



**Fachhochschule
Bielefeld**

University of Applied Sciences
Fachbereich
Elektrotechnik und
Informationstechnik

DIPLOMARBEIT

Name: Sebastian Wildfeuer
Matr.-Nr.: 204557

Thema der Arbeit: **Erweiterung eines Verfahrens zur auto-
matisierten Parameteridentifikation eines
Fahrzeugmodells**

Erstprüfer: Prof. Dr.-Ing. Ulrich Kramer
Zweitprüfer: Prof. Dr.-Ing. Stefan Dormeier

Ausgegeben am: _____
Abgegeben am: _____

Erklärung entsprechend der ADPO vom 25.06.1982 §26 Abs. 1

Wildfeuer
Name

Sebastian
Vorname

204557
Matr.-Nr.

Ich versichere, dass ich die Diplomarbeit selbstständig verfasst und keine, als die von mir angegebenen Hilfsmittel benutzt und bei Zitaten die Quelle kenntlich gemacht habe.

Mir ist bekannt, dass ich meine Diplomarbeit nach Ablauf der Aufbewahrungsfrist von 5 Jahren zurückbekommen kann.

Datum

Unterschrift

Abkürzungsverzeichnis

Γ	Systemparameter
μ	Erwartungswert
σ	Standardabweichung
$E[X]$	Erwartungswert von X
$\text{Var}(X)$	Varianz von X
$\text{Cov}(X, Y)$	Kovarianz von X und Y
δ_{ij}	Kronecker-Delta
$L(\Gamma \underline{z})$	Maximum Likelihood-Funktion
$\ell(\Gamma \underline{z})$	log-Likelihood-Funktion
∇	Nabla-Operator
D^2	Hesse-Matrix
Φ	Menge der Systemparameter
$A \setminus B$	Differenz zweier Mengen
$B \cup B$	Vereinigung zweier Mengen
$a \in A$	x ist in A
$\forall x$	Für alle x
\mathbb{R}	Menge der reellen Zahlen
\mathbb{N}	Menge der natürlichen Zahlen
\mathbb{N}^*	Menge der natürlichen Zahlen ohne die 0 (Null)
\underline{x}	x ist ein Vektor oder eine Matrix

Danksagung

Ich möchte mich ganz herzlich bedanken, bei Herrn Martin Fischer und Herrn Prof. Kramer für die Betreuung dieser Arbeit, welche mir fachlich und inhaltlich stets mit Rat und Hilfe zur Seite standen. Ich danke dem Institut für Verkehrssystemtechnik des DLR in Braunschweig welches mir die Möglichkeit geboten hat diese Arbeit im Institut zu verfassen. Weiterhin möchte ich meiner Mutter und Katrin Heinrich für das aufmerksame Lesen und Auffinden von Fehlern in dieser Arbeit danken.

Ganz besonders zu erwähnen sind meine Mutter und meinen Großeltern, dank deren Unterstützung mir dieses Studium überhaupt erst ermöglicht wurde. Vielen Dank!

Bielefeld, den 09.09.08

Sebastian Wildfeuer

Inhaltsverzeichnis

1	Einleitung und Motivation	10
1.1	Die Ausgangssituation und Problemstellung	10
1.2	Die Lösungsidee	12
2	Verwendete Software	14
2.1	CarSim	14
2.2	FitLab	16
2.3	Matlab und Simulink	17
3	Theoretische Grundlagen	19
3.1	Maximum Likelihood Schätzung	19
3.2	Die Kostenfunktion	21
3.3	Optimierungsverfahren	22
3.3.1	Newton-Raphson Verfahren	22
4	Softwareinteraktion und Schnittstellen	27
4.1	Das Gesamtkonzept	27
4.2	Die Schnittstellen im Detail	28
4.2.1	Matlab Interface-Funktion	28
4.2.2	Simulink Modell	30
5	Sensibilitätsanalyse Fahrdynamik	33
5.1	Einleitung	33
5.2	Parameterauswahl	34
5.2.1	Bekannte Parameter	34
5.2.2	Unbekannte Parameter	34
5.3	Methode der Parametervariation	35
5.4	Längsdynamik	37
5.4.1	Das Fahrmanöver	37
5.4.2	Die Parameter	37
5.4.3	Die Parametervariation	40
5.4.4	Die Ergebnisse	41

