik zu ermöglichen.

Auf der Basis der eingelesenen Sensorwerte berechnen die Steuergerätefunktionen Stellgrößen für die Aggregate des Antriebsstranges. Am Beispiel des Verbrennungsmotors sind im Kapitel 3 Methoden vorgestellt, die für die Abbildung eines nichtlinearen dynamischen Prozesses in der Simulationsumgebung eingesetzt werden. Die Modellierung des Verbrennungsmotors ist durch die Erstellung der Steller-Sensor-Streckenmodelle realisiert. Die Nachbildung des stationären und des dynamischen Verhaltens des Verbrennungsmotors kann hierbei in ausreichender Modellqualität sichergestellt werden.

Die Implementierung der Steuergerätefunktionen und der detaillierten Prozessmodelle führt zu einigen Anpassungen der Simulationsumgebung der Fahrzeuglängsdynamik. Die Anzahl der Einflussgrößen, die Anzahl und die Qualität der Optimierungskriterien und die Simulationszeit erfahren einen erheblichen Anstieg. Die Anwendung der im Kapitel 4 beschriebenen Optimierungsmethoden wirkt diesen Effekten entgegen und sichert eine strukturierte und effiziente Optimierung der Steuergeräteparameter innerhalb der Simulationsumgebung.

Aufbauend auf den Kapiteln 2, 3 und 4, d.h. unter Verwendung von:

- einer Simulation von Steuergerätefunktionen in einer Simulationsumgebung für Fahrzeuglängsdynamik
- einem nichtlinearen dynamischen Prozessmodell des Verbrennungsmotors
- den ausgewählten Optimierungsmethoden

ist im Kapitel 5 eine gemeinsame Optimierung des Verhalten des Verbrennungsmotors und der Hybrid-Betriebsstrategie realisiert. Die Optimierung beinhaltet auch den Entwurf und die Implementierung einer neuen Funktion, die die Schnittstelle zwischen der Hybrid-Betriebsstrategie und der Motorsteuerung erweitert. Um die Verbesserungspotentiale der implementierten Funktionalitäten zu berechnen, werden anschließend Optimierungsrechnungen mit unterschiedlichen Ansätzen durchgeführt. Sowohl die Optimierungsergebnisse, als auch die Eignung der eingesetzten Methoden zur Optimierung von Steuergeräteparametern sind als Arbeitsergebnis dieses Kapitels von Interesse.

Das abschließende Kapitel 6 fasst die Ziele, die wichtigsten Arbeitsschritte und die Ergebnisse der vorliegenden Arbeit zusammen.

2 Erstellung eines Werkzeugs zur Simulation von Steuergerätefunktionen

2.1 Motivation

In diesem Kapitel ist ein Werkzeug beschrieben, welches die Werkzeuge zur Optimierung der Betriebsstrategie eines Hybridfahrzeuges um die Simulationsmöglichkeit der Steuergerätefunktionen erweitert. Damit kann für die Simulationsumgebung der Fahrzeuglängsdynamik ein eindeutiger Übertragungsprozess der relevanten Steuergerätefunktionen von den Softwareerstellungs- in die Optimierungswerkzeuge definiert werden. Besonders wichtig ist an dieser Stelle die Erkenntnis, dass mit diesem Werkzeug die Freiheitsgrade eines modernen Dieselmotors in die Optimierung eingebunden werden können. Weiterhin entsprechen die optimierten Kennlinien und Kennfelder der Originalparametrierung des Steuergeräts, wodurch stets eine effiziente Applikationsmöglichkeit sichergestellt ist.

Der Einsatz solch eines Werkzeugs führt zu Änderungen im Abstimmungsprozess der Applikationsparameter, die in der Abbildung 2.1 zusammengefasst sind. Die Nachbildung der ausgewählten Funktionsumfänge wird durch eine gezielte Übertragung der originalen Steuergerätefunktionen, inklusive der Parametrierung, ersetzt. Nach den Optimierungsrechnungen entfällt hierdurch der Aufwand der Ergebnisübertragung ins Steuergerät. Es entsteht zusätzlich eine erhebliche Rückwirkung auf die Auswahl der zu optimierenden Parameter und den entsprechenden Funktionsumfang. Die simulationsgestützte Untersuchung der Hybrid-Betriebsstrategie-Parameter kann an dieser Stelle um die Parametrierung der Verbrennungskraftmaschine erweitert werden. Die relevanten Funktionsumfänge der Luftpfad- und der Einspritzungssteuerung können in die Optimierung mit einbezogen werden. Es ist hierdurch möglich Wechselwirkungen zwischen dem Verhalten des Verbrennungsmotors und der Betriebsstrategie zu untersuchen und zu optimieren. Dabei wird das Verhalten des Verbrennungsmotors gemeinsam mit der Betriebsstrategie optimiert.

Dieses Vorgehen hat nicht nur den Vorteil der direkten Übertragbarkeit der Optimierungsergebnisse ins reale Fahrzeug, sondern ist auch für die Untersuchung des transienten Verhaltens des Antriebsstranges geeignet. Der Fokus der Betrachtungen liegt hierbei auf den Momentenaufbauphasen und den Emissionswerten in den transienten Betriebsbereichen.

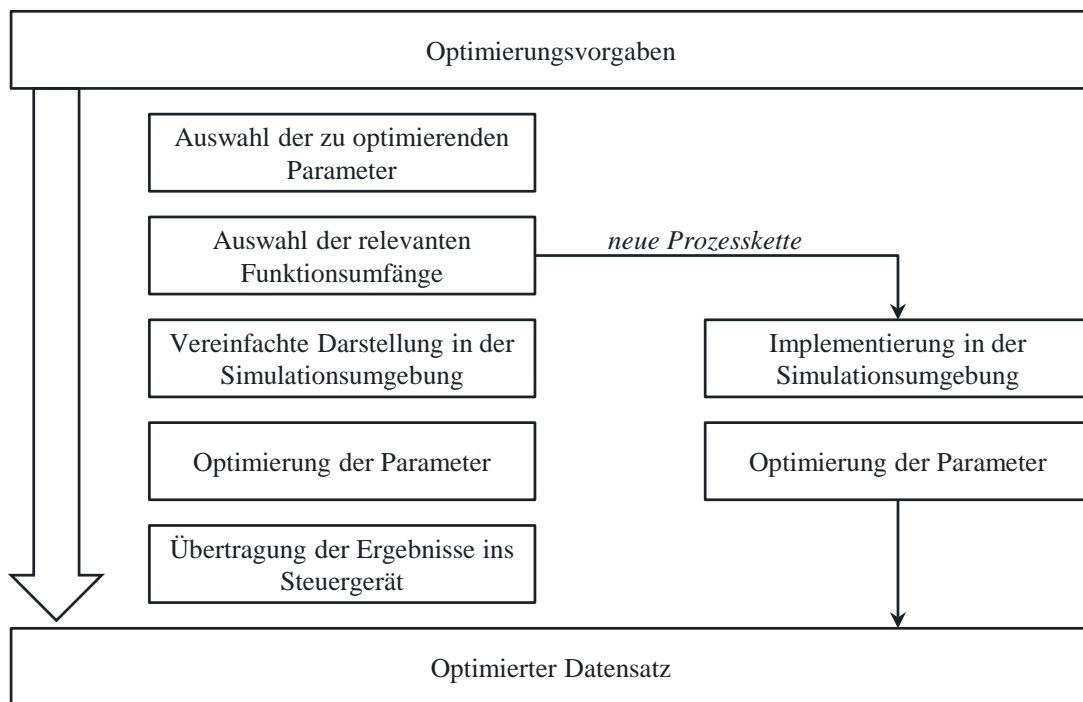


Abbildung 2.1: Anpassung des Applikationsprozesses durch die Weiterentwicklung der Simulationstools

Ein weiterer und besonders wichtiger Motivationspunkt ist die Idee, eine auf Simulink¹ basierte Testumgebung der Steuererätfunktionen für Einzel- und Integrationstests zu erstellen. Die Simulationstechniken der Simulink-Umgebung werden in diesem Fall mit den Programmier- und Testumgebungen der Ascet-Umgebung² verbunden.

Eine Betrachtung des Prozesses zur Optimierung von Betriebsstrategien eines Hybridfahrzeugs bildet den Ausgangspunkt der Betrachtungen (siehe Kapitel 2.2). Für die Realisierung der funktionalen Erweiterung ist zunächst die Einsicht in die Werkzeuge der Funktionsentwicklung und der Optimierung notwendig (siehe Kapitel 2.3). Das Kapitel 2.4 beschreibt die Vorgehensweise bei der Implementierung der gewünschten Funktionalitäten. Eine Darstellung der Simulationsergebnisse zur Verifikation des entwickelten Werkzeugs und die Beleuchtung der Eigenschaften bzw. Einsatzmöglichkeiten ist im Kapitel 2.5 gegeben.

Weiterhin werden für die simulationsgestützte Optimierung dynamische Modelle benötigt, deren Aufbau im Kapitel 3 detailliert beschrieben ist.

¹Simulink ist ein Simulationstool der Firma The MathWorks, Inc. <http://www.mathworks.de/products/simulink/>

²Ascet ist ein Tool der Firma ETAS GmbH zur Erstellung von *Embedded Software* http://www.etas.com/de/products/ascet_software_products.php

2.2 Optimierung der Betriebsstrategien von Hybrid-Fahrzeugen

In diesem Unterkapitel erfolgt eine Darstellung der Simulations- und Optimierungswerkzeuge, die bei einer Optimierung der Hybrid-Betriebsstrategie zum Einsatz kommen. Es werden Werkzeuge vorgestellt, die bei der Volkswagen AG eingesetzt werden und in den Literaturquellen [FFSZ13] und [MPW⁺13] beschrieben sind.

Im Rahmen der vorliegenden Arbeit bezieht sich die Optimierung der Betriebsstrategien von hybriden Fahrzeugen auf eine systematische, toolgestützte Optimierung von Steuergeräteeinstellungen. Den Gegenstand der Betrachtungen bildet hierbei eine Parameterschar, die für das Verhalten eines hybriden Antriebsstranges entscheidend ist. Es handelt sich um verschiedene Drehzahl- und Drehmomentschwellen, Lade- und Entladeleistungen, Schaltpunkte und weitere Applikationsparameter, die als zu optimierende Eingangsgrößen betrachtet werden. Die Zielgrößen werden auch an dieser Stelle durch die gesetzlichen Emissionsvorschriften und die kundenrelevanten Verbrauchswerte definiert. Bei den Optimierungskriterien kommen zu den Emissions- und Verbrauchsvorgaben hybridspezifische Anforderungen, wie die Einhaltung von Batterieladezuständen oder die Dauer der elektrischen Fahrphasen hinzu.

2.2.1 Gesamtfahrzeugmodell

Für die Auslegung der Betriebsstrategie eines hybriden Fahrzeugs ist in der vorliegenden Arbeit die Vorwärtssimulation von Bedeutung. Hierbei wird ein längsdynamisches Fahrzeugmodell mit Antriebsmomenten beaufschlagt. Die Antriebsmomente ergeben sich aus der Addition der einzelnen Drehmomente der Antriebsaggregate und führen zur Änderung der Fahrzeuggeschwindigkeit. Im Gegensatz zur Rückwärtssimulation, bei der eine Rückrechnung des Leistungsbedarfes durch feste Drehzahl- und Drehmomentfolgen am Rad erfolgt, wird bei der Vorwärtssimulation ein Fahrermodell benötigt. Die Vorwärtssimulation bietet dementsprechend eine hohe Ergebnisqualität, die aber nur mit entsprechendem Parametrieraufwand zu erreichen ist. Alle weiteren Betrachtungen beziehen sich somit auf diese Simulationsmethode.

An dieser Stelle fokussiert sich die Betrachtung auf ein auf Matlab/Simulink-Basis von der Volkswagen AG entwickeltes Simulationswerkzeug, welches auch in [FFSZ13] beschrieben ist. Eine schematische Darstellung dieses Werkzeugs ist in der Abbildung 2.2 gegeben. Dieses bietet die Möglichkeit, das längsdynamische Fahrverhalten eines Fahrzeugs zu simulieren und zu untersuchen.

Der modulare Aufbau erlaubt die Simulation von verschiedenen Antriebsstrangtopologien. Mit Hilfe eines Fahrermodells und eines Geschwindigkeitsprofils (Block: *Fahrer*) werden Signale der Fahreranforderung generiert. Der Vergleich der Sollgeschwindigkeit mit der simulierten Istgeschwindigkeit ist für die Bestimmung der Antriebs- bzw. der Bremsmomentanforderung entscheidend. Eine Abbildung der Steuergerätefunktionalität ist mit dem Block *Steuergerät* gegeben. Die Realisierung der Steuerung des Verbrennungsmotors beinhaltet die Implementierung von Leerlauffunktionalitäten und des Maximal- bzw. Minimalmoments. Die Abbildung der Steuergerätefunktionalitäten der Betriebsstrategie ist an dieser Stelle wesentlich genauer, weil diese die zu optimierenden Parameter

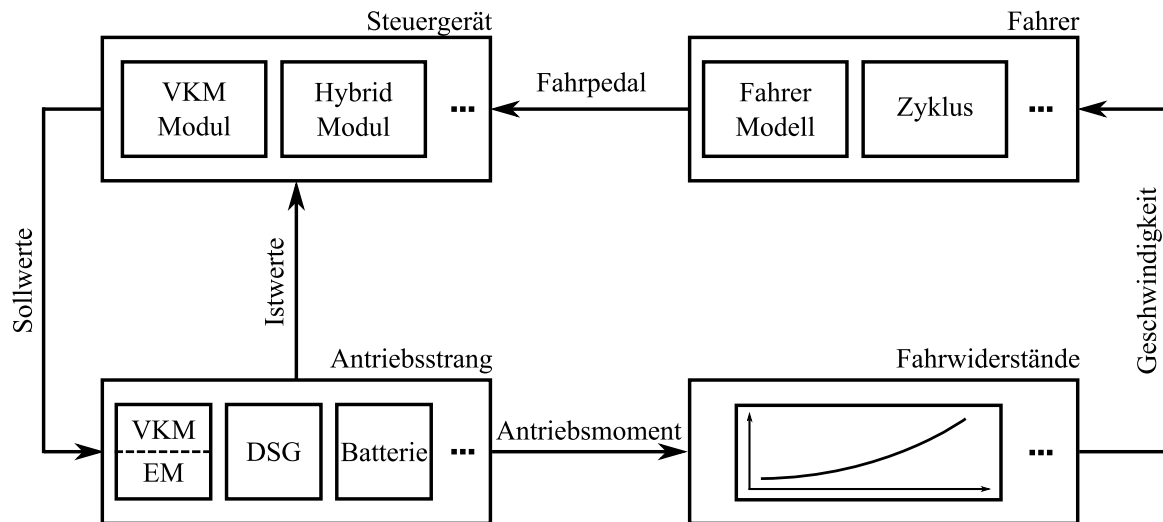


Abbildung 2.2: Aufbau des Simulationswerkzeugs zur Simulation des längsdynamischen Fahrzeugverhaltens

enthalten. Alle implementierten Steuerungsfunktionen sind auf ihre funktionale Ebene reduziert, zusammengefasst und in der Simulationsumgebung nachgebildet. Der Block *Antriebsstrang* beinhaltet die Module des Verbrennungsmotors, des Elektromotors, der Batterie, der Kupplungen usw. Der Aufbau der Module ist sowohl durch eine physikalische Grundstruktur, als auch durch eine Kennfelddarstellung realisiert. In dem Block *Antriebsstrang* werden die Sollanforderungen umgesetzt und die aktuellen Werte des Antriebsstranges berechnet.

Die Darstellung des Verbrennungsmotors ist für weitere Betrachtungen besonders signifikant. Für die Betrachtung der hybridspezifischen Eigenschaften des Antriebsstranges wird dieser oftmals durch Kennfelder dargestellt (siehe zum Beispiel [Ise10, S. 328 ff.], [Von09, S. 19 ff.]). Eine mögliche Realisierung dieses Modellblocks ist in der Abbildung 2.3 visualisiert. Um das Abgasemissionsverhalten zu bestimmen, werden die vermessenen Emissionswerte als Kennfelder in Abhängigkeit von der Motordrehzahl n_{ist} und dem Motordrehmoment M_{ist} dargestellt (Abbildung 2.3, unten). Das Drehmoment des Verbrennungsmotors M_{ist} ergibt sich aus dem begrenzten und verzögertem Wunschmoment M_{soll} . Die Motordrehzahl n_{ist} kann aus dem Integral über die Differenz aus Antriebsmoment M_{ist} und Abtriebsmoment M_w bestimmt werden.

2.2.2 Optimierung der Betriebsstrategieparameter

Es existieren verschiedene Optimierungsverfahren, die entsprechend ihrer Eigenschaften für unterschiedliche Problemstellungen geeignet sind. Einige der in diesem Umfeld eingesetzten Algorithmen sind z.B. der Simplex-Algorithmus nach Nelder-Mead [LRWW98], das Trust-Region-Verfahren [BGN00], die Partikelschwarmoptimierung [KE95], [Oli09, S. 41-58] und die evolutionären Algorithmen [Wei07].

Als Bezugszyklus für die Simulation dient ein vorgegebenes Geschwindigkeitsprofil, zum Beispiel der NEFZ- oder FTP-Zyklus. Durch die Variation der Einflussgrößen bzw. der Applikationsparameter der Betriebsstrategie ist somit eine Aussage über die modellierten Zielgrößen wie Verbrauchs-

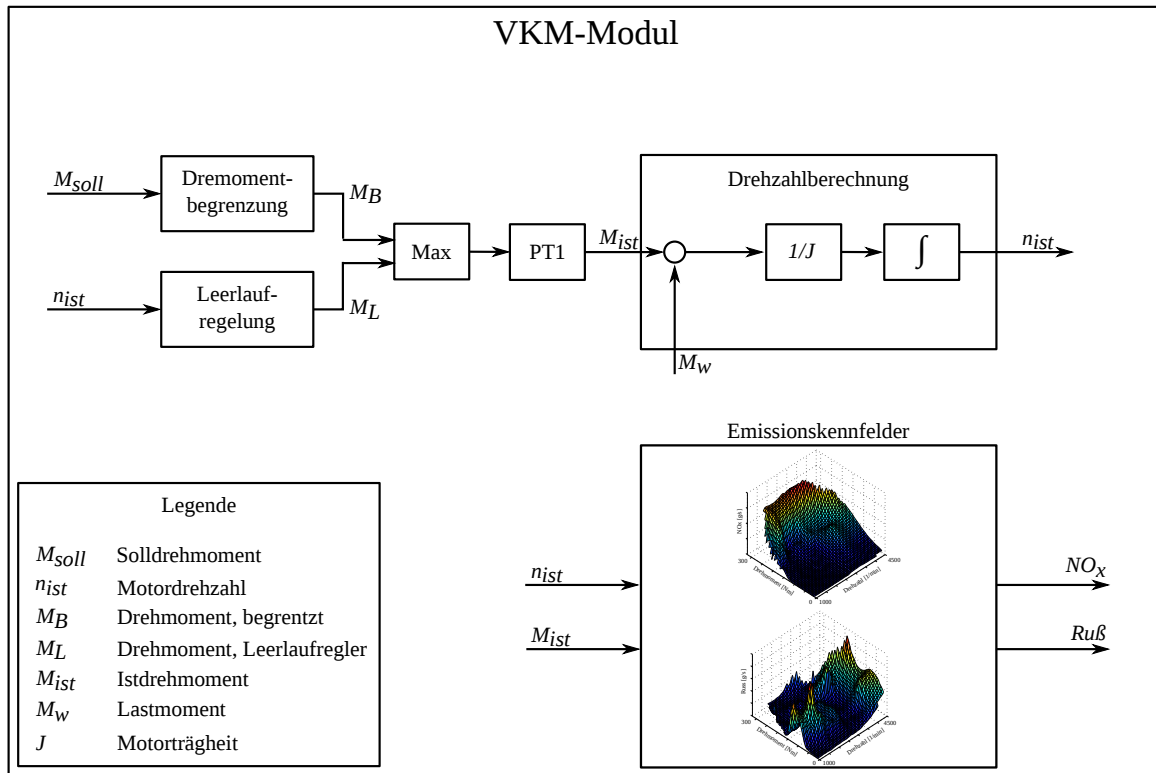


Abbildung 2.3: Beispielhafter Aufbau des Simulationblocks des Verbrennungsmotors in der Simulationsumgebung der Fahrzeuglängsdynamik

und Emissionswerte, Ladezustand der Batterie, Verlustleistungen, Dauer der elektrischen Fahrphasen usw. möglich. Die Auswertung der Zielgrößen anhand von verschiedenen Kriterien ermöglicht einem Optimierungsalgorithmus die optimale Einstellung der Parameter zu finden.

Die Implementierung der für die Optimierung relevanten Steuerungs- und Regelungsfunktionen erfordert jedoch Vereinfachungen und Vernachlässigungen im Vergleich zum originalen Code des Steuergerätes. Einerseits sind nicht alle für den originalen Steuergerätecode relevanten Betriebsphasen in der Simulation wiederzufinden. Andererseits steigt der Aufwand mit der Erhöhung des Übertragungsumfangs der Steuergerätesoftware bei der Nachmodellierung stark an. Weiterhin unterliegen die Steuerungs- und Regelungsfunktionen einer ständigen Weiterentwicklung, sodass eine Einarbeitung der funktionalen Änderungen erforderlich ist. Eine ähnliche Herausforderung im Änderungsmanagement entsteht durch viele Fahrzeugprojekte, die sich jeweils in der Softwareversion unterscheiden können. Die aus einer Betriebsstrategieoptimierung gewonnenen Ergebnisse gelten aufgrund der getroffenen Vereinfachungen und Vernachlässigungen nicht für die im Steuergerät verwendeten Funktionen. Somit stellt der Übertragungsprozess der optimierten Parameter ins Fahrzeug oftmals die Übertragung des Verhaltens des Simulationsfahrzeugs auf das Realfahrzeug dar.

Ein weiterer Aspekt, der bei den vorliegenden Simulationsmöglichkeiten nicht außer Acht gelassen werden kann, ist die Darstellung des Verbrennungsmotors als festes Kennfeldmodell. Der Vorteil dieser Implementierung ist, dass die Betriebspunkte des Verbrennungsmotors im Zusammenwirken mit der Betriebsstrategie anhand einer Kennfelddarstellung auf eine einfache Weise veranschaulicht werden können. Beispielhaft kann eine Optimierung anhand von verschiedenen Verlustleistungen hervorragend plausibilisiert werden, was durchaus zu stabilen Ergebnissen führt. Der Nachteil die-

ser Methode ist, dass die Möglichkeit einer Optimierung des Verhaltens des Verbrennungsmotors innerhalb der Simulationsumgebung nicht mehr gegeben ist. Somit können auch die Wechselwirkungen zwischen dem Verhalten der Betriebsstrategie und der Verbrennungskraftmaschine nicht untersucht bzw. optimiert werden.

Um die beschriebenen Nachteile zu verbessern, wird das Simulationswerkzeug um die Möglichkeit der Simulation von Steuererätaefunktionen erweitert. Hierfür ist zunächst die Betrachtung der Werkzeuge der Funktionsentwicklung und der Optimierung notwendig.

2.3 Werkzeuge der Funktionsentwicklung und der Optimierung

Bei der Entwicklung von Steuererätaefunktionen für die Motor- bzw. Hybridsteuerung wird bei Volkswagen vornehmlich das Entwicklungswerkzeug Ascet eingesetzt. Prinzipiell handelt es sich um ein Softwarewerkzeug, welches eine graphische Definition der einzelnen Funktionen bzw. Funktionsmodule erlaubt. Dieses Werkzeug bietet einen auf die Funktionsentwicklung abgestimmten Funktionsumfang, der seinerseits auf die Erzeugung von effizientem Programmcode für eingebettete Systeme zugeschnitten ist. Somit kann eine implementierte Funktion in einen lauffähigen C-Code überführt werden.

Die Optimierung von Funktions- bzw. Applikationsparametern der Betriebsstrategie eines Hybridfahrzeugs wird bei Volkswagen mit Hilfe des im Kapitel 2.2.1 vorgestellten und auf Matlab/Simulink-Basis aufgebauten Simulationspaketes durchgeführt. Bezogen auf das Einsatzgebiet der Simulation und Optimierung stellt dieses Werkzeug alle notwendigen Freiheiten und Funktionalitäten bereit. Die Implementierung der Vorwärtssimulation eines Fahrzeugs kann modular, skalierbar und flexibel aufgebaut werden. Die Schnittstellen für die Optimierung und die Auswertung sind schnell und effizient implementierbar.

2.4 Implementierung

2.4.1 Ziele und Anforderungen

Für die Realisierung der Weiterentwicklung der Werkzeugkette kann aus den vorherigen Betrachtungen folgendes Ziel abgeleitet werden:

Modellbasierte Optimierung der Applikationsparameter der im Steuererät verwendeten Funktionen in Matlab/Simulink

Mit diesem Ziel wird innerhalb der Vorwärtssimulation die Möglichkeit der Verwendung, von der in der Funktionsentwicklung definierten Steuererätaefunktionen, zum Zwecke der Optimierung von

Applikationsparametern, sichergestellt. Die aus der Verwendung von originalen Steuererätesfunktionen resultierende Möglichkeit der Untersuchung der transienten Betriebsphasen des Antriebsstrangs und die direkte Übertragung der Optimierungsergebnisse ins reale Fahrzeug sind wichtige Aspekte, die durchgehend zu beachten sind.

Die sich daraus ergebenden Anforderungen, die an die Werkzeugimplementierung gestellt werden, sind nachfolgend zusammengefasst:

- Simulation von dem aus Ascet erzeugtem C-Code in der Simulink-Umgebung
- Parametrierbarkeit aus Matlab/Simulink unter Verwendung von DCM-Dateien¹
- Flexibilität bezüglich der Implementierung der einzelnen Funktionen. Diese sollen einzeln implementierbar und ersetzbar sein
- Möglichst geringer Aufwand im Änderungsmanagement von Funktions- und Datenständen
- Verwendung von den im Steuergerät implementierten Routinen (z.B. Interpolationsroutinen)
- Unterstützung von verschiedenen Programmier Techniken
 - Globale Variablen, Kennfelder und Kennlinien
 - Verwendung von globalen Achsen
 - Verwendung von Systemkonstanten
 - usw.
- Robustheit, Bedienbarkeit, Konfigurierbarkeit

2.4.2 Ansatz

Für die Realisierung der beschriebenen Anforderungen ist das Konzept der Implementierung einer S-Function in der Simulink-Umgebung relevant. Dieses Konzept bietet die Möglichkeit der Anbindung von C-Code in das Konstrukt einer S-Function. Die Schnittstelle kann durch die Verwendung von dem aus Ascet erzeugtem C-Code angesprochen werden und sichert somit die grundsätzliche Machbarkeit der Übertragung der originalen Steuererätesoftware nach Matlab/Simulink. In der Abbildung 2.4 sind die wesentlichen Schritte zur Übertragung eines Steuererätes-Funktionsmoduls von Ascet nach Simulink veranschaulicht. Ausgehend von einem in der Ascet-Umgebung implementierten Funktionsmodul wird zunächst die automatische Codegenerierung benutzt, um die Beschreibung der implementierten Funktionalitäten in der Programmiersprache C zu erhalten. Das Ziel ist es, mit Hilfe einer gezielten Untersuchung des C-Codes auf eine automatisierte Weise eine C-S-Function zu erstellen und anschließend in einen passenden Simulink-Block zu integrieren.

¹Ein Dateiformat zur Speicherung von Applikationsparametern (Kennfelder, Kennlinien, Konstanten etc.). Die technische Dokumentation des Formats ist in http://www.etas.com/download-center-files/products_ASCET_Software_Products/TechNote_DCM_File_Formats.pdf zu finden.

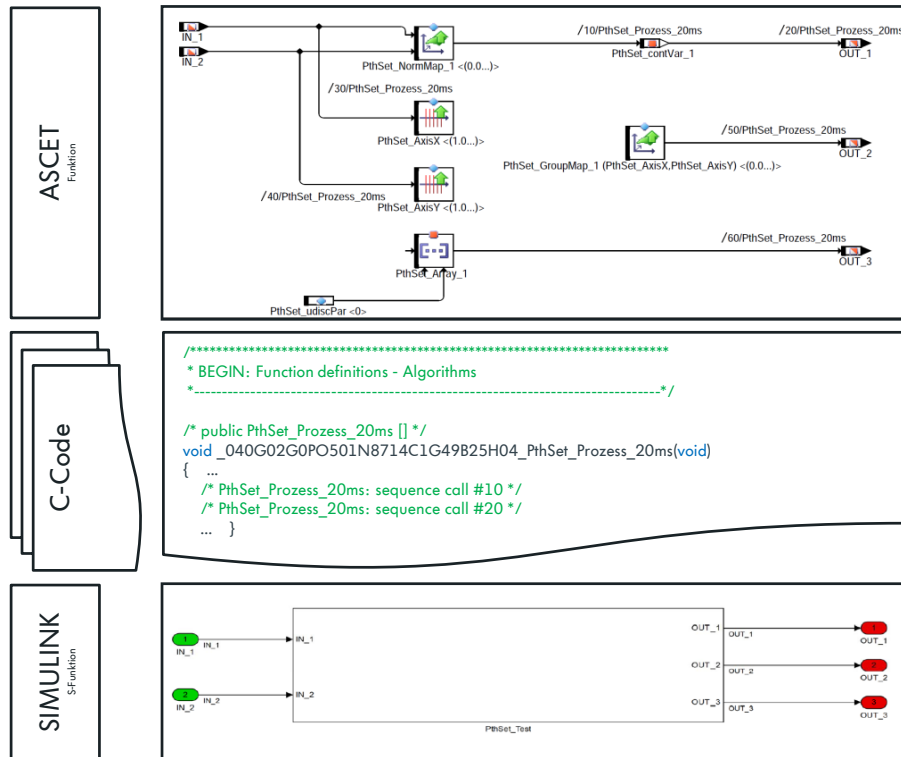


Abbildung 2.4: Grundschritte zum Transfer eines Steuergeräte-Funktionsmoduls in die Simulink Umgebung

2.4.3 Funktionale Ebene

Dieses Kapitel beschreibt das grundlegende Vorgehen bei der Implementierung des Werkzeugs. In der Abbildung 2.5 sind die wichtigsten Informationsflüsse und Arbeitsschritte der Übertragung einer Steuergerätefunktion von Ascet nach Simulink dargestellt. Die automatische Codegenerierung in Ascet bildet den ersten Schritt des Transfers einer Steuergerätefunktion nach Simulink. Der erzeugte C-Code besteht aus mehreren Einzeldateien, die wiederum auf verschiedenen Header-Dateien aufbauen. Durch gezielte Anpassungen in den Header-Dateien wird die Lauffähigkeit des erzeugten C-Codes in der Simulink-Umgebung sichergestellt. Somit können die aus der Steuergerätefunktion erzeugten C-Code Dateien unverändert beibehalten werden. Mit der gleichen Vorgehensweise wird die Übertragung der im Steuergerät implementierten Berechnungsroutinen realisiert (z.B. Interpolation).

Für die Parametrierung der Funktionen wird eine DCM-Datei benötigt, die alle in der Funktion verwendeten Parameter enthält. Diese Datei ist dem zu simulierenden Datensatz des Steuergerätes zu entnehmen. Die enthaltenen Parameterinformationen werden dann in den Matlab *Workspace* eingelesen. Spezielle Routinen, die innerhalb des S-Function-Konstrukts implementiert sind, lesen die Funktionsparametrierung direkt aus dem Matlab *Workspace* ein und verarbeiten die eingestellten Parametereinstellungen.

Um die in der Ascet Umgebung definierten globalen Variablen bzw. Variablenarrays verwenden zu können, werden diese ähnlich wie die Funktionsparameter im Matlab *Workspace* definiert. Die Quelle

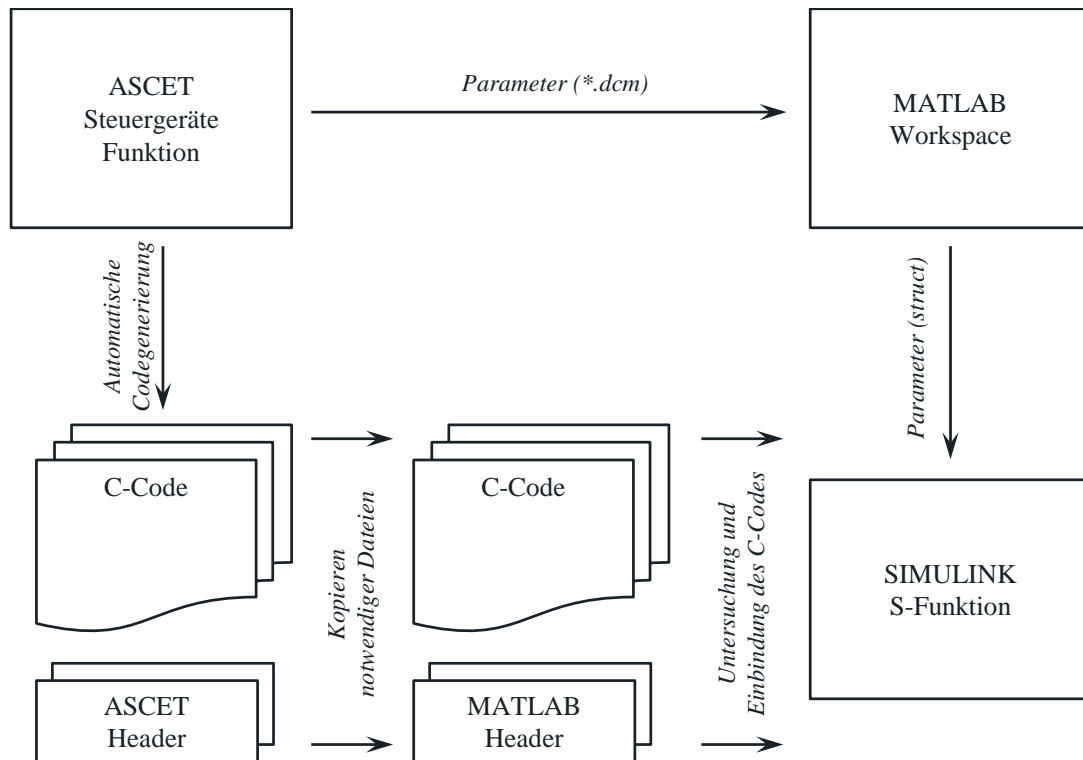


Abbildung 2.5: Schematische Darstellung des Übertragungsprozesses von Ascet nach Simulink

hierfür ist die A2L-Datei¹ des zu simulierenden Datensatzes. Die Übergabe an die einzelnen Funktionen geschieht auch in diesem Fall über die Parameterschnittstelle der S-Function. Ein wesentliches Unterscheidungsmerkmal zu den festen Parametern besteht darin, dass die global verwendeten Größen bei jedem Ausführungsschritt der Steuergerätefunktion geändert werden können.

Eine detaillierte Beschreibung der implementierten Funktionalitäten des entwickelten Werkzeugs ist im Anhang im Kapitel A.1 zu finden.

2.5 Verifikation

2.5.1 Simulationsergebnisse

Die Verifikation des vorgestellten Werkzeugs ist am Beispiel eines Dieselmotors durchgeführt. Der im Kapitel 3.4.1 vorgestellte Versuchsträger wurde für die im Kapitel 3 beschriebenen Inhalte am Verbrennungsmotorenprüfstand vermessen. Die gewonnenen Messergebnisse werden für den Test der in die Simulink-Umgebung übertragenen Steuergerätefunktionen verwendet.

Die übertragenen Umfänge des Steuergerätescodes belaufen sich hierbei auf ca. 120 Funktionsmodule. Die ausgewählten Funktionen sind für den Betrieb eines Dieselmotors entscheidend. Es sind Funktionen die zur Steuerung des Luft- und Kraftstoffpfades notwendig sind. Dementsprechend

¹Beinhaltet die Beschreibung der von einem Steuergeräteprogramm verwendeten Parameter und Messgrößen

berechnen diese Funktionen Stellgrößen wie Drosselstellungen der HD-AGR und ND-AGR, Stellung der VTG, Raildrucksollwert und Einspritzcharakteristik. Die Einspritzcharakteristik ist hierbei durch die Menge und die Zeitpunkte der Vor-, Haupt- und Nacheinspritzungen definiert.

Für die Verifikation, der in die Simulink-Umgebung übertragener Funktionsumfänge und deren Zusammenwirkung untereinander werden die Prüfstandsmessungen wie folgt verwendet:

1. Stimulation der Sensoreingänge der Funktionsmodule mit den am Prüfstand gemessenen Messsignalen
2. Vergleich der während der Prüfstandsvermessung und der Simulation berechneten Stellereollwerte (Sollwert der Zylinderfüllung, Sollwert der Abgasrückführrate, etc.)
3. Vergleich der während der Prüfstandsvermessung und der Simulation berechneten internen Berechnungsgrößen (Istwert der Zylinderfüllung, Istwert der Abgasrückführrate, etc.)

Die Auswertung der berechneten und der simulierten zeitlichen Verläufe erfolgt anhand einer direkten Gegenüberstellung. Die Beurteilung der Größe der Abweichungen erfolgt mittels der Angabe des RMSE-Fehlermaßes (siehe Kapitel A.4). Der E_{RMSE} ist ein Maß für die mittlere Summe der Quadratabweichungen. Die Abbildung 2.6 zeigt die direkten Vergleiche der im Steuergerät und in der Simulation berechneten Größen.

Die zeitlichen Verläufe der 600 Sekunden dauernden Messung bzw. Simulation beinhalten eine Anhebung der Drehzahl von 1000 auf 1250 Umdrehungen pro Minute bei der Sekunde 230. Gleichzeitig wird das Drehmoment vom maximalen auf den minimalen Wert zurückgefahren. Hierdurch kann eine Absenkung der Turbinendrehzahl und der Zylinderfüllung beobachtet werden. Im Gegensatz hierzu steigen die berechneten Massenflüsse der Hochdruck- und der Niederdruckabgasrückführung an. Die dazugehörigen Sollwertverläufe sind im unteren Bereich der Abbildung 2.6 dargestellt. Der direkte Vergleich der im Steuergerät berechneten und der in der Simulink-Umgebung simulierten Größen zeigt eine hervorragende Übereinstimmung auf. Die simulierten Werte der Sollwertverläufe treffen die während der Prüfstandsmessung berechneten Werte besonders gut. Auch bei den internen Berechnungsgrößen können nur geringe Abweichungen festgestellt werden.

Nicht Gegenstand der Betrachtungen sind an dieser Stelle verschiedene adaptive Korrekturen, die zum Beispiel die Alterung oder die Toleranzen der Bauteile beschreiben und ausgleichen. Aufgrund ihrer geringer Relevanz zum Zeitpunkt der Optimierungsrechnungen der Hybridstrategie und dem zusätzlichen Simulationsaufwand wurden diese bei dem Transfer der Steuerungssoftware des Verbrennungsmotors in die Simulationsumgebung weggelassen. Die einzelnen Abweichungen sind unter anderem auf diese, in der Simulation nicht implementierten, Funktionsmodule des Steuergeräts zurückzuführen.

2.5.2 Eigenschaften und Einsatzmöglichkeiten

Nach der Verifikation des entwickelten Werkzeugs sind in diesem Unterkapitel verschiedene Eigenschaften und Einsatzmöglichkeiten explizit beleuchtet.

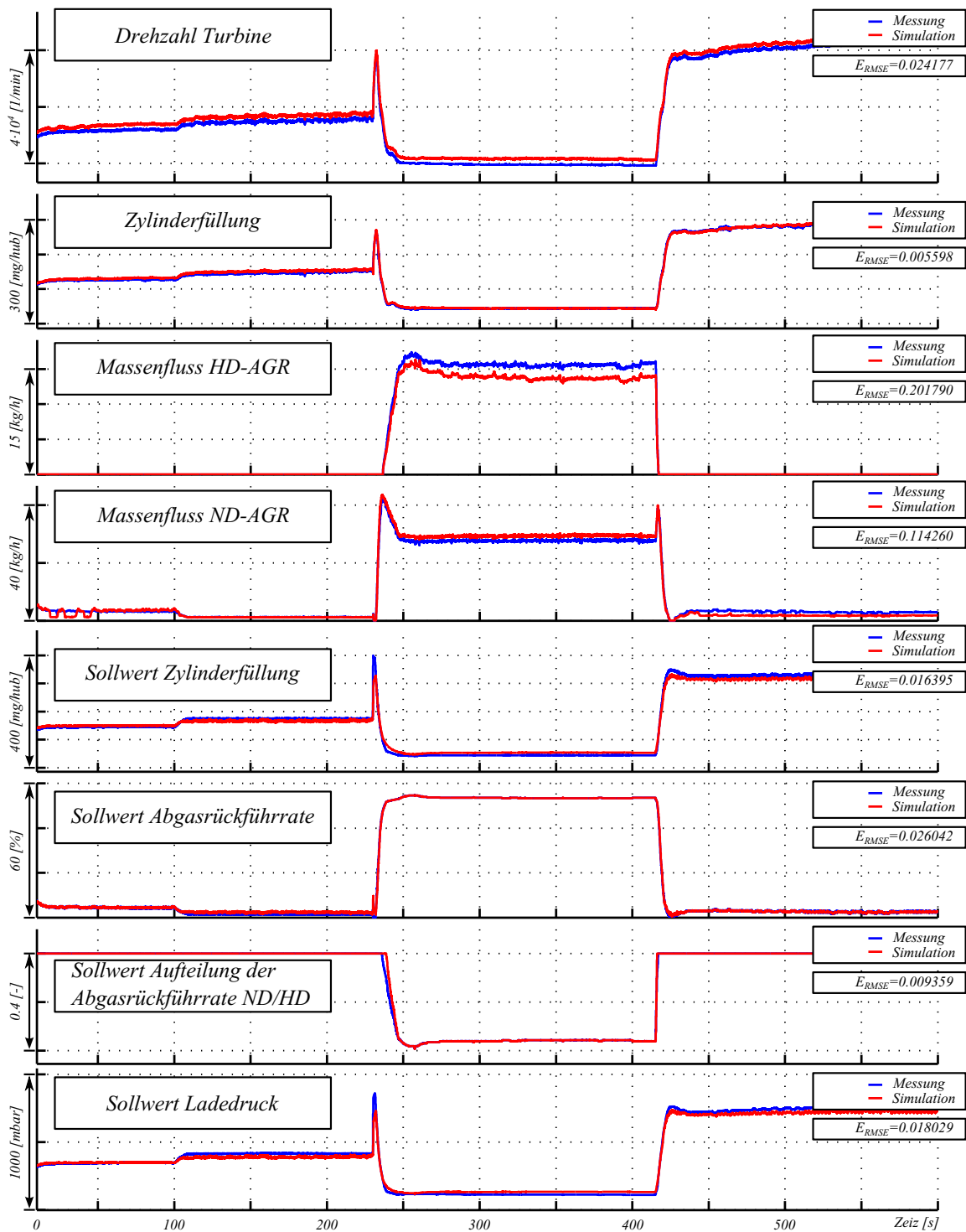


Abbildung 2.6: Test des implementierten Simulationswerkzeugs. Direkter Vergleich einiger Größen des Luftpfads

Beim Änderungsmanagement der Steuergerätefunktionen fällt der Nachpflegeaufwand in der Simulationsumgebung sehr gering aus. Ändert sich eine oder mehrere Funktionen, so ist nur die Aktualisierung des erzeugten C-Codes notwendig. Ändert sich die Parametrierung, so ist für die Aktualisierung der Simulationsumgebung die Generierung einer neuen DCM-Datei ausreichend. Als Ergebnis

steht ein Simulationswerkzeug zur Verfügung, das eine Simulation der original im Steuergerät implementierten Funktionen ermöglicht. Dieses erfüllt alle im Kapitel 2.4.1 gestellten Anforderungen. Eine Begrenzung des implementierten Werkzeugs auf die Optimierung von Applikationsparametern ist an dieser Stelle nicht zwingend erforderlich. Der Entwurf neuer Konzepte für Steuererätaefunktionen, die das Zusammenwirken der Hybridkomponenten verbessern, kann direkt in der Simulink-Umgebung erfolgen.