5.5	Querdynamik	42
5.5.1	Das Fahrmanöver	42
5.5.2	Die Parameter	42
5.5.3	Die Parametervariation	45
5.5.4	Die Ergebnisse	46
6	Die Systemidentifikation	48
6.1	Das Fahrzeug	48
6.2	Die Fahrmanöver	49
6.3	Die Rohdaten	52
6.4	Generierung der Eingangssignale	52
6.5	Vorbereitung und Einrichtung der Software	56
6.5.1	Auswahl der CarSim Fahrzeugkomponenten	56
6.5.2	Einrichtung von FitLab	57
6.6	Die Parameteridentifikation	60
6.6.1	Identifikation der Querdynamik	61
6.6.2	Identifikation der Längsdynamik	62
6.7	Ergebnisse und Ergebnisvalidierung	67
6.8	Zusammenfassung und Ausblick	76
	Literaturverzeichnis	77
A	Ergebnisse der Sensibilitätsanalyse	78
A.1	Längsdynamik	79
A.2	Querdynamik	90
B	Programme und Modelle	102
B.1	Schnittstellen	103
C	Daten und Spezifikationen	106

Abbildungsverzeichnis

1.1	Das ViewCar. Quelle: www.dlr.de	10
1.2	Das Virtual Reality Labor. Quelle: www.dlr.de	11
1.3	Der dynamische Fahrsimulator. Quelle: www.dlr.de	12
1.4	Beispiel einer Zielfunktion	13
2.3	Der CarSim Animator	14
2.1	CarSim GUI	15
2.2	CarSim WinEP	16
2.4	Das FitLab Logo. Quelle: [PM84]	16
2.5	Beispiel FitLab Plot	17
2.6	Beispiel Matlab Plot	18
2.7	Beispiel Simulinkmodell	18
3.1	Beispiel: Zweidimensionale Normalverteilung	25
3.2	Lösungsraum des Anwendungsbeispiels	26
4.1	Struktur des Gesamtsystems im Detail	27
4.2	Kommunikationspfade innerhalb des Gesamtsystem	29
4.3	Das verwendete Simulinkmodell	32
5.1	Struktur der Systemidentifikation	33
5.2	Strukturbild der Sensibilitätsanalyse	36
5.3	Längsdynamik: Zeitverlauf Gangwahl	38
5.4	Längsdynamik: Zeitverlauf Lenkwinkel	38
5.5	Längsdynamik: Zeitverlauf Gaspedal	38
5.6	Normiertes Motorkennlinienfeld. Quelle für Form des Kennlinienfeldes: CarSim 6.0	39
5.7	Querdynamik: Zeitverlauf Gangwahl	43
5.8	Querdynamik: Zeitverlauf Lenkwinkel	43
5.9	Querdynamik: Zeitverlauf Gaspedal	43
5.10	Normierte Kurve der Reifenrückstellkraft. Quelle für Form der Kenn- linie: CarSim 6.0	44

5.11	Normierte Kurve der Reifenseitenkraft. Quelle für Form der Kennlinie: CarSim 6.0	44
5.12	Normierte Kurve der Federrückstellkraft. Quelle für Form der Kennlinie: CarSim 6.0	45
6.1	Das ViewCar des Instituts für Verkehrssystemtechnik. Quelle: [Sun07b]	48
6.2	Das ViewCar des Instituts für Verkehrssystemtechnik. Quelle: [Sun07b]	48
6.3	VW-Testgelände in Ehra. Quelle: [Sun07b]	49
6.4	Berechnung der Gangwahl	54
6.5	Dialogfenster zur Einrichtung des dynamischen Modells	57
6.6	Dialogfenster laden der Rohdaten	59
6.7	Dialogfenster, Wahl der Ein- und Ausgangskanäle.	59
6.8	Dialogfenster, Wahl der Parameter.	60
6.9	Normierte Kurve der Federrückstellkraft. Quelle für Form der Kennlinie: CarSim 6.0	61
6.10	Anpassung des normierten Motorkennlinienfeld. Quelle für Form des Kennlinienfeld: CarSim 6.0	63
6.11	Abdeckung des Motorkennlinienfeld durch Eingangsdaten Nr. 1	65
6.12	Abdeckung des Motorkennlinienfeld durch Eingangsdaten Nr. 2	66
6.13	Die Längsbeschleunigung des Simulationsmodell im Vergleich mit den Realdaten.	69
6.14	Die gefilterte Längsbeschleunigung des Simulationsmodell im Vergleich mit den Realdaten.	70
6.15	Die Längsbeschleunigung des Simulationsmodell im Detail und im Vergleich mit den Realdaten.	71
6.16	Die Querschleunigung des Simulationsmodell im Vergleich mit den Realdaten.	72
6.17	Die Geschwindigkeit des Simulationsmodell im Vergleich mit den Realdaten.	73
6.18	Die Gierrate des Simulationsmodell im Vergleich mit den Realdaten. .	74
6.19	Die Motordrehzahl des Simulationsmodell im Vergleich mit den Realdaten.	75
A.1	Ausgangsverläufe Sens. Analyse Längsdynamik Nr. 001	80
A.2	Ausgangsverläufe Sens. Analyse Längsdynamik Nr. 002	81
A.3	Ausgangsverläufe Sens. Analyse Längsdynamik Nr. 003	82
A.4	Ausgangsverläufe Sens. Analyse Längsdynamik Nr. 004	83
A.5	Ausgangsverläufe Sens. Analyse Längsdynamik Nr. 005	84
A.6	Ausgangsverläufe Sens. Analyse Längsdynamik Nr. 006	85
A.7	Ausgangsverläufe Sens. Analyse Längsdynamik Nr. 007	86
A.8	Ausgangsverläufe Sens. Analyse Längsdynamik Nr. 008	87
A.9	Ausgangsverläufe Sens. Analyse Längsdynamik Nr. 009	88
A.10	Ausgangsverläufe Sens. Analyse Längsdynamik Nr. 010	89