Die Implementierung einer Konzeptfunktion direkt in der Simulationsumgebung und die Zusammenführung dieser mit einem Verbund aus schon vorhandenen Steuererätaefunktionen bieten die besten Voraussetzungen für den Test eines Konzepts (siehe Abbildung 2.7, *Voruntersuchung*). Ist der Test vielversprechend, so kann aus dem Konzept eine Steuererätaefunktion in Ascet definiert werden. Die Übertragung dieser Funktion in die Simulationsumgebung erlaubt nicht nur den grundsätzlichen Machbarkeitstest, sondern eröffnet auch die Möglichkeit die Funktion in realitätsnaher Umgebung zu testen. Das Signaleingangs- und Ausgangsverhalten, die Wahl der Parametrierung, die Wahl der Stützstellenzahl und des Stützstellenabstandes sowie die Rechengenauigkeit und das Prozessverhalten der Funktion sind nur einige Eigenschaften, die in der Simulationsumgebung getestet werden können (siehe Abbildung 2.7, *Test*).

Insgesamt ergibt sich eine effiziente Entwicklungsmethodik, die die Möglichkeiten zum Testen, Optimieren und Kalibrieren von Steuererätaefunktion stark erweitert.

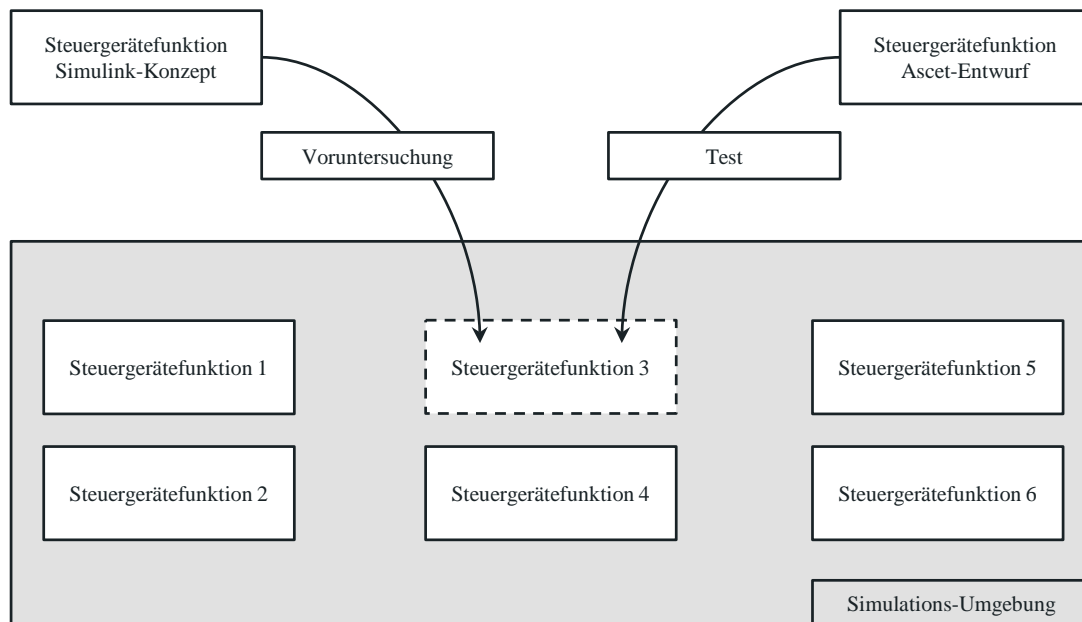


Abbildung 2.7: Einsatzmöglichkeiten des implementierten Simulationswerkzeugs

3 Modellierung des untersuchten Verbrennungsmotors

3.1 Motivation

Im Kapitel 2.2.1 ist das, für die Optimierung von Betriebsstrategien eingesetzte, Gesamtfahrzeug-Simulationsmodell vorgestellt. Anhand der übergreifenden Blockdarstellung in der Abbildung 2.2 ist weiterhin die Funktionsweise des Modells erläutert. Durch das entwickelte Werkzeug erfährt der Block *Steuergerät* eine grundlegende Modifikation. Es werden alle relevanten Funktionsmodule des Steuergerätes in die Simulationsumgebung übertragen.

Dieses Kapitel konzentriert sich auf den Block *Antriebsstrang*. Wie auch der Abbildung 2.2 zu entnehmen ist, besteht dieser Block aus mehreren Komponenten bzw. Modulen. Die zu den rotierenden Massen beitragenden Komponenten, wie der Verbrennungsmotor, die elektrische Maschine, das Getriebe oder das Differentialgetriebe beinhalten jeweils einen sogenannten Rotationsmassenblock. Dieser ist für den Beitrag der jeweiligen Massenträgheit im Antriebsstrang zuständig. Durch das Ein- bzw. Auskuppeln, die verschiedenen Gänge oder das Ein- bzw. Ausschalten des Verbrennungsmotors kann so die passende Massenträgheit konfiguriert werden. Weiterhin wird an dieser Stelle die Initialisierung der Komponenten mit der jeweils passenden Drehzahl durchgeführt. Entsprechend diesen Überlegungen ist in jedem Modul eine passende Steuerung implementiert, die verschiedene komponentenspezifische Aufgaben übernimmt. Hierzu zählen unter anderem die Einhaltung der Momentengrenzen oder die Steuerung bestimmter Abläufe (Kuppeln, Schalten usw.).

Durch die Konfiguration verschiedener Konstruktionsdaten und Parameter, wie Wirkungsgradkennfelder, Verlustleistungen, Schleppmomente usw., können die einzelnen Komponenten an die jeweiligen Fahrzeugprojekte angepasst werden.

Der Verbrennungsmotor stellt innerhalb eines hybriden Antriebsstranges die wesentliche Antriebs-einheit dar. Die Energiewandlung in dieser Antriebsquelle ist immer mit einem Kraftstoffverbrauch und der Erzeugung von Abgasemissionen verbunden. Der optimale Betrieb dieser Einheit ist ein wichtiger Bestandteil bei der Optimierung der Betriebsstrategie eines Hybridfahrzeugs. Entsprechend den im Kapitel 1.3 vorgestellten Zielen werden die Freiheitsgrade des Verbrennungsmotors in die Optimierung von der Hybrid-Betriebsstrategie eingebunden.

Bei der Modellierung des Verbrennungsmotorverhaltens wird oft von der Erstellung der Zielgrößenmodelle als Funktion der verstellten Einflussgrößen ausgegangen [Ise10, S. 130 ff.]. Diese Zielgrößenmodelle beschreiben hierbei die Strecke *Verstellparameter – Zielgröße* als eine geschlossene mathematische Gleichung. Es erfolgt keine Aufteilung des modellierten Prozesses. Sowohl der im Steuergerät

implementierter Softwareteil, als auch der als Motor verfügbarer Teil der Strecke werden zusammengefasst. Der Softwareteil ist hierbei durch die Strecke zwischen den Verstellparametern und den Stell Sollwerten definiert. Das Verhalten des Verbrennungsmotors ist durch die Strecke zwischen den Stellwerten und den gemessenen Sensorwerten beschrieben. Die einheitliche Betrachtung des Prozesses führt zu einer Abhängigkeit des Modells vom aktuellen Funktions- und Datenstand. Eine Änderung des Softwareteils könnte durchaus die Gültigkeit der erstellten Modelle einschränken. Im Rahmen der vorliegenden Arbeit werden für die Modellierung des Verbrennungsmotorverhaltens die in der Simulink-Simulationsumgebung implementierten Funktionalitäten betrachtet. Im Kapitel 2 ist ein Werkzeug vorgestellt, welches die Übertragung der Funktionsmodule des Steuergeräts in die Simulink-Umgebung ermöglicht. Die damit erstellte Abbildung der Steuergerätfunktionen in der Simulationsumgebung wird für Simulations- bzw. Optimierungsaufgaben verwendet. Eine getrennte Betrachtung des Softwareteils und der Motorstrecke kann an dieser Stelle realisiert werden. Hierdurch können die diskutierten Nachteile bei Änderungen des Funktions- bzw. Datenstandes behoben werden.

Als Eingangssignale benötigen die Funktionen verschiedene Sensorsignale, die vom Fahrzeug zur Verfügung gestellt werden. Aus den Sensorsignalen berechnen die Funktionen Stellsignale für die Ansteuerung der Aktuatoren, die ihrerseits eine Rückwirkung auf die Sensorsignale ausüben. Die beschriebene *Steller-Sensor* Strecke bildet das Modell des Verbrennungsmotors ab und ist bis jetzt in der im Einsatz befindlichen Simulationsumgebung nicht implementiert. Die Abbildung 3.1 verdeutlicht die diskutierten Zusammenhänge.

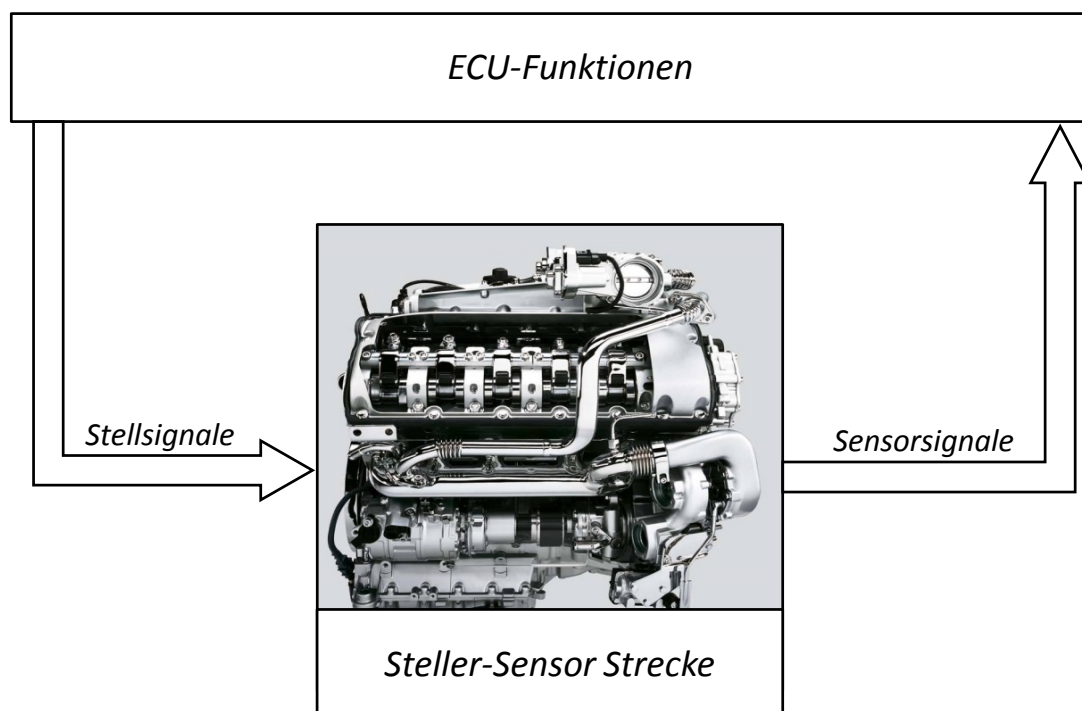


Abbildung 3.1: Ansatz zur Implementierung des Motormodells in der Simulationsumgebung

Das Ziel ist es, die von den Funktionsmodulen benötigten Signale der Druck- und der Temperatur-

sensoren in ihren stationären und dynamischen Eigenschaften nachzubilden. Damit wird in der Simulationsumgebung der Fahrzeuglängsdynamik das implementierte Kennfeldmodell des Verbrennungsmotors (siehe Kapitel 2.2.1) durch ein nichtlineares, dynamisches Motormodell ersetzt. Entsprechend der implementierten Struktur verfügt dieses Modell über alle Freiheitsgrade, die ein moderner Dieselmotor bietet. Eine gemeinsame Optimierung der Hybrid-Betriebsstrategie und des Verhaltens des Verbrennungsmotors wird hierdurch ermöglicht.

Nachfolgend sind die für die Modellbildung notwendigen Methoden vorgestellt. Im Kapitel 3.2 sind die Methoden der Messdatengewinnung beschrieben. Die Auswahl eines geeigneten Modellansatzes und einer geeigneten Modellierungsmethode ist im Kapitel 3.3 dokumentiert. Zusätzlich ist die Beschreibung der Umsetzung der Modellierungsansätze ein Bestandteil dieses Kapitels. Ein Modellierungsbeispiel und eine Darstellung ausgewählter Modellierungsergebnisse sind im Kapitel 3.4 zusammengefasst.

3.2 Versuchsplanung

Für die Vermessung eines modernen Verbrennungsmotors mit vielen Einflussgrößen stehen mehrere Methoden zur Wahl. In Abhängigkeit vom zulässigen Messaufwand, den einzusetzenden Modellierungsverfahren, der Anzahl der Einflussgrößen und weiteren Kriterien wird das eine oder andere Verfahren favorisiert.

Bei der Basisapplikation am Motorenprüfstand können zum Beispiel quasistationäre Vermessungsverfahren eingesetzt werden [Boe13, S. 44 ff.]. Bei dieser Vermessungsmethode werden die Einflussgrößen rampenförmig, mit unterschiedlichen Steigungen, verstellt. Diese Vermessungsart ist immer dann vorteilhaft, wenn die Zeitkonstanten der zu untersuchenden Größen klein sind. Bei der Betrachtung von verbrennungsnahen Größen, die sehr schnell auf ihre Endwerte einschwingen, entsteht ein relativ geringer Schleppfehler¹. Diese Fehlerart kann unter anderem durch Hin- und Rückmessung mit anschließender Mittelwertbildung kompensiert werden [Boe13, S. 44 ff.], [Ise10, S. 176 ff.]. In der vorliegenden Arbeit konzentrieren sich die Betrachtungen auf die stationäre Vermessungsmethode. Hierbei werden einzelne stationäre Messpunkte am Prüfling eingestellt. Nach Ablauf einer Einregelzeit und einer Beharrzeit wird der stationäre Messwert aufgenommen. Viele der für die Modellbildung benötigten Größen schwingen nur langsam auf ihre stationäre Endwerte ein. Beispielhaft seien an dieser Stelle die Temperaturen des Abgasstranges nach Partikelfilter genannt. Dieses Verhalten begünstigt die Anwendung der vorliegenden Vermessungsart. Die stationäre Motorvermessung trägt somit entscheidend zur Modellqualität bei.

Aufgrund der enormen Komplexität eines modernen Dieselmotors ist die Menge der zu parametrierenden Einflussgrößen sehr hoch. Es ergibt sich hierdurch ein mehrdimensionaler Parameterraum, der durch einen Versuchsplan abgedeckt werden muss. Dieser Parameterraum unterliegt infolge des nichtlinearen Verbrennungsprozesses und der Wechselwirkungen der Einflussgrößen einer starken Unsymmetrie.

¹Infolge einer Systemträgheit entsteht bei einer rampenförmigen Verstellung der Einflussgrößen eine Differenz, zwischen dem gemessenen Wert einer Messgröße und dem Wert der sich bei einer stationären Vermessung einstellen würde. Der Begriff *Schleppfehler* steht an dieser Stelle für die beschriebene Differenz.

Die Abbildung 3.2 stellt einen Überblick über die wichtigsten Schritte bei der Planung und Durchführung einer Motorvermessung zusammen. Somit beinhalten die folgenden Unterkapitel 3.2.1 bis

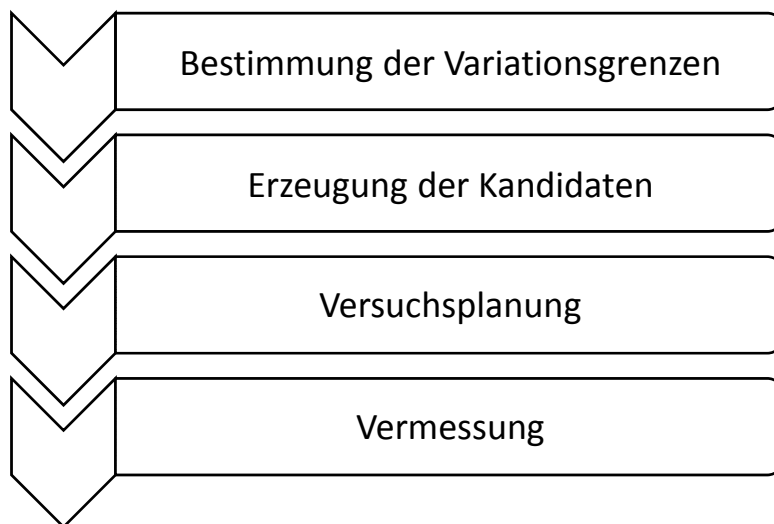


Abbildung 3.2: Die wichtigsten Schritte der Versuchsplanung für die Vermessung eines modernen Verbrennungsmotors

3.2.4 die Detailbeschreibungen der einzelnen Schritte der Motorvermessung.

Bedingt durch die dargestellten Zusammenhänge bildet die Bestimmung der Verstellgrenzen eines Versuchsplans eine entscheidende Rolle für die Modellqualität und den Versuchsaufwand. Es existieren in der Literatur unterschiedliche Ansätze und Strategien zur Bestimmung der Verstellgrenzen eines Parameterraums im Kontext der Optimierung von Verbrennungsmotoren. Einige dieser Ansätze sind im nachfolgenden Kapitel 3.2.1 beschrieben.

Nachdem die Grenzen des Parameterraums bestimmt sind, werden geeignete Kandidaten für den zu erstellenden Versuchsplan benötigt. Eine Auswahl an geeigneten Methoden der Kandidatenerzeugung ist im Kapitel 3.2.2 dargestellt.

Um bei einem möglichst geringen Messaufwand den Versuchsraum auszuleuchten, haben sich die Methoden des *Design of Experiments* (DoE) herauskristallisiert [SBH10], [Ise10, S. 130 ff.]. Ein statistischer Versuchsplan, der an den gewählten Modelltyp angepasst ist, deckt den gesamten Versuchsraum ab, wodurch der Messaufwand minimiert wird [KKG⁺13]. Eine vertiefte Darstellung der Methoden der Versuchsplanung ist dementsprechend im Kapitel 3.2.3 zu finden.

Die benötigten Vermessungen, die eine Abbildung des Verbrennungsmotorverhaltens in der Simulationsumgebung erlauben, weisen einige Besonderheiten auf. Das Kapitel 3.2.4 beschreibt diese Besonderheiten und gibt einen Überblick über den notwendigen Versuchsaufwand.

Eine gute Übersicht über das Vorgehen bei der Vermessung ist in [Ise10, S. 131 ff.] gegeben. Die theoretischen Grundlagen der Versuchsplanung können in [SBH10] eingesehen werden.

3.2.1 Bestimmung der Verstellgrenzen

Vor dem Entwurf eines Versuchsplans ist die Kenntnis über den Verstellbereich der Einflussgrößen notwendig. Die Nichtbeachtung der Verstellgrenzen hat direkte Auswirkungen auf die Modellqualität.

Für die Bestimmung der Verstellgrenzen sind mehrere Strategien und Methoden bekannt (ein Überblick ist in [Ise10, S. 170 ff.] gegeben). Um den fahrbaren Betrieb des Verbrennungsmotors zu erfassen, ist eine Vorvermessung des Prozesses notwendig. Am Beispiel eines Dieselmotors geht es hauptsächlich um die Ermittlung der betriebspunktabhängigen Einstellgrenzen der Zylinderfüllung, der Abgasrückführrate, des Raildruckes und der Einspritzcharakteristik. Als nicht fahrbar werden Versuchsbereiche angesehen, bei denen eine vom Anwender definierte Grenzverletzung vorliegt. Die Grenzen unterteilen sich hierbei in *harte* und *weiche* Grenzbedingungen. Zu den *harten* Grenzen zählen Kriterien, die den Schutz des Versuchsträgers und dessen Bauteile bzw. der Messtechnik umfassen. Zum Beispiel Abgastemperaturen vor Turbine, Turboladerdrehzahl und Zylinderdruckbegrenzung. Zu den *weichen* Grenzen zählen Beschränkungen, die den Versuchsraum auf einen sinnvollen Bereich reduzieren. Beispielfhaft werden nur Betriebspunkte zugelassen, die eine Rußzahl- oder eine HC-Emissionsobergrenze nicht überschreiten.

Bei der Ermittlung der Verstellgrenzen, wird an einem Betriebspunkt $\{n_{Mot}, M_{Mot}\}$ eine Online-Verstellung der Einflussgrößen vorgenommen, bis eine Grenzverletzung vorliegt. In [HTA⁺02] ist die *adaptive online DoE* Methode zur Optimierung eines Ottomotors mit Direkteinspritzung vorgestellt. Ausgehend von einem stabilen Betriebspunkt ermittelt diese Methode die Grenzen des Versuchsraums mittels einer gleichzeitigen sternförmigen Verstellung der Einflussgrößen. Die Verstellung erfolgt mit einer definierten Schrittweite. Falls eine Grenzverletzung vorliegt, kann die letzte erlaubte Kombination der Einflussgrößen gespeichert und das D-optimale Design des Versuchsplans angepasst werden. Die Anwendung dieser Methode an einem Dieselmotor ist in [JPM03] beschrieben.

Es existieren weitere Methoden zur Bestimmung der Versuchsraumgrenzen. Vor allem wird mit einer adaptiven Schrittweitenanpassung oder einer rampenförmigen Verstellung gearbeitet. Durch die verschiedenen Verstellstrategien wird eine messzeitoptimale Vorvermessung angestrebt. Unabhängig von der Wahl der Verstellmethode, sind die Totzeiten und die Dynamik der zur Grenzwertfassung verwendeter Messtechnik zu beachten. Ein zu großer Schleppfehler, kann zu einer verspäteten Detektion einer Grenzverletzung führen. Ist die Verstellung zu langsam, wird die angestrebte Messzeitminimierung nicht erreicht. Ein guter Kompromiss wird an dieser Stelle gesucht und kann durch Erfahrungswerte gefunden werden.

Das Ergebnis dieser Voruntersuchung sind Erkenntnisse über die betriebspunktabhängigen Verstellgrenzen der Einflussgrößen, in die zu einem späteren Zeitpunkt der Versuchsplan hinein skaliert wird. Beispielfhaft sind die Verstellgrenzen der Zylinderfüllung des untersuchten Dieselmotors sind in der Abbildung 3.3 dargestellt. Für die Bestimmung der eingezeichneten Grenzen dieser Einflussgröße wurden drei Drehzahlstufen bei mittlerer Drehmomentbelastung des Verbrennungsmotors ausgesucht. Die Darstellung erfolgt in Abhängigkeit von der berechneten Abgabeleistung an der Kurbelwelle.

Es ist durchaus denkbar, die Minimal- und Maximal-Werte der Einflussgrößen als Funktionen dar-

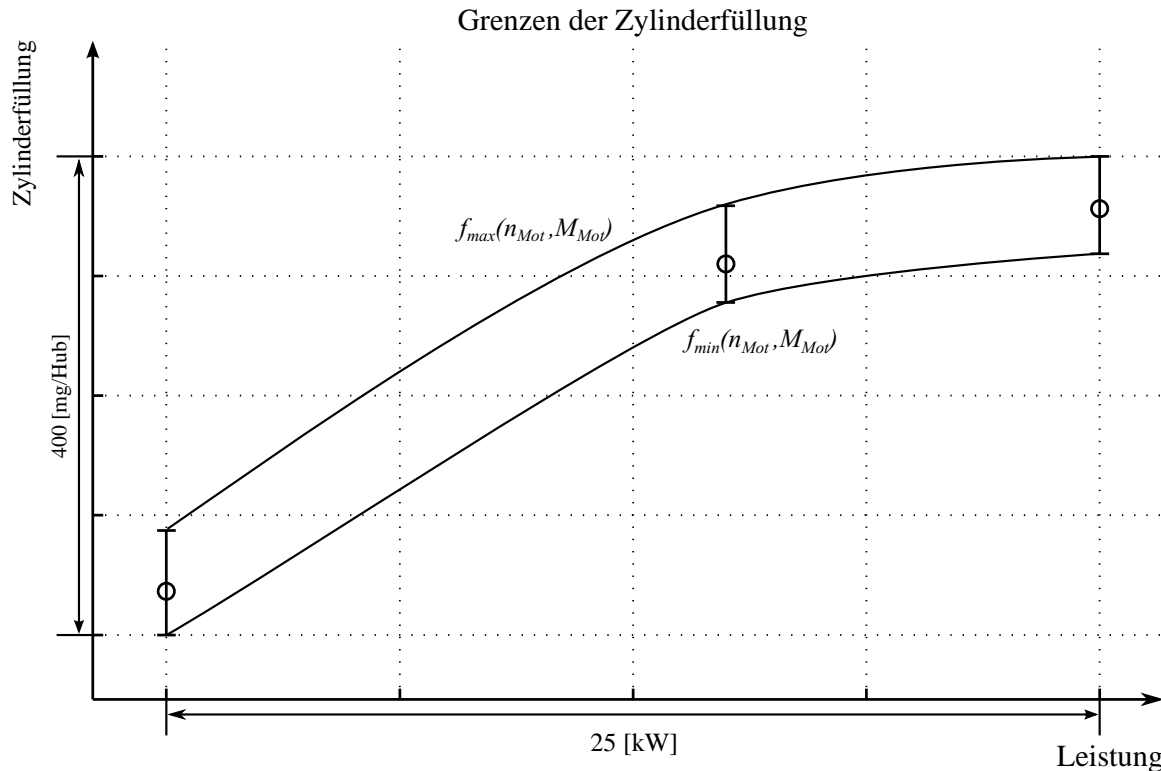


Abbildung 3.3: Änderung der Verstellgrenzen der Zylinderfüllung in Abhängigkeit von der berechneten Abgabeleistung an der Kurbelwelle

zustellen:

$$x_{min} = f_{min}(n_{Mot}, M_{Mot})$$

$$x_{max} = f_{max}(n_{Mot}, M_{Mot})$$

Hierfür reichen Vermessungen der Grenzwerte an wenigen Betriebspunkten, bei definierten Drehzahlen n_{Mot} und definierten Drehmomenten M_{Mot} , aus. Für die Darstellung der Funktionen f_{min} und f_{max} sind zum Beispiel einfache Polynommodelle geeignet.

Bei der Erzeugung der Kandidaten für den Versuchsplan, sind die in diesem Prozessschritt erstellten Modelle der Grenzwerte zu beachten.

3.2.2 Erzeugung der Kandidaten

Für die Erstellung eines Versuchsplans ist es notwendig, einen Satz an möglichen Kombinationen der Einflussgrößen x zu generieren und in einem sogenannten Kandidatensatz K zusammenzufassen. Aus den erstellten Kandidaten werden bei der späteren Versuchsplanung Stellgrößenkombinationen ausgesucht und in den Testplan T aufgenommen. Nachfolgend sind einige mathematische Methoden zur Erzeugung eines gleichverteilten Kandidatensatzes K vorgestellt. Für weiterführende Informationen sei an dieser Stelle auf [SBH10, S. 168 ff.] verwiesen.

Rasterung

Im einfachsten Fall, d.h. ohne die Beachtung der Verstellgrenzen, kann eine Rasterung des Eingangsraums zur Generierung des Kandidatensatzes verwendet werden. Jede Einflussgröße wird in l_i mögliche Einstellstufen aufgeteilt. Die Aufteilung kann äquidistant erfolgen. Eine Aufteilung in Stufen unter Berücksichtigung von *a priori* Kenntnissen ist empfehlenswert. Hierbei können die nichtlinearen Eigenschaften der einzelnen Einflussgrößen berücksichtigt werden. Zu den Nachteilen dieser Methode zählt unter anderem der exponentielle Anstieg der möglichen Kandidaten. Eine Betrachtung der Dimensionen des Kandidatensatzes ergibt:

$$K \in \mathbb{R}^{L \times N}$$

$$L = \prod_{i=1}^N l_i \quad (3.1)$$

wobei

L = Die Anzahl der Testpunkte im Kandidatensatz K

N = Die Dimension des Eingangsraums

l_i = Anzahl möglicher Einstellstufen des i -ten Eingangs

Bei einer Aufteilung jeder Einflussgröße x_i in eine feste Anzahl der Stufen l folgt:

$$L = l^N \quad (3.2)$$

Entsprechend der Gleichung (3.2) kann für $N = 10$ Einflussgrößen mit jeweils $l = 5$ Verstellstufen die Dimension des Kandidatensatzes K berechnet werden:

$$L = 5^{10} = 9765625$$

bzw.:

$$K \in \mathbb{R}^{9765625 \times 10}$$

Die Berechnung von Matrizenoperationen mit der vorliegenden Kandidatenmatrix K ist sehr zeintensiv. Zusätzlich führt diese Methode der Kandidatenerzeugung in Verbindung mit einem stark unsymmetrischen Verstellbereich zur mangelhaften Abdeckung des Versuchsraums durch die Kandidatenpunkte. In [Ise10, S. 171 f.] wird dieses anhand eines plakativen Beispiels dargestellt.

Radiale Rasterung

Für eine bessere Abdeckung des Versuchsraums kann eine sphärische bzw. radiale Rasterung angewendet werden [Ise10, S. 171 f.]. Durch die Erzeugung der Kandidaten innerhalb der verallgemeinerten Kugelkoordinaten [BSMM07, S. 209 ff.] kann der Anteil der erzeugten Kandidatenpunkte innerhalb des Versuchsraums erhöht werden. Bei der Implementierung ist dieses Vorgehen für jede zu untersuchende Drehzahl- /Lastkombination durchzuführen. Durch die schrittweise Erhöhung des

sphärischen Radiusess bei verschiedenen Winkelstellungen füllt sich den Versuchsraum mit Kandidatenpunkten.

Aus mathematischer Sicht ergibt sich durch die Anwendung dieses Verfahrens keine Absenkung der Anzahl der möglichen Kandidatenpunkte. Lediglich die Verteilung und die Verfügbarkeit im Versuchsraum ändern sich. Ein erheblicher Nachteil dieser Rastermethode ist die Ungleichmäßigkeit der Verteilung der Kandidatenpunkte im Versuchsraum. Die Abbildung 3.4 zeigt dieses anhand eines 2 – D Beispiels. Wie der Abbildung zu entnehmen ist, nimmt die Dichte der Punkte mit dem

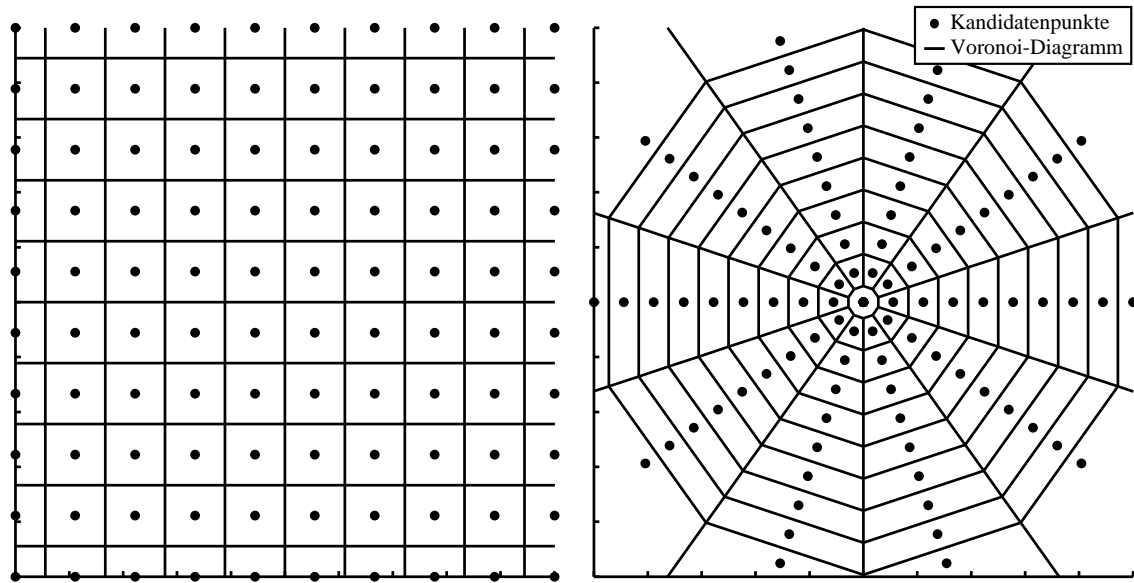


Abbildung 3.4: Vergleich der orthogonalen und der radialen Rasterung am Beispiel eines 2 – D Versuchsplans

Radius stark ab. Mit der Berechnung und Darstellung des Voronoi-Diagramms [Kle05, S. 209 ff.] kann diese Tatsache visualisiert werden.

Methoden zur Erzeugung von gleichverteilten Kandidaten im Versuchsraum

Für die Erzeugung gleichverteilter Kandidatenpunkte existieren viele unterschiedliche Methoden bzw. mathematische Algorithmen. Eine Zusammenstellung verschiedener Bewertungskriterien und eine gute Übersicht über die im Regelfall angewendeten Methoden ist in [SBH10, S. 168 ff.] zu finden.

Das Ziel der hier diskutierten Methoden ist die Erzeugung einer gleichverteilter Punktemenge in einem mehrdimensionalen Raum, auf Basis von Quasi-Zufallsfolgen von Zahlen. Die in der Literatur veröffentlichten Methoden und deren Bewertungskriterien dienen der numerischen Annäherung eines N -dimensionalen Integrals einer Verteilungsdichtefunktion $f : \mathbb{R}^N \rightarrow \mathbb{R}$:

$$\int_0^1 \dots \int_0^1 f(x_1 \dots x_N) dx_1 \dots dx_N \approx \frac{1}{L} \sum_{i=0}^{L-1} f(P_i) \quad (3.3)$$

wobei $P_i = P_1 \dots P_L$ Punkte aus dem Eingangsraum \mathbb{R}^N sind [Sob67], [Hal60].

Eine effiziente Möglichkeit der Erzeugung von gleichverteilten Kandidatensets im Versuchsraum ist die Halton-Sequenz [Hal60]. Eine ausführliche Beschreibung der Funktionsweise dieses Verfahrens findet sich unter anderem in [SBH10, S. 169 ff.]. Analog hierzu wird an dieser Stelle eine Zusammenfassung des Vorgehens gegeben.

Für einen Eingangsraum \mathbb{R}^N gelten folgende Definitionen:

$$\begin{aligned} \text{Eingangsfaktoren: } & 1 \dots i \dots N \\ \text{Testpunkte: } & 1 \dots j \dots L \end{aligned}$$

Jedem Eingangsfaktor i wird nun eine Primzahl b_i zugeordnet.

$$\begin{aligned} \text{Faktor } 1 & \implies b_1 = 2 \\ \text{Faktor } 2 & \implies b_2 = 3 \\ \text{Faktor } 3 & \implies b_3 = 5 \\ & \vdots \\ \text{Faktor } i & \implies b_i \\ & \vdots \\ \text{Faktor } N & \implies b_N \end{aligned} \tag{3.4}$$

Im Anschluss kann für jeden betrachteten Eingangsfaktor i und Testpunkt j die Zahl j einer Zerlegung zur Basis b_i unterzogen werden. Hierdurch ergibt sich folgende Darstellung:

$$j = \sum_k a_{i,j_k} b_i^k \tag{3.5}$$

wobei $a_{i,j_k} \in \mathbb{N}[0, 1 \dots b_i - 1]$.

Am Beispiel von $j = 6$ ergibt sich:

$$\text{Basis } b_i = 2: \implies j = 0 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^2 = 6 \tag{3.6}$$

$$\text{Basis } b_i = 3: \implies j = 0 \cdot 3^0 + 2 \cdot 3^1 + 0 \cdot 2^2 = 6 \tag{3.7}$$

Der quasi-zufälliger Wert $x_{i,j}$ errechnet sich wie folgt:

$$x_{i,j} = \sum_K \frac{a_{i,j_k}}{b_i^{k+1}} \tag{3.8}$$

Bezogen auf die Beispiele (3.6) und (3.7) kann geschrieben werden:

$$\begin{aligned} \text{Basis } b_i = 2: \implies j &= 0 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^2 \\ x_{2,6} &= 0 \cdot \frac{1}{2^1} + 1 \cdot \frac{1}{2^2} + 1 \cdot \frac{1}{2^3} = \frac{3}{8} \end{aligned} \tag{3.9}$$

bzw.

$$\begin{aligned} \text{Basis } b_i = 3: \quad &\implies \quad j = 0 \cdot 3^0 + 2 \cdot 3^1 + 0 \cdot 2^2 \\ &x_{3,6} = 0 \cdot \frac{1}{3^1} + 2 \cdot \frac{1}{3^2} + 0 \cdot \frac{1}{3^3} = \frac{2}{9} \end{aligned} \quad (3.10)$$

Aus der Formel (3.4) folgt, dass mit steigender Anzahl der Eingangsfaktoren auch eine immer größere Primzahl b_N gewählt werden muss. Aus der Betrachtung der Gleichung (3.8) wird ersichtlich, dass mit steigender Primzahl b_N ein immer kleinerer Abstand zwischen den Punkten erzeugt wird. Dieses erfordert, mit der steigenden Dimension des Eingangsraums, eine durchaus größere Anzahl an Punkten. Laut [SBH10, S. 171] wird $L > \prod b_i$ gefordert. Zusätzlich sind die Berechnungen mit verschiedenen Primzahlen rechenintensiver als bei einem Verfahren, welches zum Beispiel nur die Primzahl 2 nutzt.

Genau an dieser Stelle setzt das in [Sob67] veröffentlichte Sobol-Sequenz-Verfahren auf. Die Sobol-Sequenz ist eine effektive Methode, zur Erzeugung von gleichverteilten Quasi-Zufallsfolgen von Zahlen, in Räumen höherer Dimension. Für die Berechnung der Zahlenfolge in einem Rechner werden keine Multiplikationen und Additionen benötigt, nur die Schiebeoperation zur Basis 2 [Sob67].

Unabhängig vom eingesetzten Verfahren bildet eine händelbare Anzahl an Testpunkten $P \in \mathbb{R}^N$ das Ergebnis dieses Arbeitsschrittes. Die Punkte P des erzeugten Kandidatensatzes K liegen hierbei in den Versuchsraumgrenzen, sodass die Einstellung dieser bei der Versuchsdurchführung ohne Grenzverletzungen möglich ist.

$$K = \begin{bmatrix} P_1 \\ P_2 \\ \vdots \\ P_j \\ \vdots \\ P_L \end{bmatrix} = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,i} & \cdots & x_{1,N-1} & x_{1,N} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,i} & \cdots & x_{2,N-1} & x_{1,N} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{j,1} & x_{j,2} & \cdots & x_{j,i} & \cdots & x_{j,N-1} & x_{j,N} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{L,1} & x_{L,2} & \cdots & x_{L,i} & \cdots & x_{L,N-1} & x_{L,N} \end{bmatrix} \quad (3.11)$$

wobei

$$i = 1 \dots \text{bis} \dots N$$

$$j = 1 \dots \text{bis} \dots L$$

3.2.3 Versuchsplanung

Aus den vorliegenden L Testpunkten des Kandidatensatzes K , so wie diese in der Gleichung (3.11) definiert sind, müssen in diesem Arbeitsschritt M Messpunkte für den Testplan bzw. Versuchsplan T ausgesucht werden. Für diese Aufgabe werden in der vorliegenden Arbeit Methoden der statistischen Versuchsplanung herangezogen. Die Methoden des *Design of Experiments (DoE)* bieten die Möglichkeit den Messaufwand zu reduzieren, die Qualität der Modelle zu erhöhen und den gesamten Versuchsraum auszuleuchten. Aufgrund der starken Einschränkungen des Versuchsraums durch die Verstellgrenzen werden im Weiteren die D-optimalen Versuchspläne betrachtet. Weitere Entwurfsmethoden für Testpläne und zusätzliche Informationen über statistische Methoden in der Versuchsplanung können z.B. in [SBH10] eingesehen werden.

Für die Darstellung der D-Optimalität werden an dieser Stelle Polynommodelle betrachtet. Das Optimalitätskriterium ist jedoch nicht auf die polynomialen Modellansätze beschränkt.

Die Gleichung (3.12) gibt einen Überblick über einen Modellansatz mit zwei Eingangsparametern.

$$\begin{aligned}
 y &= a_m x_1^m + b_m x_2^m + a_{m-1} x_1^{m-1} + b_{m-1} x_2^{m-1} + \dots \\
 &\dots + c_{j,k} x_1^j x_2^k + \dots + c_{1,2} x_1 x_2 + \dots \\
 &\dots + a_1 x_1 + b_1 x_2 + c_0
 \end{aligned} \tag{3.12}$$

Für die Bestimmung der Modellkoeffizienten $a_i, b_i, c_i \dots$ sind somit mindestens n_k Messungen notwendig, wobei n_k die Anzahl der Modellkoeffizienten ist. Hierdurch wird die Durchführung einer Regressionsanalyse ermöglicht [BSMM07, S. 781 ff.]. Für die Bewertung von verschiedenen Eigenschaften und Kennwerten, wie die Messstreuung oder die Modellgüte, ist die messtechnische Gewinnung eines Verifikationsdatensatzes notwendig. Somit wird die Größe des geforderten Vermessungsumfanges $M > n_k$ gefordert.