A.11 Ausgangsverläufe Sens. Analyse Querdynamik Nr. 001	91
A.12 Ausgangsverläufe Sens. Analyse Querdynamik Nr. 002	92
A.13 Ausgangsverläufe Sens. Analyse Querdynamik Nr. 003	93
A.14 Ausgangsverläufe Sens. Analyse Querdynamik Nr. 004	94
A.15 Ausgangsverläufe Sens. Analyse Querdynamik Nr. 005	95
A.16 Ausgangsverläufe Sens. Analyse Querdynamik Nr. 006	96
A.17 Ausgangsverläufe Sens. Analyse Querdynamik Nr. 007	97
A.18 Ausgangsverläufe Sens. Analyse Querdynamik Nr. 008	98
A.19 Ausgangsverläufe Sens. Analyse Querdynamik Nr. 009	99
A.20 Ausgangsverläufe Sens. Analyse Querdynamik Nr. 010	100
A.21 Ausgangsverläufe Sens. Analyse Querdynamik Nr. 011	101
C.1 Zeichnung: Doppelter Spurwechsel. Quelle [Sun07b]	112

KAPITEL 1

Einleitung und Motivation

1.1 Die Ausgangssituation und Problemstellung

Das Institut für Verkehrssystemtechnik des Deutschen Zentrums für Luft und Raumfahrt in Braunschweig beschäftigt sich, vor dem Hintergrund der vielen Verkehrsunfälle verursacht durch Fehlverhalten des Fahrers im Straßenverkehr, mit der Untersuchung von Fahrerassistenzsystemen unter ergonomischen, physiologischen sowie technischen Gesichtspunkten.

Unter den, dem Institut zur Verfügung stehenden Mitteln zur praktischen Erprobung und Untersuchung der Assistenzsysteme, befinden sich das ViewCar und drei Fahrsimulatoren.

„Das ViewCar ist ein Messfahrzeug zur Analyse der Wahrnehmungsprozesse und des Verhaltens von Fahrern im Straßenverkehr. Es ist mit Sensoren zur Messung und Aufzeichnung der Verkehrsumgebung, der Bedienung des Fahrzeugs und des resultierenden Fahrzeugverhaltens ausge-



Abbildung 1.1: Das ViewCar des Instituts für Verkehrssystemtechnik. Quelle: www.dlr.de

stattet. Damit ermöglicht das ViewCar Untersuchungen zum Verstehen und zur Modellierung des Fahrerverhaltens.“

- (Quelle: www.dlr.de)

Computersimulationen spielen eine immer größer werdende Rolle. Sie ermöglichen die kostengünstige Erprobung von technischen Systemen, die Berechnung von Ausgangsgrößen für diese, welche in der Realität nur schwer oder gar nicht messtechnisch erfassbar sind (man denke beispielsweise an die Temperatur in der Mitte eines Schmelzofens) und die Durchführung von Versuchen unter stets den selben Voraussetzungen.

In beispielsweise [KN98] wird Simulation wie folgt definiert:

Simulation: Simulation im engeren Sinne ist die Nachahmung realer Prozesse mittels Computern auf der Grundlage mathematischer Modelle.

Die ersten beiden Fahr simulatoren des Instituts, das Virtual-Reality-Labor (VR-Lab) und das Human-Machine-Interface-Labor (HMI-Lab) eignen sich zur schnellen und flexiblen Erprobung von Assistenzsystemen.

Der dritte Fahr simulator verfügt über ein komplettes integriertes Fahrzeug, welches über ein hydraulisches Hexapods system dynamisch bewegt werden kann. Auf diese Weise werden Fahr situationen, neben den visuellen Eindrücken über das 270° Bildgebungssystem und der akustischen Rückmeldungen des Surround-Soundsystems, um die Bewegungswahrnehmung ergänzt.

Die Technischen Eckdaten des dynamischen Fahr simulators sind in Tab. (1.1) aufgeführt.

Alle Fahr simulatoren haben gemein, dass sie die selbe Software für die Fahrdynamiksimulation verwenden. Diese Software heisst CarSim und wird später in Kap. 2 genauer vorgestellt.

In dieser Software sind die oben erwähnten mathematischen Modelle implementiert. Problematisch ist hierbei, dass das mathematische Modell nicht genau bekannt ist.

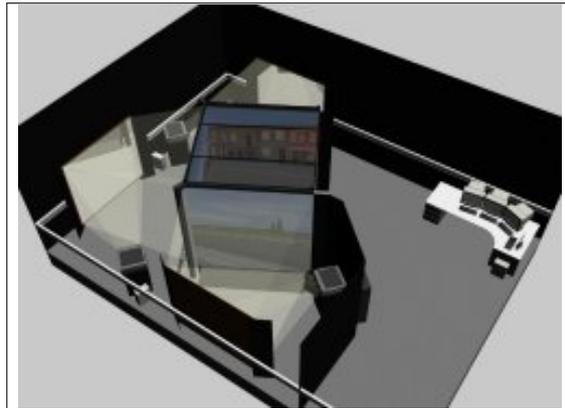


Abbildung 1.2: Das Virtual-Reality-Labor.
Quelle: www.dlr.de

Generell gilt für alle Fahrdynamikmodelle, dass sie eine gewisse Menge an Parametern benötigen, durch welche das zu simulierende Fahrzeug spezifiziert wird. Das kann im einfachsten Fall ein halbes Dutzend Parameter sein, oder, wie im Fall von CarSim, mehr als hundert.

Der aktuell in den Fahr simulatoren verwendete Parametersatz wurde so ausgewählt, dass er dem ViewCar möglichst nahe kommt, heisst, es wurde bei der Zusammenstellung aus den von CarSim mitgelieferten Fahrzeugkomponenten jene ausgewählt, welche denen des ViewCar am ähnlichsten sind.

Wünschenswert ist es an dieser Stelle jedoch, über einen Parametersatz zu verfügen, welcher das ViewCar, innerhalb gewisser Toleranzen, genau beschreibt!

Die dafür benötigten Kenngrößen und Parameter lassen sich jedoch nur unter größeren Aufwand Messtechnisch erfassen. Die wenigsten sind von vornherein bekannt.

	Weg	Geschwindigkeit	Beschleunigung
Längs	$\pm 1,5$ m	± 2 m/s	± 10 m/s ²
Quer	$\pm 1,4$ m	± 2 m/s	± 10 m/s ²
Vertikal	$\pm 1,4$ m	± 2 m/s	± 10 m/s ²
Rollen	$-20^\circ / + 21^\circ$	± 50 °/s	± 250 °/s ²
Nicken	$\pm 21^\circ$	± 50 °/s	± 250 °/s ²
Gieren	$\pm 21^\circ$	± 50 °/s	± 250 °/s ²

Tabelle 1.1: Die technischen Daten des dynamischen Fahrsimulators. Quelle: www.dlr.de



Abbildung 1.3: Der dynamische Fahrsimulator. Quelle: www.dlr.de

1.2 Die Lösungsidee

Der in dieser Arbeit behandelte Lösungsansatz ist die Identifikation des Parametersatz $\underline{\Gamma}$, über einen iterativen, numerischen Optimierungsprozess.

Hierzu werden bestimmte Kenngrößen wie beispielsweise Beschleunigungen und Geschwindigkeiten des realen Fahrzeugs während eines Fahrversuches, zusammen mit den Eingangsgrößen, also Lenkwinkel, Gangwahl, Kupplung, Gas- und Bremspedalstellung aufgenommen.

Die Problemstellung ergibt sich demnach wie folgt:

Man verfügt über eine Fahrdynamiksimulation, welche einen gewissen Parametersatz erwartet.

Des Weiteren hat man ein Fahrzeug zur Verfügung, welches in der Simulation abgebildet werden soll. Es ist nun der Parametersatz gesucht, für welche die Simulation das Realsystem, eben das Fahrzeug, möglichst genau widerspiegelt.

Anschliessend wird eine Simulation des Fahrzeugs mit einem beliebigen, vom Benutzer gewählten Startparametersatz $\underline{\Gamma}_{Start}$ durchgeführt. Die resultierenden Ausgangsverläufe werden miteinander verglichen. Über diesen Vergleich kann man die Ähnlichkeit beider Verläufe bewerten. Ausgehend vom Startparametersatz $\underline{\Gamma}_{Start} = \underline{\Gamma}_k(k = 0)$, wird dann versucht einen neuen Parametersatz $\underline{\Gamma}_{k+1}$ zu identifizieren, welcher das System besser beschreibt und die Ähnlichkeit der Ausgänge beider Systeme größer werden lässt.