Der vorliegende Modellansatz kann in eine Matrixgleichung überführt werden¹:

$$y = \begin{bmatrix} x_1^m & x_2^m & x_1^{m-1} & x_2^{m-1} & \dots & x_1 & x_2 & 1 \end{bmatrix} \cdot \begin{bmatrix} a_m \\ b_m \\ a_{m-1} \\ b_{m-1} \\ \vdots \\ a_1 \\ b_1 \\ c_0 \end{bmatrix} \tag{3.13}$$

Anschließend sind die M Messungen eines Versuchsplans T für die Berechnung der Modellkoeffizienten in die folgende Matrixform zu integrieren:

$$\underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_i \\ \vdots \\ y_M \end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix} x_{1,1}^m & x_{2,1}^m & x_{1,1}^{m-1} & x_{2,1}^{m-1} & \dots & x_{1,1} & x_{2,1} & 1 \\ x_{1,2}^m & x_{2,2}^m & x_{1,2}^{m-1} & x_{2,2}^{m-1} & \dots & x_{1,2} & x_{2,2} & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{1,i}^m & x_{2,i}^m & x_{1,i}^{m-1} & x_{2,i}^{m-1} & \dots & x_{1,i} & x_{2,i} & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{1,M}^m & x_{2,M}^m & x_{1,M}^{m-1} & x_{2,M}^{m-1} & \dots & x_{1,M} & x_{2,M} & 1 \end{bmatrix}}_{\mathbf{X}} \cdot \underbrace{\begin{bmatrix} a_m \\ b_m \\ a_{m-1} \\ b_{m-1} \\ \vdots \\ a_1 \\ b_1 \\ c_0 \end{bmatrix}}_{\mathbf{w}} \tag{3.14}$$

$$\mathbf{y} = \mathbf{X} \cdot \mathbf{w} \tag{3.15}$$

Hierbei ist \mathbf{y} die Zusammenstellung der gemessenen Zielgrößenwerte, \mathbf{X} enthält die Informationen über die Verstellung der Einflussgrößen und den Modellansatz und \mathbf{w} beinhaltet die gesuchten Mo-

¹Die Terme der Wechselwirkungen c_i wurden aus Gründen der Übersichtlichkeit weggelassen.

dellkoeffizienten. Für die Berechnung der im Vektor \mathbf{w} zusammengestellten Koeffizienten ist die Lösung der Gleichung (3.16) erforderlich.

$$\begin{aligned}\mathbf{y} &= X\mathbf{w} \\ X^T\mathbf{y} &= X^T X\mathbf{w} \\ \mathbf{w} &= (X^T X)^{-1} X^T \mathbf{y}\end{aligned}\tag{3.16}$$

Durch die Minimierung der Determinante des Terms $(X^T X)^{-1}$ bzw. die Maximierung der Determinante von $(X^T X)$ entsteht ein D-optimaler Versuchsplan. Bildlich gesprochen wird der Informationsgehalt der Informationsmatrix $(X^T X)$ erhöht [SBH10, S. 54]. Die Maximierung der Determinante trägt entscheidend zur Stabilität der berechneten Koeffizienten bei, wobei der Einfluss des Messrauschens auf die Berechnungsergebnisse reduziert wird. Anhand eines 2-Parameter-Beispiels werden die Aussagen verdeutlicht werden.

Bei einer unbekanntem Funktion $y = f(x)$ und einem linearem Modellansatz $y = w_1 \cdot x + w_0$ müssen mindestens zwei Einträge in der Versuchsmatrix X vorhanden sein.

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \end{bmatrix} \cdot \begin{bmatrix} w_0 \\ w_1 \end{bmatrix}\tag{3.17}$$

Entsprechend ergibt sich für die Determinante $(X^T X)$:

$$\det(X^T X) = (x_1 - x_2)^2\tag{3.18}$$

Durch das Erhöhen des Abstands zwischen den zu vermessenden Stützstellen x_1 und x_2 wird die Determinante von $(X^T X)$ maximiert. Die Stabilität der Berechnung der Modellkonstanten w_0 und w_1 wird hiermit verbessert und eine Steigerung der Modellqualität erreicht. Die Abbildung 3.5 visualisiert den Einfluss des Messrauschens auf die Modellkoeffizienten und die Auswirkungen der D-Optimalität.

Für die Berechnung der D-Optimalität werden verschiedene Verfahren eingesetzt. Im einfachsten Fall werden aus dem Kandidatensatz K einzelne Kandidaten ausgesucht und in einem Versuchsplan T zusammengestellt. Durch ein iteratives Tauschen der Kandidatenpunkte ändert sich bei jeder Iteration auch die Informationsmatrix $(X^T X)$. Das Ziel ist es, durch eine geeignete Wahl der Kandidatenpunkte die Determinante der Matrix $(X^T X)^{-1}$ zu minimieren.

3.2.4 Vermessung

Nach der Erstellung eines Versuchsplans T kann mit der Vermessung des Prozesses begonnen werden. Im vorliegenden Fall eines Dieselmotors werden die Führungsgrößen des Luft- und Kraftstoffpfads nach dem entworfenen Testplan T verstellt. Entsprechend den im Kapitel 1.3 definierten Zielen der Arbeit wird ein dynamischen Verbrennungsmotormodell erstellt. Den Modellen zugrunde liegenden Vermessungen des Verbrennungsmotors müssen somit Informationen über das stationäre und das dynamische Motorverhalten beinhalten. Das stationäre Verhalten der betrachteten Steller-Sensor-Strecke (s. Kapitel 3.1, Abbildung 3.1) kann durch die Aufnahme von stationären Punkten

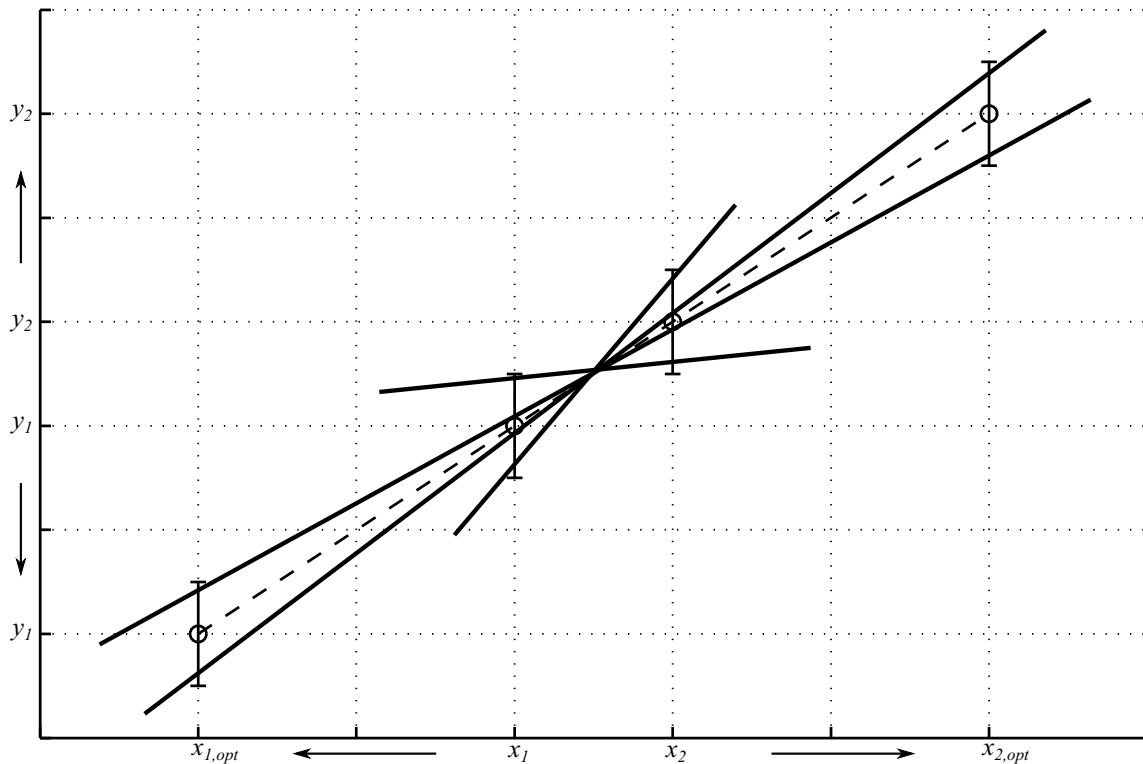


Abbildung 3.5: Reduktion des Einflusses des Messrauschens bei Anwendung der DoE-Methoden

einer DoE-Vermessung erfasst werden. Für die Gewinnung der Informationen über das dynamische Verhalten sind die Übergänge zwischen den stationären Punkten interessant. Ein Teil, der bei einer stationären DoE-Vermessung verstellten Einflussgrößen, wird über die in der Steuergerätesoftware implementierten Regler eingestellt. Aufgrund der modellbasierten Regelung der aktuellen Dieselmotorfamilie ([NKJ⁺12, S. 109], [Nö10, S.73 ff.]) sind die dynamischen Übergänge zwischen den stationären Messpunkten einer durchgeführten DoE-Vermessung nicht Teil der Eigendynamik des Systems. Somit wird eine eigenständige, dynamische Vermessung notwendig. Hierbei werden die Steller des Systems direkt angeregt.

Stationäre Vermessung Diese Vermessungsart beinhaltet die Aufnahme der stationären Messpunkte des Vermessungsplans. Die Verstellgrößen werden hierbei durch die im Steuergerät implementierte Regler eingestellt. Die dynamischen Übergänge zwischen den stationären Punkten sind dementsprechend nicht durch die Eigendynamik des Systems bestimmt und können nicht für die Vermessung des dynamischen Systemverhaltens herangezogen werden.

Für diese Vermessungsart werden bei der aktuellen Komplexität des Dieselmotors ca. 500 – 1000 Messpunkte benötigt.

Dynamische Vermessung Für die Aufnahme des dynamischen Systemverhaltens werden aus den aufgenommenen stationären Messpunkten wenige Messpunkte ausgesucht, mit dem Ziel die dynamischen Übergänge zwischen den Messpunkten aufzuzeichnen. Im Unterschied zur stationären Vermessung werden bei dieser Vermessungsart die Stellereinstellungen des Verbrennungsmotors direkt eingestellt, sodass das Einschwingverhalten der Eigendynamik des Systems ent-

spricht. Ein mögliches Kriterium für die Auswahl der Messpunkte könnte der Ladedruck sein. An einem Betriebspunkt (n_{Mot}, M_{Mot}) könnten zum Beispiel zwei Punkte mit möglichst großem Ladedruckunterschied ausgesucht werden.

Für die vorliegende Vermessungsart sind ca. 50 – 100 Messpunkte und deren Übergänge ausreichend.

3.3 Modellierung des Verbrennungsmotorverhaltens

3.3.1 Modellierungsverfahren

Für die Modellierung des vermessenen Prozesses, stehen unterschiedliche Methoden zur Wahl. Der Standardprozess des DoE-Methodenkastens sieht die Modellierung der Zielgröße mit einem Polynommodell vor. Dieses ist für das weitere Vorgehen nicht zwingend erforderlich. An dieser Stelle werden die Eigenschaften von einigen Modellierungsverfahren vorgestellt. Die Polynome, die neuronalen Netze und die Gaußprozessmodelle bilden den Schwerpunkt der Betrachtungen. Die Modellierungsverfahren weisen, bezogen auf das Einsatzgebiet der Motorapplikation, unterschiedliche Eigenschaften auf [KKG⁺13], [RSK⁺13].

Die Polynommodelle sind im Vergleich zu den anderen beiden Verfahren sehr robust. Sie zeigen jedoch Schwächen bei starken nichtlinearen Zusammenhängen bzw. Sprungstellen und unter Einbeziehung von größeren Betriebsbereichen [KKG⁺13].

Unter dem Oberbegriff „neuronalen Netze“ existiert eine Vielzahl verschiedener Strukturen. Die Multilayer-Perceptron (MLP) Struktur ist eine verbreiterte Modellierungsmethode. Diese Struktur weist theoretisch eine hohe Genauigkeit auf und ist prinzipiell für die Abbildung von starken Nichtlinearitäten geeignet. Um gute Interpolationseigenschaften sicherzustellen und Overfitting zu vermeiden, ist unter anderem eine höhere Anzahl an Messdaten notwendig, was mit dem Konzept einer DoE-Vermessung nicht immer konform ist.

Die statistischen Lernverfahren auf Basis von Gaußprozessen verbinden eine hohe Genauigkeit, Reproduzierbarkeit und Interpolationsfähigkeit in einem Ansatz [RSK⁺13]. Bei schwach durchleuchteten Versuchsräumen kann für jeden Datenpunkt eine Gaußfunktion definiert und parametrisiert werden. Die Parametrisierung ist so zu bestimmen, dass die zu modellierenden Zusammenhänge mit der höchsten Wahrscheinlichkeit wiedergegeben werden. Dieser Ansatz hat sich im vorliegenden Einsatzgebiet als geeignet erwiesen und erlaubt eine automatisierte Bestimmung der gesuchten Funktionenschar [KKG⁺13]. Aufgrund ihres Aufbaus weisen diese Modellierungsansätze jedoch einen erhöhten Rechenaufwand auf [RSK⁺13].

Der mathematische Modellansatz und die Methoden zur Bestimmung der Modellkoeffizienten sind für die Polynommodelle, für die Neuronalen Netze und für die Gaußprozessmodelle nachfolgend zusammengefasst.

Für alle Modellierungsmethoden wird von einem Messdatensatz $D = \{(\mathbf{x}_i, y_i) \mid i = 1 \dots M\}$ ausgegangen. Der Vektor \mathbf{x}_i besteht aus einer Zusammenfassung der Einflussgrößen x_1 bis x_N . Der zum Vektor \mathbf{x}_i zugehörige Messwert der Zielgröße ist durch y_i abgebildet.

Polynome

Analog zur Gleichung (3.15) werden für die polynomiale Abbildung der im Messdatensatz D enthaltenen Informationen folgende Definitionen getroffen:

$$f(\mathbf{x}) = X\mathbf{w} \quad y = f(\mathbf{x}) + \epsilon \quad (3.19)$$

wobei \mathbf{x} die Einflussgrößen und y die Zielgröße repräsentieren. Die Designmatrix ist durch X , die Modell- bzw. Polynomkoeffizienten sind durch \mathbf{w} und der Modellfehler durch ϵ dargestellt.

Die gesuchten Koeffizienten können mit dem Least-Squares-Verfahren [BSMM07, S. 277 ff.] bestimmt werden, wobei folgende Gleichung gelöst wird:

$$\sum_{i=1}^L (y_i - f(\mathbf{x}_i))^2 = \min! \quad (3.20)$$

Die Formulierung des linearen Gleichungssystems in der Matrizenform entspricht den Gleichungen (3.15) und (3.16).

$$X^T X \mathbf{w} = X^T \mathbf{y} \quad (3.21)$$

Bei der Lösung des Gleichungssystems (3.21) und der Bestimmung von \mathbf{w} mit Hilfe der symmetrischen Matrix $X^T X$ werden im Allgemeinen numerische Lösungsmethoden angewendet, so wie diese in [BSMM07, S. 278 ff.] beschrieben sind.

Bei der praktischen Umsetzung eines Polynommodells wird zusätzlich die Relevanz der einzelnen Polynomterme betrachtet, so wie diese in der Gleichung (3.12) definiert sind. Durch das Hinzunehmen und das Ausschließen der einzelnen Terme wird die Designmatrix X iterativ angepasst. Für die Erstellung eines qualitativ hochwertigeren Modells werden verschiedene Bewertungskriterien betrachtet, wie zum Beispiel das Bestimmtheitsmaß (Kapitel A.5) oder der RMSE (Kapitel A.4). Unerlässlich an dieser Stelle ist auch die Auswertung verschiedener statistischer Kennwerte und Darstellungen, wie die Darstellung der Residuen als Funktion der Einflussgrößen bzw. der Zielgröße oder die Betrachtung der Streuung der Messwerte bezogen auf den Modellausgang. Hierdurch werden unter anderem die systematischen Fehler vermieden und eine Aussage über die Größenordnung des Modellfehlers ermöglicht.

Künstliche Neuronale Netze

Im Rahmen dieser Arbeit sind stellvertretend für die Künstlichen Neuronalen Netze (KNN) die dreischichtigen Multi-Layer-Perceptron (MLP) Netze betrachtet. Eine gute Übersicht über verschiedene Strukturen der Neuronalen Netze kann unter anderem in [Sch10b, S. 38 ff.], [Win02] oder [Aye98] eingesehen werden.

Mit einem MLP-Netz lassen sich *theoretisch* beliebig genaue Approximationen einer *stetigen* oder sogar einer *messbaren* Funktion realisieren [HSW89], [Aye98, S. 8, 9], [Win02, S. 22]. Wobei der Nachweis der theoretischen Approximationsfähigkeit der MLP-Netze keine Hinweise auf die Anzahl der

notwendigen Neuronen in der Zwischenschicht liefert. Weiterhin ist ein MLP-Netz mit nur einer Zwischenschicht für die theoretischen Betrachtungen ausreichend, obwohl beim praktischen Einsatz die Aufstellung einer mehrschichtigen Struktur sinnvoll sein kann. Die Anwendung eines iterativen Lernverfahrens könnte hierdurch Vorteile bei der Berechnung der Wichtungen aufweisen [Aye98, S. 9].

Der Aufbau eines MLP-Netzes mit zwei Eingängen, drei versteckten Neuronen und einem Ausgang (2|3|1) ist in der Abbildung 3.6 dargestellt.

Die Berechnung des Ausgangs kann durch folgende Formel beschrieben werden:

$$y = b_2 + \begin{bmatrix} w_1^2 & w_2^2 & w_3^2 \end{bmatrix} \cdot \delta \left(\begin{bmatrix} b_1^1 \\ b_2^1 \\ b_3^1 \end{bmatrix} + \begin{bmatrix} w_{1,1}^1 & w_{1,2}^1 \\ w_{2,1}^1 & w_{2,2}^1 \\ w_{3,1}^1 & w_{3,2}^1 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \right) \quad (3.22)$$

oder in der allgemeinen Form

$$y = B_2 + W_2 \delta(B_1 + W_1 \mathbf{x}) \quad (3.23)$$

wobei B_2 , W_2 , B_1 und W_1 als Modellkoeffizienten \mathbf{w} bezeichnet werden.

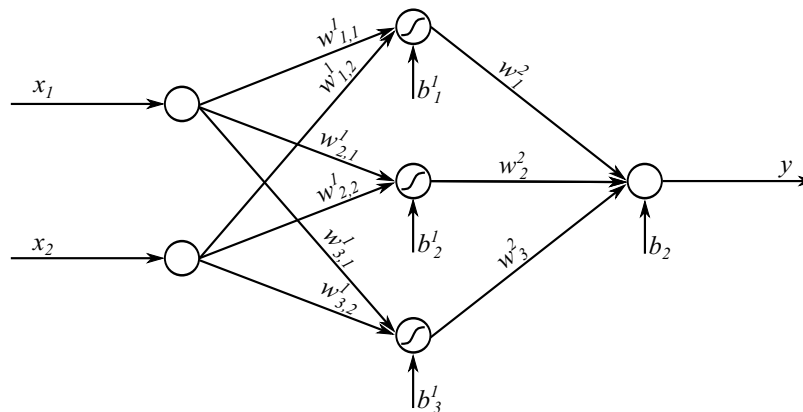


Abbildung 3.6: Beispielhafter Aufbau eines dreischichtigen MLP-Netzes, bestehend aus zwei Eingangsneuronen, drei Neuronen in der verdeckten Schicht und einem Ausgangsneuron

Für die Aktivierungsfunktion $\delta(x)$ kann zum Beispiel die Log-Sigmoidale-Funktion (3.24) oder die hyperbolische Tangens-Funktionen (3.25) eingesetzt werden [Aye98, S. 7], [Win02, S. 14], [SBH10, S. 210 f.]

$$\delta(x) = \frac{1}{1 + e^{-x}} \quad (3.24)$$

$$\delta(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (3.25)$$

Die Anwendung der Aktivierungsfunktion auf einen Vektor erfolgt elementweise.

Für die Bestimmung der Modellkoeffizienten \mathbf{w} ist die Anwendung eines Lernverfahrens notwendig

[Win02, S. 19], [Aye98, S. 17]. Hierbei wird die Minimierung einer Kostenfunktion durchgeführt. Die Kostenfunktion beschreibt analog zur Gleichung (3.20) die Abweichungen zwischen dem Modellausgang und den Messdaten.

$$J(\mathbf{w}) = \frac{1}{2} e(\mathbf{w})^T e(\mathbf{w}) \quad (3.26)$$

wobei $e(\mathbf{w})$ die Residuen repräsentiert und sich wie folgt zusammensetzt:

$$e(\mathbf{w}) = \begin{bmatrix} y_1 - f(\mathbf{x}_1, \mathbf{w}) \\ y_2 - f(\mathbf{x}_2, \mathbf{w}) \\ \vdots \\ y_i - f(\mathbf{x}_i, \mathbf{w}) \\ \vdots \\ y_M - f(\mathbf{x}_M, \mathbf{w}) \end{bmatrix} \quad (3.27)$$

Durch eine gezielte Anpassung der Kostenfunktion können weiterhin die Residuen an ausgewählten Datenpunkten unterschiedlich gewichtet werden. Zusätzlich kann ein explizites Extrapolationsverhalten aufgeprägt werden, sodass die Berechnung plausibler Modellwerte auch in diesem Fall sichergestellt ist [Win02, S. 25].

$$J(\mathbf{w}) = \frac{1}{2} e(\mathbf{w})^T V e(\mathbf{w}) \quad (3.28)$$

Die Bewertungsmatrix V ist wie folgt definiert:

$$V = \text{diag}([\mu_1 \dots \mu_L]) \quad (3.29)$$

Werden nun die Bewertungskoeffizienten $\mu_i = 1$ gewählt, so ergibt sich eine gleichmäßige Bewertung der Residuen. Für den Fall der Aufnahme von Extrapolationsdaten in den Lerndatensatz, können die zugehörigen Bewertungskoeffizienten $\mu < 1$ gewählt werden. Dieses Vorgehen ist nicht auf die neuronalen Netze beschränkt und kann bei anderen Modellierungsverfahren angewendet werden. Weitere Modifikationen der Kostenfunktion $J(\mathbf{w})$ sind durch die Aufnahme zusätzlicher Bewertungsterme möglich. Beispielhaft kann ein Term zur Gewichtung bestimmter Wichtungen bezüglich ihrer Maximalwerte definiert werden.

Für die Minimierung der Kostenfunktion sind in der Literatur mehrere Ansätze beschrieben. Die Backpropagation-Methode [RHW86] ist eine der Standardmethoden, die bei der Bestimmung der Modellkoeffizienten \mathbf{w} eingesetzt wird. Ausgehend von einer Initialbedatung der Modellkoeffizienten wird iterativ in Richtung eines lokalen Minimums der Kostenfunktion vorgegangen.

$$\mathbf{w}_{neu} = \mathbf{w}_{alt} - \alpha \frac{dJ(\mathbf{w})}{d\mathbf{w}} \quad (3.30)$$

Für die Änderung der einzelnen Wichtungen $w_{i,j}$ ergibt sich somit folgende Berechnungsvorschrift:

$$\Delta w_{i,j} = -\alpha \frac{dJ(\mathbf{w})}{dw_{i,j}} \quad (3.31)$$

Die Wahl einer geeigneten Schrittweite α ist für die Schnelligkeit der Konvergenz des Verfahrens entscheidend. Folglich wird diese mit Hilfe einer Adaptionstrategie angepasst.

Eine effizientere Methode zur Lösung des nichtlinearen Optimierungsproblems ist der Levenberg-Marquardt-Algorithmus [Mar63], [Mor77]. Dieser weist im Vergleich zur Backpropagation-Methode eine schnellere Konvergenz auf [Win02, S. 20].

Weiterhin ist die Anzahl der im Modell verwendeten Neuronen, und damit zur Verfügung stehender Freiheitsgrade, für die Qualität des Modells entscheidend. Eine Methode zur automatischen Ermittlung der optimalen Neuronenzahl ist in [ALWT98] und [Win02, S. 23 f.] beschrieben. Hierbei erfolgt eine Aufteilung des zur Verfügung stehenden Messdatensatzes in einen Trainings- und einen Validierungsdatensatz. Das Lernverfahren wird mit verschiedenen Anzahlen der Neuronen mit den Trainingsdaten durchgeführt. Zur Bewertung des Modells wird der SSE-Fehler (Kapitel A.3) bezüglich der Trainings- und Validierungsdaten betrachtet.

Die Abbildung 3.7 visualisiert einen typischen Verlauf des Fehlermaßes über die Neuronenzahl. Der Tiefpunkt des Validierungsfehlers markiert die optimale Neuronenzahl. Die Wahl einer größeren Anzahl der Neuronen führt zum Overfitting. Bei kleineren Neuronenzahlen können die gegebenen Abhängigkeiten nicht in der optimalen Qualität modelliert werden. Die Anwendung der beschriebenen Methode wurde am Beispiel der Modellierung eines Verbrennungsmotors mit Saugrohreinspritzung in [ALWT98] veröffentlicht.

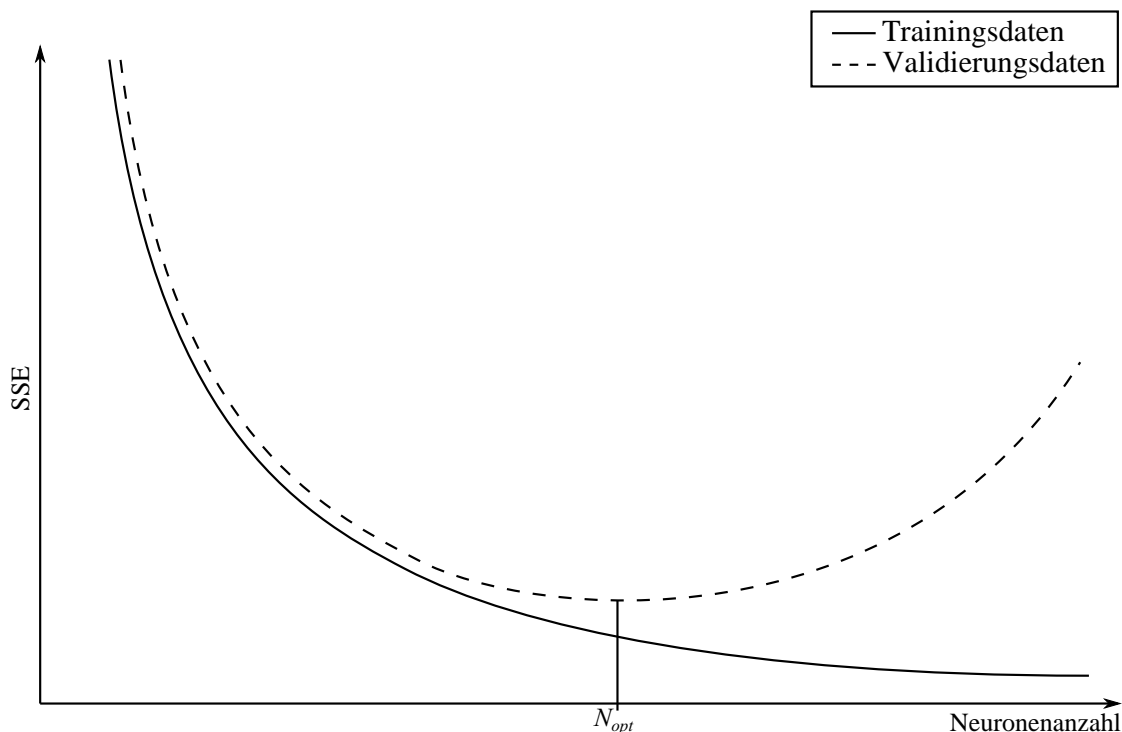


Abbildung 3.7: SSE-Fehler der Trainings- und Validierungsdaten in Abhängigkeit von der Neuronenzahl

Gaußprozessmodelle

In diesem Unterkapitel wird die Approximation des gemessenen Datensatzes D mit statistischen Lernverfahren behandelt. Dieser Themenkomplex ist recht umfangreich, für weiterführende Informationen sei an dieser Stelle auf [Bis06], [RW06], [Kus06] oder [Kra11] verwiesen. Eine Anwendung dieser Verfahren im Bereich der Motorapplikation ist in [KKG⁺13] beschrieben.

Bei der Modellierung von physikalischen Zusammenhängen mit Gaußprozessen können zusätzlich zu den Schätzwerten im Interpolationsbereich auch Informationen über das Vertrauensintervall jedes Interpolationswertes angegeben werden. Im Sinne des Ansatzes von Bayes wird vor der Modellierung eine *a priori* Verteilung für die zu approximierende Funktion $f(\mathbf{x})$ ausgewählt. Ohne die Messdaten des Datensatzes D zu betrachten, spiegelt die *a priori* Verteilung das Vorwissen über den zu modellierenden Prozess wieder. Erst durch das Hinzunehmen der Messdaten aus D wird im Anschluss die *a posteriori* Verteilung bestimmt. Diese beschreibt die gesuchten Zusammenhänge und stellt somit das gesuchte Modell dar.

In [RW06, S. 13] ist ein Gaußprozess als eine Zusammenstellung von Zufallsvariablen definiert, die über eine gemeinsame Gaußverteilung verfügen. Weiterhin ist die vollständige Spezifikation von einem Gaußprozess durch eine Erwartungsfunktion $m(\mathbf{x})$ und eine Kovarianzfunktion $k(\mathbf{x}, \mathbf{x}')$ eines Prozesses $f(\mathbf{x})$ beschrieben [RW06, S. 13],[Kra11, S. 8]:

$$f(x) \sim GP(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')) \quad (3.32)$$

Im Grunde handelt es sich um eine Verallgemeinerung der mehrdimensionalen bzw. multivariaten Gaußverteilung. Im einfachsten 1 – D Fall entspricht diese der Normalverteilung $\mathcal{N}(\mu, \sigma_n^2)$ mit dem Mittelwert μ und der Varianz σ_n^2 .

$$f(x) \sim \mathcal{N}(\mu, \sigma_n^2) \quad (3.33)$$

In einem N -dimensionalen Fall gilt somit folgender Zusammenhang:

$$f(\mathbf{x}) \sim \mathcal{N}(m(\mathbf{x}), \Sigma(\mathbf{x}, \mathbf{x})) \quad (3.34)$$

wobei

$$\begin{aligned} m(\mathbf{x}) &: \mathbb{R}^N \rightarrow \mathbb{R} \\ \Sigma(\mathbf{x}, \mathbf{x}) &\in \mathbb{R}^{N \times N} \quad \text{mit} \quad \Sigma_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j) : \mathbb{R}^N \rightarrow \mathbb{R} \end{aligned}$$

Häufig werden lineare Erwartungswertfunktionen $m(\mathbf{x})$ und stationäre Kovarianzfunktionen $k(\mathbf{x}, \mathbf{x}')$ festgelegt [Kra11, S. 9]. Beispielhaft kann für $m(\mathbf{x})$ eine Konstante oder eine lineare Funktion mit einem frei wählbaren Parameter (Hyperparameter) gewählt werden. Das einfachste Beispiel für eine Kovarianzfunktion $k(\mathbf{x}, \mathbf{x}')$ ist die sogenannte *squared exponential* Funktion:

$$k(\mathbf{x}, \mathbf{x}') = e^{-\frac{1}{2l}|\mathbf{x}-\mathbf{x}'|^2} \quad (3.35)$$

wobei l einen Hyperparameter für die charakteristische Längsskalierung des Prozesses darstellt. Im Allgemeinen können die Erwartungs- und Kovarianzfunktionen über mehrere Hyperparameter

verfügen, die zu einem Parametervektor θ zusammengefasst werden.

Bei der Wahl einer *a priori* Verteilung werden die Hyperparameter θ definiert und damit ein erwartetes Verhalten der zu modellierenden Funktion $f(\mathbf{x})$ festgelegt. Um diesen Prozessschritt zu visualisieren werden in der Abbildung 3.8 *oben* an den Stützstellen x_1, x_2, x_3 und x_4 jeweils mehrere zufällige Funktionswerte erzeugt. Anschließend werden für einen ausgewählten Satz an Hyperparametern θ mehrere mögliche bzw. erwartete Funktionsverläufe einer noch nicht gemessenen Funktion $f(\mathbf{x})$ berechnet und eingezeichnet. Somit wird mit der *a priori* Verteilung eine Aussage über die Eignung der Hyperparameter θ für die Darstellung des zu modellierenden Prozesses generiert. Wenn die möglichen Funktionsverläufe den Erwartungen entsprechen wird mit der Berechnung der *a posteriori* Verteilung begonnen. An dieser Stelle sind dementsprechend die Messdaten $[\mathbf{x}_i, f_i | i = 1 \dots M]$ zu betrachten. Gedanklich werden aus den *a priori* festgelegten Funktionen diejenigen übernommen, die mit den Beobachtungen $(\mathbf{x}, f(\mathbf{x}))$ übereinstimmen (s. Abbildung 3.8 *unten*) [RW06, S. 15].

Ein Gaußprozessmodell liefert nicht nur Informationen über einen möglichen Verlauf der zu modellierenden Funktion, sondern auch über das Vertrauensintervall der Modellwerte. In der Abbildung 3.9 ist die Approximation eines beispielhaften Messdatensatzes dargestellt. Zusätzlich zu den Messpunkten und dem Verlauf der Modellfunktion $f(\mathbf{x})$, ist in den Interpolationsbereichen eine Visualisierung des Vertrauensintervalls durchgeführt worden (an dieser Stelle 0.5σ - Umgebung). Für die Berechnung neuer Funktionswerte $f_*(\mathbf{x})$ bzw. f_* im Interpolationsbereich mit dem Gaußprozessmodell sind sowohl die definierten Hyperparameter θ , als auch die Messdaten $[\mathbf{x}_i, f_i | i = 1 \dots M]$ von Bedeutung. Für die multivariate Verteilung der Trainingsdaten f gilt entsprechend der Gleichung (3.34):

$$f(\mathbf{x}) \sim \mathcal{N}(m(\mathbf{x}), \Sigma(\mathbf{x}, \mathbf{x}))$$

Für die multivariate Verteilung der Mess- bzw. Trainingsdaten f und der Interpolationsdaten f_* kann dementsprechend geschrieben werden:

$$\begin{bmatrix} f \\ f_* \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} m(\mathbf{x}) \\ m_*(\mathbf{x}) \end{bmatrix}, \begin{bmatrix} \Sigma(\mathbf{x}, \mathbf{x}) & \Sigma(\mathbf{x}, \mathbf{x}_*) \\ \Sigma(\mathbf{x}_*, \mathbf{x}) & \Sigma(\mathbf{x}_*, \mathbf{x}_*) \end{bmatrix} \right) \quad (3.36)$$

Die bedingte Verteilung der Interpolationswerte $(f_* | \mathbf{x}_*, \mathbf{x}, f)$ unter der Annahme der durchgeführten Beobachtungen (\mathbf{x}, f) und der Werte der Interpolationsstützstellen \mathbf{x}_* wird durch folgende Formel bestimmt:

$$f_* | \mathbf{x}_*, \mathbf{x}, f \sim \mathcal{N} \left(\Sigma(\mathbf{x}_*, \mathbf{x}) \Sigma(\mathbf{x}, \mathbf{x})^{-1} f, \Sigma(\mathbf{x}_*, \mathbf{x}_*) - \Sigma(\mathbf{x}_*, \mathbf{x}) \Sigma(\mathbf{x}, \mathbf{x})^{-1} \Sigma(\mathbf{x}, \mathbf{x}_*) \right) \quad (3.37)$$

siehe [RW06, S. 15, 16, 200] und [Kra11, S. 12].

Die Wahl der Hyperparameter θ ist für die Modellierungsergebnisse entscheidend. Die Berechnung eines optimalen Parametersatzes erfolgt durch die Anwendung des Satzes von Bayes [BSMM07, S. 750 ff.] und einiger Methoden zur Bestimmung der bedingten Wahrscheinlichkeit $p(\theta | \mathbf{x}, f(\mathbf{x}))$. Für weitere Informationen zur Optimierung von Hyperparameter sei an dieser Stelle auf [RW06, S. 105 ff.] verwiesen. Als Ergebnis der Optimierung wird nicht nur ein verbesserter Verlauf der Modellfunk-

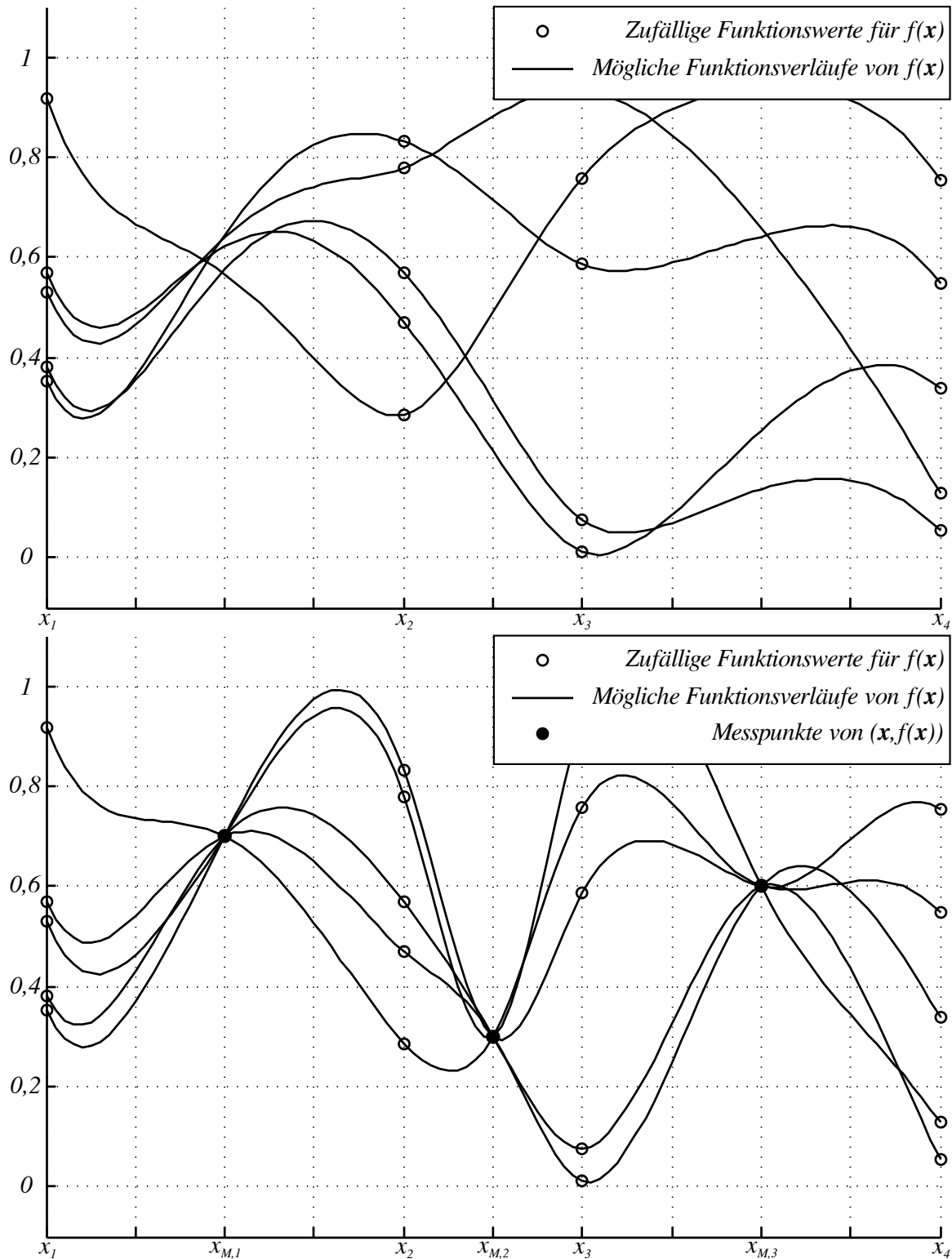


Abbildung 3.8: Übersicht über die Erstellung eines Gaußprozessmodells. **Oben** *a priori* Verteilung: An den Stützstellen x_1, x_2, x_3 und x_4 sind jeweils mehrere zufällige Funktionswerte $f(\mathbf{x})$ aufgetragen. Die Funktionswerte sind durch mögliche Funktionsverläufe von $f(\mathbf{x})$ verbunden. **Unten** *a posteriori* Verteilung: zu den Stützstellen x_1, x_2, x_3 und x_4 kommen die Messwertstützstellen $x_{M,1}, x_{M,2}$ und $x_{M,3}$ hinzu. Alle möglichen Funktionsverläufe von $f(\mathbf{x})$ gehen durch die Messwerte $(\mathbf{x}, f(\mathbf{x}))$ durch.

tion, sondern auch eine Optimierung der Vertrauensintervalle erreicht. Für das in der Abbildung 3.9 dargestellte Beispiel werden durch die Optimierung vor allem starke Verbesserungen im Verlauf des Vertrauensintervalls erzielt. Die Abbildung 3.10 zeigt den Verlauf der erstellten Funktion mit den optimierten Hyperparametern. Aus der Sicht des Anwenders kann für die endgültige Beschrei-

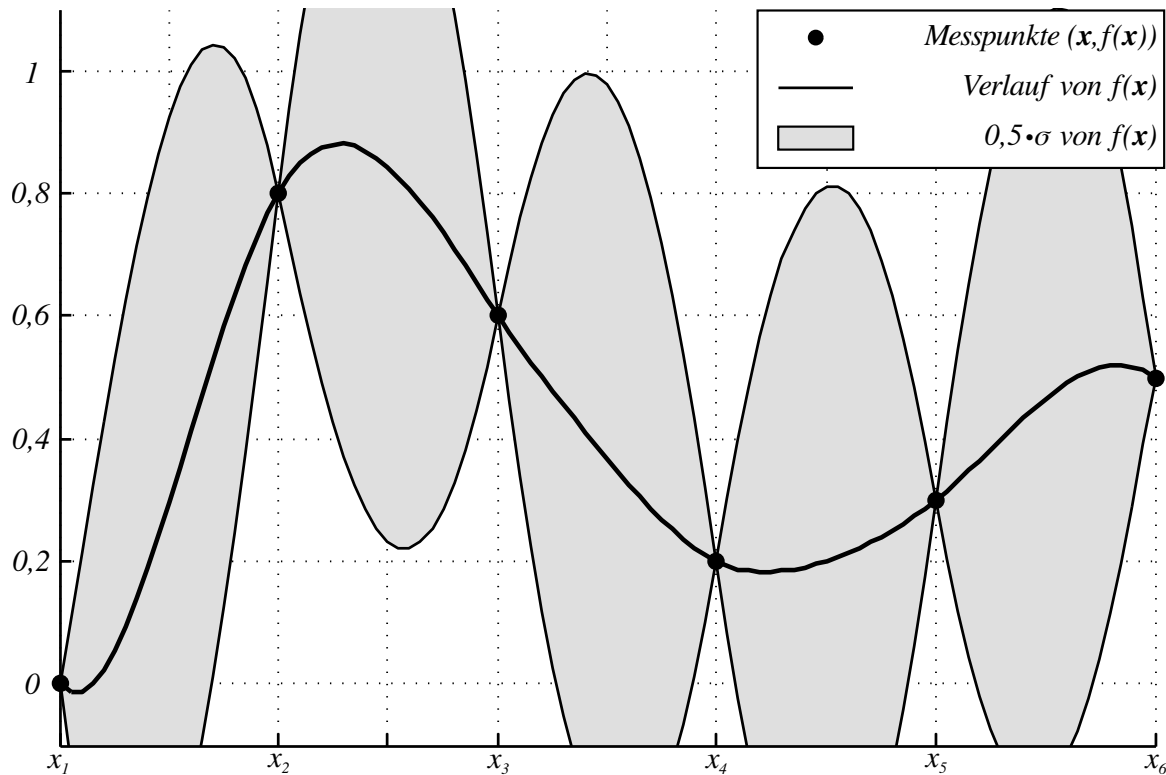


Abbildung 3.9: Beispiel: *posterior* Verteilung für den 2 – D Fall

bungsformel $y = f(\mathbf{x})$ geschrieben werden:

$$y = \sum_{i=1}^M \theta_i e^{-\frac{1}{2} \sum_{j=1}^N \frac{(x_{i,j} - x_j)^2}{l_j}} \quad (3.38)$$

wobei M die Anzahl der Trainingsdaten und N die Eingangsdimension darstellt. θ_i sind die Hyperparameter und $X_{i,j}$ die Trainingsdaten [KKG⁺13], [ETA14b].