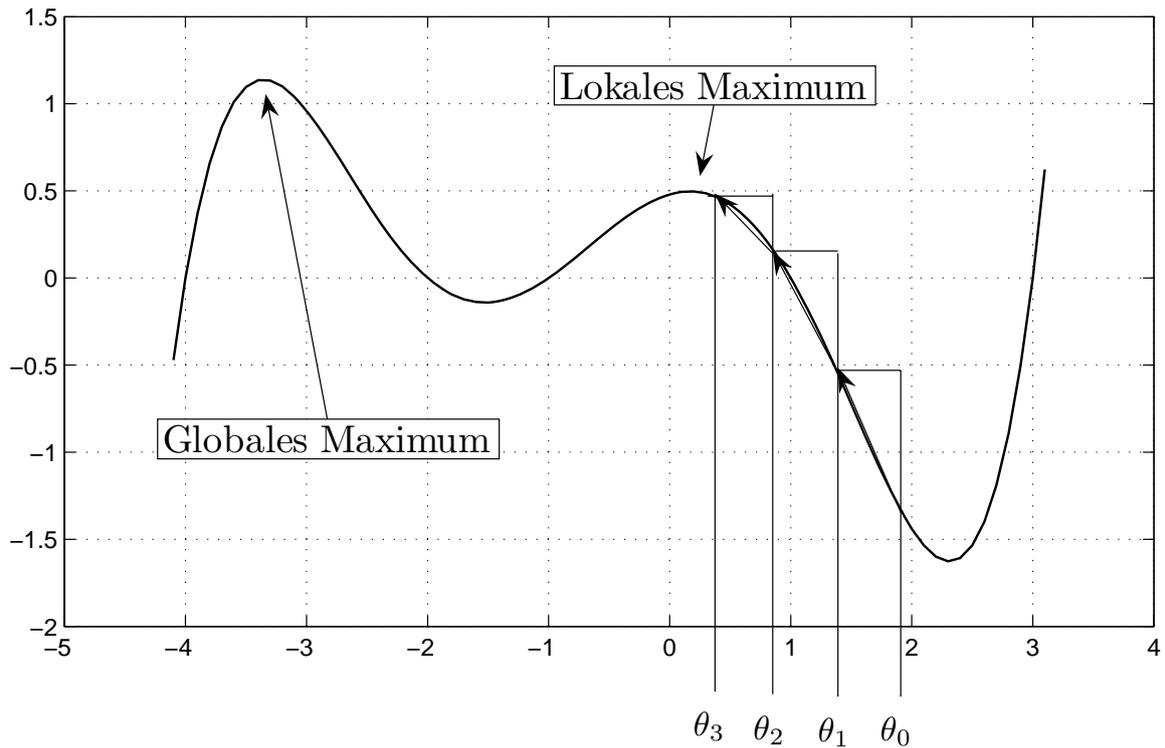


Abbildung 1.4: Beispiel einer Zielfunktion mit einem lokalen und einem globalen Maximum und der gegen Maximum strebenden Lösung θ_k

Auf diese Weise nähert man sich im zum System gehörenden Lösungsraum schrittweise dem Maximum. Da jedoch keine Analytische Beschreibung des Systems vorliegt, ist nicht entscheidbar, ob ein lokales Maximum gefunden wurde, oder das globale Maximum.

ob es zu den selben Ergebnissen führt, wenn man sofort nach allen Parametern optimiert, wird sich zeigen.

Als Vorgehensweise wird eine sequenzielle Optimierung der Parameter gewählt. Das System wird nach dem ersten Parameter optimiert. Anschliessend wird dieser konstant gehalten und es wird nach dem nächsten Parameter optimiert. Auf diese Weise wird versucht das System kontrolliert dem realen System anzunähern. Ob ein solches Vorgehen notwendig ist, oder

KAPITEL 2

Verwendete Software

2.1 CarSim

Die Simulationssoftware CarSim von Mechanical Simulation ist eine, in der Industrie sehr verbreitete, kommerzielle Software für Fahrdynamiksimulationen.

Die Software realisiert für ein Fahrzeug mit 6 räumlichen Freiheitsgraden eine nichtlineare, durch bis zu 150 Größen parametrierbare Systemdynamik, die in der Lage ist 700 bis 1200 Kenngrößen des Fahrzeugzustandes zu berechnen und zur Verfügung zu stellen.

Ein großer Vorteil von CarSim sind die sehr flexible Schnittstellen zu anderen Softwareprodukten wie Matlab/Simulink oder selbst geschriebenen Anwendungen in C oder C++.

Beides erlaubt es, Teilkomponenten der Fahrdynamik durch externe, den eigenen Bedürfnissen angepassten Modellen zu ersetzen, neue hinzu zufügen, oder die Dynamiksimulation in eigenen, angepassten Anwendungen zu verwenden, wie beispielsweise für Echtzeit-Berechnungen in Fahrzeugsimulatoren, wie es am Deutschen Zentrum für Luft und Raumfahrt

in Braunschweig oder der Fachhochschule Bielefeld geschehen ist.

CarSim setzt sich insgesamt aus 4 Modulen zusammen, die getrennt voneinander agieren.



Abbildung 2.3: CarSim Animator

GUI Das Graphical User Interface, z.B. Abb. (2.1), erlaubt es, Fahrzeugmodelle bequem zu parametrieren, die Teilkomponenten eines Fahrzeuges auszuwählen (z.B.: automatische Gangschaltung, manuelle Gangschaltung, Kupplungsart, usw.) und diese zu einem Gesamtmodell zusammenzuführen. Das Ergebnis dieses Design- und Parametrierungsvorganges ist eine Parameterdatei, in welcher alle Einstellungen, die über das GUI gemacht wurden, gespeichert werden.

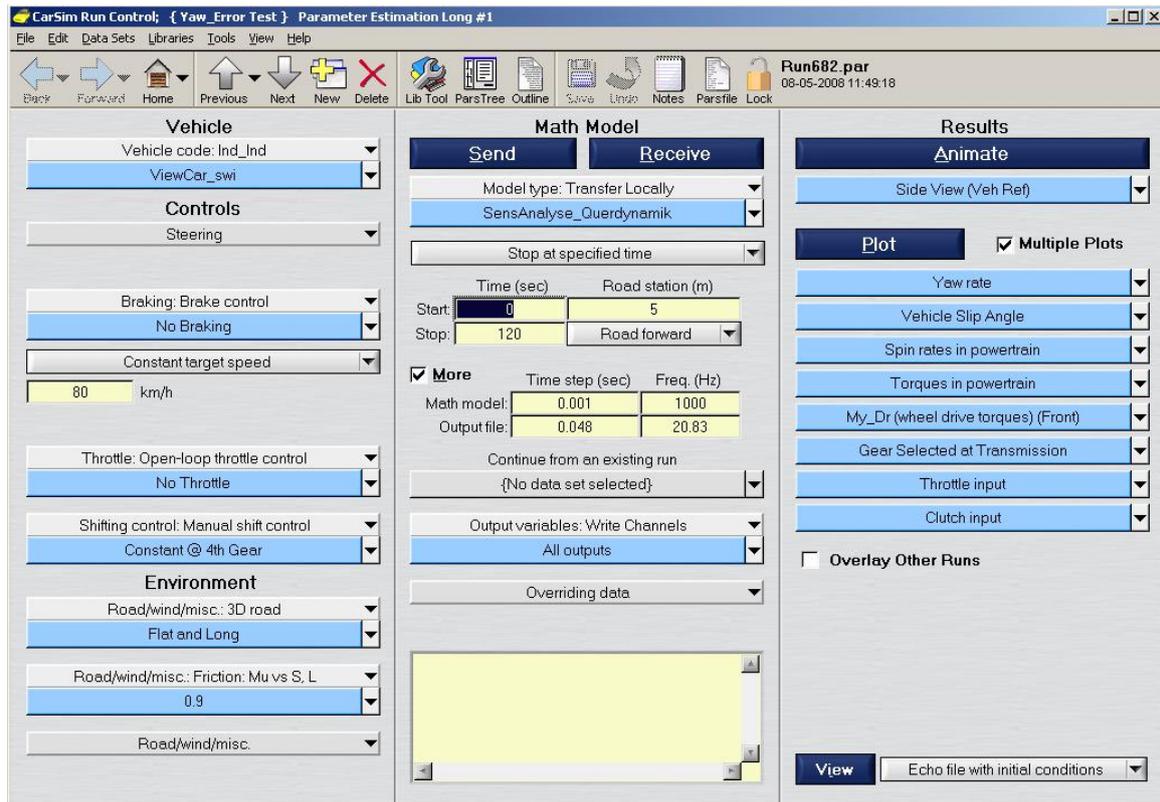


Abbildung 2.1: CarSim Graphical User Interface

Solver Der Solver ist das Programm, welches numerisch die Differentialgleichungen des Dynamikmodells löst. Als Eingang erwartet der Solver eine Parameterdatei, in welcher die Parametrierung der Systemdynamik zu finden ist und bestimmt wird, welche Ein- und Ausgangsgrößen das Modell hat. Neben dem internen Solver von CarSim steht eine .dll zur Verfügung welche von Matlab/Simulink angesprochen werden kann und eine .lib C/C++ Library, die in eigene C/C++ Projekte eingebunden werden kann.

Animator Der Animator ist eine Grafikanwendung die in der Lage ist aus den, durch den Solver berechneten Ausgangsdaten, eine 3D-Animation des Fahrzeuges und der Fahrumgebung zu generieren. Auf diese Weise lassen sich die Ergebnisse der Dynamiksimulation anschaulich darstellen und präsentieren.