3.3.2 Modellierungsansatz

Mit den in den vorherigen Kapiteln beschriebenen Vermessungs- und Modellierungsmethoden wird das statische Verhalten eines nichtlinearen Prozesses erfasst und nachgebildet. Für die Darstellung der Steller-Sensor-Strecken ist gemäß dem Kapitel 3.1 die Modellierung sowohl des stationären, als auch des dynamischen Verhaltens notwendig. Mit Hilfe der blockorientierten Modellierungsansätze nach Wiener und Hammerstein [DP04, S. 183], [Sch10b, S. 262 ff.] wird das dynamische Verhalten eines Prozesses dem stationären Verhalten aufgeprägt. Hierbei werden spezielle dynamische Erweiterungsblöcke entweder am Eingang oder am Ausgang des Prozesses aufgeschaltet. Die Abbildung

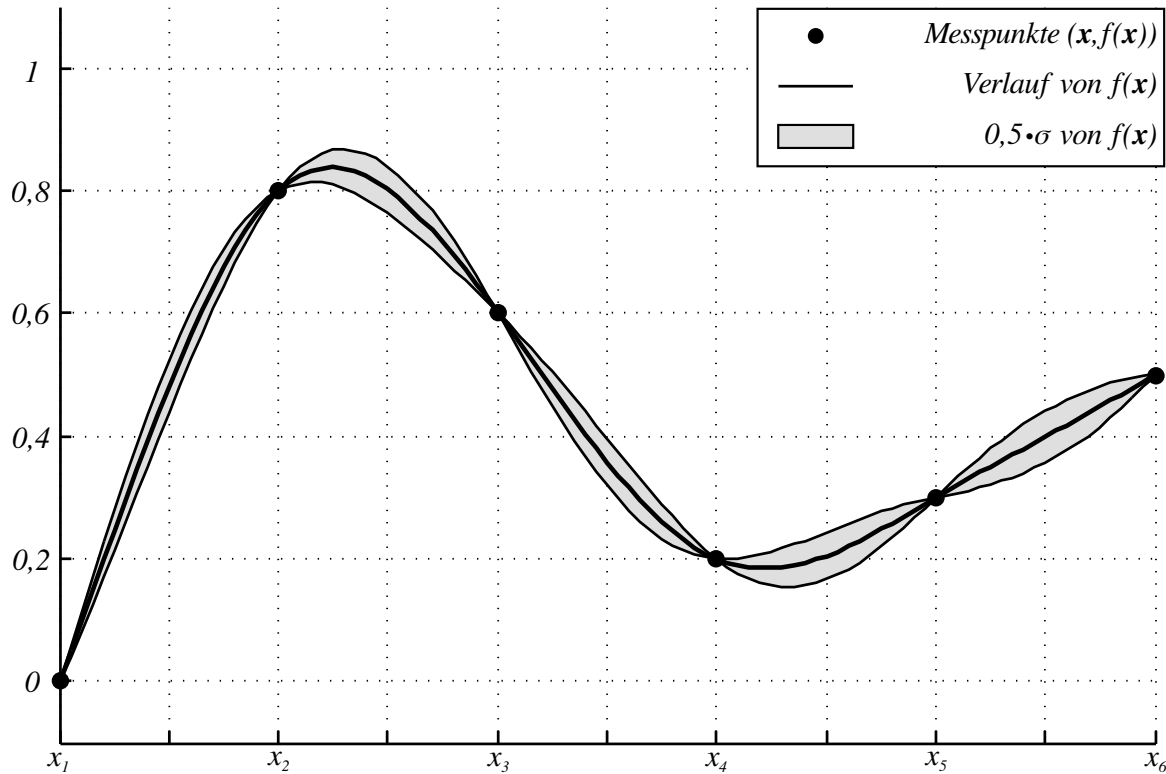


Abbildung 3.10: Beispiel: Darstellung der *posterior* Verteilung für den 2 – D Fall mit optimierten Hyperparametern

3.11 visualisiert die Blockschaltbilder der beiden Modellansätze.

Eine weitere Möglichkeit der Nachbildung eines dynamisches Prozessverhaltens ist die Verwendung eines Modellansatzes, der sowohl statische als auch dynamische Eigenschaften in einer geschlossener Form beschreibt. Hierfür sind verschiedene Formen der dynamischen neuronalen Netze besonders geeignet. Ein detaillierter Überblick über den Aufbau und die Einsatzmöglichkeiten von *Time-Delay-Neuronal-Networks (TDNN)*, *Recurrent-Neuronal-Networks (RNN)* und *Diagonal-Recurrent-Neuronal-Networks (DRNN)* ist in [Aye98, S. 9 ff.] und [Win02, S. 52 ff.] zu finden.

Im betrachteten Kontext der Modellierung des Verbrennungsmotorverhaltens ist von einer stetig steigenden Anzahl an Einflussgrößen des dynamischen Prozesses auszugehen. Ein moderner Dieselmotor verfügt durchaus über 14 bis 18 Freiheitsgrade [KKG⁺13], die zur Konfiguration des Luft- und Kraftstoffpfades zur Verfügung stehen¹.

Aufgrund des erhöhten Bedarfs an dynamischen Messdaten bei den dynamischen neuronalen Netzen, konzentriert sich die Auswahl des Modellierungsansatzes auf die Wiener- und Hammerstein-Modellstruktur. Weiterhin wird bei der Hammerstein-Modellstruktur im Vergleich zur Wiener-Modellstruktur mit geringem Messaufwand ein größerer Betriebsbereich des Motors abgedeckt. Für die Bestimmung der Koeffizienten der linearen Dynamik werden die Einflussgrößen hierbei gleichzeitig und nicht nacheinander verstellt. Dementsprechend wird im Rahmen dieser Arbeit die Hammerstein-Modellstruktur für die Modellierung der Steller-Sensor-Strecke des Verbrennungsmotors gewählt. Die Eignung einer dynamischen Erweiterung am Ausgang der Modellierungsstrecke im Kontext der

¹Der Aufbau der aktuellen Generation der Dieselmotoren von Volkswagen kann in [NKJ⁺12] eingesehen werden.

Modellbildung des Verbrennungsmotors wurde im Rahmen der *Strukturierten Prozessnachbildung* gezeigt [Sch10a, S. 24 ff.].

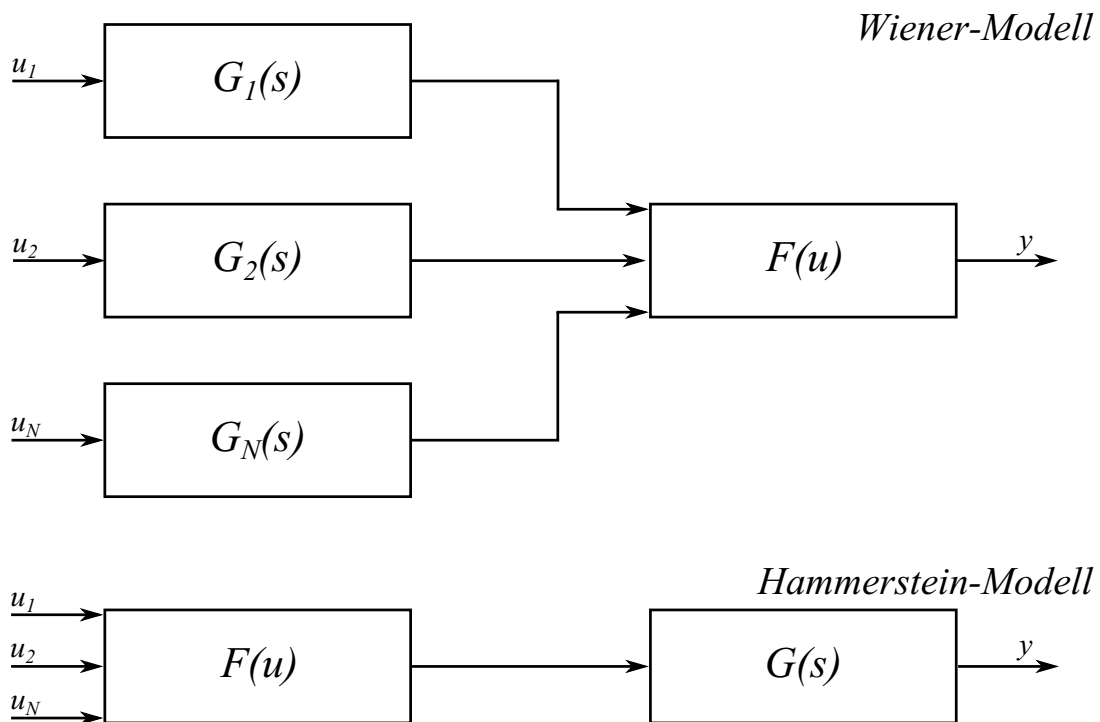


Abbildung 3.11: Prinzipskizze eines Wiener-Modells (*oben*) und eines Hammerstein-Modells (*unten*), wobei $G(s)$ eine lineare Übertragungsfunktion und $F(u)$ eine statische Nichtlinearität darstellen.

3.3.3 Reallisierung

Für die Realisierung des Hammerstein-Modellansatzes liegen entsprechend dem Kapitel 3.2.4 zwei unterschiedliche Vermessungen vor. Einerseits eine DoE-Vermessung, die die Informationen über das stationäre Motorverhalten beinhaltet. Andererseits eine dynamische Vermessung, die durch eine direkte Anregung der Steller des Systems die Aufnahme des dynamischen Verhaltens ermöglicht. Basierend auf diesen Vermessungen haben sich folgende Modellierungsmethoden für die Erstellung der Steller-Sensor-Streckenmodelle als geeignet erwiesen:

Statische Nichtlinearität Für die Implementierung der statischen Nichtlinearität werden die neuronalen Netze als MLP-Struktur verwendet. Die Auswahl der optimalen Neuronenanzahl ist nach dem im Kapitel 3.3.1 beschriebenen Prinzip durchgeführt. Zusätzlich zum SSE wurde die statistische Verteilung der relativen Fehler (Kapitel A.2) betrachtet. Die Berechnung der Standardabweichung σ (Kapitel A.6) und des Mittelwerts μ sind hierbei von Bedeutung. Bei der Annahme einer Gaußverteilung der Residuen¹ sind somit statistische Aussagen über die

¹Die Annahme ist durchaus mit den Annahmen der Gaußprozess-Modellierung konform [RW06, S. 8].

Verteilung der relativen Fehler möglich. Im 2σ -Intervall der Fehlerverteilung liegen dementsprechend $\approx 95,5\%$ der Fehler. Für eine Aussage über die Modellgüte ist die obere Grenze des Intervalls interessant. Hier kann ein Grenzwert definiert werden, der bei der Modellerstellung einzuhalten ist, zum Beispiel 3%. Falls die gewünschte Modellgenauigkeit nur mit einer Neuronenanzahl erreicht werden kann, die mit einem verstärktem Overfitting-Effekt verbunden ist, können die Interpolationseigenschaften mit Hilfe von Gaußprozess-Modellen verbessert werden. Bei dieser praktisch orientierten Methode wird vor dem Training des neuronalen Netzes ein Gaußprozess-Modell mit den Trainingsdaten erstellt. Das GP-Modell beschreibt hierbei, bezogen auf die gewählten Hyperparameter und die Trainingsdaten, den wahrscheinlichsten Verlauf des zu modellierenden Zusammenhangs im Interpolationsbereich. Mit diesem Modell können anschließend zusätzliche Punkte erzeugt werden, die das neuronale Netz im Interpolationsbereich stabilisieren und erheblich zur Vermeidung des Overfitting-Effekts beitragen. Die Aufnahme der erzeugten Punkte in den Trainingsdatensatz erfolgt nach dem gleichen Prinzip, wie die Aufnahme von Extrapolationsdaten. Hierbei ist mittels Bewertungsfaktoren die Gewichtung der Residuen dieser Punkte in der Kostenfunktion viel schwächer bewertet, als es bei den Trainingsdaten der Fall ist.

Dynamische Erweiterung Für die Erstellung der dynamischen Erweiterung sind lineare Übertragungsfunktionen in Form von $PT1$ - oder $PT2$ -Gliedern verwendet. Um die Modellierungsergebnisse der statischen Nichtlinearität nicht zu verfälschen, sind die Koeffizienten der Übertragungsfunktionen so zu definieren, dass die Eigenschaft $F(s=0) \stackrel{!}{=} 1$ immer erfüllt ist. Die Werte der Koeffizienten können anhand von den dynamischen Messschrieben und einem Optimierungsalgorithmus bestimmt werden. Bei der Minimierung der Abweichung zwischen dem Modellausgang und den Messergebnissen wird wieder das 2σ -Intervall der relativen Fehlerverteilung betrachtet.

Falls notwendig, sind die Koeffizienten als betriebspunktabhängige Funktionen $f(n_{Mot}, M_{Mot})$ zu implementieren.

Die Abbildung 3.12 gibt einen Überblick über die implementierte Modellstruktur. Die in die Simulink-Umgebung übertragenen Steuerungsfunktionen des untersuchten Dieselmotors sind im *ECU-Funktionen* Block zusammengestellt. Zusätzlich zu den Vorgaben der Drehzahl und Fahrpedalstellung liest dieser Block die von den erstellten Modellen berechneten Sensorsignale ein. Die Funktionen des *Momentenpfad*-Moduls berechnen aus den Eingangssignalen eine Drehmomentvorgabe $M_{VKM,Soll}$. Die Umsetzung der Drehmomentvorgabe erfolgt in den Modulen *Luftpfad* und *Kraftstoffpfad*. Abhängig von der eingelesenen Parametrierung sind die Sollwertvorgaben der einzelnen Steller das Ergebnis der Funktionsberechnungen. Die Verarbeitung der Sollwertvorgaben erfolgt in der *Steller-Sensor Strecke*. In diesem Block sind die, in diesem Kapitel beschriebenen, Modelle implementiert. Aus den Vorgaben für die einzelnen Steller des Verbrennungsmotors berechnen die Modelle entsprechende Sensorsignale, Abgasemissionswerte und das vom Motor gelieferte Drehmoment.

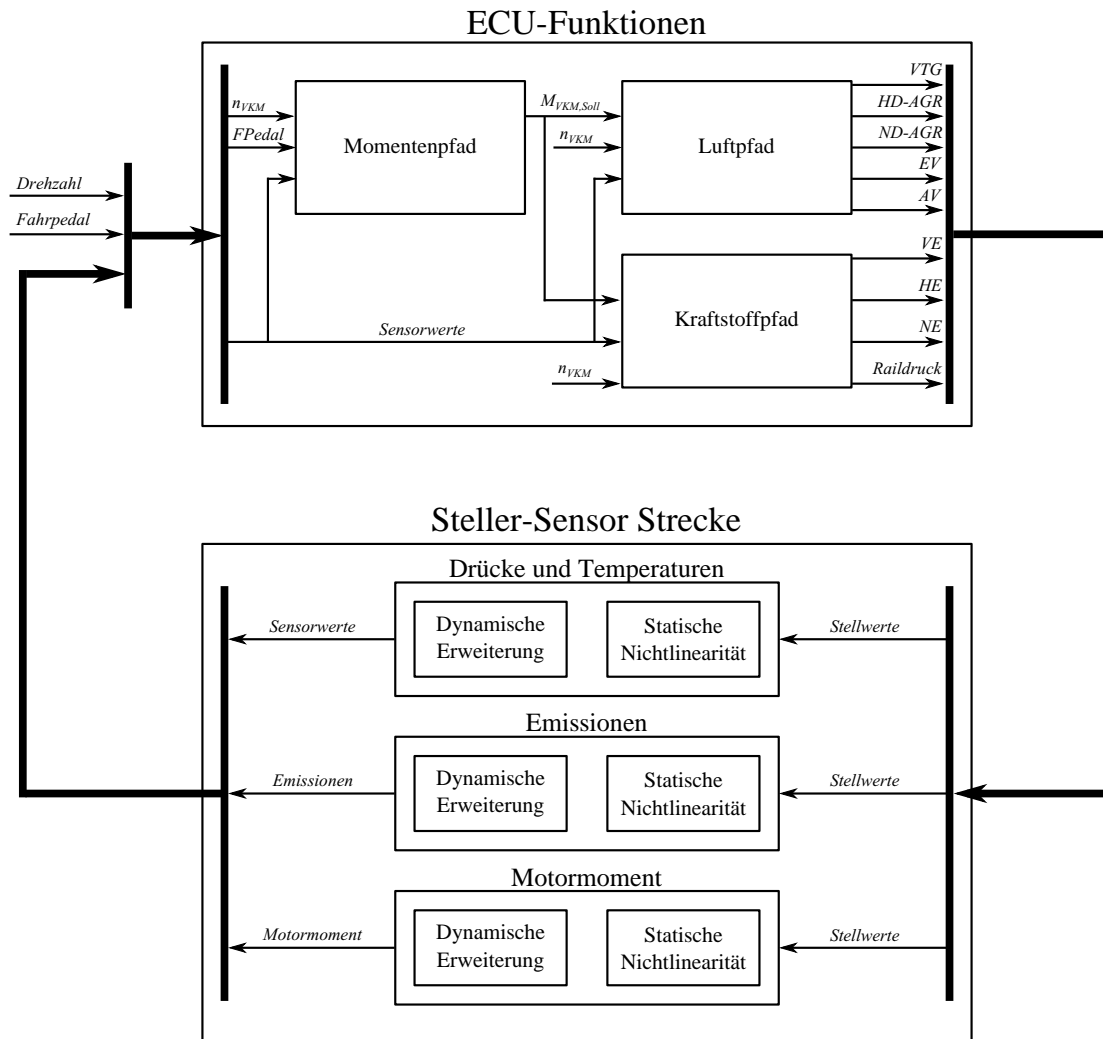


Abbildung 3.12: Überblick über die erstellte Modellstruktur des Verbrennungsmotors. **Oben:** Die Aus dem Steuergerät übertragenen Funktionsmodule. **Unten:** Die implementierten Modelle der *Steller-Sensor Strecke* des Verbrennungsmotors.

3.4 Modellierungsbeispiel

3.4.1 Versuchsträger

Beim Versuchsträger handelt es sich um einen Dieselmotor der aktuellen TDI-Generation von Volkswagen. Das untersuchte Aggregat gehört zum modularen Dieselbaukasten (MDB) des Volkswagen-Konzerns und ist für den Einsatz in Klein-, Kompakt- und Mittelklassefahrzeugen konzipiert. Eine ausführliche Vorstellung dieser Motorengeneration, inklusive der Darstellung der technischen Besonderheiten, ist in [NKJ⁺12] zu finden. An dieser Stelle erfolgt nur eine kurze Übersicht der technischen Daten des untersuchten und modellierten Triebwerks (siehe Tabelle 3.1).

Tabelle 3.1: Technische Daten des modellierten Dieselmotors

2,0 TDI MDB		
Zylinderanzahl /-anordnung	R4	
Hubraum	1968	cm^3
Ventile pro Zylinder	4	
Verdichtung	16,2	
Nennleistung	ca. 105	kW
Nenndrehmoment	ca. 320	Nm
Aufladung	VTG-Turbolader mit einem im Saugrohr integrierten Ladeluftkühler.	
Ventiltrieb	Variable Verstellung der Einlassnockenwelle.	
Abgasrückführung	Hoch- und Niederdruck Abgasrückführung. Beide Rückführstrecken gekühlt. Elektromotorische Abgasklappe.	
Software	Gassystemmodellierung mit der Abbildung aller notwendigen Luftpfadgrößen wie Temperaturen, Drücke und Massenströme. Gassystemregelung mit Hauptführungsgrößen: Zylinderführung und Abgasrückführrate.	
Abgasstufe	EU6	

3.4.2 Modellierungsergebnisse

Die nachfolgenden Abbildungen 3.13 bis 3.15 geben einen Überblick über die erzielten Modellierungsergebnisse. Stellvertretend für die Größen des Luftpfades ist in der Abbildung 3.13 die Auswertung des Ladedruckmodells dargestellt. An dieser Stelle sei darauf hingewiesen, dass es sich um ein empirisches Modell einer Steller-Sensor-Strecke handelt. Somit ist nur der Zusammenhang zwischen den vom Steuergerät verstellten Stellwerten und den gemessenen Sensorwerten modelltechnisch approximiert. Stellvertretend für die Größen des Abgastraktes ist in der Abbildung 3.14 die Auswertung des Modells der Temperatur vor Turbine zusammengefasst. Abschließend folgt die Darstellung der Modellierungsergebnisse der Abgasemissionen. Das Modell der Stickstoffoxide ist für diese Größengruppe in der Abbildung 3.15 visualisiert.

Auf der *linken* Seite der Abbildungen ist jeweils ein beispielhafter dynamischer Übergang eingezeichnet. Es handelt sich hierbei um einen Betriebspunktwechsel aus der dynamischen Vermessung

(s. Kapitel 3.2.4). Auf der *rechten* Seite der Abbildungen ist die Bewertung der stationären Modellierung dargestellt. An dieser Stelle erfolgt eine Gegenüberstellung der gemessenen und der simulierten Daten. Das Bestimmtheitsmaß R^2 wird hierbei zur Beurteilung der Modellqualität hinzugezogen. Es erfolgt jeweils eine Angabe für die Trainings- und die Verifikationsdaten.

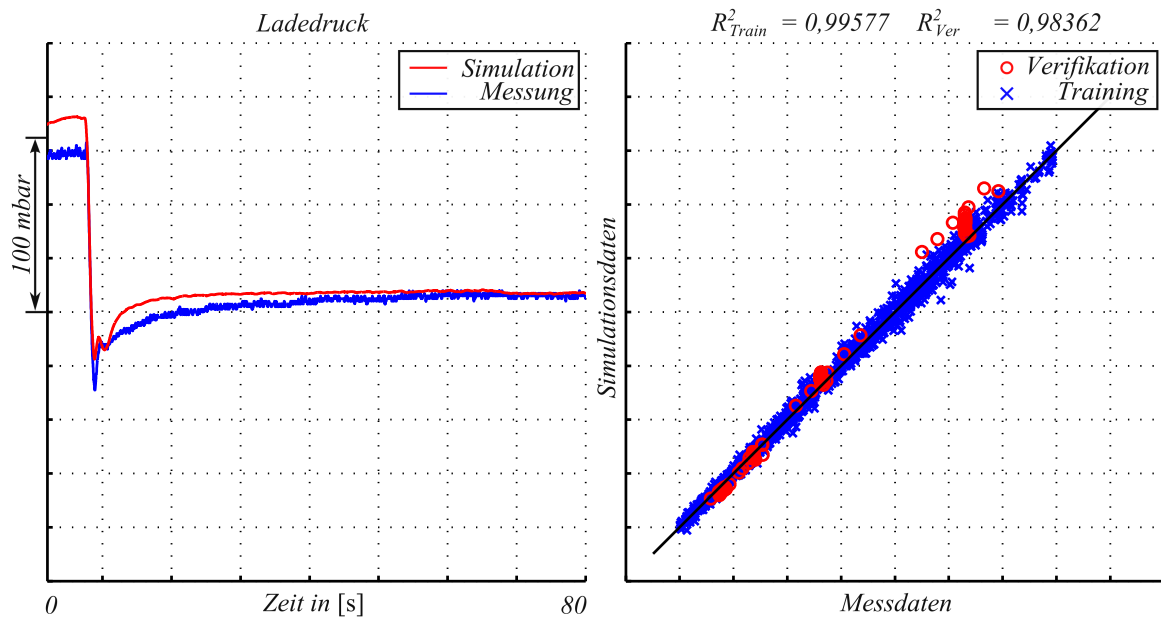


Abbildung 3.13: Modellierungsergebnisse des Ladedruckmodells. **Links:** Ein beispielhafter dynamischer Übergang, zwischen zwei Betriebspunkten. **Rechts:** Bewertung der stationären Modellierung anhand einer direkten Gegenüberstellung der gemessenen und der simulierten Betriebspunkte.

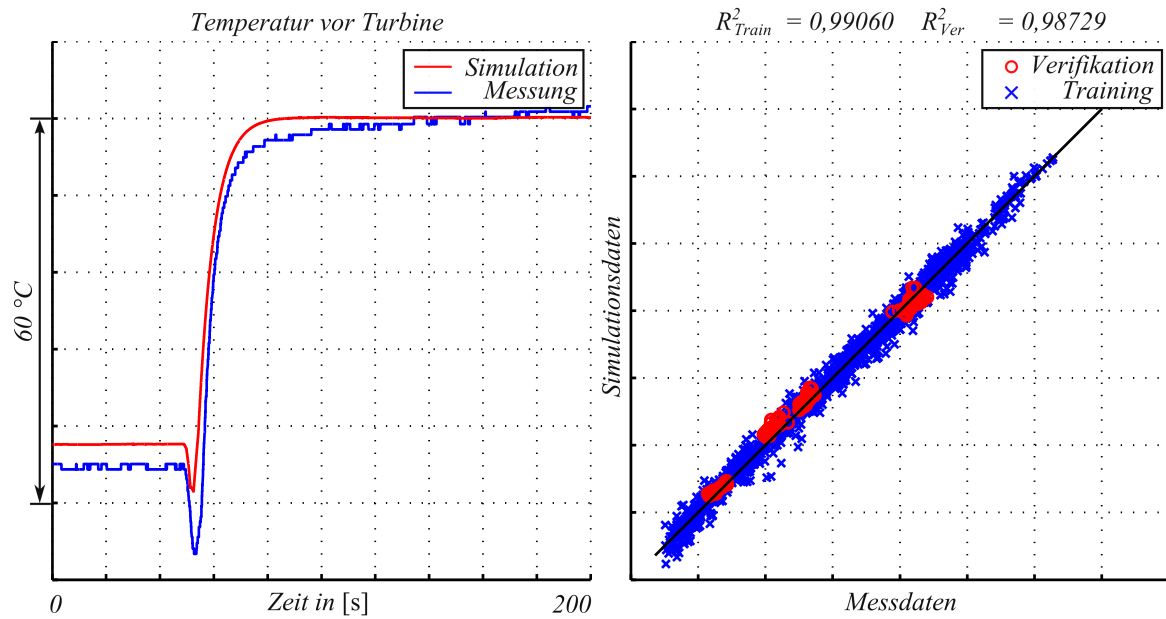


Abbildung 3.14: Modellierungsergebnisse der Temperatur vor Turbine. **Links:** Ein beispielhafter dynamischer Übergang, zwischen zwei Betriebspunkten. **Rechts:** Bewertung der stationären Modellierung anhand einer direkten Gegenüberstellung der gemessenen und der simulierten Betriebspunkte.

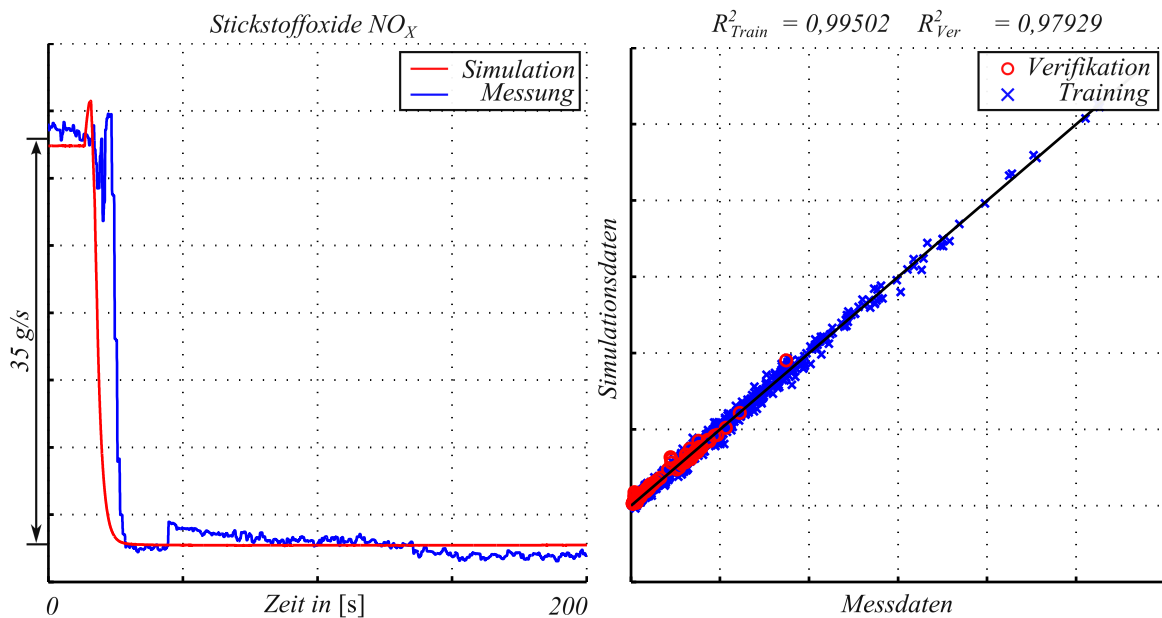


Abbildung 3.15: Modellierungsergebnisse der Stickstoffoxide NO_x . **Links:** Ein beispielhafter dynamischer Übergang, zwischen zwei Betriebspunkten. **Rechts:** Bewertung der stationären Modellierung anhand einer direkten Gegenüberstellung der gemessenen und der simulierten Betriebspunkte.

4 Optimierungswerkzeuge

4.1 Motivation

4.1.1 Basisoptimierung der Steuerungs- und Regelungsfunktionen

Nach dem Fertigstellen des Prozessmodells wird in einem nächsten Schritt mit der Optimierung des Prozesses anhand von gewählten Optimierungskriterien begonnen. Den Ausgangspunkt der Motivationsaspekte für das folgende Kapitel bilden die häufig eingesetzten Modelle der Strecke *Einflussgröße-Zielgröße* als eine geschlossene mathematische Beschreibung (siehe zum Beispiel [Ise10, S. 131 ff.]). Beispielhaft sei an dieser Stelle ein Modell der Stickoxidemissionen in Abhängigkeit von Einflussgrößen wie zum Beispiel der Zylinderfüllung oder der Abgasrückführrate (AGR) genannt.

Die Kriterien zur Optimierung des Verhaltens eines Verbrennungsmotors bzw. eines Antriebsstrangs sind direkt von den gesetzlichen Emissionsvorschriften und den kundenrelevanten Verbrauchswerten abgeleitet. Ein Lösungsansatz ist durch die Aufstellung einer Kostenfunktion, die die Kriterien durch gewichtete Addition zusammenfasst, gegeben. Ein anderer Lösungsansatz betrachtet eine multikriterielle Optimierung. Das Ziel hierbei ist die Berechnung einer Paretofront der definierten Optimierungskriterien.

Eine Möglichkeit, die Einstellung der Applikationsparameter bezüglich der gewählten Kriterien zu optimieren, ist die Betrachtung der relevanten Betriebspunkte eines Fahrzyklus. Die Auswahl und die Gewichtung der charakteristischen Betriebspunkte sind für die Zyklushochrechnung der Abgasemissionen und des Verbrauchs entscheidend (siehe zum Beispiel [Ise10, S. 155]). Durch die Verstellung der Applikationsparameter und die Auswertung der Optimierungskriterien in den Betriebspunkten ist ein Optimierungsalgorithmus in der Lage Parametereinstellungen zu berechnen, die den Optimierungsanforderungen genügen.

Eine weitere Möglichkeit, das Verhalten des Antriebsstranges zu optimieren, ist die sogenannte Rückwärtssimulation. Die für eine Antriebseinheit relevanten gewichteten Betriebspunkte werden durch die Drehzahl-/Drehmomentfolgen eines Fahrzyklus ersetzt. Dadurch kann die Aussagekraft der Optimierungsergebnisse verbessert werden.

Das im Kapitel 2.2.1 vorgestellte Simulationsmodell implementiert eine Vorwärtssimulation der Fahrzeuglängsdynamik eines Fahrzeugs. Die Drehzahl-/Drehmomentfolgen werden hierbei durch ein Soll-Geschwindigkeitsprofil ersetzt. Für die Einhaltung des Profils wird ein Fahrermodell eingesetzt. Wie im Kapitel 2.2.1 beschrieben, liefert das Simulationsmodell eine bessere Genauigkeit, die aber nur mit größerem Parametrierungsaufwand zu erreichen ist.

Als Ergebnis stehen die optimierten Kennfelder zur Verfügung. Die Betrachtung einzelner Betriebspunkte bei der Optimierung kann zu größeren Differenzen zwischen den benachbarten Kennfeldstützstellen führen [Kla08, S. 4, 36, 89]. Im dynamischen Betrieb des Verbrennungsmotors werden diese nacheinander durchlaufen und verursachen Sprünge in den Stellersollwerten. Dieses ist aus Optimalitäts- und Komfortgründen nicht erwünscht und kann durch die Anwendung von Glättungsalgorithmen behoben werden. Bei der Anwendung dieser Algorithmen ist weiterhin zu beachten, dass die geglätteten Kennfelder keine großen Abweichungen zu den Optimierungsergebnissen aufweisen, da sonst die Ergebnisse der Optimierung verfälscht werden.

Die Anpassung der Kennfelder während der Optimierung führt weiterhin zur Änderung des Wirkungsgrades des Verbrennungsmotors. Aus der Sicht des Gesamtsystems verschieben sich die Stützstellen der für die Zyklusbetrachtungen angenommenen Betriebspunkte. Dieses wird durch die Änderung des Drehmomentes an der Kurbelwelle bei gleicher Drehzahl und Fahrpedalstellung bzw. Einspritzmenge hervorgerufen. Folglich werden für die Erzeugung, des im Zyklus angenommenen Momentes, auch andere Einspritzmengen benötigt. Eine Anpassung bzw. Rückrechnung der Optimierungsergebnisse auf die geänderten Einspritzmengen ist somit notwendig.

4.1.2 Änderungen im Simulationsmodell

Im Folgenden werden Werkzeuge betrachtet, die bei der Optimierung, des im Rahmen dieser Arbeit erstellten Prozessmodells, zum Einsatz kommen. Mit Hilfe des im Kapitel 2.4 vorgestellten Werkzeuges werden Steuererätefunktionen in der Simulationsumgebung des Gesamtfahrzeugs aus dem Kapitel 2.2.1 integriert. Der Transfer der Module umfasst Funktionen zur Koordination und Steuerung eines hybriden Antriebsstranges und zur Steuerung und Regelung eines Dieselmotors. Die *Steller-Sensor*-Streckenmodelle sind mit den im Kapitel 3.3.3 vorgestellten Methoden erstellt.

Die Implementierung der Steuererätesoftware und der Modelle führen zu Änderungen im Simulationsmodell. Bezogen auf die Basisoptimierung, mit der Implementierung der Streckenmodelle *Einflussgröße-Zielgröße*, ergeben sich hierdurch einige Vorteile. Die im Kapitel 2.5.2 beschriebenen Vorteile der Unempfindlichkeit gegenüber Daten- und Funktionsstandsänderungen und die im Kapitel 2.1 behandelte Vereinfachung der Ergebnisübertragung ins Steuergerät sind nur einige aussagekräftige Motivationsaspekte der Auftrennung der Modellierungsstrecke in einen Softwareteil und eine Steller-Sensor Strecke. Durch die Verwendung der Vorwärtssimulation kann weiterhin auf die Anpassung der Optimierungsergebnisse bezüglich der geänderten Einspritzmengen verzichtet werden. Bei einer Änderung des Wirkungsgrades greift das Fahrermodell und passt die Fahrpedalstellung an. Der somit angepasste Betriebspunkt geht automatisch in die Optimierungsrechnungen ein. Diesen positiven Eigenschaften des geänderten Simulationsmodells stehen durchaus einige, mit den implementierten Änderungen verbundene, Nachteile gegenüber. Die vorteilhafte Möglichkeit der Optimierung der Hybrid-Betriebsstrategie gemeinsam mit dem Verbrennungsmotorverhalten bringt entsprechend einen Anstieg der Einflussgrößen mit sich. Durch die steigende Simulationszeit und eine Vergrößerung des Parameterraums verändert sich auch der Einsatzrahmen der Optimierungsalgorithmen erheblich. Algorithmen, die bei der Minimumsuche eine höhere Anzahl an Iterationen benötigen oder nur eine lokale Minimierung einer Kostenfunktion durchführen, erfahren hierbei einen Effizienzverlust.

Auch bei den betrachteten Zielgrößen treten einige Änderungen ein. Die Genauigkeit der Modelle ist im Vergleich zur Kennfelddarstellung deutlich erhöht. Zu den betrachteten Zielgrößen des Verbrauchs und der Abweichung des Batterieladezustandes rücken die Abgasemissionen und der Anteil der elektrischen Fahrphasen stärker in den Fokus der Betrachtungen.

Um den neuen Herausforderungen gerecht zu werden, wird der Einsatz nachfolgend vorgestellter Werkzeuge vorgeschlagen. Die vorgestellte Methode der Gütevektroptimierung (s. Kapitel 4.2) soll hierbei eine systematische Betrachtung und Optimierung der Zielgrößen sicherstellen. Gleichzeitig soll die im Rahmen der vorliegenden Arbeit neu entwickelte Methode *DelaunaySearch* (s. Kapitel 4.3) den gestiegenen Simulationszeiten entgegenwirken.

4.2 Gütevektroptimierung

Das Verfahren der Gütevektroptimierung (*Systematische Auslegung von Reglern durch Optimierung eines vektoriellen Gütekriteriums*) [KS79] [Ack93, S. 356 ff.] beschreibt eine iterative Methode zur systematischen Auslegung von Reglerparametern. Bei einer gewählten Reglerstruktur mit freien Reglerparametern wird das Optimierungsergebnis systematisch, von Entwurfsschritt zu Entwurfsschritt, soweit verbessert, bis der bestmögliche Kompromiss der einzelnen Gütekriterien gefunden ist. Der praktische Einsatz des Verfahrens ist zum Beispiel in [KS83] beschrieben. In dieser Veröffentlichung wird die Auslegung des Flugreglers eines Flugzeuges mit dem Verfahren der Gütevektroptimierung durchgeführt.

Das ursprünglich aus dem Bereich der Regelungstechnik stammende Verfahren kann als eine universale Methode zur Lösung von Problemstellungen mit vielen Entwurfsspezifikationen¹ eingesetzt werden. Durch den Einsatz des Verfahrens werden unter anderem gleichzeitig viele unterschiedliche und zum Teil gegenläufige Eigenschaften systematisch betrachtet und optimiert. Die Zielwerte für die einzelnen Eigenschaften sind im Allgemeinen nicht im Voraus bekannt. Die iterative Vorgehensweise bei der Optimierung ermöglicht an dieser Stelle eine schrittweise Annäherung an das erreichbare Optimum. Zusätzlich kann durch die Berechnung der einzelnen Gütekriterien eine Vergleichbarkeit unterschiedlicher Entwürfe sichergestellt werden.

Nachfolgend erfolgt eine Zusammenfassung, der in [KS79] veröffentlichten, Informationen zur Beschreibung des Verfahrens.

Die freien Parameter, des zu optimierenden Systems, werden zunächst zu einem Parametervektor $K = [k_1, k_2, \dots, k_n]$ zusammengefasst. Jede Entwurfsspezifikation¹ ist als Gütekriterium $g_i = f(K)$ in Form eines Minimierungsproblems zu definieren. Die einzelnen Gütekriterien werden in zwei Kategorien eingeteilt.

Typ 1 Gütekriterien, die minimiert werden sollen

Typ 2 Gütekriterien, die nach der Optimierung einen Wert unter einer Obergrenze aufweisen müssen

¹In der vorliegenden Arbeit werden Anforderungen wie Abgasemissionsgrenzen, Verbrauchsangaben usw. als Entwurfsspezifikationen aufgefasst

Eine Zusammenfassung der Gütekriterien g_i in einem Gütevektor G ist nicht nur für weitere Optimierungsrechnungen notwendig, sie bietet auch die Möglichkeit, die Optimierungsergebnisse systematisch zu vergleichen.

$$G(K) = \begin{bmatrix} g_1(K) \\ g_2(K) \\ \vdots \\ g_N(K) \end{bmatrix} \quad (4.1)$$

Mit einer plausiblen Einstellung des Parametervektors K kann ein Vorgabevektor $C = [c_1, c_2, \dots, c_n]$ mit folgender Eigenschaft bestimmt werden:

$$G(K) < C \quad (4.2)$$

bzw.

$$g_i(K) < c_i \quad \text{wobei} \quad i = 1, 2, \dots, n$$

Das Entwurfsverfahren beinhaltet eine iterative Optimierung. Das Ziel jedes Iterationsschrittes ν ist die Bestimmung eines Parametervektors K^ν , der den Gütevektor G bezogen auf den Vorgabevektor C^ν minimiert. Hierfür wird die Beziehung zwischen G und C wie folgt definiert:

$$G(K^\nu) \leq \alpha \cdot C^\nu \quad (4.3)$$

bzw.

$$\alpha \geq \frac{G(K^\nu)}{C^\nu} \quad (4.4)$$

wobei $\alpha < 1$ ist und ν die Nummer der aktuellen Iteration repräsentiert. Somit kann für das kleinstmögliche $\alpha(K)$, welches die Ungleichung (4.3) noch erfüllt, geschrieben werden:

$$\alpha(K) = \max_i \frac{g_i(K)}{c_i} \quad (4.5)$$

Für jeden Iterationsschritt ν ergibt sich hiermit folgende Optimierungsaufgabe:

$$\min_K \{\alpha(K)\} \quad (4.6)$$

$$\min_K \left\{ \max_i \frac{g_i(K)}{c_i} \right\} \quad (4.7)$$

Die einzelnen Elemente c_1, c_2, \dots, c_n gehören zum Vorgabevektor C^ν .

Im ersten Iterationsschritt ist der Vorgabevektor C^0 durch eine plausible Einstellung des Parametervektors K^0 bekannt. Für weitere Iterationen $\nu = 1, 2, \dots$ sind die Vorgabevektoren nach dem Prinzip $C^\nu < \dots < C^1 < C^0$ zu wählen. Für das Ergebnis nach ν Iterationen ergibt sich:

$$G(K^\nu) < C^\nu < C^{\nu-1} < \dots < C^0 \quad (4.8)$$

Die monoton fallende Folge der Vorgabevektoren erzielt eine schrittweise Annäherung an das Optimum des Systemverhaltens, bezüglich der definierten Gütekriterien. Die iterative Vorgehensweise

wird soweit fortgesetzt, bis keine signifikante Verbesserung des Systemverhaltens erzielt wird.