WinEP WinEP ist eine Plotting und Analysesoftware zur grafischen Darstellung der Ein- und Ausgangsdaten der Simulation

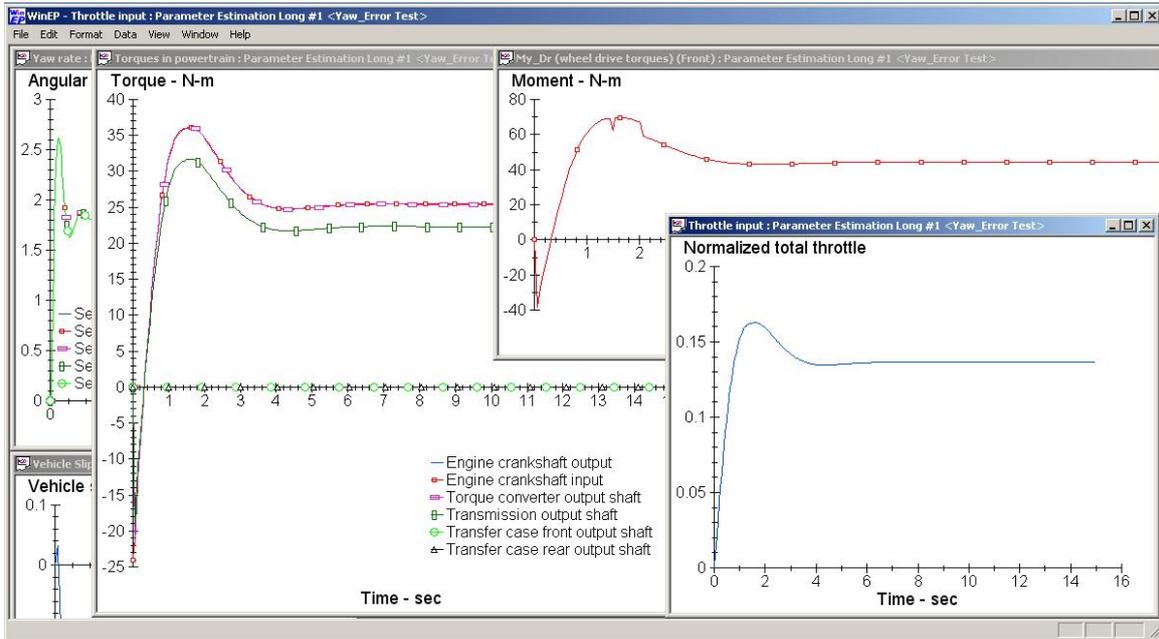


Abbildung 2.2: CarSim WinEP

2.2 FitLab

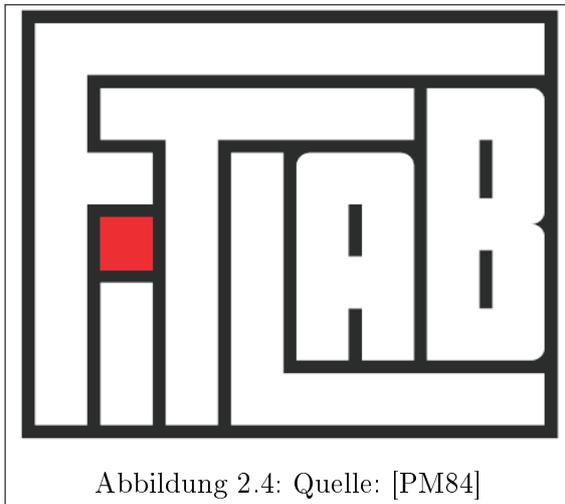


Abbildung 2.4: Quelle: [PM84]

In der Toolbox sind die in Kapitel 3 beschriebenen Algorithmen und Verfahren des Systemidentifikationsprozesses implementiert.

Dabei besitzt FitLab Schnittstellen und Verfahren zur Identifikation von linearen und nicht linearen Systemen im Zeitbereich, sowie linearen Systemen im Frequenzbereich.

Als Eingangsgrößen erwartet die Toolbox aufgenommene Messdaten des zu identifizierenden Systems und ein simulierbares Modell, welches über eine vordefinierte Schnittstelle verfügen muss.

FitLab ist eine Toolbox für Matlab welche im Rahmen von Forschungstätigkeiten am Institut für Flugsystemtechnik des Deutschen Zentrums für Luft und Raumfahrt entstanden ist.

Über diese Schnittstelle wird die Simulation parametrisiert und mit den zeitlichen Verläufen der Eingänge versorgt. Die Ausgänge der Simulation nimmt FitLab wiederum entgegen und berechnet aus dem