In der Praxis ist bei der Wahl der Vektoren C^v folgende Strategie einzuhalten:

1. Die Gütekriterien g_i vom *Typ 2* müssen unter die durch c_i definierten Obergrenzen fallen.
2. Die Entwurfskoeffizienten c_i sind so zu wählen, dass die Gütekriterien g_i vom *Typ 1* in ihren Werten reduziert werden, ohne dass die Gütekriterien g_i vom *Typ 2* ihre Obergrenzen verletzen.

Weiterhin wird bei der praktischen Implementierung des Verfahrens empfohlen, die Optimierungsaufgabe aus der Gleichung (4.7) zu modifizieren. Hintergrund ist die angestrebte Differenzierbarkeit der Funktion $\alpha(K)$ über den gesamten Parameterraum K . Diese Eigenschaft wird oft von Optimierungsalgorithmen gefordert und kann gleichzeitig für numerische Stabilität sorgen.

$$\alpha(K) = \max_i \frac{g_i(K)}{c_i}$$

wird ersetzt durch

$$\begin{aligned} \tilde{\alpha}(K) &= \frac{1}{\rho} \ln \left(\sum_{i=1}^n \exp \left[\rho \frac{g_i(K)}{c_i} \right] \right) \quad \rho > 0 \\ \tilde{\alpha}(K) &= \frac{1}{\rho} \ln \left(\sum_{i=1}^n \exp \left[\rho \frac{g_i(K)}{c_i} \right] \right) + \alpha(K) - \alpha(K) \\ \tilde{\alpha}(K) &= \alpha(K) + \frac{1}{\rho} \ln \left(\sum_{i=1}^n \exp \left[\rho \frac{g_i(K)}{c_i} \right] \right) + \frac{1}{\rho} \ln (\exp [-\rho \alpha(K)]) \\ \tilde{\alpha}(K) &= \alpha(K) + \frac{1}{\rho} \left\{ \ln \left(\sum_{i=1}^n \exp \left[\rho \frac{g_i(K)}{c_i} \right] \right) + \ln (\exp [-\rho \alpha(K)]) \right\} \\ \tilde{\alpha}(K) &= \alpha(K) + \frac{1}{\rho} \left\{ \ln \left(\sum_{i=1}^n \exp \left[\rho \frac{g_i(K)}{c_i} \right] \right) \cdot \exp [-\rho \alpha(K)] \right\} \\ \tilde{\alpha}(K) &= \alpha(K) + \frac{1}{\rho} \ln \left(\sum_{i=1}^n \exp \left[\rho \left(\frac{g_i(K)}{c_i} - \alpha(K) \right) \right] \right) \end{aligned} \quad (4.9)$$

Die Kostenfunktion $\alpha(K)$ aus der Gleichung (4.5) ist jeweils nur von einem Gütekriterium abhängig. Durch die Anwendung des *max*-Operators wird bei der Wahl dieser Funktion jeweils nur das Gütekriterium $g_i(K)$ optimiert, welches den kleinsten Abstand zum zugehörigen c_i aufweist. Im Gegensatz hierzu ist die Kostenfunktion $\tilde{\alpha}(K)$ nicht nur im Parameterraum K stetig differenzierbar, sondern auch von allen Komponenten des Gütevektors G abhängig. Als Optimierungsergebnis steht daher eine Paretofront der $\frac{g_i}{c_i}$ zur Verfügung.

Mit dieser Entwurfsstrategie kann der bestmögliche Kompromiss zwischen den einzelnen Gütekriterien gefunden werden [Ack93, S. 356 ff.]. Die Abbildung 4.1 visualisiert die Zusammenhänge zwischen G , C und α bzw. $\tilde{\alpha}$. Als Beispiel dient an dieser Stelle die simulationsgestützte Variation der Abgasrückführrate an einem festgehaltenen Betriebspunkt des modellierten Dieselmotors. Die Gütevektoreinträge sind durch Stickstoffoxidemissionen, Rußemissionen und den Kraftstoffverbrauch definiert. Die eingezeichnete Kostenfunktion $\tilde{\alpha}(K)$ bildet jeweils das Maximum der einzelnen Gütekriterien $g_i(K)$, bezogen auf die in einer Iteration v gewählten Vorgabekoeffizienten c_i .

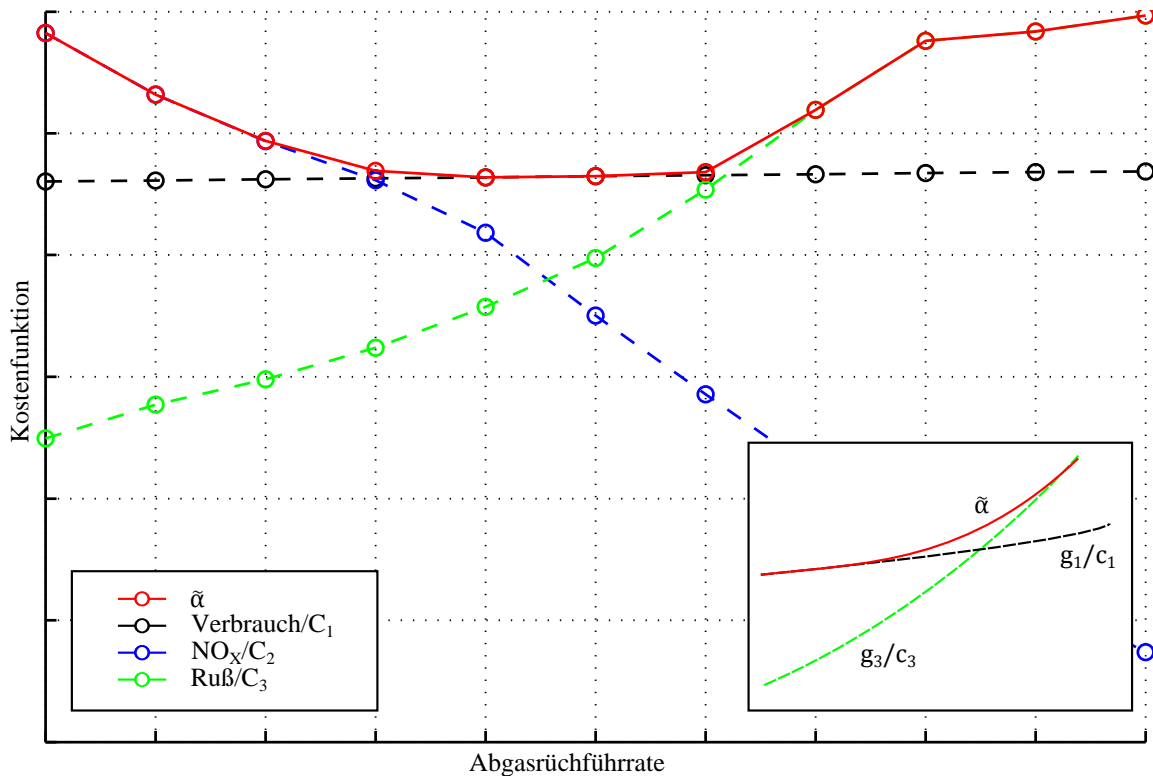


Abbildung 4.1: Darstellung der Kostenfunktion innerhalb der Gütevektoroptimierung

Zusätzlich ist der Unterschied zwischen $\alpha(K)$ und $\tilde{\alpha}(K)$ verdeutlicht (siehe Bildausschnitt unten rechts).

4.3 Entwicklung des Vermessungsverfahrens

DelaunaySearch

4.3.1 Motivation und Ziele

Die im Kapitel 4.1.2 beschriebenen Anpassungen des Simulationswerkzeuges führen zu steigenden Simulationszeiten und einer damit verbundenen Anpassung des Einsatzrahmens der Optimierungsalgorithmen. Ein Minimumsuchverfahren, welches eine Suche mit vielen Iterationsaufrufen durchführt, verliert seine Eignung mit steigender Simulationsdauer. Um der Problematik des steigenden Bedarfes an Optimierungszeit entgegenzuwirken, wird nach einem Verfahren gesucht, welches eine Minimumsuche mit möglichst wenig Iterationen ermöglicht. Die Grundidee ist hierbei einen maximalen Gewinn an Informationen aus der Lage der durchgeführten Iterationen zu realisieren.

Eine Zufallsinitialisierung, wie sie am Anfang vieler Optimierungsalgorithmen definiert ist, führt zur einer Abhängigkeit des Optimierungsstartwertes von einer Zufallsvariablen. Diese Abhängigkeit führt entweder zur Eingrenzung der betrachteten Möglichkeiten oder zu einer mehrmaligen Initialisierung des Verfahrens. Dadurch wird die Anzahl an notwendigen Iterationen bei der Lösung ei-

ner Optimierungsaufgabe erhöht. Vor allem bei der Partikelschwarmoptimierung (PSO) [Oli09, S. 41-58] und den evolutionären Algorithmen (EVO) [Oli09, S. 13-40] wird nicht nur mit einer Zufallsinitialisierung gearbeitet, die stochastische Komponente ist ein grundsätzlicher Bestandteil dieser Verfahren. Bei den PSO-Verfahren ist diese Komponente ein Teil der Anpassung der Partikelposition bzw. der Partikelgeschwindigkeit [Oli09, S. 48]. Bei den EVO-Algorithmen wird die Rekombination bzw. die Mutation durch die stochastische Komponente gesteuert [Oli09, S. 17]. Die Anwendung der stochastischen Methoden der beiden Verfahren dient insbesondere einer Globalisierung der Suche im Parameterraum.

Im Gegensatz hierzu setzt die neuentwickelte Methode *DelaunaySearch* bei der Globalisierung der Suche im Parameterraum auf eine gezielte und systematische Analyse der durchgeführten Simulationsergebnisse. Mit Hilfe der bei der Analyse gewonnenen Informationen können Bereiche mit den bestmöglichen Voraussetzungen für ein Minimum identifiziert werden. Somit kann das iterative Verfahren bei der Bestimmung des nächsten Iterationsschrittes auf die Verwendung einer Zufallsvariable verzichten. Der zu simulierende Punkt liegt hierbei in der Mitte des identifizierten Bereiches. Durch die systematische Analyse der zur Verfügung stehender Informationen wird eine Einsparung der Anzahl der Iterationen angestrebt. Als Ergebnis der Berechnungen ist ein gut gewählter Startwert für eine lokale Optimierung zu erwarten.

Die Wiederverwendung der Simulationsergebnisse, auch bei der Anpassung der Kostenfunktion durch das Verfahren der Gütevektroptimierung, trägt zur Einsparung der Iterationen bei.

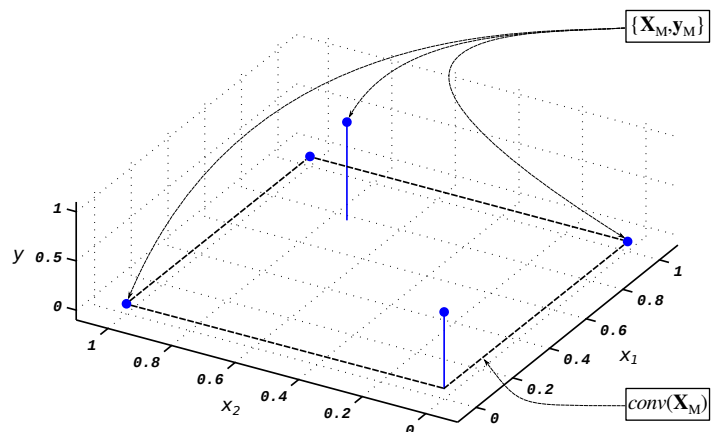
4.3.2 Funktionsweise

Nachfolgend ist die Funktionsweise der entwickelten Methode *DelaunaySearch* schrittweise, in einer tabellarischen Form erläutert. Für die Darlegung des Funktionsprinzips wird von einem Datensatz $D = \{\mathbf{X}_M, \mathbf{y}_M\}$ mit den Einflussgrößen $\mathbf{X}_M = [x_1 \dots x_M]$ und den dazugehörigen Werten einer Kostenfunktion $\mathbf{y}_M = [y_1 \dots y_M]$ ausgegangen.

Beschreibung

Bildliche Darstellung

Schritt 1 Als Ausgangspunkt der Optimierungsrechnungen dient eine Grundvermessung des Systems. Das Verfahren zieht nur Punkte in Betracht, die innerhalb der konvexen Hülle des Verstellbereiches der Einflussparameter $conv(\mathbf{X}_M)$ liegen.



Fortsetzung auf der folgenden Seite

Beschreibung

Bildliche Darstellung

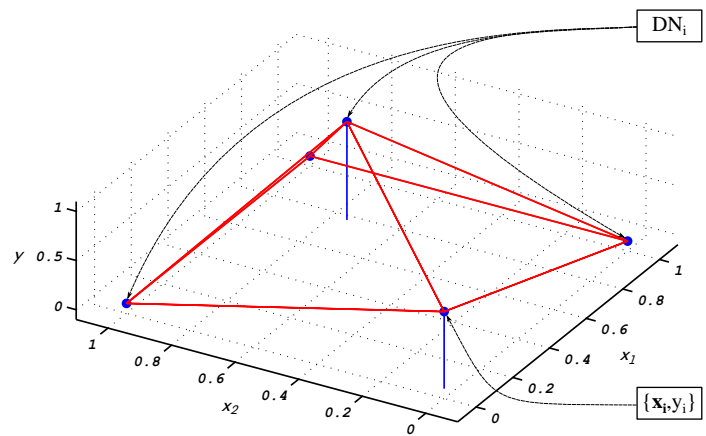
Schritt 2 Der aktuelle Datensatz \mathbf{X}_M wird einer Delaunay-Triangulation [Kle05, S. 231 ff.] unterzogen

$$XDT = DT(\mathbf{X}_M) \quad (4.10)$$

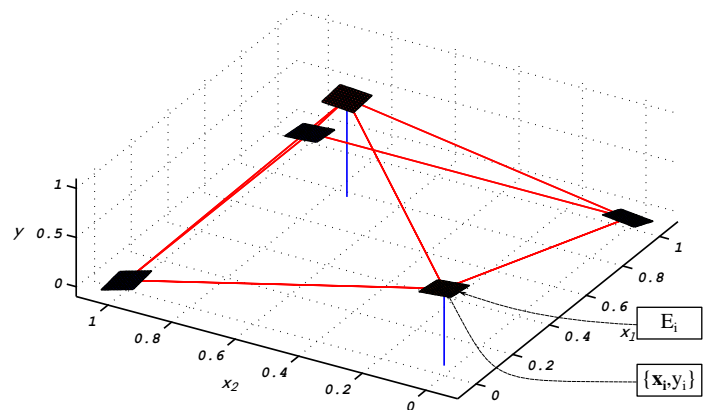
wobei M die Anzahl der untersuchten Parametereinstellungen darstellt. Hierdurch sind die direkten Nachbarn

$$DN_i = \{\mathbf{X}_{DN_i} | y_{DN_i}\} \quad (4.11)$$

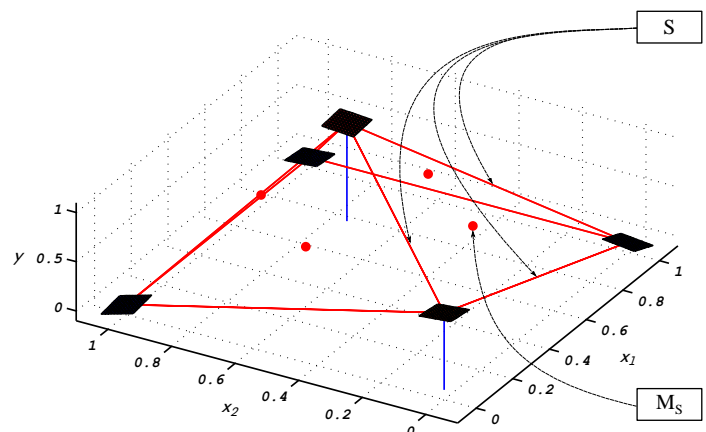
jeder untersuchten Parametereinstellung $\mathbf{x}_i, i = 1 \dots M$ bekannt.



Schritt 3 Mit Hilfe der direkten Nachbarn DN_i wird zu jeder untersuchten Parametereinstellung \mathbf{x}_i und dem dazugehörigen Wert der Kostenfunktion y_i eine Gradientenfläche E_i berechnet. Die Grundlage der Berechnung bildet eine Polynomapproximation der untersuchten Parametereinstellung $\{\mathbf{x}_i, y_i\}$ und deren Nachbarpunkte DN_i .



Schritt 4 Für jedes aus der Delaunay-Triangulation entstandene Simplex S wird die Stelle des Mittelpunktes M_S bestimmt.



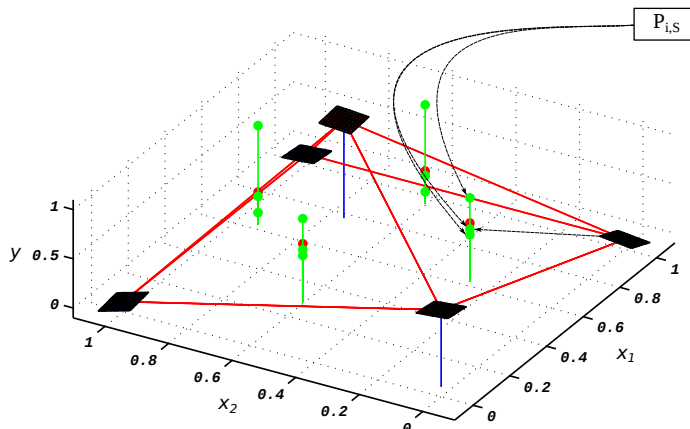
Fortsetzung auf der folgenden Seite

Beschreibung

Bildliche Darstellung

Schritt 5 Es folgt die Berechnung der Werte der Gradientenflächen E_i der Eckpunkte von S an der Stelle des Mittelpunktes M_S

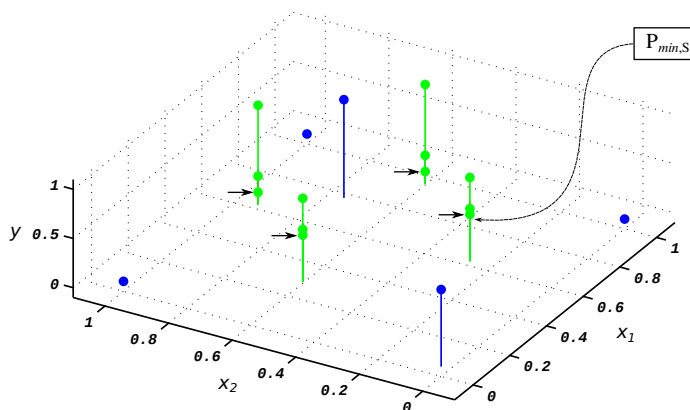
$$P_{i,S} = E_i(M_S) \quad (4.12)$$



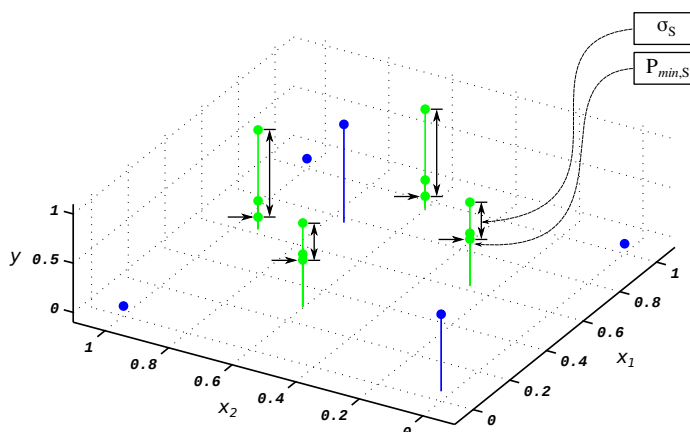
Schritt 6a Aus der Lage der im Schritt 5 berechneten Punkte $P_{i,S}$, kann für jedes Simplex S der mögliche Minimalwert der Kostenfunktion an der Stelle des Mittelpunktes M_S geschätzt werden.

$$P_{min,S} = \min(P_{i,S}) \quad (4.13)$$

Dieser ist durch den kleinsten Wert der für einen Simplex S berechneten Punkte $P_{i,S}$ gegeben.



Schritt 6b Die Lage der berechneten Punkte $P_{i,S}$ liefert auch Informationen über die *Sicherheit* σ_S in jedem Simplex S . Ist die Streuung der Punkte $P_{i,S}$ klein, so ist die Wahrscheinlichkeit, dass der wahre Wert der zu minimierenden Kostenfunktion in der Nähe der Punkte $P_{i,S}$ liegt, größer.

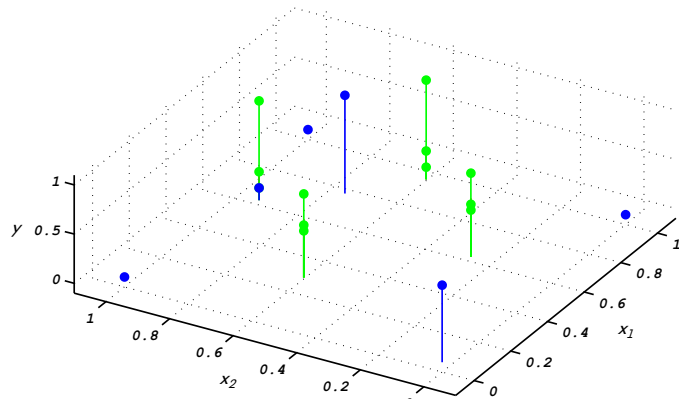


Fortsetzung auf der folgenden Seite

Beschreibung

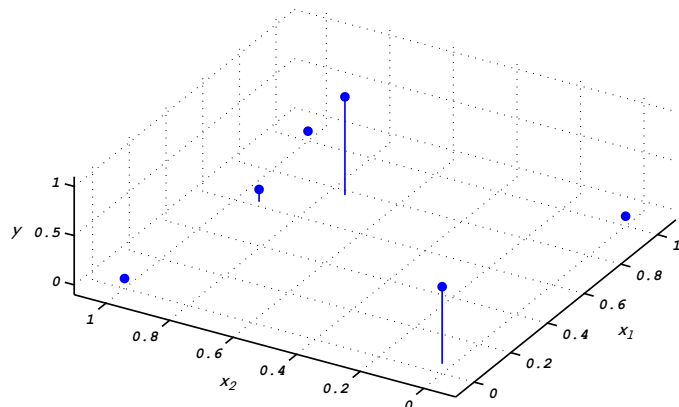
Bildliche Darstellung

Schritt 7 Aus den Informationen über die Lage der kleinsten Werte $P_{min,S}$ und der Sicherheiten σ_S aller untersuchten Simplexe S wird für die nächste Iteration ein neuer Punkt $\{\mathbf{x}_{M+1}, y_{M+1}\}$ synthetisiert und zum Datensatz $D = \{\mathbf{X}_M, \mathbf{y}_M\}$ hinzugefügt. Wenn einer oder mehrere der in den Simplexen berechneten kleinsten Werte $P_{min,S}$ eine definierte Schwelle ξ unterschreiten, wird der dazugehörige Mittelwert M_S für die nächste Iteration gewählt. Andernfalls wird für die nächste Iteration der Mittelwert M_S des Simplexes mit der größten Unsicherheit verwendet.



Schritt 8 Das Verfahren wird mit der nächsten Iteration und dem aktualisierten Datensatz $D = \{\mathbf{X}_{M+1}, \mathbf{y}_{M+1}\}$ im Schritt 2 fortgesetzt.

Nach dem Ablauf einer vorgegebenen Iterationszahl wird der beste in den Schritten 2 bis 7 bestimmte Iterationspunkt als Startwert für eine lokale Optimierung verwendet.



4.3.3 Diskussion

Anwendungsbeispiel

Nachfolgend ist ein plakatives Beispiel für die Anwendung des vorgestellten Verfahrens gegeben. Für den Test des Verfahrens wird auf das erstellte Modell des Dieselmotors zurückgegriffen. Der technische Aufbau des Triebwerkes und des Modells ist im Kapitel 3.4 beschrieben.

Der Umfang des übertragenen Steuergerätecodes beträgt 120 Funktionsmodule. Verbunden sind diese Module mit 10 Modellen der *Steller-Sensor*-Strecken.

Für die zu optimierenden Einflussgrößen wurden 6 Parameter einer Kennfeldvariation ausgewählt.

Die Parameter $P = \{p_1, p_2, \dots, p_6\}$ bestimmen die Form der Sollkennfelder des Dieselmotors durch Modifikationen der Originalkennfelder. Die Modifikationen sind in Form von verschiedenen Skalierungen implementiert. Die aus *a priori* Wissen bestimmte Grundform kann hierbei beibehalten werden. Die Implementierung bietet einerseits ausreichend Variationsraum für die Optimierungsrechnungen und verhindert gleichzeitig eine ungültige Vorgabe der Sollwerte.

Die Tabelle 4.2 beinhaltet eine Zuordnung der einzelnen Parameterwerte p_i und der zugehörigen Sollkennfelder des Dieselmotors.

Tabelle 4.2: Zusammenstellung der Einflussparameter und der Sollkennfelder des modellierten Dieselmotors

Parameter	Kennfeld
p_1	Zylinderfüllung
p_2	AGR-Rate
p_3	AGR-Aufteilungsfaktor (HD/ND)
p_4	AGR-Aufteilungsfaktor (HD/ND)
p_5	Lage der Haupteinspritzung
p_6	Lage der Haupteinspritzung

Die Zielgrößen der Optimierung sind durch 3 Kenngrößen der Abgasemissionen und des Verbrauchs dargestellt und zu einem Gütevektor $G = [G_1, G_2, G_3]$ zusammengefasst. Ausgehend von einer Initialbedingung ist das Ziel der Optimierung ein NO_x -Konzept abzuleiten. Hierbei wird bei einem ungefähr gleichbleibenden Verbrauch eine starke Einsparung der NO_x -Emissionen gefordert, wobei die Ruß-Emissionen eine vorgegebene Schranke nicht überschreiten dürfen.

In der Tabelle 4.3 sind die Zielgrößen, der Ausgangszustand vor der Optimierung und der Zielzustand nach der Optimierung dargestellt. Auf der rechten Seite der Tabelle sind die Optimierungsergebnisse zusammengetragen. Die Optimierung, mit einem in der Matlab-Umgebung implementiertem Standardwerkzeug, *fmincon*¹ ist mit dem hier vorgestellten Verfahren *DelaunaySearch* und einer anschließenden lokalen Optimierung mit *fmincon* verglichen.

Für die Optimierungsrechnungen beider Ansätze sind 85 Iterationen angesetzt. Die Optimierung mit *fmincon* wurde durch das Verfahren selbst nach 57 Iterationen mit dem Erreichen eines lokalen Minimums abgebrochen. Bei der Optimierung mit *DelaunaySearch* + *fmincon* wurden die Iterationen wie folgt aufgeteilt: 12 Iterationen *Grundvermessung*, 60 Iterationen *DelaunaySearch* und 13 Iterationen *fmincon*. Die Wirksamkeit dieses Verfahrens kann anhand des Vergleichs der beiden Ansätze beurteilt werden. Der Einsatz des beschriebenen Verfahrens weist eine deutliche Verbesserung der Optimierungsergebnisse auf, bei G_1 um 1.4 %, bei G_2 um 12.77 % und bei G_3 um 11.86 %. Die vorgestellten Ergebnisse des *DelaunaySearch*-Verfahrens fallen unter die definierten Zielwerte des Gütevektors [140, 100, 200].

¹<http://www.mathworks.de/de/help/optim/ug/fmincon.html>

Tabelle 4.3: Zielgrößen und Randbedingungen der Optimierung

Zielgröße	Gütekriterium	Ausgangszustand	Zielzustand	f_{\mincon}	$DelaunaySearch$ + f_{\mincon}	Differenz
Verbrauch	G_1	138	140	140.94	138.99	-1.4 %
NO_x	G_2	150	100	106.86	94.76	-12.77 %
$Ru\beta$	G_3	159	200	222.98	199.34	-11.86 %

Vor- und Nachteile

Ausgehend vom vorgestellten Vergleich der Optimierungsrechnungen konnte ein besseres Optimierungsergebnis erzielt werden. Durch eine systematische Analyse der Lage der durchgeführten Iterationspunkte im Raum konnte im vorliegenden Fall eine Globalisierung der Suche realisiert werden ohne eine Zufallsvariable zu verwenden.

Jedoch ergeben sich einige offene Fragen, die bei der Verwendung des Verfahrens diskutiert werden müssen.

Delaunay-Triangulation Die Berechnungszeit der Delaunay-Triangulation steigt vor allem mit der Dimension N des Parameterraumes $\mathbf{x} = [x_1 \dots x_N]^T$ und der Anzahl M der Punkte im Datensatz $D = \{\mathbf{X}_M, \mathbf{y}_M\}$. Die Implementierung der Triangulation in der Matlab-Umgebung ist mit der Qhull-Bibliothek [BDH96], [Qhu14] realisiert. Bei jeder Iteration der beschriebenen Methode wird somit eine neue Berechnung der Triangulation für alle Punkte des Datensatzes D durchgeführt. Dieses verlangsamt das Verfahren und führt zur Ineffizienz. Andere Methoden sind für die Anwendung innerhalb der entwickelten Methode besser geeignet. Der Einsatz der inkrementellen Methode zur Berechnung der Delaunay-Triangulation [Kle05, S. 272 ff.] kann mit dem vorliegenden iterativen Verfahren zur Effizienzsteigerung führen. Ausgehend aus einer vorliegenden Delaunay-Triangulation $XDT = DT(\{\mathbf{X}_M, \mathbf{y}_M\})$ wird der Fall der Aufnahme eines neuen Punktes $\{\mathbf{x}_{M+1}, y_{M+1}\}$ in den Datensatz $D = \{\mathbf{X}_M, \mathbf{y}_M\}$ betrachtet und eine Aktualisierung der Delaunay Triangulation XDT durchgeführt. Genau diese Vorgehensweise wird ausgehend aus dem **Schritt 7** im **Schritt 2** des entwickelten Verfahrens benötigt. Durch die Verwendung einer Hilfsstruktur in Form eines gerichteten Graphen und der Speicherung von Konstruktionsinformationen der Delaunay-Dreiecke erreicht diese Implementierung eine bessere Berechnungseffizienz [Kle05, S. 277 ff.]. Die Anbindung dieser Berechnungsmethode an die Matlab-Umgebung könnte mit der CGAL-Bibliothek [CGA14] und dem Konstrukt der C-Mex Funktionen [The14b] realisiert werden.

Abbruchkriterien Bezugnehmend auf die beschriebene Logik zur Auswahl des nächsten zu simulierenden Punktes, im **Schritt 7** des *DelaunaySearch*-Verfahrens, gestaltet sich die Definition eines Abbruchkriteriums als schwierig. Falls sich kein aussichtsreicher Punkt für ein Minimum findet, wird der Mittelpunkt des Simplexes mit der größten Unsicherheit vermessen. Mit

dieser Auswahlstrategie ist das Verfahren so konzipiert, dass die Konvergenz zu einem lokalen Minimum nicht zwingend erforderlich ist. In der Praxis wird das Verfahren nach einer vordefinierten Iterationsanzahl N_{DS} abgebrochen. Ausschlaggebend für die Definition der maximalen Iterationsanzahl ist maximale akzeptable Dauer der Optimierungsrechnungen T_g . Diese setzt sich zusammen aus der Dauer einer Grundvermessung T_{Init} , der Anwendungsdauer des *DelaunaySearch* Verfahrens T_{DS} und der anschließenden lokalen Optimierungsdauer T_{Opt} . In der Praxis hat sich folgende Aufteilung als Sinnvoll erwiesen:

$$\begin{aligned} T_{Init} &\approx 0.25 \cdot T_g \\ T_{DS} &\approx 0.5 \cdot T_g \\ T_{Opt} &\approx 0.25 \cdot T_g \end{aligned} \quad (4.14)$$

Ausgehend von der bekannten Dauer für eine Iteration T_{Iter} des Modells folgt:

$$N_{DS} = \frac{T_{DS}}{T_{Iter}} \quad (4.15)$$

Mit der Bestimmung von N_{DS} ist ein vorläufiges Abbruchkriterium des Verfahrens definiert. Bei Bedarf kann das Verfahren jederzeit unterbrochen oder weitergeführt werden.

Wahl der Grundvermessung Die Wahl der Grundvermessung ist ein weiteres wichtiges Kriterium beim Einsatz des *DelaunaySearch*-Verfahrens. Folgend dem **Schritt 1** des Verfahrens werden bei der Suche nach einem Minimum nur Punkte der konvexen Hülle der Grundvermessung in Betracht gezogen $conv(X_N)$. Erfahrungsgemäß lassen sich alle Eckpunkte der Verstellparameter $X = \{x_1, \dots, x_N\}$ aufgrund der höheren Anzahl der benötigten Simulationen nicht immer in die Grundvermessung aufnehmen. Die Anzahl wächst exponentiell mit der Anzahl der Eingänge entsprechend der Formel (3.1) bzw. (3.2). Alternativ kann an dieser Stelle eine DoE-Vermessung verwendet werden. Bei einem einfachen Modellansatz kann eine Grundvermessung erzeugt werden, die den kompletten Versuchsraum abdeckt. Diese Vermessung kann mit einer Kreuzvariation der Verstellparameter kombiniert werden. Hierbei wird ausgehend von der Normaleinstellung nacheinander jeweils nur ein Parameter auf seine *Min/Max*-Werte eingestellt.

5 Optimierung der Betriebsstrategie eines Hybridfahrzeugs

5.1 Motivation und Ziele

Entsprechend den im Kapitel 1.3 vorgestellten Zielen der Arbeit wird eine gemeinsame Optimierung der Betriebsstrategie eines Hybridfahrzeugs und des Verbrennungsmotorverhaltens durchgeführt. Aufbauend auf folgenden Arbeitsergebnissen aus den Kapiteln 2, 3 und 4 wird in diesem Kapitel die Parametrierung der Steuerung des Verbrennungsmotors und der Hybrid-Betriebsstrategie gemeinsam optimiert.

1. Die Simulation von Steuererätefunktionen in einer Simulationsumgebung für Fahrzeuglängsdynamik (s. Kapitel 2)
2. Die Verwendung eines nichtlinearen dynamischen Prozessmodells des Verbrennungsmotors (s. Kapitel 3)
3. Den Einsatz von den vorgestellten Optimierungsmethoden (s. Kapitel 4)

Die Optimierung beinhaltet weiterhin den Entwurf und die Implementierung einer neuen Funktion, die die Schnittstelle zwischen der Hybrid-Betriebsstrategie und der Motorsteuerung erweitert. Um Verbesserungspotentiale der implementierten Funktion abzuschätzen, werden mehrere Optimierungsansätze untersucht und ausgewertet. Die Optimierungsansätze beinhalten die simulationsgestützte Optimierung des Fahrzeugverhaltens, mit und ohne den zusätzlichen Funktionalitäten. Durch einen direkten Vergleich der Optimierungsergebnisse werden die gesuchten Verbesserungspotentiale aufdeckt. Hierbei ist es wichtig, die beschriebenen Werkzeuge auf ihre Eignung im Kontext des Optimierungsvorhabens zu untersuchen. Das im Kapitel 2 realisierte Werkzeug zur Übertragung der Steuereräte-Software in die Fahrzeugsimulationsumgebung konnte am Beispiel des Verbrennungsmotors untersucht und bewertet werden. An dieser Stelle sind nicht die Simulationseigenschaften der übertragenen Module von Interesse, vielmehr ist die Eignung der bereitgestellten Softwareumfänge zur Erstellung und zum Test einer neuen Funktion von Bedeutung. Analog hierzu sind die im Kapitel 4 beschriebenen Optimierungswerkzeuge bezüglich ihrer Eigenschaften bei der Optimierung des Fahrzeugverhaltens zu untersuchen. Die Qualität der Ergebnisse und die Anzahl der benötigten Iterationen, bzw. die benötigte Optimierungsdauer stehen hierbei im Fokus der Betrachtungen.

Für das weitere Vorgehen und die Ableitung der Ziele des Optimierungsvorhabens ist eine grundlegende Betrachtung der Optimierungspotentiale notwendig. Hierzu gehören vor allem die Wechselwirkungen zwischen der Betriebsstrategie des Hybridfahrzeugs und dem Verhalten des Verbrennungsmotors. Eine der Grundverknüpfungen ist die Drehmoment-Schnittstelle. Hierbei stellt die Betriebsstrategie in Abhängigkeit vom Ladezustand der Traktionsbatterie (SoC) unterschiedliche Drehmoment-Anforderungen an den Verbrennungsmotor. Somit sind bei einer entladenen Batterie eher hochlastige und bei einer voll geladenen Batterie eher niederlastige Betriebspunkte des Verbrennungsmotors zu erwarten. Bei der Betrachtung eines Fahrzyklusses führen dementsprechend unterschiedliche Einstellungen der Betriebsstrategie zu unterschiedlichen Zeitverläufen des Batterieladezustandes und zu unterschiedlichen Drehmoment-Anforderungen an den Verbrennungsmotor. Wird während eines Fahrzyklusses viel elektrisch gefahren, was durchaus zu den kundenrelevanten Kriterien zählt, so sinkt der Batterieladezustand kontinuierlich ab und führt zu längeren und vor allem intensiveren Nachladephasen. Dieses Verhalten hat eine entsprechende Auswirkung auf die NO_x -Emissionen, die ihrerseits in den höherlastigen Betriebspunkten ansteigen. Das Verhalten ist beispielhaft für das untersuchte Simulationsfahrzeug in der Abbildung 5.1 abgebildet. Im Teil *a*) der Abbildung ist die Fahrt im NEFZ-Zyklus mit Ausgangseinstellungen dargestellt. Im Teil *b*) der Abbildung kann die Fahrt mit erweiterten elektrischen Fahrphasen begutachtet werden. Im unteren Teil der Abbildung *c*) ist der direkte Vergleich der NO_x -Emissionen für beide Einstellungen visualisiert. Die Differenz zwischen den beiden Fällen beträgt hierbei ca. 25 %.

Um das Emissionsverhalten des Verbrennungsmotors im gesamten Betriebsraum zu beurteilen, sind in der Abbildung 5.2 die NO_x - und $Ru\beta$ -Emissionen über die Drehzahl n_{Mot} und das Drehmoment M_{Mot} des Motors dargestellt. Grundlage der Darstellung bildet eine Vermessung des Verbrennungsmotors mit der Standardparametrierung des Steuergeräts. Während die NO_x -Emissionen in den Hochlastbereichen stark zunehmen, sinken die $Ru\beta$ -Emissionen stark ab. In den Niederlastbereichen ist der umgekehrte Zusammenhang erkennbar. Die unterschiedlichen Einstellungen der Betriebsstrategie und die damit verbundenen Änderungen der Drehmomentanforderungen an den Verbrennungsmotor modifizieren das Abgasemissionsverhalten des Fahrzeugs. Bezogen auf das Beispiel in der Abbildung 5.1 kann hierdurch der Anstieg der NO_x -Emissionen verifiziert werden.

Durch das implementierte Verbrennungsmotormodell kann das Verhalten der Verbrennungskraftmaschine an die jeweiligen Anforderungen angepasst werden. Im Gegensatz zu einer festen Kennfeldimplementierung, reagiert das erstellte Modell variabel auf verschiedene Einstellungen des Luft- und Kraftstoffpfads und bietet somit die für eine gemeinsame Optimierung benötigte Variabilität an. Folglich wird der Hybrid-Betriebsstrategie die Möglichkeit geboten, nicht nur eine Momentenanforderung zu stellen, sondern auch über die Art der Momentenerzeugung zu bestimmen. Beispielhaft können bei niedrigen Ladezuständen der Traktionsbatterie und hochlastigen Betriebsanforderungen des Verbrennungsmotors gezielt Maßnahmen zur Vermeidung von NO_x -Emissionen realisiert werden. Zusammenfassend lässt sich für das Optimierungsvorhaben folgendes Ziel ableiten:

Eine gemeinsame Optimierung der Hybrid-Betriebsstrategie und des Verhaltens des Verbrennungsmotors soll dazu beitragen den Verbrauch zu minimieren, die elektrischen Fahrphasen zu maximieren und die Abgasemissionen unter vorgegebenen Schranken zu halten.

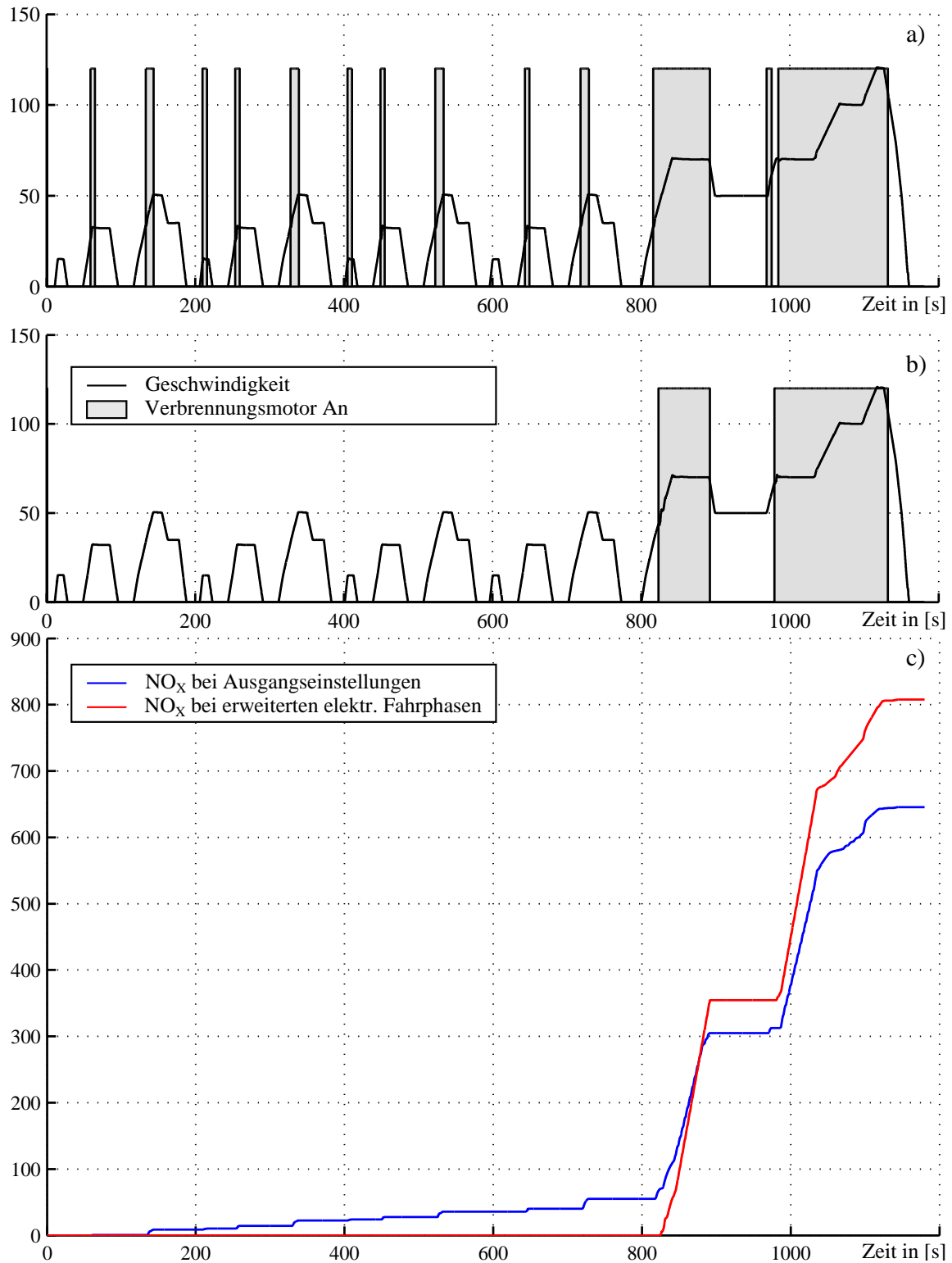


Abbildung 5.1: Simulation einer Fahrt des NEFZ-Zyklus mit verschiedenen Einstellungen der Betriebsstrategie. **Teil a):** Simulation mit Ausgangseinstellungen. **Teil b):** Simulation mit Einstellungen für erweiterte elektrische Fahrphasen. **Teil c):** Direkter Vergleich der NO_x-Emissionen für beide Einstellungen der Betriebsstrategie

Bei der Realisierung soll eine Funktion implementiert werden, welche die Schnittstelle zwischen der Betriebsstrategie und der Motorsteuerung erweitert. In Abhängigkeit vom Ladezustand der Trak-

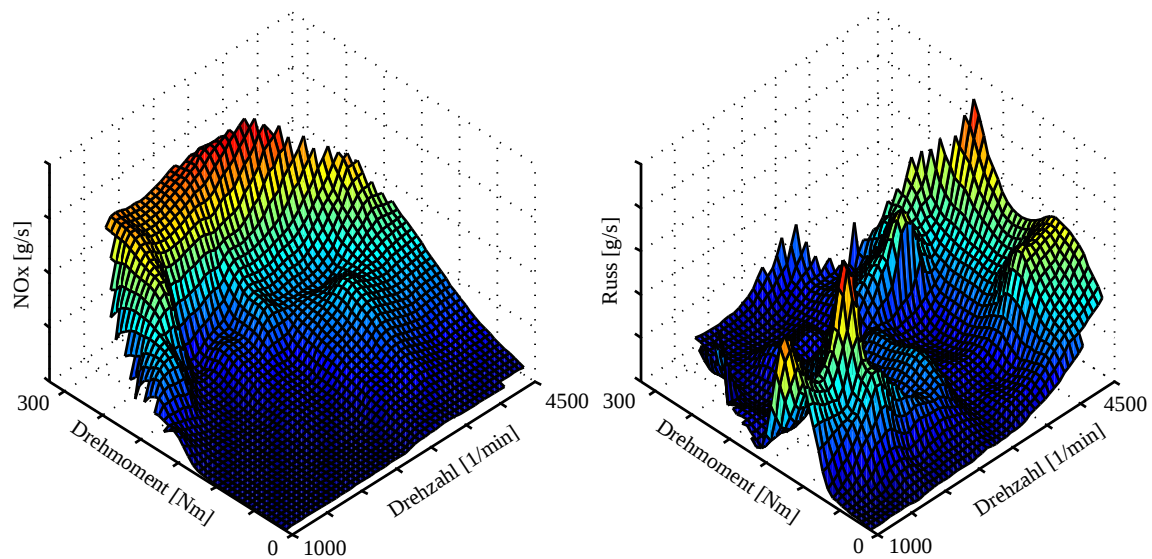


Abbildung 5.2: Darstellung der gemessenen NO_x und $Ru\beta$ -Emissionen im gesamten Betriebsraum des Verbrennungsmotors

tionsbatterie soll das Verhältnis zwischen den NO_x - und den $Ru\beta$ -Emissionen des Verbrennungsmotors so angepasst werden, dass auch bei erweiterten elektrischen Fahrphasen die Grenzvorgaben nicht überschritten werden.

5.2 Implementierung

5.2.1 Ausgangszustand

Der Ansatzpunkt bei der Implementierung der notwendigen Funktionsumfänge, ist die Betrachtung des Ausgangszustands. Hierzu gehören vor allem die Funktionsumfänge der Hybrid-Betriebsstrategie, die Steuerung des Verbrennungsmotors und des Getriebes. Die in die Gesamtfahrzeugsimulation übertragenen Funktionsumfänge des Steuergeräts sind in der Abbildung 5.3 zusammengefasst. Die Funktionen der Hybrid-Betriebsstrategie sind im Hybrid-Modul zusammengefasst. Diese umfassen die Berechnung folgender Größen und Fahrzeugzustände:

- Verbrennungsmotor an/aus
- Start- und Stoppart des Verbrennungsmotors
- Die Nachlade- bzw. Entladeleistung, falls der Verbrennungsmotor an ist

Für die Realisierung der Berechnung werden verschiedene Bedingungen überprüft, die den Fahrzeugzustand beschreiben und einen Start- bzw. einen Stoppvorgang des Verbrennungsmotors auslösen.

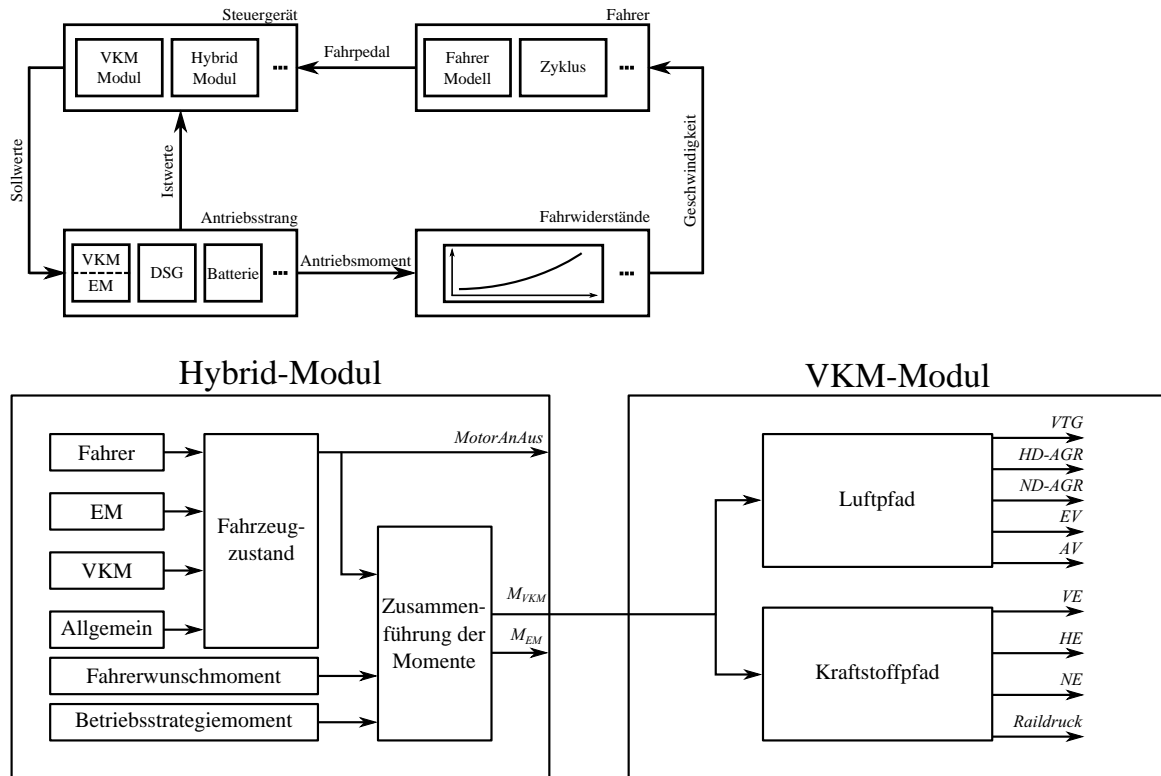


Abbildung 5.3: Darstellung der Softwarestruktur der Hybrid-Betriebsstrategie und der Dieselmotorsteuerung

Die einzelnen Bedingungen sind in Form von verschiedenen Schwellen, Kennfeldern und Kennlinien realisiert und können in unterschiedlichen Kategorien zusammengeführt werden. Unter anderem sind an dieser Stelle plakativ vier verschiedene Kategorien aufgeführt. In der Kategorie *Fahrer* können Bedingungen zusammengefasst werden, die das Verhalten des Fahrers auswerten. Beispielfähig seien an dieser Stelle die Auswertung der Fahrpedalstellung und des dazugehörigen Gradienten denkbar. Die Kategorie *EM* beinhaltet dementsprechend Bedingungen, die den Zustand des Elektromotors prüfen. Zu dieser Kategorie würden zum Beispiel die maximalen Drehzahl- und Drehmomentgrenzen zählen, ab denen der Start des Verbrennungsmotors notwendig ist. Eine Zusammenstellung der, für den Verbrennungsmotor zutreffenden, Bedingungen kann in der entsprechenden Kategorie *VKM* implementiert werden. Hier würden die Temperatur- und Mindestlaufzeitbedingungen des Verbrennungsmotors zusammenpassen. Als letztes Beispiel ist die Kategorie *Allgemein* aufgeführt. In dieser Kategorie ist die Zusammenstellung der restlichen Bedingungen, wie zum Beispiel die Geschwindigkeits- oder Außentemperaturvorgaben zusammenzufassen. Da es sich an dieser Stelle um plakative Beispiele handelt, sind weitere Kategorien und Bedingungen nicht nur denkbar, sondern eher realistisch.

Aufgrund zusätzlicher Bedingungen könnten bei einem angeforderten Start bzw. Stopp des Verbrennungsmotors unterschiedliche Start- bzw. Stopparten realisiert werden. Für weitere Betrachtungen im Rahmen dieser Arbeit ist dieses nicht relevant. Es wird nur eine Start- und eine Stoppart betrachtet.

Falls die Verbrennungskraftmaschine an ist, bestimmt die Hybrid-Betriebsstrategie die Lade- bzw. Entladeleistung der Traktionsbatterie. Bei der Berechnung sind Kenngrößen wie Momentanforde-

rung des Fahrers, Ziel-SoC, Bordnetzbedarf, Fahrzeuggeschwindigkeit etc. relevant.

Die Steuerung des Verbrennungsmotors (s. Abbildung 5.3, VKM-Modul) ist für die Einstellung des, von der Betriebsstrategie geforderten, Momentes zuständig. Da ein moderner Verbrennungsmotor viele Anforderungen aus den unterschiedlichen Bereichen, wie Verbrauch, Emissionen, Komfort etc. erfüllen muss, verfügt dieser über viele Stellgrößen. Vor allem die Ansteuerung der Stellgrößen des Luft- und Kraftstoffpfades ist durchaus umfangreich und komplex. Zur Konfiguration des Luftpfades zählen unter anderem Stellgrößen wie die variable Turbinengeometrie (VTG), die Menge des zurückgeführten Abgasstromes und die Aufteilung dieses auf die Hochdruck- bzw. Niederdruck-Strecke (HD- bzw. ND-AGR) und die Berechnung der Einlass- bzw. Auslassventilsteuerzeiten (EV bzw. AV). Der Kraftstoffpfad umfasst die Konfiguration der Einspritzcharakteristik und des Raildrucks. Die Einspritzung unterteilt sich im vorliegenden Fall auf die Einstellungen von mehreren Voreinspritzungen (VE), der Haupteinspritzung (HE) und einer Nacheinspritzung (NE). Innerhalb eines Hybrid-Fahrzeugs erfährt der Verbrennungsmotor Anforderungen, die sich je nach Einstellung der Betriebsstrategie von den Anforderungen eines konventionellen Fahrzeuges unterscheiden. Somit liegt die Überlegung nahe, das Verhalten dieser Antriebseinheit an die jeweiligen Anforderungen der Betriebsstrategie anzupassen.

5.2.2 Funktionsentwurf

Ausgehend von den vorangegangenen Überlegungen wird in diesem Unterkapitel der Entwurf einer neuen Steuererätefunktion beschrieben. Hierfür werden die in der Abbildung 5.1 vorgestellten Simulationsergebnisse einer weiteren, genaueren Prüfung unterzogen (s. Abbildung 5.4). Aus der Analyse der Untersuchungsergebnisse können dann Anforderungen für eine Erweiterung der in der Abbildung 5.3 vorgestellten Funktionsumfänge abgeleitet werden.

Im Teil *a*) der Abbildung 5.4 sind die vom Verbrennungsmotor in einem NEFZ-Zyklus angefahrenen Betriebspunkte $\{n_{Mot}, M_{Mot}\}$ in einem Diagramm zusammengetragen. Die Auswertung der dargestellten Betriebspunkte in einem Histogramm ist im Teil *b*) der Abbildung 5.4 zu finden. Die Leistungsstufen sind hierbei in 5 kW-Schritten eingeteilt. Im Vergleich zu den *Ausgangseinstellungen* führt die Implementierung von *erweiterten elektrischen Fahrphasen* zur Verschiebung des Leistungsspektrums zu höheren Leistungen. Die Ursache hierfür stellen die intensiveren Nachladephase der Traktionsbatterie dar. Entsprechend der Abbildung 5.4 *b*) werden Leistungen ab 40 kW viel öfter vom Verbrennungsmotor zur Verfügung gestellt. Die Betriebspunkte aus der Leistungsklasse 60 – 70 kW werden mit den Ausgangseinstellungen nicht erreicht. Es ergeben sich abhängig von den getroffenen Einstellungen der Hybrid-Betriebsstrategie und dem daraus resultierenden Verlauf des SoC unterschiedliche Anforderungsprofile des Verbrennungsmotors.

Um das Verhalten des Verbrennungsmotors an die variablen Anforderungen anzupassen, werden in der Steuerung des Verbrennungsmotors zusätzliche Betriebsmodi eingeführt und damit die Schnittstelle zwischen der Hybrid-Betriebsstrategie und der Steuerung des Verbrennungsmotors erweitert. Die Betriebsmodi stellen bei vergleichsweise gleichbleibenden Verbrauchswerten unterschiedliche Emissionskonzepte dar. Für eine entladene Batterie und die erwarteten hochlastigen Anforderungen des Verbrennungsmotors, stehen Betriebsmodi zur Verfügung, in denen der Motor weniger NO_x -Emissionen emittiert und gleichzeitig die Ruß-Emissionen unter einer vorgegebenen Schranke hält

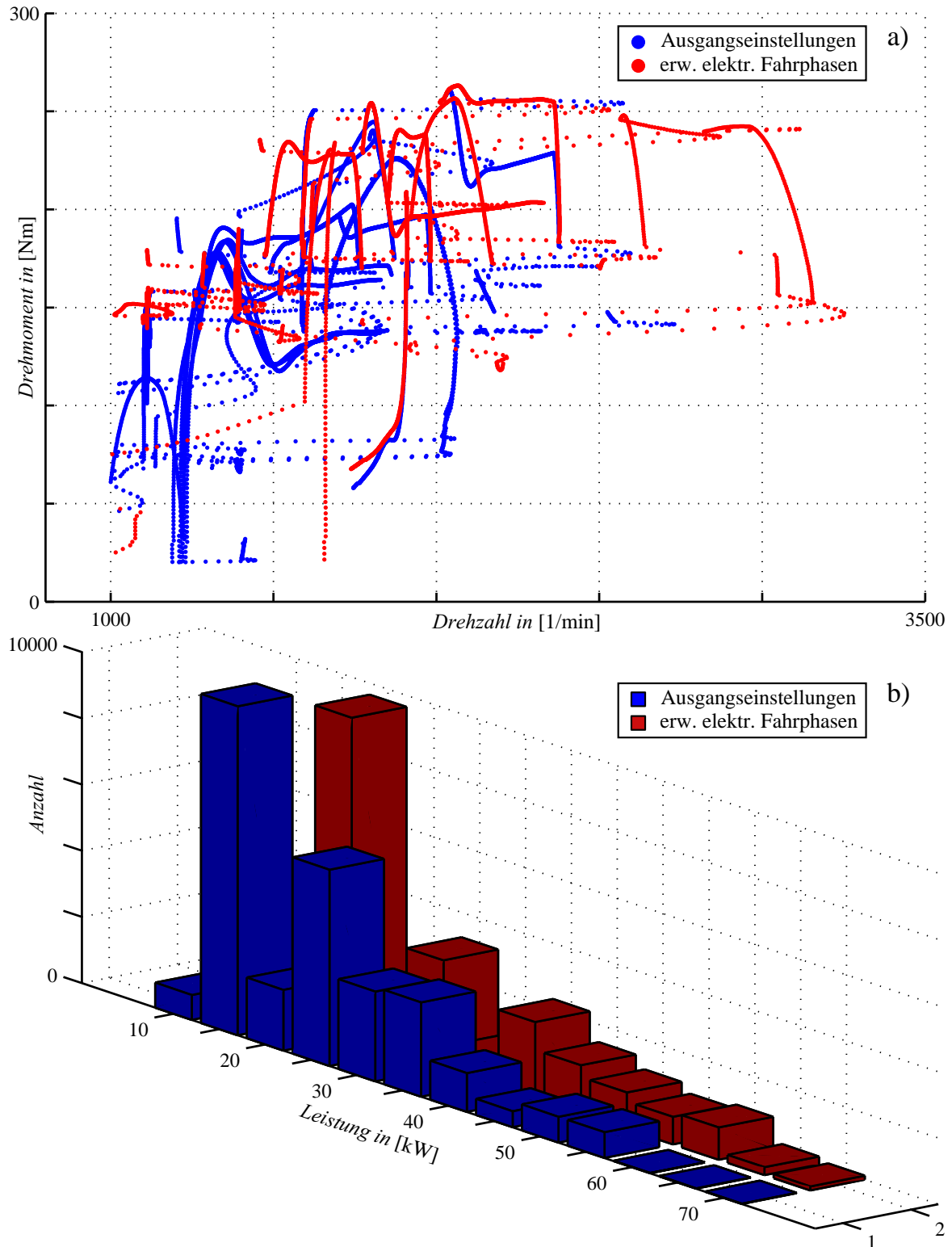


Abbildung 5.4: Unterschiedliche Anforderungsprofile des Verbrennungsmotors in einem NEFZ-Zyklus in Abhängigkeit von den Einstellungen der Hybrid-Betriebsstrategie. **Teil a):** Direkter Vergleich der simulierten Betriebspunkte $\{n_{Mot}, M_{Mot}\}$ des Verbrennungsmotors. **Teil b):** Auswertung der Betriebspunkte in einem Histogramm.

(sog. NO_X -Konzept). Für eine geladene Batterie und die erwarteten niederlastigen Anforderungen sind Betriebsmodi implementiert, die zur Vermeidung der $Ru\beta$ -Emissionen und der Einhaltung ei-

ner Obergrenze für die NO_x -Emissionen führen (sog. *Ruß-Konzept*). Eine Entscheidung über die Verwendung des einen oder des anderen Betriebsmodus erfolgt über die Hybrid-Betriebsstrategie. Innerhalb dieser ist die Berechnung des einzusetzenden Betriebsmodus über eine variable Anbindung an den Ladezustand der Traktionsbatterie realisiert.

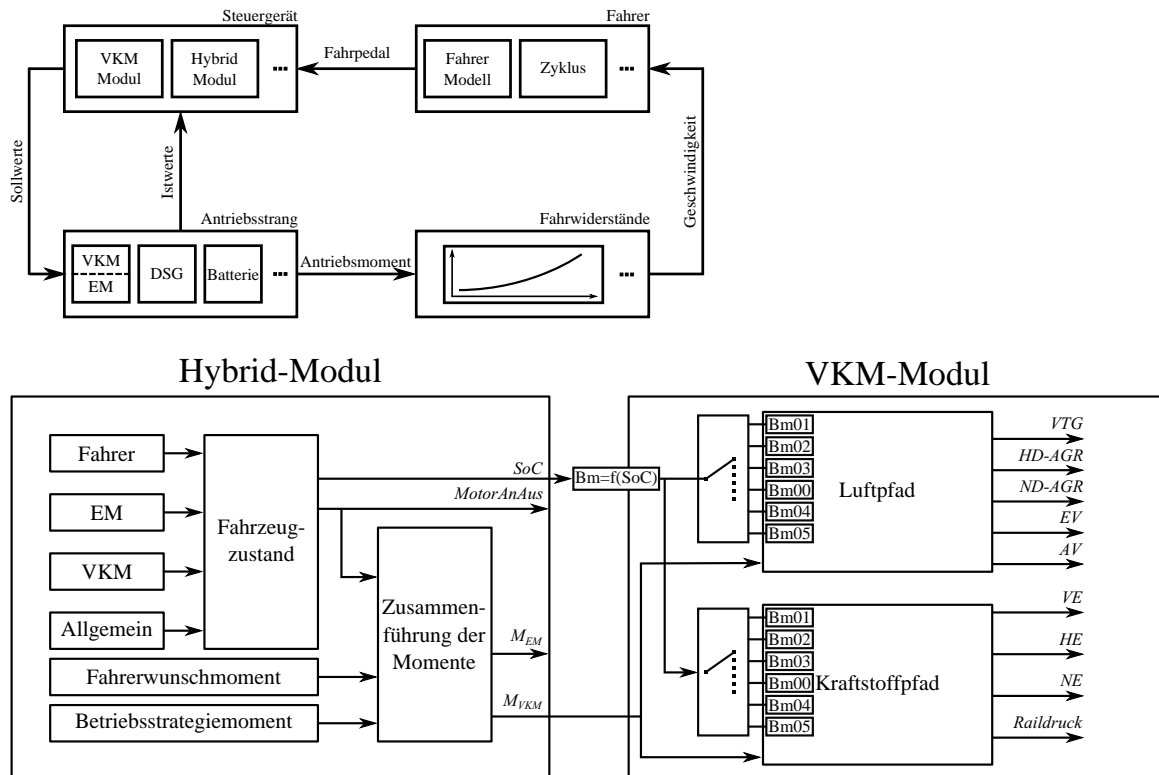


Abbildung 5.5: Einbindung des Funktionsentwurfes in die Softwarestruktur der Hybrid-Betriebsstrategie und der Dieselmotorsteuerung. Zusätzlich zu der, in der Abbildung 5.3 dargestellten, Softwarestruktur erfolgt an dieser Stelle die Auswertung des SoC -Signals zur Berechnung eines passenden Betriebsmodi des Verbrennungsmotors ($Bm = f(SoC)$). Im *VKM-Modul* sind weiterhin die Betriebsmodi des Verbrennungsmotors $Bm00$ bis $Bm05$ eingezeichnet

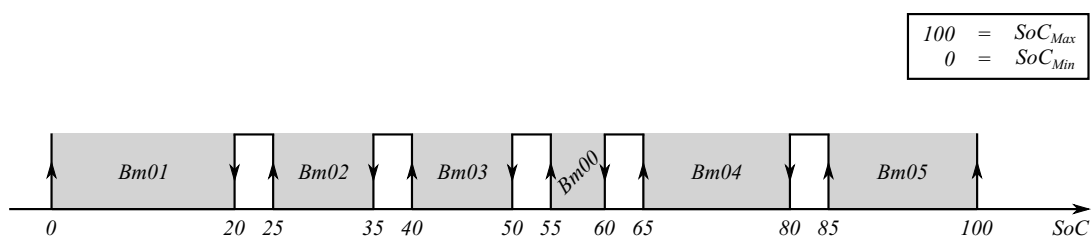


Abbildung 5.6: Schwellwerte der Betriebsmodi der Verbrennungskraftmaschine aufgetragen über den Ladezustand der Traktionsbatterie. Die schattierten Flächen stellen die gültigen Bereiche für den jeweiligen Betriebsmodus dar. Die Pfeile deuten ein Schmitt-Trigger Verhalten an, was ein stabiles umschalten zwischen den Betriebsmodis gewährleistet.

Die Implementierung des Funktionsentwurfes ist mit dem Softwareentwicklungswerkzeug Ascet

durchgeführt. Anschließend erfolgt die Übertragung des Funktionsmoduls in die Fahrzeugsimulationsumgebung mit dem im Kapitel 2 entwickelten Werkzeug. Es ergibt sich der in der Abbildung 5.5 dargestellte Aufbau des Steuergerätmoduls.

Die Auswahl der Betriebsmodi des Verbrennungsmotors (VKM-Modul, $Bm00$ bis $Bm05$) ist über mehrere Schwellenwerte mit dem Ladezustand der Traktionsbatterie verbunden. Somit entsteht eine weitere Schnittstelle zwischen der Hybrid-Betriebsstrategie und der Verbrennungsmotorsteuerung ($Bm = f(SoC)$). Die Abbildung 5.6 visualisiert die Verteilung der Schaltschwellen zwischen den einzelnen Betriebsmodi in Abhängigkeit vom Ladezustand der Traktionsbatterie bei der Ausgangsparametrierung. Die Bedeutung der einzelnen Betriebsmodi $Bm00$ bis $Bm05$ ist durch die Angaben der Tabelle 5.5 gegeben. Ist der Ladezustand der Batterie kleiner als der Schwellwert des neutralen Betriebsmodus $Bm00$, so wird ein entsprechender Betriebsmodus zur Vermeidung der NO_x -Emissionen ausgewählt ($Bm01$, $Bm02$ oder $Bm03$). Auf der anderen Seite des $Bm00$ sind hingegen Betriebsmodi zur Einsparung von $Ru\beta$ -Emissionen vorgesehen ($Bm04$ und $Bm05$).

5.3 Optimierung

5.3.1 Einflussparameter und Gütekriterien

Für die Optimierung des Fahrzeugverhaltens erfordert ein Optimierungsalgorithmus die Definition von Einflussparametern und Gütekriterien. Die Einflussparameter werden im Rahmen der Optimierung vom Optimierungsalgorithmus variiert. Die Bewertung des modifizierten Systemverhaltens erfolgt anschließend mit Hilfe von entsprechenden Gütekriterien. In den nachfolgenden Abschnitten sind die, für die gemeinsame Optimierung der Hybrid-Betriebsstrategie und des Verhaltens des Verbrennungsmotors, implementierten Einflussparameter und Gütekriterien ausführlich beschrieben. Alle beschriebenen Einflussparameter dienen der Modifikation entsprechender Kennlinien und Kennfeldern der Funktionsmodule der Hybrid-Betriebsstrategie und der Steuerung des Verbrennungsmotors. Aufgrund der Simulation von Steuergerätesoftware¹ entsprechen die modifizierten Kennlinien und Kennfelder stets der original im Steuergerät verwendeten Parametrierung. Im Unterschied zur Verwendung von modellierten bzw. vereinfachten Funktionsmodulen des Steuergeräts in der Simulationsumgebung, ist durch diese Vorgehensweise eine reibungslose Übertragbarkeit der Ergebnisse ins Fahrzeug sichergestellt. Somit ergibt sich die im Kapitel 2.1 beschriebene Prozessänderung bei der Kalibrierung der verschiedenen Funktionsmodule des Steuergeräts (s. Abbildung 2.1). Weiterhin bietet diese Vorgehensweise die Möglichkeit zusätzlich zu einer energetischen Betrachtung des Antriebsstrangs², das Fahrzeugverhalten bezüglich der Abgasemissions- und des Komfortverhaltens zu optimieren.

In der Tabelle 5.1 sind die implementierten Einflussparameter der Hybrid-Betriebsstrategie und der Steuerung des Verbrennungsmotors zusammengestellt. Wie auch im Optimierungsbeispiel im Kapitel 4.3.3, wird bei der Modifikation betreffender Kennlinien und Kennfelder durch die Parameter p_i die Grundform stets beibehalten. Die Implementierung bietet ausreichend Variationsraum für die

¹Entsprechend der im Kapitel 2 beschriebenen Simulation der Funktionsmodule eines Steuergeräts

²Hierbei liegt die Optimierung des Verbrauchs im Fokus der Betrachtungen

Modifikation der Steuergeräteparametrierung und sichert gleichzeitig eine ungültige Vorgabe ab.

Tabelle 5.1: Einflussgrößen der Hybrid-Betriebsstrategie und der Steuerung des Verbrennungsmotors bei der Optimierung.

	Parameter	Beeinflusste Kennlinie/Kennfeld
Verbrennungsmotor	p_1	Zylinderfüllung
	p_2	AGR-Rate
	p_3	AGR-Aufteilungsfaktor (HD/ND)
	p_4	AGR-Aufteilungsfaktor (HD/ND)
	p_5	Lage der Haupteinspritzung
	p_6	Lage der Haupteinspritzung
	p_7	Lage der Voreinspritzung 2
	p_8	Lage der Voreinspritzung 1
	p_9	Lage der Nacheinspritzung 3
	p_{10}	Menge der Nacheinspritzung 3
	p_{11}	Raildrucksollwert
Betriebsstrategie	p_{12}	Nach- bzw. Entladekennfeld
	p_{13}	Fahrpedalstellung beim Start des Verbrennungsmotors
	p_{14}	Drehzahl der E-Maschine beim Start des Verbrennungsmotors
	p_{15}	Schaltkennfeld im Hybridmodus
	p_{16}	Schaltkennfeld im Elektromodus
	p_{17}	Schwellwerte der VKM-Betriebsmodi

Die Einflussparameter der Verbrennungsmotorsteuerung bewirken verschiedene Anpassungen der Steller des Luft- und Kraftstoffsystems. Eine Übersicht über die Anordnung der Eingriffsmöglichkeiten ist im Strukturbild des zur Verfügung stehenden Verbrennungsmotors dargestellt (s. Abbildung 5.7). Es folgt eine detaillierte Beschreibung der verwendeten Einflussparameter.

Zylinderfüllung

Der Parameter p_1 beeinflusst das Kennfeld der gewünschten Zylinderfüllung des Verbrennungsmotors. Diese setzt sich zusammen aus der Frischluft, der Abgasrückführrate und der Restgasmasse im Zylinder. Die Auswirkungen der Verstellung dieser Größe wirken vor allem auf die Einstellung des Abgasturboladers. Im vorliegenden Fall wird die VTG-Stellung angepasst (s. Abbildung 5.7, Position 2).

Abgasrückführrate, Aufteilungsfaktor

Der Parameter p_2 modifiziert das Kennfeld für die Vorgabe der Abgasrückführrate (AGR-Rate) im Zylinder. Diese setzt sich zusammen aus der Summe der zurückgeführten Abgasmenge über die Hochdruck- und die Niederdruckschleife des Luftpfads (Abbildung 5.7, Gasfluss über die Positionen 7 und 10). Die Aufteilung der Gasmenge auf die beiden Schleifen ist über einen Aufteilungsfaktor gesteuert. Die Beeinflussung des Vorgabekennfelds dieses Faktors ist durch die Parameter p_3 und p_4

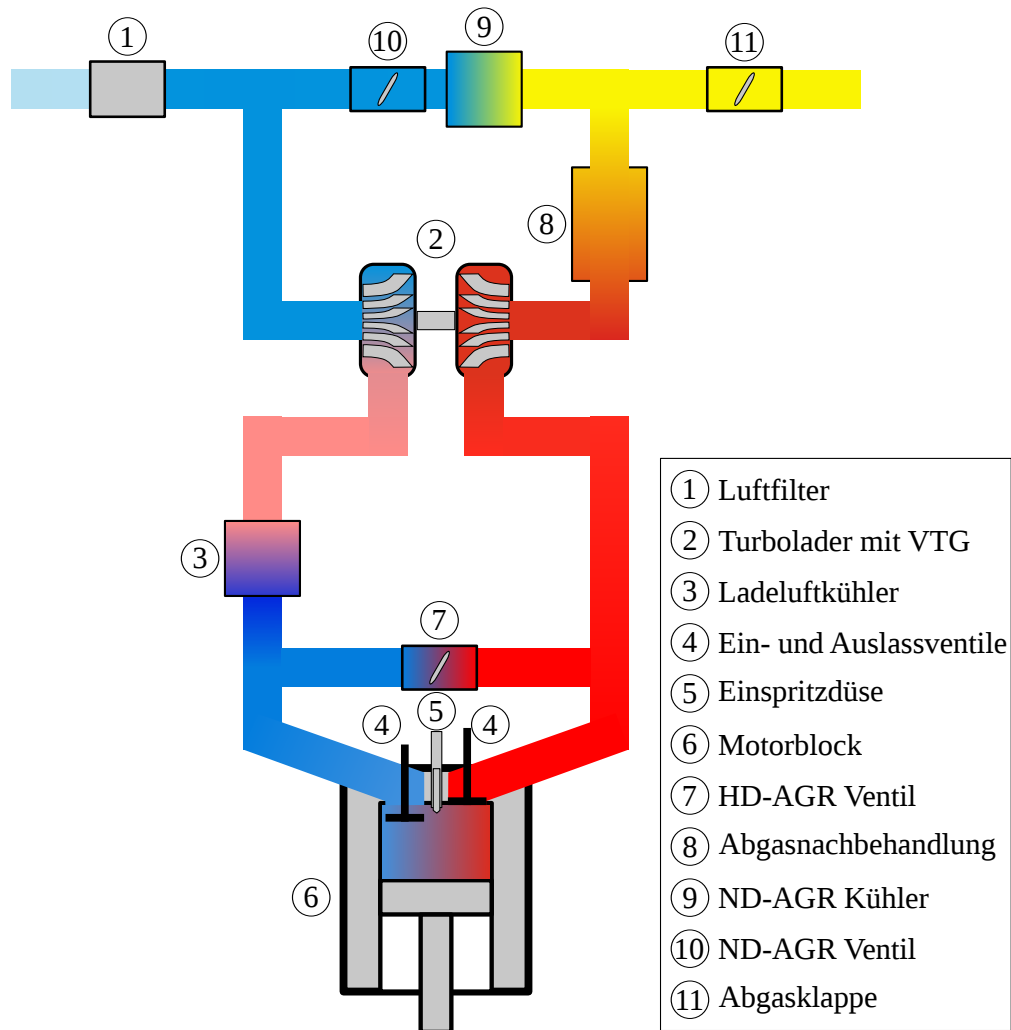


Abbildung 5.7: Übersicht über die Variationsgrößen des Luft- und Einspritzsystems des Verbrennungsmotors, die während der Optimierung vom Optimierungsalgorithmus verstellt werden

realisiert.

Die Verstellung der Parameter p_2 , p_3 und p_4 wirkt hauptsächlich auf die Stellung der Abgasrückführventile der Hoch- und der Niederdruckschleife und die Stellung der Abgasklappe (s. Abbildung 5.7, Position 11).

Konfiguration des Einspritzsystem

Eine Übersicht über die variierten Größen des Einspritzsystems ist in der Abbildung 5.8 abgebildet. Um die Lage der Haupteinspritzung zu beeinflussen, verstellt der Optimierungsalgorithmus die Parameter p_5 und p_6 . Hierdurch variiert das Vorgabekennfeld dieser wichtigen Verbrennungseinflussgröße. Die Parameter p_7 und p_8 beeinflussen die Lage der Voreinspritzungen 2 und 1. Die Variation erfolgt hierbei immer relativ zur Lage der Haupteinspritzung. Analog hierzu bewirkt der Verstellparameter p_9 eine Anpassung des Vorgabekennfelds für die relative Lage der Nacheinspritzung 3. Zusätzlich erfolgt durch Anpassung des Parameters p_{10} die Änderung der Einspritzmenge

dieser Nacheinspritzung.

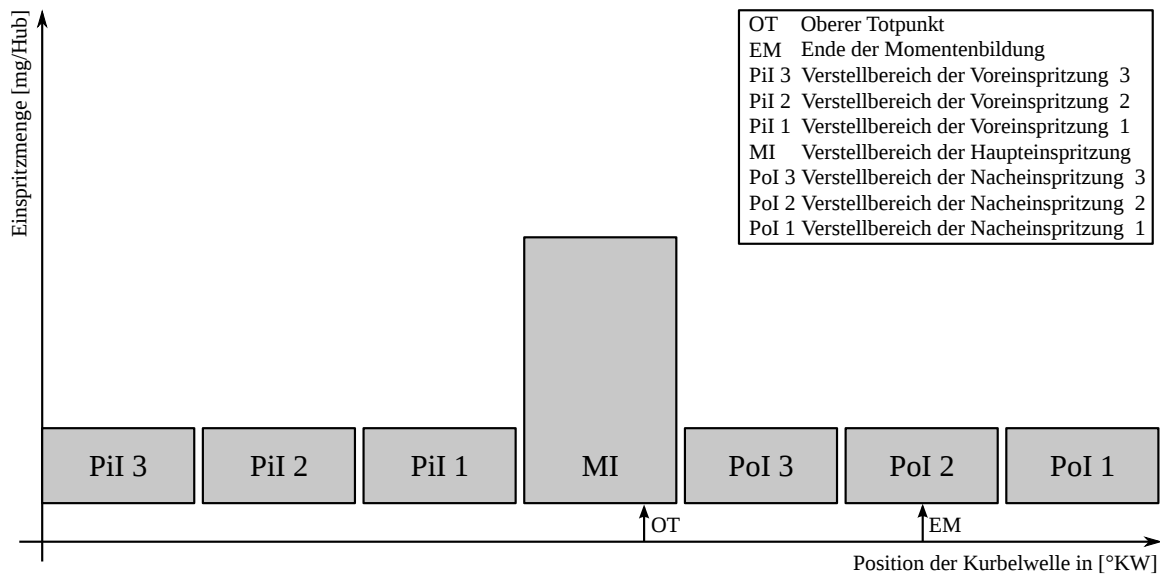


Abbildung 5.8: Variationsgrößen des Einspritzsystems, die während der Optimierung der Betriebsstrategie vom Optimierungsalgorithmus verstellt werden

Raildruck

Mit der Verstellung des Parameters p_{11} verfügt der Optimierungsalgorithmus über die Möglichkeit, den Sollwert des Kraftstoffdrucks im Einspritzsystem zu verstellen. Auch an dieser Stelle erfolgt eine Anpassung des Sollwertekennfelds.

Ladekennfelder der Hybridstrategie

Wie in den vorherigen Kapiteln 5.1 und 5.2 beschrieben, stellt die Betriebsstrategie des Hybridfahrzeugs Drehmomentanforderungen an den Verbrennungsmotor. In Abhängigkeit vom Ladezustand der Traktionsbatterie werden sowohl Lade- als auch Entlademomente angefordert. Im vorliegenden Fall wird bezogen auf den Ladezustand zunächst ein Lade- bzw. Entladekennfeld gewählt, das die Sollwertvorgaben des Verbrennungsmotormoments enthält. Falls der Verbrennungsmotor im eingeschalteten Zustand ist, wird der Drehmomentwunsch der Hybridstrategie zu dem Fahrerwunschmoment addiert, begrenzt und an den Verbrennungsmotor geleitet.

Mit der Anpassung des Parameters p_{12} erfolgt eine Skalierung der Lade- und Entladekennfelder und somit eine Anpassung des angeforderten Drehmoments.

Startanforderungen des Verbrennungsmotors

Im Kapitel 5.2.1 ist ein Konzept vorgestellt, wie die Betriebsstrategie durch die Auswertung verschiedener Bedingungen über eine Start- bzw. Stoppanforderung des Verbrennungsmotors entscheidet.

Eine der Bedingungen ist die Schwelle der Fahrpedalstellung, die zum Start des Verbrennungsmotors führt. Durch einen Offset entscheidet dieser Applikationsparameter auch über eine entsprechende Stoppfreigabe. Der Parameter p_{13} ist mit dem Schwellenwert der Fahrpedalstellung verknüpft und ermöglicht dem Optimierungsalgorithmus die Bedingungen der Start- und Stoppanforderungen anzupassen.

Eine andere Bedingung, die zu einer Startanforderung führen kann, ist von der Drehzahl der E-Maschine abhängig. Übersteigt diese eine definierte Schwelle, so wird von der Betriebsstrategie eine Startanforderung des Verbrennungsmotors ausgelöst. Die Stoppfreigabe erfolgt auch hier über einen definierten Offset. Für die Verstellung der beschriebenen Drehzahlschwelle ist der Parameter p_{16} zuständig.

Schwellwerte der VKM-Betriebsmodi

Im vorangegangenen Kapitel 5.2.2 wurde der Entwurf einer neuen Funktion beschrieben, die die Schnittstelle der Betriebsstrategie und des Verbrennungsmotors erweitert. Hierbei wurden auf der Seite des Verbrennungsmotors Betriebsmodi $Bm00$ bis $Bm05$ eingeführt und der Betriebsstrategie als ein zusätzlicher Freiheitsgrad zur Verfügung gestellt.

Durch die Variation des Parameters p_{17} wird eine Anpassung der Schaltschwellwerte der einzelnen Betriebsmodi erreicht.

Gütekriterien

Die Bewertung der Zielgrößen ist durch die Auswertung verschiedener Gütekriterien realisiert, die sich jeweils auf das gewählte Geschwindigkeitsprofil beziehen. Eine Zusammenstellung der Bewertungskriterien ist in der Tabelle 5.2 zu finden. Die Auswertung der Kriterien erfolgt anhand von einem definiertem Punktesystem.

Das G_1 beschreibt die Abweichung des Ladezustandes der Traktionsbatterie am Anfang und am Ende des simulierten Fahrzyklusses.

$$\Delta SoC = |SoC_{End} - SoC_{Start}| \quad (5.1)$$

Der Verbrauch G_2 kann anhand der Integration der Gesamteinspritzmenge beurteilt werden.

$$G_2 \sim \int \dot{Q} dt \quad (5.2)$$

wobei \dot{Q} die Gesamteinspritzmenge in $[g/s]$ ist.

Analog hierzu basiert die Emissionsbewertung G_3 bzw. G_4 , auf dem Integral der emittierten NO_X - bzw. Ruß-Werte.

Die beschriebenen Gütekriterien G_1 bis G_4 sind alle als Minimierungsproblem definiert. Die gleiche Anforderung muss auch für die Bewertung des Anteils der elektrischen Fahrphasen gelten. Dement-

sprechend wird für die Bewertung dieser Eigenschaft das Gütekriterium G_5 wie folgt definiert:

$$G_5 = \left(1 - \frac{S_{EV}}{S_{Ges}}\right) \cdot 100 \quad (5.3)$$

wobei S_{EV} der Anteil der zurückgelegten Strecke im elektrischen Fahrmodi von der gesamten Strecke S_{Ges} ist.

Tabelle 5.2: Zusammenstellung der verwendeten Gütekriterien

Gütekriterium	Bedeutung
G_1	ΔSoC
G_2	Verbrauch
G_3	NO_X -Emissionen
G_4	Ruß-Emissionen
G_5	Anteil der elektrischen Fahrphasen

5.3.2 Strategie

Bei der gemeinsamen Optimierung der Applikationsparameter der Hybrid-Betriebsstrategie und der Steuerung des Verbrennungsmotors werden verschiedene Ansätze verfolgt. Nachfolgend erfolgt eine kurze Beschreibung der einzelnen Schritte:

Ausgangszustand Die Betrachtung des Ausgangszustandes bildet den Bezugspunkt der Optimierungsrechnungen. Hierzu gehört die Darstellung der Werte des Gütefunktional für eine simulierte NEFZ-Fahrt mit folgenden Einstellungen der Betriebsstrategie:

- Verbrennungsmotor an
- Hybrid-Modus

Die Einstellung *Verbrennungsmotor an* impliziert die Simulation einer Fahrt des Hybridfahrzeugs entlang des NEFZ-Geschwindigkeitsprofils mit durchgehend eingeschaltetem Verbrennungsmotor. Analog hierzu beinhaltet die Simulation mit den *Hybrid-Modus* Einstellungen eine Fahrt des Hybridfahrzeugs mit den nicht optimierten Einstellungen der Betriebsstrategie. An dieser Stelle ist weiterhin jedes andere Geschwindigkeitsprofil denkbar.

Ansatz 1: Optimierung des Ausgangszustands Die Berechnung des optimalen Fahrzeugverhaltens ohne eine Kopplung an die verschiedenen Betriebsmodi, d.h. nur mit der im Ausgangszustand verfügbaren Drehmomentschnittstelle, soll einer besseren Einschätzung der erzielten Ergebnisse dienen.

Ansatz 2: Verifikation des Funktionsentwurfs In diesem Optimierungsvorhaben wird die implementierte Funktion verifiziert. Bezugnehmend auf die ausgesuchten Bewertungskriterien wird das Verbesserungspotenzial betrachtet, das durch die entwickelte Funktion entsteht. Die Optimierung erfolgt hierbei in zwei Schritten:

1. Die verschiedenen Betriebsmodi der NO_x - und $Ru\beta$ -Konzepte werden anhand eines typischen Anforderungsprofils des Verbrennungsmotors bestimmt.
2. Die Hybrid-Betriebsstrategie wird unter Berücksichtigung der, durch die Betriebsmodi verfügbaren, Freiheitsgrade gemeinsam mit der Steuerung des Verbrennungsmotors optimiert. Die Kopplung zwischen dem Ladezustand der Batterie und den Betriebsmodi ist hierbei variabel.

Ansatz 3: gesamtheitliche Optimierung Bei diesem Ansatz werden die Applikationsparameter der Hybrid-Betriebsstrategie und des Verbrennungsmotors zusammen optimiert. Die entwickelte Funktion wird hierfür deaktiviert. Um die Anzahl der Einflussgrößen zu begrenzen, konzentriert sich dieser Ansatz auf die *wichtigsten* Parameter des Verbrennungsmotors und der Hybrid-Betriebsstrategie.

Ansatz 4: vertiefte gesamtheitliche Optimierung Dieser Optimierungsansatz entspricht dem Ansatz 3, mit dem Unterschied, dass hierbei *alle verfügbaren* Parameter verstellt werden.

5.3.3 Ergebnisse

Ausgangszustand

Als Ausgangszustand werden die Einstellungen der Applikationsparameter vor den Optimierungsrechnungen bezeichnet. Das Fahrzeugverhalten bei einer simulierten NEFZ-Fahrt mit durchgehend eingeschaltetem Verbrennungsmotor und im Hybrid-Modus wird untersucht. Die Bewertung erfolgt anhand der deklarierten Gütekriterien.

Die Tabelle 5.3 stellt die Werte der Gütekriterien für die ausgewählten Einstellungen der Hybrid-Betriebsstrategie zusammen.

Tabelle 5.3: Ausgangszustand vor der Optimierung der Hybrid-Betriebsstrategie und des Verbrennungsmotorverhaltens

	Beschreibung	VKM An	Hybrid
G1	ΔSoC	1.41	7.73
G2	Verbrauch	535	508
G3	NO_x	502	730
G4	$Ru\beta$	110	66
G5	E-Fahrt	100	51.93

Eine besondere Aufmerksamkeit sei an dieser Stelle den Werten des Gütekriteriums G1 gewidmet. Die Abweichung zwischen dem SoC -Stand am Anfang und am Ende der NEFZ-Fahrt ist bei der Simulation im Hybrid-Modus größer als 5%. Hierdurch entsteht eine Beeinträchtigung der Vergleichbarkeit der Ergebnisse. Trotz dieser Abweichung und den nicht optimalen Einstellungen der Hybrid-Betriebsstrategie kann, im direkten Vergleich zu den Ergebnissen der *VKM An*-Simulation, eine Einsparung des Kraftstoffverbrauchs beobachtet werden. Weiterhin lässt sich ein starker Anstieg der NO_x -Emissionen und eine Absenkung der $Ru\beta$ -Emissionen feststellen. Der Wert des Gütekriteriums

G5, der mit steigendem Anteil der elektrischen Fahrphasen sinkt, ist mit dem kontinuierlich eingeschalteten Verbrennungsmotor erwartungsgemäß 100. Im Hybrid-Modus sinkt dieser auf einen Wert von 51.93. Zur besseren Einschätzung des an dieser Stelle erzielten Resultates kann die Visualisierung in der Abbildung 5.10 Teil a) betrachtet werden. Zusätzlich zu der simulierten Geschwindigkeit des Fahrzeugs im NEFZ-Zyklus, sind die Fahrphasen eingezeichnet, in denen der Verbrennungsmotor an ist.

Ansatz 1: Optimierung des Ausgangszustands

Für die Optimierung des Fahrzeugverhaltens mit den im Ausgangszustand verfügbaren Freiheitsgraden werden 5 Einflussgrößen

$$\{p_{12}, p_{13}, p_{14}, p_{15}, p_{16}\}$$

verwendet (siehe Tabelle 5.1). Die Einflussgrößen des Verbrennungsmotors und die Verstellung der Schwellwerte der VKM-Betriebsmodi bleiben auf den jeweiligen Standardwerten unverändert. Das entwickelte Funktionsmodul ist deaktiviert.

Die Optimierungsergebnisse sind in der Tabelle 5.4 dargestellt. Als Bezugsgröße der Optimierungsrechnungen dienten an dieser Stelle die im Ausgangszustand ermittelten Werte des Gütefunktionalis einer simulierten NEFZ-Fahrt im Hybrid-Modus.

Tabelle 5.4: Optimierungsergebnisse Ansatz 1: Optimierung des Ausgangszustands (nur Hybridstrategie)

	Beschreibung	Sollwert	Hybrid-Modus	Ansatz 1
G1	ΔSoC	≤ 5	7.73	3.26
G2	Verbrauch	<i>min</i>	508	490
G3	NO_x	≤ 600	730	607
G4	<i>Ruß</i>	≤ 120	66	48
G5	E-Fahrt	≤ 55	51.93	54.61

Folgende Aussagen lassen sich aus der Analyse der erzielten Optimierungsergebnisse ableiten:

- Die Werte von G1 bzw. ΔSoC liegen beim nicht optimierten Hybrid-Modus oberhalb der geforderten 5%-Grenze. Dieses schränkt die Aussagekraft des Vergleiches ein.
- Eine Absenkung der Verbrauchswerte G2 um ca. 3.5% kann dennoch beobachtet werden.

$$\Delta G2 = \frac{508 - 490}{508} = 0.0354$$

- Eine deutliche Überschreitung der NO_x -Obergrenze von 600 Punkten kann bei den entsprechenden G3-Werten des Hybrid-Modus festgehalten werden. Im Gegensatz hierzu liegen die

NO_x -Emissionen nach der Optimierung nur geringfügig über dem Zielwert.

$$\Delta G3 = \frac{730 - 600}{600} = 0.217 \quad \& \quad \Delta G3 = \frac{607 - 600}{600} = 0.0117$$

- Die Werte der $Ru\beta$ -Emissionen sind unkritisch und bleiben deutlich unter den Anforderungen.
- Trotz der Einhaltung der Obergrenze für das Gütekriterium $G5$ tritt eine deutliche Verschlechterung dieses Kriteriums ein (um ca. 5.2%).

$$\Delta G3 = \frac{54.61 - 51.93}{51.93} = 0.0516$$

Ein direkter Vergleich in der Abbildung 5.10 Teil *a)* und *b)* verdeutlicht die Unterschiede. Im innerstädtischer Teil des Zyklus beträgt die Verlängerung der hybridisch gefahrenen Abschnitte jeweils 2.5 Sekunden pro Motorstart. Im Überlandteil wird um ca. 15.5 Sekunden länger mit einem eingeschaltetem Verbrennungsmotor gefahren.

Die Nichteinhaltung des ΔSoC relativiert jedoch die Aussagen bezüglich der Abgasemissionen und den Angaben über die elektrischen Fahrphasen.

Ansatz 2: Verifikation des Funktionsentwurfs

Bei diesem Ansatz erfolgt die Verifikation der entwickelten Funktion. Der zweistufige Optimierungsansatz umfasst zunächst die Berechnung der verschiedenen Betriebsmodi für die Verbrennungskraftmaschine. Hierfür sind die definierten Einflussgrößen des Verbrennungsmotors relevant (siehe Tabelle 5.1):

$$\{p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8, p_9, p_{10}, p_{11}\}$$

Anschließend werden die Betriebsmodi der Hybrid-Betriebsstrategie zur Verfügung gestellt. In Abhängigkeit von SoC -Schwellen kann diese einen passenden Modus der Verbrennungskraftmaschine einstellen. Die Schwellen selbst können über den Applikationsparameter bzw. die Einflussgröße p_{17} modifiziert werden. Somit ergeben sich die 6 Einflussgrößen im zweiten Schritt dieser Optimierung zu:

$$\{p_{12}, p_{13}, p_{14}, p_{15}, p_{16}, p_{17}\}$$

1. Berechnung der Betriebsmodi für die Verbrennungskraftmaschine Für die Berechnung der verschiedenen Betriebsmodi des Verbrennungsmotors sind die Gütekriterien G_2 , G_3 und G_4 von Bedeutung. Mittels eines typischen Drehzahl-/Drehmomentprofils des Hybridfahrzeugs aus dem NEFZ werden die einzelnen NO_x - und $Ru\beta$ -Konzepte bestimmt. Die Tabelle 5.5 fasst die Optimierungsziele der Gütekriterien zusammen.

Tabelle 5.5: Optimierungsziele der Gütekriterien G_2 , G_3 und G_4 bei der Berechnung der Betriebsmodi $Bm00$ bis $Bm05$ des Verbrennungsmotors

Betriebsmodus	Zielwerte	Bedeutung
	$[C_2 \ C_3 \ C_4]$	
$Bm01$	$[min \ 425 \ 260]$	starkes Nachladen
$Bm02$	$[min \ 450 \ 200]$	mittleres Nachladen
$Bm03$	$[min \ 475 \ 150]$	schwaches Nachladen
$Bm00$	$[min \ 500 \ 60]$	neutral Modus
$Bm04$	$[min \ 700 \ 50]$	schwaches Entladen
$Bm05$	$[min \ 900 \ 30]$	starkes Entladen

Der Betriebsmodus $Bm00$ bildet einen neutralen Modus. Der Verbrennungsmotor wird in diesen Modus nur dann versetzt, wenn der Ladezustand der Traktionsbatterie geringe bis gar keine Abweichung vom Ziel-SoC aufweist. Ausgehend von diesem Betriebsmodus werden drei NO_X -Konzepte anvisiert, $Bm01$, $Bm02$ und $Bm03$. Diese implementieren eine stufenweise Reduktion der NO_X -Emissionen. Auf der anderen Seite sind die Betriebsmodi $Bm04$ und $Bm05$, die bei einer vollen Traktionsbatterie aktiviert werden. Hierbei liegt der Fokus der Betrachtungen auf der Einsparung der $Ru\beta$ -Emissionen.

Die Tabelle 5.6 fasst die erzielten Optimierungsergebnisse für die einzelnen Betriebsmodi zusammen.

Tabelle 5.6: Optimierungsergebnisse der Betriebsmodi des Verbrennungsmotors

Betriebsmodus	Zielwerte	Optimierungsergebnisse
	$[C_2 \ C_3 \ C_4]$	$[G_2 \ G_3 \ G_4]$
$Bm01$	$[min \ 425 \ 260]$	$[445 \ 408 \ 76]$
$Bm02$	$[min \ 450 \ 200]$	$[440 \ 445 \ 62]$
$Bm03$	$[min \ 475 \ 150]$	$[439 \ 472 \ 60]$
$Bm00$	$[min \ 500 \ 60]$	$[438 \ 496 \ 58]$
$Bm04$	$[min \ 700 \ 50]$	$[432 \ 686 \ 29]$
$Bm05$	$[min \ 900 \ 30]$	$[431 \ 889 \ 22]$

Um die Wechselwirkungen zwischen den einzelnen Gütekriterien zu visualisieren, sind diese für die einzelnen Betriebsmodi in Abbildung 5.9 zusammengefasst.

2. Optimierung der Betriebsstrategie mit der implementierten Funktion Die Ergebnisse der Optimierungsrechnungen sind in der Tabelle 5.7 zusammengestellt. Als Vergleichswerte dienen an dieser Stelle die Werte der Gütekriterien $G1$ bis $G5$ aus dem Ansatz 1.

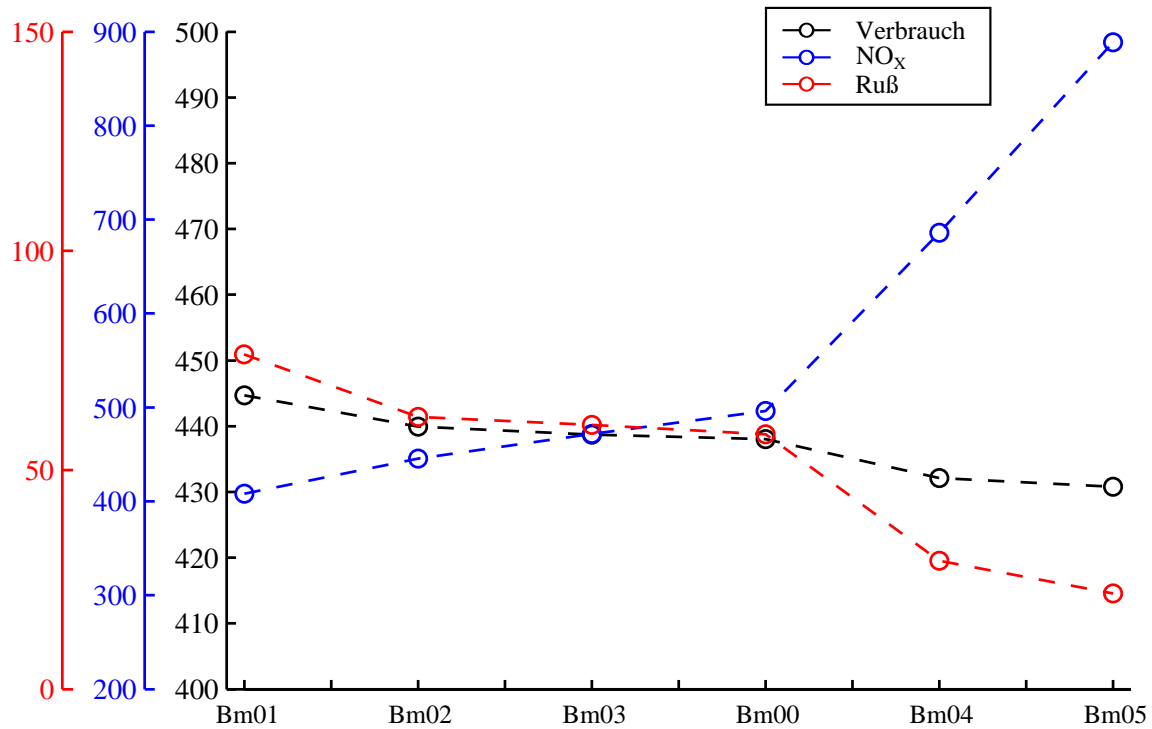


Abbildung 5.9: Graphische Darstellung der verschiedenen Betriebsmodi des Verbrennungsmotors

Tabelle 5.7: Optimierungsergebnisse Ansatz 2: Verifikation des Funktionsentwurfs (Hybrid-Betriebsstrategie mit der entwickelten Funktion)

	Beschreibung	Sollwert	Hybrid-Modus	Ansatz 1	Ansatz 2
G1	ΔSoC	≤ 5	7.73	3.26	4.84
G2	Verbrauch	<i>min</i>	508	490	500
G3	NO_x	≤ 600	730	607	284
G4	$Ru\beta$	≤ 120	66	48	88
G5	E-Fahrt	≤ 55	51.93	54.61	53.13

Aus einem direkten Vergleich der beiden Optimierungsansätze 1 und 2 können folgende Aussagen abgeleitet werden:

- Die Optimierungslösungen liegen unter der 5%-Grenze des ΔSoC und sind somit vergleichbar.
- Die Verbrauchswerte sind auf einem vergleichbaren Niveau. Durch die Freigabe der implementierten Betriebsmodianpassung ist dennoch ein geringer Anstieg der Verbrauchswerte zu beobachten.

$$\Delta G2 = \frac{500 - 490}{490} = 0.02$$

Die Änderungen können mit der starken Verminderung der NO_x -Emissionen plausibilisiert werden, liegen jedoch im Bereich der Modellgenauigkeit.

- Die implementierten Betriebsmodi der Verbrennungskraftmaschine bewirken eine starke Absenkung der NO_X -Emissionen und einen gleichzeitigen Anstieg der $Ru\beta$ -Emissionen.
- Der direkte Vergleich des Anteils der elektrischen Fahrphasen ergibt eine Verbesserung dieses kundenrelevanten Kriteriums:

$$\Delta G5 = \frac{54.61 - 53.13}{54.61} = 0.027$$

Obwohl die Änderung auf den ersten Blick nicht signifikant erscheint, bewirkt die Empfindlichkeit dieses Gütekriteriums im vorliegenden Bereich eine merkliche Anpassung des Fahrzeugverhaltens. Für einen Vergleich der beiden Lösungen ist die Betrachtung der Teile *b*) und *c*) aus der Abbildung 5.10 wichtig. Die Unterschiede werden erst bei der genaueren Betrachtung des innerstädtischen Teils (s. Abbildung 5.11, Teil *a*)) und des Überlandteils (s. Abbildung 5.11, Teil *b*)) des NEFZ deutlich. Im innerstädtischen Teil erfolgt bei der Beschleunigung auf 50 km/h der Start des Verbrennungsmotors ca. 3 Sekunden später. Im Überlandteil ergibt sich eine Verschiebung des elektrischen Fahrbetriebes von der ersten auf die zweite Beschleunigung auf die 70 km/h.

Ansatz 3: gesamtheitliche Optimierung

Eine direkte Optimierung der Hybrid-Betriebsstrategie und des Verbrennungsmotorverhaltens ist bei diesem Ansatz realisiert. Hierfür ist die implementierte Funktion, die der Betriebsstrategie verschiedene Betriebsmodi des Verbrennungsmotors zur Verfügung stellt, abgeschaltet. Im Gegensatz zum Ansatz 4 konzentriert sich die Wahl der Einflussgrößen auf alle Einstellparameter der Betriebsstrategie und die *wichtigsten* Einstellparameter des Verbrennungsmotors (insgesamt 10 Parameter, Tabelle 5.1):

$$\{p_1, p_2, p_5, p_6, p_{11}, p_{12}, p_{13}, p_{14}, p_{15}, p_{16}\}$$

Die Ergebnisse dieses Optimierungsansatzes sind in der Tabelle 5.8 zusammengetragen. Zur besseren Einschätzung der Ergebnisse sind auch an dieser Stelle weiterhin die Ergebnisse der Ansätze 1 und 2 aufgeführt.

Tabelle 5.8: Optimierungsergebnisse Ansatz 3: gesamtheitliche Optimierung (Hybrid-Betriebsstrategie und die wichtigsten Einstellparameter des Verbrennungsmotors)

	Beschreibung	Sollwert	Ansatz 1	Ansatz 2	Ansatz 3
G1	ΔSoC	≤ 5	3.26	4.84	3.41
G2	Verbrauch	<i>min</i>	490	500	507
G3	NO_X	≤ 600	607	284	487
G4	$Ru\beta$	≤ 120	48	88	102
G5	E-Fahrt	≤ 50	54.61	53.13	48.06

Folgende Erkenntnisse lassen sich aus dem Vergleich gewinnen:

- Bezogen auf den Ansatz 2 steigen die Verbrauchswerte um 1.4 %. Analog ergibt sich eine Verschlechterung zum Ansatz 1 um 3.5 %.
- Die Emissionen unterschreiten die definierten Obergrenzen mit einem deutlichem Abstand.
- Bei diesem Optimierungsansatz konnte der Sollwert für das Gütekriterium G_5 von 55 auf 50 abgesenkt werden. Bezogen auf den Ansatz 1 beträgt die Absenkung ca. 12 % und auf den Ansatz 2 entsprechend ca. 9.5 %. Diese starke Verbesserung ist in der Abbildung 5.10 Teil *d*) visualisiert. Die Fahrt im innerstädtischem Teil des NEFZ konnte komplett im elektrischen Fahrmodus realisiert werden. Trotz dieser langen elektrischen Fahrphase erfahren die Verbrauchswerte nur einem geringen Anstieg und die Abgasemissionen übersteigen nicht die definierten Obergrenzen.

Ansatz 4: vertiefte gesamtheitliche Optimierung

Als eine Erweiterung bzw. Vertiefung des Ansatzes 3 erfolgt an dieser Stelle eine Optimierung mit allen verfügbaren Einflussgrößen der Betriebsstrategie und des Verbrennungsmotors. Es ergeben sich insgesamt 16 zu optimierende Parameter (siehe Tabelle 5.1):

$$\{p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8, p_9, p_{10}, p_{11}, p_{12}, p_{13}, p_{14}, p_{15}, p_{16}\}$$

Der Parameter p_{17} ist aufgrund der Deaktivierung der implementierten Funktion bei diesem Ansatz nicht relevant.

Die nachfolgende Tabelle 5.9 beinhaltet eine direkte Gegenüberstellung der Ansätze 1 bis 4.

Tabelle 5.9: Optimierungsergebnisse Ansatz 4: vertiefte gesamtheitliche Optimierung (Hybrid-Betriebsstrategie und alle Einstellparameter des Verbrennungsmotors)

	Beschreibung	Sollwert	Ansatz 1	Ansatz 2	Ansatz 3	Ansatz 4
G1	ΔSoC	≤ 5	3.26	4.84	3.41	4.82
G2	Verbrauch	<i>min</i>	490	500	507	505
G3	NO_x	≤ 600	607	284	487	465
G4	<i>Ruß</i>	≤ 120	48	88	102	76.04
G5	E-Fahrt	≤ 50	54.61	53.13	48.06	48.07

Zur Analyse wird an dieser Stelle der Vergleich der Ansätze 3 und 4 durchgeführt.

- Durch das Hinzunehmen von zusätzlichen Einflussgrößen konnten die Verbrauchswerte im Ansatz 4 um ca. 0.4 % reduziert werden. Diese Verbesserung liegt jedoch unterhalb der Modellgenauigkeit.

- Die Abgasemissionen liegen auch in diesem Fall deutlich unter den definierten Obergrenzen. Eine Verbesserung der Emissionswerte kann dennoch festgestellt werden. Vor allem eine Reduktion der Ruß-Emissionen um ca. 25 % konnte realisiert werden.
- Der Anteil elektrischer Fahrphasen im Zyklus ist in diesem Fall konstant geblieben. Eine Visualisierung wird nicht durchgeführt.

Diskussion

Nach der Vorstellung der erreichten Optimierungsergebnisse erfolgt an dieser Stelle zunächst die Betrachtung der erforderlichen Optimierungsdauer der einzelnen Ansätze. Die Tabelle 5.10 beinhaltet eine Gegenüberstellung der benötigten Iterationen.

Tabelle 5.10: Aufteilung der für die Optimierungsrechnungen benötigten Simulationen

	Ansatz 1	Ansatz 2	Ansatz 3	Ansatz 4
Anzahl der Einflussgrößen	5	6	10	16
Grundvermessung	129 Iter.	241 Iter.	233 Iter.	256 Iter.
DelaunaySearch	100 Iter.	100 Iter.	– Iter.	– Iter.
Lokale Optimierung	250 Iter.	100 Iter.	210 Iter.	200 Iter.
Summe	479 Iter.	449 Iter.	443 Iter.	456 Iter.

Für die Berechnung der Ergebnisse wurden im Schnitt ca. 450 Iterationen benötigt. Bei einer Simulationsdauer von ca. 18 Minuten pro Iterationsschritt ergibt sich eine Optimierungsdauer von ca. 135 Stunden. Die zur Optimierung des Fahrzeugverhaltens benötigte Zeit stellt eine weiche Obergrenze dar, die als akzeptabel gilt. Somit wurde die Suche nach einem Optimum nach der Durchführung der angegebenen Iterationsanzahlen nicht mehr fortgeführt.

Bei den Ansätzen 1 und 2 wurde das entwickelte Verfahren *DelaunaySearch* erfolgreich eingesetzt. Bei den Optimierungsansätzen konnte mit dem Algorithmus stets ein guter Startwert für die abschließende lokale Optimierung bestimmt werden. Ein Einsatz des Verfahrens bei den Ansätzen 3 und 4 war aufgrund der Anzahl der Einflussgrößen nicht möglich. Die im Kapitel 4.3.3 vorgestellten Nachteile des implementierten Verfahrens zur Berechnung der Delaunay-Triangulation führen zu einer Ineffizienz des *DelaunaySearch*-Algorithmus. Somit erfordert die Berechnung der Delaunay-Triangulation (siehe Schritt 2 des *DelaunaySearch*-Algorithmus) mit steigender Anzahl der Einflussgrößen unverhältnismäßig viel Zeit. Nach der Berechnung der Grundvermessungspunkte hat sich der Einsatz des Verfahrens somit als nicht sinnvoll erwiesen. In diesem Fall wurde eine umfangreiche Grundvermessung durchgeführt. Diese bestand aus einem DoE-Vermessungsplan und einer Kreuzvariation, so wie diese im Kapitel 4.3.3 beschrieben ist. Der bezüglich der gewählten Kriterien beste Simulationspunkt wurde anschließend als Startwert für eine lokale Optimierung verwendet. Mit diesem Vorgehen konnte ein durchaus zufriedenstellendes Optimierungsergebnis gefunden werden.

Weiterhin ist zu beachten, dass für die Erstellung der verschiedenen Betriebsmodi im Ansatz 2

mehrere Optimierungsrechnungen mit einem typischen Drehzahl-/Drehmomentprofil durchgeführt wurden. Die Dauer einer Iteration beträgt hierbei ca. 4 Minuten, da nur der Verbrennungsmotor simuliert wird. Mit einer Grundvermessung von 746 Iterationen und einer anschließenden lokalen Optimierung im Umfang von 100 Iterationen pro Betriebsmodi ergibt sich eine zusätzliche Optimierungsdauer von ca. $(746 + 5 \cdot 100) \cdot 4/60 = 83$ Stunden.

Analog hierzu muss darauf hingewiesen werden, dass beim Ansatz 4 die Optimierungsergebnisse des Ansatzes 3 in die Grundvermessung aufgenommen wurden. Das somit eingeflossene *a priori* Wissen hat entscheidend zur Bestimmung des optimalen Fahrzeugverhaltens beigetragen. Trotz einer umfangreichen Grundvermessung wurden genau diese Ergebnisse als Startwert der lokalen Optimierung vom Algorithmus ausgewählt.

Die entwickelten Werkzeuge und die eingesetzten Methoden haben sich bei der Lösung der vorliegenden Optimierungsaufgabe als geeignet erwiesen. Bei allen Optimierungsansätzen konnte stets eine den Anforderungen entsprechende Lösung gefunden werden.

Gesamtheitlich betrachtet, ist der Einsatz der im Kapitel 5.2.2 entwickelten Funktion nicht zwingend erforderlich. Durch die direkte Optimierung der Stellgrößen der Hybrid-Betriebsstrategie und der Verbrennungskraftmaschine sind ähnliche Ergebnisse erreichbar. Die gewählten Kriterien in den Optimierungsrechnungen enthalten jedoch keine Bewertung der Robustheit und der Allgemeingültigkeit der Ergebnisse. An dieser Stelle könnte die beschriebene Funktion durchaus Vorteile bringen. Die unterschiedlichen Betriebsmodi des Verbrennungsmotors, die der Betriebsstrategie zur Verfügung gestellt werden, sind nicht direkt von einem gewählten Geschwindigkeitsprofil bzw. Drehzahl-/Lastprofil abhängig. Dadurch ist auch das Optimierungsergebnis nicht direkt auf einen Geschwindigkeitsprofil fixiert. Besonders im Hinblick auf die RDE-Prüfverfahren¹ und die damit verbundene Bewertung der Abgasemissionen im realen Fahrbetrieb wird eine verbesserte Robustheit der Ergebnisse erwartet.

Mit Hilfe der Simulation der Fahrzeuglängsdynamik und dem entwickelten Werkzeug zur Übertragung der Funktionsmodule in die Simulations-Umgebung konnte eine effiziente Testumgebung für Steuergerätefunktionen realisiert werden. Die Implementierung eines Funktionsentwurfes, der Transfer des Entwurfs in die Simulationsumgebung und eine anschließende Optimierung ermöglichte sowohl eine Analyse des Zusammenspiels mit anderen Funktionsmodulen des Steuergeräts als auch eine Abschätzung des Verbesserungspotentials. Der optimierte Datensatz kann weiterhin direkt in einem Fahrzeugsteuergerät eingesetzt werden.

Die in der Praxis ermittelte Optimierungsdauer lässt offensichtlich eine Verbesserung zu, ist aber gleichzeitig für einen Test der eingesetzten Simulations- und Optimierungsmethoden mehr als akzeptabel. Eine Verlagerung der Berechnungen von lokalen Rechenkapazitäten in ein leistungsfähigeres Rechenzentrum bei gleichzeitiger Parallelisierung der Iterationsläufe ist wesentlich effizienter. Dieser Schritt würde nicht nur die Gesamtoptimierungsdauer signifikant verkürzen, sondern auch gleichzeitig mehr Iterationen und mehr Einflussgrößen erlauben.

¹RDE steht für *Real Driving Emissions*. Unter diesem Begriff werden vom Gesetzgeber Prüfverfahren definiert, die die Abgasemissionen eines Fahrzeugs im realen Fahrbetrieb bewerten. Die Erfüllung der RDE-Grenzwerte wird künftig für die Zulassung eines Fahrzeugs relevant sein.

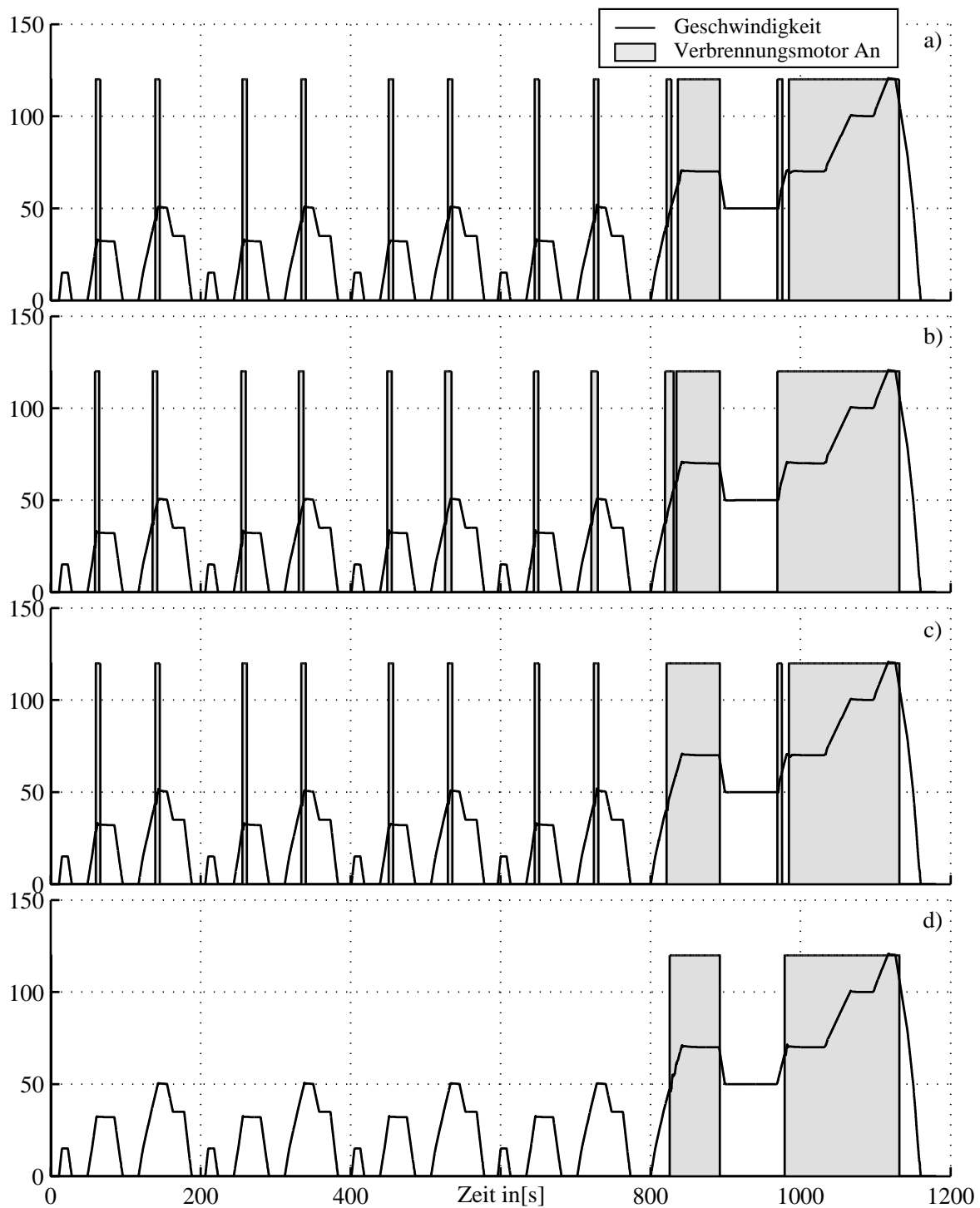


Abbildung 5.10: Vergleich des Fahrzeugverhaltens in einer NEFZ-Simulation bei verschiedenen Optimierungsansätzen. **Teil a)** Ausgangszustand **Teil b)** Ansatz 1: *Optimierung des Ausgangszustands* **Teil c)** Ansatz 2: *Verifikation des Funktionsentwurfs* **Teil d)** Ansatz 3: *gesamtheitliche Optimierung* und Ansatz 4: *vertiefte gesamtheitliche Optimierung*

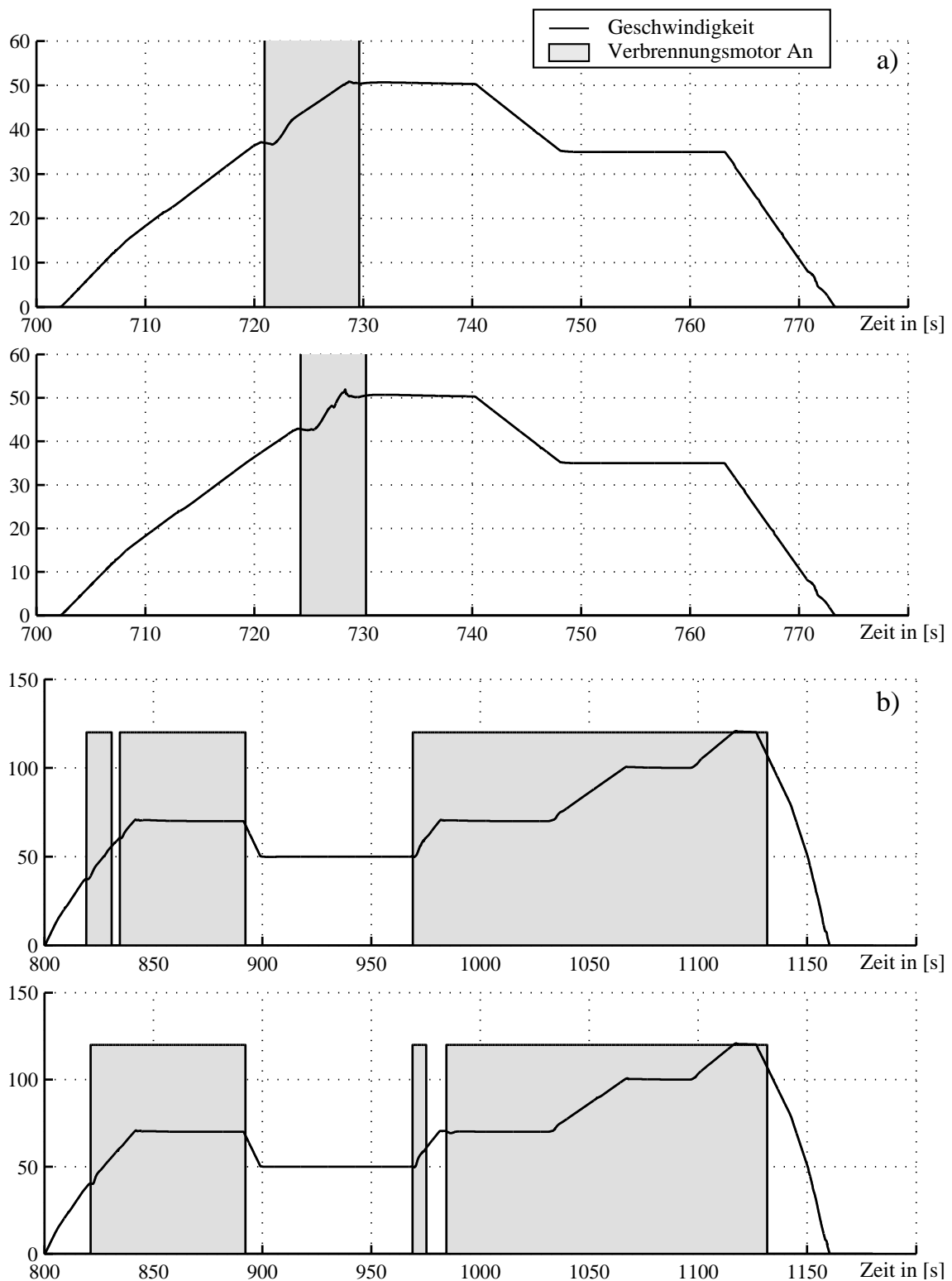


Abbildung 5.11: Unterschiede des Fahrzeugverhaltens für die Optimierungsansätze 1 und 2.

Teil a): Ausschnitt aus dem Städteteil des NEFZ. Oben sind die Simulationsergebnisse für den Ansatz 1: *Optimierung des Ausgangszustands*, unten für den Ansatz 2: *Verifikation des Funktionsentwurfs* dargestellt. **Teil b):** Darstellung der Simulationsergebnisse für den Überlandteil des NEFZ (analog zu Teil a))

6 Zusammenfassung und Ausblick

6.1 Zusammenfassung

Aus den Bestrebungen der Automobilhersteller Produkte anzubieten, die kundenorientiert und umweltfreundlich sind, resultiert ein bemerkenswerter Anstieg in der Komplexität und der Vielfalt der automobilen Antriebsstränge. Nicht nur das Zusammenspiel mehrerer Antriebsaggregate, sondern auch die stetige Weiterentwicklung der einzelnen Antriebskomponenten führen zu einem besseren Gesamtwirkungsgrad. Mit dieser Entwicklung ist die Aufnahme neuer Einflussgrößen, und der damit verfügbaren neuen Freiheitsgrade zur Ansteuerung der Antriebskomponenten, fest verbunden. Auf der Seite der Steuerungs- und Regelungsalgorithmik kann ein entsprechender Komplexitätsanstieg beobachtet werden. Mit den aufwendigen Methoden wird eine optimale Ansteuerung der Antriebskomponenten in allen Betriebsbereichen angestrebt.

Die Bestimmung der optimalen Einstellungen der Applikationsparameter der Steuerungs- und Regelungsfunktionen, durch eine Online-Optimierung an einem Prüfstand oder in einem Fahrzeug, ist kaum durchführbar. Eine simulationsgestützte Parameteroptimierung, die auf der Vermessung, Modellierung und Optimierung eines Prozesses aufgebaut ist, stellt einen effizienten Weg zur Lösung der Problemstellung dar. Gleichzeitig trägt eine geeignete Simulationsumgebung zur Effizienzsteigerung der Validierung der implementierten Steuerungs- und Regelungsalgorithmen bei.

Vor dem Hintergrund der dargestellten Zusammenhänge, wurde im Rahmen dieser Arbeit das Ziel einer gemeinsamen Optimierung der Hybrid-Betriebsstrategie und des Verhaltens des Verbrennungsmotors verfolgt. Die Übernahme von den im Steuergerät verwendeten Funktionsmodulen in die Simulationsumgebung für Fahrzeuglängsdynamik stellt eine effiziente Applikationsmöglichkeit der Originalparametrierung dar. Gleichzeitig ist es notwendig, das Verhalten des Verbrennungsmotors derart nachzubilden, dass das stationäre und das dynamische Verhalten, inklusive aller relevanten Einflussmöglichkeiten, wiedergegeben werden kann.

Ausgehend von der Betrachtung der Werkzeuge zur Optimierung der Hybridstrategie und der Funktionsentwicklung der Steuergerätesoftware sind die Rahmenbedingungen und Zielwerte für die Implementierung eines neuen Werkzeugs abgeleitet. Ein eindeutiger Übertragungsprozess der in Ascet¹ definierten Steuergerätefunktionen in die Simulink-Simulationsumgebung² mit der Absicht einer modellbasierten Optimierung der Applikationsparameter war das primäre Ziel der Werkzeugentwicklung. Durch die automatische Codegenerierung ist zu jedem in Ascet definiertem Funktionsmodul

¹Ascet ist ein Werkzeug der Firma ETAS GmbH zur Erstellung von *Embedded Software* http://www.etas.com/de/products/ascet_software_products.php

²Simulink ist ein Simulationswerkzeug der Firma The MathWorks, Inc. <http://www.mathworks.de/products/simulink/>

eine entsprechende Beschreibung in der Programmiersprache C erzeugt. Über die in der Simulink-Umgebung zur Verfügung stehende S-Funktion-Programmierschnittstelle kann der erzeugte C-Code auf eine automatisierte Art eingebunden werden. Die Untersuchung des C-Codes auf die Anzahl und Art der Ein- bzw. Ausgänge, der verwendeten Parameter und der globalen Variablen stellen nur einige wichtige Aspekte bei der Werkzeugrealisierung dar. Das entwickelte Werkzeug ist nicht nur in der Lage die Ascet-Steuergerätefunktionen in die Simulink-Umgebung zu übertragen, sondern es erfüllt auch weitere wichtige Eigenschaften, die beim Einsatz dieser Methode ausschlaggebend sind. Eine erhöhte Flexibilität bezüglich der Daten- und Funktionsstandänderungen, sowie die Parametrierbarkeit der Funktionsmodule direkt aus einer DCM-Datei¹ sind Verbesserungen die an dieser Stelle zu nennen sind. Tritt eine Änderung des Datenstands auf, so ist der Import einer neuen DCM-Beschreibungsdatei ausreichend. Die Weiterentwicklung eines Funktionsmoduls kann durch die Aktualisierung und Einbindung des C-Codes berücksichtigt werden.

Die Modellierung des Verbrennungsmotorverhaltens ist durch die Abbildung der *Steller-Sensor*-Strecken in der Simulationsumgebung realisiert. Die von den Funktionsmodulen benötigten Signale der Druck- und der Temperatursensoren sind in ihren stationären und dynamischen Eigenschaften modelliert. Die implementierten Steuergerätefunktionen sind hierdurch in der Lage auf ein nichtlineares, dynamisches Modell zuzugreifen, das im Gegensatz zu einem Kennfeldmodell über alle Freiheitsgrade, die ein moderner Dieselmotor bietet, verfügt.

Die Gewinnung der Messdaten für das Modell erfolgt aufgrund der gewählten Wiener-Modellstruktur in zwei Schritten. Durch die Anwendung der DoE-Vermessungsmethoden sind die stationären Systemeigenschaften erfasst. Anschließend wird zur Vermessung der Eigendynamik des Systems die Aufnahme der dynamischen Systemeigenschaften durchgeführt. Die gesamte Motorvermessung umfasste hierbei nicht nur die Gewinnung eines Trainingsdatensatzes, sondern auch eines Verifikationsdatensatzes.

Bei der Modellierung des stationären Systemverhaltens erfolgt der Einsatz der künstlichen neuronalen Netze. Die Auswahl der optimalen Neuronenanzahl erfolgt durch die Betrachtung des SSE für die Trainings- und die Verifikationsdaten. Falls notwendig, wird zur Sicherstellung der angestrebten Modellqualität, das Interpolationsverhalten durch die Erstellung von Gauß-Prozess-Modellen verbessert. Mit den Gauß-Prozess-Modellen werden hierbei zusätzliche Stützpunkte erzeugt und mit einer verminderten Priorität in die Modellierung eingebunden. Für die Modellierung des dynamischen Systemverhaltens werden lineare Übertragungsfunktionen in Form von *PT1*- oder *PT2*-Gliedern verwendet. Bei der Minimierung der Abweichung zwischen dem Modellausgang und den Messergebnissen wird zusätzlich zum SSE das 2σ -Intervall der relativen Fehlerverteilung betrachtet.

Die Implementierung der Steuergerätefunktionsmodule und der erstellten *Steller-Sensor*-Streckenmodelle in der Simulationsumgebung für Fahrzeuglängsdynamik führte zum Anstieg der Simulationszeit und einer Vergrößerung des Parameterraums. Ausgehend von der Betrachtung dieser Änderungen, sind neben dem Einsatz von bekannten Optimierungsverfahren auch neue Algorithmen entwickelt und implementiert.

¹Ein Dateiformat zur Speicherung von Applikationsparametern (Kennfelder, Kennlinien, Konstanten etc.). Die technische Dokumentation des Formats ist in http://www.etas.com/download-center-files/products_ASCET_Software_Products/TechNote_DCM_File_Formats.pdf zu finden.

Das aus der Regelungstechnik bekannte Verfahren der Gütevektroptimierung¹ trägt entscheidend zu einer systematischen Betrachtung und Optimierung der Zielgrößen bei. Das Ergebnis des Verfahrens ist durch das Optimum der Paretofront der einzelnen Entwurfsspezifikationen gekennzeichnet. Die steigenden Simulationszeiten benachteiligen Minimumsuchverfahren, die eine Vielzahl an Iterationen benötigen. Um die Verwendung einer Zufallsvariablen, die maßgeblich zur Steigerung der Iterationanzahl beiträgt, zu vermeiden und gleichzeitig eine Globalisierung der Suche im Parameterraum zu ermöglichen wird die entwickelte Methode *DelaunaySearch* eingesetzt. Im Gegensatz zu den bekannten Algorithmen, wie die Partikelschwarmoptimierung [Oli09, S. 41-58] oder die evolutionären Algorithmen [Oli09, S. 13-40], setzt die neu entwickelte Methode bei der Suche nach dem Minimum einer Kostenfunktion auf eine systematische Analyse der durchgeführten Simulationsergebnisse. Mit Hilfe der bei der Analyse gewonnenen Informationen werden Bereiche mit den bestmöglichen Voraussetzungen für ein Minimum identifiziert. Somit verzichtet das iterative Verfahren bei der Bestimmung des nächsten Iterationsschrittes auf die Verwendung einer Zufallsvariable. Als Ergebnis der Berechnungen steht ein gut gewählter Startwert für eine lokale Optimierung zur Verfügung.

Aufbauend auf der Simulation der Fahrzeuglängsdynamik, der Steuergerätefunktionen und der *Steller-Sensor*-Streckenmodelle in einer Simulationsumgebung wird die Hybrid-Betriebsstrategie gemeinsam mit der Steuerung des Verbrennungsmotors optimiert. Mit der Entwicklung und Implementierung einer neuen Funktion wird weiterhin die Verbindung zwischen der Betriebsstrategie und der Motorsteuerung erweitert. Die vorgestellten Werkzeuge ermöglichten hierbei nicht nur einen Test der neuen Funktionalitäten, sondern auch eine Abschätzung der Verbesserungspotentiale beim Verbrauch und Abgasemissionen. Insgesamt konnte eine effiziente Testumgebung für eine gemeinsame Optimierung der Betriebsstrategie und des Verbrennungsmotorverhaltens eines Hybridfahrzeugs realisiert werden. Die Simulation der Fahrzeuglängsdynamik und der Steuergerätefunktionen in einer Simulationsumgebung ergänzt die Werkzeugkette zur Applikation der Steuergeräteparameter.

6.2 Ausblick

Eine feste Einbindung des entwickelten Werkzeugs, zur Übertragung der Steuergerätefunktionen in die Simulationsumgebung, in den Optimierungsprozess der Applikationsbedatung stellt einen nächsten logischen Schritt dieser Arbeit dar. Weiterer Aufwand in den Bereichen Entwicklung, Bedienbarkeit und Support führt zu einer erhöhten Stabilität des Übertragungsprozesses und einer größeren Akzeptanz der Funktionsentwickler und Applikateure. Die investierten Kapazitäten führen gleichzeitig zur Einsparung des Prüfstandsbedarfs und der Steigerung der Applikationsqualität.

Weitere Entwicklungsschritte sind zur Ausschöpfung zusätzlicher Potentiale, der in der Matlab/Simulink-Umgebung verfügbaren Simulationstechniken, denkbar. Die zeitaufwendigen Simulationen können hierbei von einer lokalen Rechenleistung auf die Kapazitäten eines Rechenzentrums verlagert werden. Die Möglichkeiten einer Parallelisierung der Berechnungen spielt an dieser Stelle die zentrale Rolle. Mit diesen Voraussetzungen ist die Simulationen mit verschiedenen Parametereinstellungen gleichzeitig möglich. Auch das entwickelte Simulationswerkzeug *DelaunaySearch*, lässt

¹Systematische Auslegung von Reglern durch Optimierung eines vektoriiellen Gütekriteriums [KS79]

sich auf den Einsatz von Parallelsimulationen optimieren. Die im Schritt 7 des Verfahrens definierte Berechnung des nächsten Simulationsschritts kann ohne zusätzlichen Aufwand auf eine Ausgabe mehrerer zu simulierender Punkte umgestellt werden.

Weiterhin lassen sich auch Methoden der Partikelschwarmoptimierung und der evolutionären Algorithmen unter der Anwendung von Parallelsimulationen optimal einsetzen. Hierbei wird jedes Partikel in einem eigenen Prozess simuliert werden.

Auf der Seite der Modellierung des Antriebsstrangs ergeben sich weitere Optimierungspotentiale, die zur Fortführung der vorliegenden Arbeit beitragen. Die Anwendung der vorgestellten Vermessungs- und Modellierungsmethoden bewirkt eine Verbesserung der Modellqualität weiterer Komponenten des Antriebsstrangs. Im Fokus der Betrachtungen stehen vor allem die Traktionsbatterie, die E-Maschine und das Getriebe. Nicht nur die stationären und die dynamischen Systemeigenschaften stehen hierbei im Vordergrund. Vor allem die Temperaturabhängigkeit bzw. das Aufwärmverhalten der Komponenten sind für die Berechnung der Zielgrößen entscheidend.

Für die Traktionsbatterie sind zusätzlich zum Einfluss der Temperatur auf den Wirkungsgrad, der Einfluss auf die Kapazität, die Untersuchung bzw. Einstellung der optimalen Aufwärmstrategie und der maximalen Strombelastung interessant.

Bei der E-Maschine sind sowohl die Betrachtungen der Verlustleistung, als auch die maximalen stationären bzw. dynamischen Drehmomentbelastungen relevant.

Weiterhin ergibt sich beim Verbrennungsmotor mit fallender Temperatur nicht nur eine Verschlechterung des Wirkungsgrads, sondern auch ein überproportionaler Anstieg der Abgasemissionen.

Die Abarbeitung der aufgezeigten Handlungsfelder verleiht der modellgestützten Applikation der Steuergeräteparameter mehr Aussagekraft. Eine gemeinsame Optimierung der Hybrid-Betriebsstrategie und des Verhaltens eines Dieselmotors wird in diesem Fall in weiteren, wichtigen Betriebsbereichen sichergestellt.

A Anhang

A.1 Werkzeug zur Simulation von Steuergerätefunktionen, implementierte Funktionsumfänge

Im Kapitel 2 ist ein Werkzeug zur Übertragung von Steuergerätefunktionen aus der Ascet-Funktionsentwicklungsumgebung in die Simulink-Simulationsumgebung beschrieben. Dieses Kapitel liefert einen detaillierten Einblick in die implementierten Strukturen des erstellten Werkzeugs. Zunächst werden die wesentlichen Eigenschaften des aus Ascet erzeugten C-Codes vorgestellt. Anschließend erfolgt die Beleuchtung des S-Function-Konzepts in Simulink. Aufbauend auf diesen Darstellungen kann die Implementierung der verschiedenen Teile des entwickelten Werkzeuges erläutert werden.

Erläuterungen zum Aufbau des aus Ascet erzeugten C-Codes

Bei der Implementierung der Funktionsmodule für ein Steuergerät in der Ascet-Umgebung ist der Zugriff auf viele verschiedene Programmier Techniken möglich. Einige, dem Kontext dieser Arbeit entsprechende, Eigenschaften sind nachfolgend zusammengefasst. Für weiterführende Informationen sei an dieser Stelle auf die von der Firma ETAS zur Verfügung gestellten Informationen verwiesen [ETA14a].

Message Konzept Jedes implementierte Funktionsmodul kommuniziert innerhalb seiner Projektumgebung durch sogenannte Ein- bzw. Ausgangsmessages. Bei der Abarbeitung wird von jeder empfangener Eingangsmessage zunächst eine Kopie angelegt. Somit ergeben sich bei einer Änderung des Eingangssignals zwischen den Ausführungszeiten des Moduls, keine Auswirkungen auf die Prozessberechnungen der implementierten Funktion.

Berechnungsreihenfolge Innerhalb eines Funktionsmoduls sind Prozesse definiert, die mit einer bestimmten Taktrate aufgerufen werden und somit die Berechnung der implementierten Funktionalitäten ermöglichen. Die Berechnungsreihenfolge innerhalb eines Prozesses kann vom Anwender frei bestimmt werden.

Gruppenkennlinien und Gruppenkennfelder Diese Programmier Technik erlaubt die Definition von Achsen, die gleichzeitig von mehreren Kennlinien bzw. Kennfelder benutzt werden.

Globale Parameter Verschiedene Parameter können durch ihre Import- bzw. Export-Eigenschaften in einem Modul definiert werden und später in anderen Modulen verwendet werden.

Globale Variablen Die Verwendung von importierten bzw. exportierten Variablen entspricht der Programmiertechnik der globalen Parameter, mit dem Unterschied, dass diese Variablen bei jedem Ausführungsschritt des Moduls geändert werden können.

Bei dem Übertragungsprozess eines Ascet-Funktionsmoduls in die Simulink-Umgebung ist die Unterstützung der vielen Programmiertechniken sicherzustellen. Die Abbildung A.1 gibt einen allgemeinen Überblick über den Aufbau des aus Ascet erzeugten C-Codes eines Funktionsmoduls.

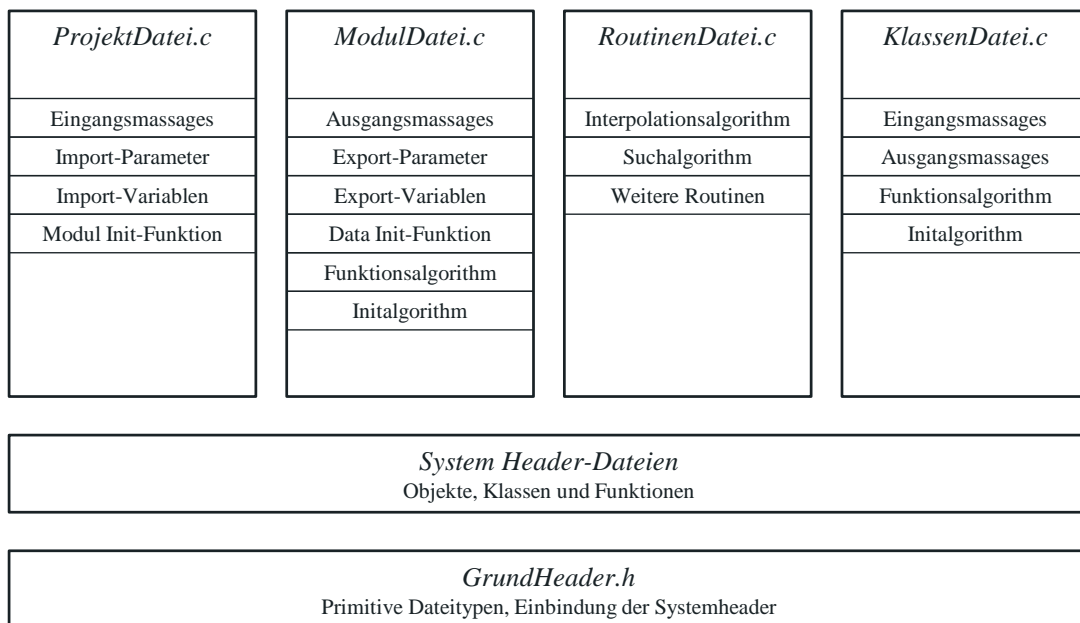


Abbildung A.1: Überblick über den aus Ascet erzeugten C-Code

In der *GrundheaderDatei.h* sind die primitiven Dateitypen definiert, grundlegende Funktionen deklariert und die *System-Header* Dateien eingebunden. Der eigentliche C-Code des Funktionsmoduls ist im Wesentlichen auf den *System-Header* Dateien aufgebaut. Hier sind die im C-Code verwendeten Dateitypen, Objekte, Klassen und Funktionen definiert bzw. deklariert.

Das implementierte Funktionsmodul unterteilt sich nach der Codegenerierung auf vier verschiedene Dateien, *ProjektDatei.c*, *ModulDatei.c*, *RoutinenDatei.c* und *KlassenDatei.c*.

In der *ProjektDatei.c* sind die Eingangsmessages und die vom Funktionsmodul importierten Parameter und globale Variablen definiert. Zusätzlich ist in dieser Datei die Modul-Init-Funktion implementiert, die für die Definition aller notwendigen Objekte und Daten im Speicher zuständig ist.

Die *ModulDatei.c* beinhaltet die Ausgangsmessages und die vom Funktionsmodul exportierten Parameter, zusammen mit den globalen Variablen. Eine Funktion für die Initialisierung der lokal verwendeten Parameter mit Standardwerten ist ein weiterer Bestandteil dieser Datei. Alle im Funktionsmodul verwendeten Initialisierungs- und Berechnungsprozesse sind an dieser Stelle definiert.

Innerhalb der Routinen-Dateien sind die für die Berechnung notwendigen Berechnungsmechanismen definiert und stehen den Funktionsmodulen zur Verfügung. Hierzu zählen unter anderem

Interpolations- und Suchroutinen, etc. In den Klassen-Dateien sind die vom Anwender erstellten Bibliotheksmodule zusammengefasst.

Die Ausführung des Funktionsmoduls fängt erwartungsgemäß mit dem Aufruf der Modul-Init Funktion aus der der *ProjektDatei.c* an. Diese definiert den für die Ausführung des Funktionsmoduls notwendigen Speicherbereich. Vor der Berechnung der im Modul implementierten Funktionalitäten ist der Aufruf des Init-Prozesses notwendig. Hierbei werden die vom Anwender definierten Initialisierungsaufgaben berechnet. Anschließend folgt die an ein festes Zeitraster gekoppelte Ausführung der Berechnungsprozesse des Funktionsmoduls. Die vom Anwender definierten Prozesse lesen die in der *ProjektDatei.c* deklarierte Eingangssignale ein. Ein Zugriff auf die eingebundenen Routinen und Bibliotheksklassen ist bei den Berechnungen möglich. Die Ausgangsgrößen des Funktionsmoduls werden nach den Berechnungen direkt in der *ModulDatei.c* bedatet.

Simulink S-Function-Konzept

Die Simulink-Simulationsumgebung bietet die Möglichkeit der Implementierung von den in der Programmiersprache C/C++ definierten S-Funktionen. Die im Rahmen der vorliegenden Arbeit relevanten Schnittstellen zwischen dem *Simulink-Engine* und dem zu implementierenden C-Code sind nachfolgend kurz beschrieben. Detaillierte Informationen zum Thema C-S-Funktionen sind unter anderem auf der Webseite der Firma The MathWorks einzusehen [The14a].

mdlInitializeSizes Spezifikation der Anzahl, der Dimension, des Datentyps und weiterer Eigenschaften der Ein- und Ausgänge der S-Funktion. Weiterhin wird hier die Anzahl der Parameter und der Abtastzeiten definiert. Dieses ist die erste Methode die vom Simulink bei der Berechnung einer S-Function aufgerufen wird.

mdlInitializeConditions Eine optionale Methode, die bei der Initialisierung der S-Function aufgerufen wird. Vorgesehen ist an dieser Stelle die Initialisierung der Zustände der Funktion.

mdlInitializeSampleTimes Die Definition der Eigenschaften der Abtastzeit stehen innerhalb dieser Methode im Vordergrund

mdlOutputs Diese Methode berechnet die Ausgangssignale einer S-Funktion und wird bei jedem Iterationsschritt aufgerufen.

mdlTerminate Hier werden Methoden zusammengefasst, die für die Speicherfreigebe beim Beenden der Simulation notwendig sind.

Anpassung der Headerdateien

Für die Realisierung der Simulationsmöglichkeit der Ascet-Funktionsmodule in der Simulink-Umgebung ist zunächst eine Anpassung der Headerdateien notwendig. Die durchgeführten Anpassungen ermöglichen die Einbindung eines ungeänderten C-Codes in das S-Function-Konzept. Die Änderungen in den Headerdateien betreffen hauptsächlich folgende Themengebiete:

- Neudefinition der Dateitypen

- Neudefinition der abstrakten Objekt- und Klassenstrukturen
- Deklaration und Definition von Grundfunktionen
- Definition von Konstanten
- Definition von Funktionsmakros

Die angepassten Headerdateien sind zusammen mit den Simulink-Headerdateien in das S-Funktion-Konstrukt eingebunden.

Implementierung der Ein- und Ausgänge

Für die Definition der einem Funktionsmodul entsprechenden S-Funktion werden weiterhin Informationen über die Ein- und Ausgänge benötigt. Durch gezielte Untersuchung der *ProjektDatei.c* werden die Informationen über die Anzahl, die Namen, die Dateitypen etc. der Eingänge bestimmt. Ähnlich lassen sich die gleichen Informationen über die Ausgänge des Moduls aus der *ModulDatei.c* einholen.

Diese Informationen werden in den Methoden des S-Funktion-Konstrukts verwendet. Die Methode *mdlInitializeSizes* definiert hiermit die Anzahl und die Eigenschaften der Ein- und Ausgänge. Weiterhin werden die gewonnenen Informationen bei dem Einlesen der Eingangssignale und der Berechnung der Ausgangssignale in der die Methode *mdlOutput* benötigt.

Implementierung der Parametrierung

Die Parametrierung der Module erfolgt durch die Auswertung einer DCM-Datei, die alle für das vorliegende Funktionsmodul notwendigen Parameter beinhaltet. Grundsätzlich werden bei der Initialisierung die einzelnen Parameter mit Standardwerten bedatet, sodass das Fehlen eines Parameters in der DCM-Datei keine fatalen Auswirkungen auf die Funktionsweise des Moduls hat. Die eingelesenen Parameterinformationen stehen im Matlab-*Workspace* in Form einer Struktur zur Verfügung. Das Einlesen der Parametrierung erfolgt über die Parametrierungsschnittstelle des S-Function-Blocks. Die Parameterstruktur wird innerhalb der Methode *mdlInitializeConditions* ausgewertet. Die einzelnen Parameter werden anhand ihrer Namen identifiziert und bedatet. Weiterhin werden innerhalb dieser Methode, die vom Anwender implementierten und in der *ModulDatei.c* definierten, Initialisierungsprozesse aufgerufen.

Implementierung des Berechnungsprozesses

Der in der *ModulDatei.c* definierte Berechnungsprozess des Funktionsmoduls wird innerhalb der *mdlOutput* Methode aufgerufen. Das Einlesen der Eingangssignale ist vor dem Prozessaufruf und das Setzen der Ausgangssignale nach dem Prozessaufruf definiert.

Implementierung der globalen Variablen

Die globalen Variablen werden von mehreren Funktionsmodulen gleichzeitig verwendet. Diese können aus einer passenden A2L-Datei eingelesen und im Matlab-*Workspace* in Form einer Struktur zur Verfügung gestellt werden. Während der iterativen Berechnung der *mdlOutput*-Methode werden diese entsprechend vor der Ausführung des Berechnungsprozesses eingelesen und nach Ausführung gesetzt.

Zusammenführung

Eine weitere Herausforderung stellt die Zusammenführung mehrerer, in die Simulink-Umgebung übertragener, Funktionsmodule in einem Simulationsmodell dar. Das in Ascet angewendete Messagekonzept zur intermodularen Kommunikation ist in der Simulink-Umgebung durch die Anwendung der *From*- und *Goto*-Blöcke realisiert. Die Entstehung von algebraischen Schleifen wird durch eine gezielte Ein-/Ausgangsuntersuchung der einzelnen Funktionsmodule vermieden. Die gleiche Untersuchung liefert auch die Informationen über die gesamten Systemeingänge und Systemausgänge.

A.2 Relativer Fehler

Der relative Fehler E_{Rel} zwischen dem gemessenen Prozessausgang \mathbf{y} und dem Modellausgang $\tilde{\mathbf{y}}$ kann wie folgt bestimmt werden (s. auch [LK11, S. 15]):

$$E_{Rel} = \frac{y_i - \tilde{y}_i}{y_i} \quad (\text{A.1})$$

mit

$$i = 1 \dots M$$

$M = \text{Anzahl der Messpunkte}$

$$\mathbf{y} = [y_1 \quad y_2 \quad \dots \quad y_i \quad \dots \quad y_M]^T$$

$$\tilde{\mathbf{y}} = [\tilde{y}_1 \quad \tilde{y}_2 \quad \dots \quad \tilde{y}_i \quad \dots \quad \tilde{y}_M]^T$$

Entsprechend gilt für den absoluten relativen Fehler:

$$E_{RelAbs} = \frac{|y_i - \tilde{y}_i|}{|y_i|} \quad (\text{A.2})$$

$$(\text{A.3})$$

Eine Relativierung auf den Maximalwert führt zu:

$$E_{RelAbsMax} = \frac{|y_i - \tilde{y}_i|}{\max_{1 \leq i \leq M} \{|y_i|\}} \quad (\text{A.4})$$

A.3 Normierte Summe der Quadratabweichungen

Die normierte Summe der Quadratabweichungen E_{SSE} lässt sich nach folgender Berechnungsvorschrift ermitteln [Win02, S. 23 f.], [Sch10a, S. 127]:

$$E_{SSE} = \frac{1}{\max_{1 \leq i \leq M} \{|y_i|\}} \sqrt{\frac{1}{M} \sum_{i=1}^M (y_i - \tilde{y}_i)^2} \quad (\text{A.5})$$

A.4 Mittlere Summe der Quadratabweichungen

Die mittlere Summe der Quadratabweichungen E_{RMSE} lässt sich nach folgender Berechnungsvorschrift ermitteln:

$$E_{RMSE} = \frac{1}{\bar{y}} \sqrt{\frac{1}{M} \sum_{i=1}^M (y_i - \tilde{y}_i)^2} \quad (\text{A.6})$$

mit

$$\bar{y} = \frac{1}{M} \sum_{i=1}^M y_i \quad (\text{A.7})$$

A.5 Bestimmtheitsmaß

Das Bestimmtheitsmaß R^2 stellt eine Angabe für den Anteil der Varianz von \mathbf{y} dar, der durch das Modell $\tilde{\mathbf{y}}$ erklärt wird. Die Definition erstreckt sich auf den Bereich zwischen 0 und 1, wobei 0 für *keine Übereinstimmung* und 1 für *volle Übereinstimmung* steht (s. auch [Koh05, S. 147]).

$$R^2 = \frac{\sum_{i=1}^M (\tilde{y}_i - \bar{y})^2}{\sum_{i=1}^M (y_i - \bar{y})^2} \quad (\text{A.8})$$

A.6 Standardabweichung

Die Standardabweichung σ kann nach folgender Vorschrift berechnet werden [BSMM07, S. 771 ff.]:

$$\sigma = \sqrt{\frac{1}{M-1} \sum_{i=1}^M (y_i - \bar{y})^2} \quad (\text{A.9})$$

A.7 Gaußverteilung

Die Gauß- bzw. Normalverteilung ist wie folgt definiert [BSMM07, S. 758 f.], [RW06]:

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \quad (\text{A.10})$$

oder in einer Kurzform

$$x \sim \mathcal{N}(\mu, \sigma_n^2)$$

mit dem Erwartungswert

$$E(x) = \int_{-\infty}^{\infty} x p(x) dx = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} x \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) dx = \mu \quad (\text{A.11})$$

und der Varianz

$$V(x) = E[(x-\mu)^2] = \int_{-\infty}^{\infty} (x-\mu)^2 p(x) dx = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} (x-\mu)^2 \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) dx = \sigma^2 \quad (\text{A.12})$$

Analog hierzu ergibt sich für die Betrachtung der N -dimensionalen Gauß-Verteilung:

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{N/2} |\Sigma|^{1/2}} \exp\left(-\frac{1}{2} (\mathbf{x}-\mu)^T \Sigma^{-1} (\mathbf{x}-\mu)\right) \quad (\text{A.13})$$

Kurzform

$$\mathbf{x} \sim \mathcal{N}(\mu, \Sigma)$$

mit

$$\begin{aligned} \mu &\in \mathbb{R}^{1 \times N} \\ \Sigma &\in \mathbb{R}^{N \times N} \end{aligned}$$

Eine Verallgemeinerung zum Gauß-Prozess ist dementsprechend wie folgt definiert:

$$p(\mathbf{x}) \sim GP(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')) \quad (\text{A.14})$$

mit

$$\begin{aligned} m(\mathbf{x}) &= E[f(\mathbf{x})]: \mathbb{R}^N \rightarrow \mathbb{R} \\ k(\mathbf{x}, \mathbf{x}') &= E[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))]: \mathbb{R}^N \rightarrow \mathbb{R} \end{aligned}$$

A.8 Delaunay-Triangulation

Bei einer gegebenen Punktemenge $P \in \mathbb{R}^N$ ist die Delaunay-Triangulation definiert als die Triangulation $DT(P)$ mit folgender Eigenschaft [JEJ04, S. 520], [MOMM04, S. 196 ff.]:

Innerhalb der N -dimensionalen Sphäre, die den Umkreis jedes Simplexes aus $DT(P)$ umfasst, existieren keine weiteren Punkte aus P

Die Delaunay-Triangulation stellt die winkeloptimale Triangulation der Punktmenge P dar. Hierbei werden die Winkel jedes Simplexes aus $DT(P)$ maximiert.

Für den 2 – D -Fall kann die Delaunay-Triangulation visualisiert werden (s. Abbildung A.2).

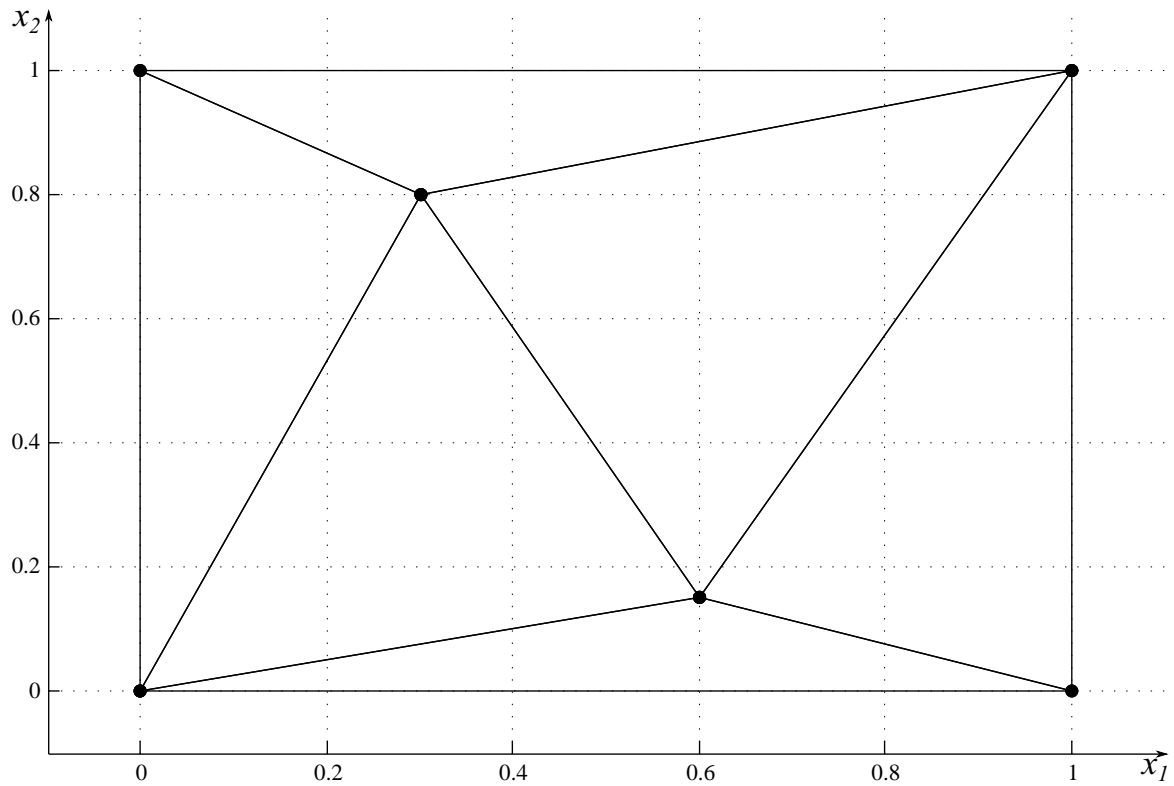


Abbildung A.2: Darstellung der Delaunay-Triangulation im 2 – D -Fall

Literaturverzeichnis

- [Ack93] ACKERMANN, Jürgen: *Robuste Regelung*. Springer, 1993
- [ALWT98] AYEYB, M. ; LICHTENTHÄLER, D. ; WINSEL, T. ; THEUERKAUF, H. J.: SI Engine Modeling Using Neural Networks. In: *SAE Technical Paper 980790* (1998)
- [Aye98] AYEYB, Mohamed: *Zur modellbasierten Regelung technischer Prozesse mit dynamischen neuronalen Netzen*, Universität Kassel, Diss., 1998
- [BDH96] BARBER, Bradford C. ; DOBKIN, David P. ; HUHDANPAA, Hannu: The quickhull algorithm for convex hulls. In: *ACM Transactions on Mathematical Software (TOMS)* 22 (1996), Nr. 4, S. 469–483
- [BGN00] BYRD, R.H. ; GILBERT, J.C. ; NOCEDAL, J: A trust region method based on interior point techniques for nonlinear programming. In: *Mathematical Programming* (2000), S. 149–185
- [Bis06] BISHOP, Christopher M.: *Pattern Recognition and Machine Learning*. Springer, 2006
- [Boe13] BOEHME, Dominik: *Methoden zur effizienten Basisapplikation für Luftpfad- und Abgastemperaturmodelle von Ottomotoren*, TU Darmstadt, Diss., 2013
- [BSMM07] BRONSHTEIN, Ilja N. ; SEMENDYAYEV, Konstantin A. ; MUSIOL, Gerhard ; MUEHLIG, Heiner: *Handbook of Mathematics*. Springer, 2007
- [CGA14] CGAL: *Computational Geometry Algorithms Library*. <https://www.cgal.org/>. Version: 24.04.2014
- [DP04] DITTMAR, Rainer ; PFEIFFER, Bernd-Markus: *Modellbasierte prädiktive Regelung: eine Einführung für Ingenieure*. Oldenbourg, 2004
- [ETA14a] ETAS GMBH: *ASCET Produktfamilie*. http://www.etas.com/de/products/ascet_software_products.php. Version: 10.04.2014
- [ETA14b] ETAS GMBH: *ASCMO Software-Produkte*. <http://www.etas.com/de/products/ascmo.php>. Version: 10.04.2014
- [FFSZ13] FELSCH, Christian ; FUGEL, Markus ; SCHOLZ, Norbert ; ZILLMER, Michael: Fuel efficiency and driving performance comparison of different hybrid powertrain topologies. In: *11. Symposium: Hybrid- und Elektrofahrzeuge*. Braunschweig, 2013, S. 326–334
- [Hal60] HALTON, J.H.: On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. In: *Nummerische Mathematik* 2 (1960), Nr. 1, S. 84–90

- [HSW89] HORNİK, Kurt ; STINCHCOMBE, Maxwell ; WHITE, Halbert: Multilayer feedforward networks are universal approximators. In: *Neural networks 2* (1989), Nr. 5, S. 359–366
- [HTA⁺02] HARALD, Stuhler ; THOMAS, Kruse ; AXEL, Stuber ; KURT, Gschweidl ; WALTER, Piock ; HORST, Pfluegl ; PETER, Lick: Automated Model-Based GDI Engine Calibration Adaptive Online DoE Approach. In: *SAE Technical Paper 2002-01-0708* (2002)
- [Ise10] ISERMANN, Rolf: *Elektronisches Management motorischer Fahrzeugantriebe*. Springer, 2010
- [JEJ04] JACOB E., Goodman ; JOSEPH, O'Rourke: *Handbook of Discrete and Computational Geometry*. Chapman & Hall/CRC, 2004
- [JPM03] JOHANN, Bachler ; PETER, Mathis ; MARTIN, Rzehorska: Calibration Methodologies for On-line Optimisation of Diesel Engines with Regard to Emissions, NVH and Performance. In: *SAE Technical Paper 2003-26-0008* (2003), S. 197–204
- [KE95] KENNEDY, James ; EBERHART, Russell: Particle Swarm Optimization. In: *IEEE* (1995), S. 1942–1948
- [Ken10] *Kapitel Particle swarm optimization*. In: KENNEDY, James: *Encyclopedia of Machine Learning*. Springer, 2010
- [KKG⁺13] KLAR, H. ; KLAGES, B. ; GUNDEL, D. ; KRUSE, T. ; HUBER, T. ; ULMER, H.: Neue Verfahren zur effizienten modellbasierten Motorapplikation. In: *5. Internationales Symposium für Entwicklungsmethodik*. Wiesbaden, 2013, S. 158–167
- [Kla08] KLAR, Henning: *Reduktion der dieselmotorischen Abgasemissionen in den transienten Betriebsphasen mittels modellbasierter Optimierung und Echtzeitsimulation*, Universität Kassel, Diss., 2008
- [Kle05] KLEIN, Rolf: *Algorithmische Geometrie*. Springer, 2005
- [Koh05] KOHN, Wolfgang: *Statistik, Datenanalyse und Wahrscheinlichkeitsrechnung*. Springer, 2005
- [Kra11] KRACKER, Hilke: *Modellierung und Kalibrierung von Computermodellen mit Anwendung auf einen Umformprozess*, TU Dortmund, Diss., 2011
- [KS79] KREISSELMEIER, Gerhard ; STEINHAUSER, Reinhold: Systematische Auslegung von Reglern durch Optimierung eines vektoriiellen Gütekriteriums. In: *Regelungstechnik* (1979), S. 76–79
- [KS83] KREISSELMEIER, Gerhard ; STEINHAUSER, Reinhold: Application of vector performance optimization to a robust control loop design for a fighter aircraft. In: *Control* 37 (1983), Nr. 2, S. 251–284
- [Kus06] KUSS, Malte: *Gaussian Process Models for Robust Regression, Classification, and Reinforcement Learning*, TU Darmstadt, Diss., 2006
- [LK11] LEON, Fernando ; KIENCKE, Uwe: *Messtechnik, Systemtheorie für Ingenieure und Informatiker*. Springer, 2011

- [LRWW98] LAGARIAS, J.C. ; REEDS, J. A. ; WRIGHT, M. H. ; WRIGHT, P. E.: Convergence Properties of the Nelder-Mead Simplex Method in Low Dimensions. In: *SIAM Journal of Optimization* (1998), S. 112–147
- [Mar63] MARQUARDT, Donald W.: An Algorithm for Least-Squares Estimation of Nonlinear Parameters. In: *Journal of the Society for Industrial and Applied Mathematics* 11 (1963), Nr. 2, S. 431–441
- [MOMM04] MARK, de B. ; OTFRIED, Cheong ; MARC, van K. ; MARK, Overmars: *Computational Geometry: Algorithms and Applications*. Springer, 2004
- [Mor77] MORE, Jorge J.: The Levenberg-Marquardt algorithm: implementation and theory. In: *Conference on Numerical Analysis University of Dundee Scotland* (1977)
- [MPW⁺13] MÜNNING, Daniel ; PIEWEK, Jan ; WAGNER, Alexander ; FELSCH, Christian ; ZILLMER, Michael ; GÜHMANN, Clemens ; ELTS, Peter: Application of a Hybrid Optimization Method in the Diesel Hybrid Drive Train Development Process. In: *7. Tagung Design of Experiments (DoE) in der Motorenentwicklung*. Berlin, 2013, S. 269–288
- [NKJ⁺12] NEUSSER, Heinz-Jakob ; KAHRSTEDT, Jörn ; JELDEN, Hanno ; ENGLER, Hermann-Josef ; DORRENKAMP, Richard ; JAUNS-SEYFRIED, Stefanie ; KRAUSE, Andreas: Die neue modulare TDI-Generation von Volkswagen. In: *33. Internationales Wiener Motorensymposium 2012*. Wien, 2012, S. 85–110
- [Nö10] NÖTHEN, Christian: *Strategien zur Gassystemregelung von Pkw-Dieselmotoren*, Universität Magdeburg, Diss., 2010
- [Oli09] OLIVER, Kramer: *Computational Intelligence*. Springer, 2009
- [PST⁺08] PISCHINGER, S. ; SCHERNUS, C. ; THEUERKAUF, H. ; WINSEL, T. ; AYEB, M. ; SCHMIDGAL, J. ; WILHELM, C.: Erforschung eines Motormodells zur Applikationshilfe am Beispiel des Kaltstart- und Warmlaufverhaltens (Phase II). In: *Abschlussbericht des Forschungsvorhaben VFI 2007*, 2008
- [Qhu14] QHULL: *Qhull*. <http://www.qhull.org/>. Version: 24.04.2014
- [RHW86] RUMELHART, D.E. ; HINTON, G.E. ; WILLIAMS, R.J.: Learning representations by back-propagating errors. In: *Nature* 323 (1986), S. 533–536
- [RSK⁺13] RANGO, Job ; SCHNORBUS, Thorsten ; KWEE, Henry ; BECK, Ralf ; KINOO, Bert ; ARTHOZOUL, Simon ; ZHANG, Miao: Comparison of Different Approaches for Global Modeling of Combustion Engines. In: *7. Tagung Design of Experiments (DoE) in der Motorenentwicklung*. Berlin, 2013, S. 70–91
- [RW06] RASMUSSEN, C. E. ; WILLIAMS, C. K. I.: *Gaussian Processes for Machine Learning*. MIT Press, 2006
- [RW14] RASMUSSEN, C. E. ; WILLIAMS, C. K. I.: *Gaussian Processes for Machine Learning*. <http://www.gaussianprocess.org/gpml/>. Version: 08.04.2014

- [SBH10] SIEBERTZ, Karl ; BEBBER, David van ; HOCHKIRCHEN, Thomas: *Statistische Versuchsplanung*. Springer, 2010
- [Sch10a] SCHMIDGAL, Jurj: *Verfahren zur echtzeitfähigen Simulation dynamischer Emissionsverläufe ausgewählter Schadstoffe von Ottomotoren*, Universität Kassel, Diss., 2010
- [Sch10b] SCHRÖDER, Dierk: *Intelligente Verfahren: Identifikation und Regelung nichtlinearer Systeme*. Springer, 2010
- [Sob67] SOBOL, I.M.: On the distribution of points in a cube and the approximate evaluation of integrals. In: *Computational Mathematics and Mathematical Physics* 7 (1967), Nr. 4, S. 784–802
- [The14a] THE MATHWORKS, INC: *Create C/C++ S-Functions*. <http://www.mathworks.de/de/help/simulink/create-cc-s-functions.html>. Version: 04.03.2014
- [The14b] THE MATHWORKS, INC: *MEX-File Creation API*. <http://www.mathworks.de/de/help/matlab/write-cc-mex-files.html>. Version: 24.04.2014
- [Von09] VON GRUNDHERR ZU ALTENTHAN UND WEIYHERHAUS, Johannes: *Ableitung einer heuristischen Betriebsstrategie für ein Hybridfahrzeug aus einer Online-Optimierung*, TU München, Diss., 2009
- [WBVK04] WARD, M. ; BRACE, C. ; VAUGHAN, N. ; KENNEDY, G.: Automated Limit Space Characterisation & Mapping. In: *SAE Technical Paper 2004-01-1616* (2004)
- [Wei07] WEICKER, K.: *Evolutionäre Algorithmen*. Springer, 2007
- [Win02] WINSEL, Thomas: *Stabile neuronale Prozessmodelle: automatisierte Generierung echtzeitfähiger Modelle zur Nachbildung des dynamischen Verhaltens von Verbrennungsmotoren*. VDI-Verlag, 2002