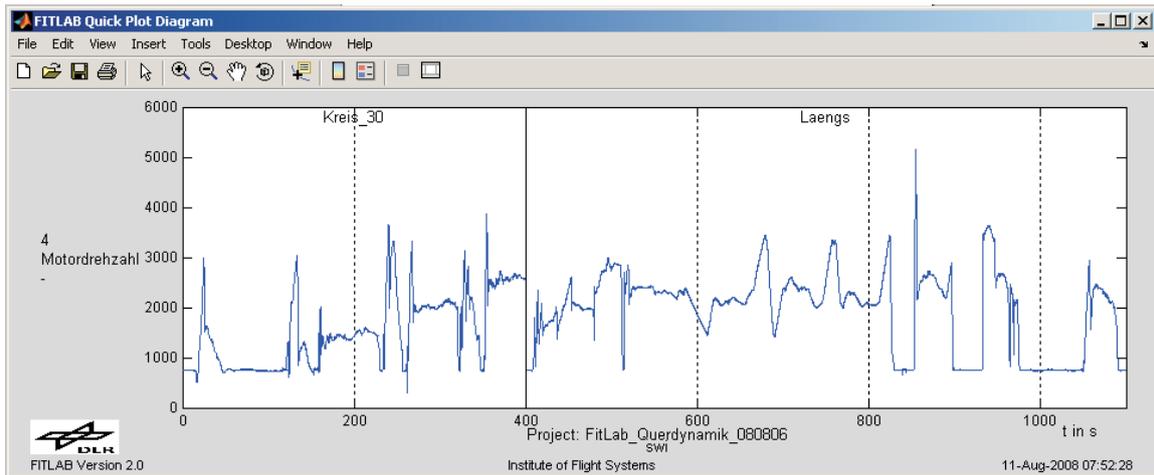


Abbildung 2.5: FitLab Plot von Eingangsdaten

Vergleich von Soll- und Istverläufen in welcher Art die Parametrierung geändert werden muss.

Insbesondere ist FitLab in der Lage über eine große Zahl von Parametern und verschiedenen Messdatensätzen gleichzeitig zu optimieren.

Dies ist essenziell wenn man das Simulationsmodell nicht nur für einzelne Manöver optimieren will, sondern für ein möglichst großes Spektrum von Eingangsverläufen.

2.3 Matlab und Simulink

Matlab steht für MATrix LABoratory und ist eine kommerzielle Software aus dem Hause The Mathworks zur Lösung numerischer Probleme und der Analyse und Darstellung von Daten.

Die im Programm integrierte Skriptsprache, welche zur Laufzeit interpretiert wird, ermöglicht die unkomplizierte Implementierung von kleineren Programmen und Algorithmen.

Der Funktionsumfang der Software lässt sich modular durch Zukauf von sogenannten Toolboxen erweitern, was kostspielig ist, Matlab aber in die Lage versetzt, bei allen möglichen technischen Problemen und Aufgabenstellungen aus diversen Bereichen, wie beispielsweise Reglerentwurf, Filterdesign, Bild- und Videoverarbeitung, Bioinformatik, Entwicklung von Realtime-Software oder Modellierung und Analyse von Prozessen des Finanzwesens, behilflich zu sein.

Besonders erwähnenswert ist die Software Simulink, welche auf Matlab aufsetzt und ein grafisches Modellierungstool für dynamische Systeme ist. Es bringt eine große Anzahl an vorgefertigten Blöcken mit, die es einem erlauben Simulationen

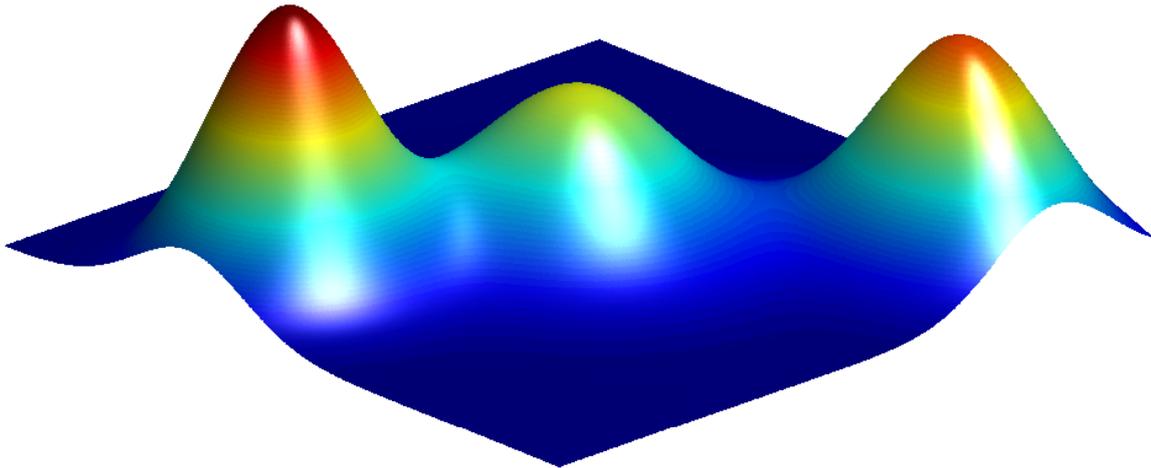


Abbildung 2.6: Matlab Beispielplot

schnell und auf einfache Art und Weise auf zu bauen.

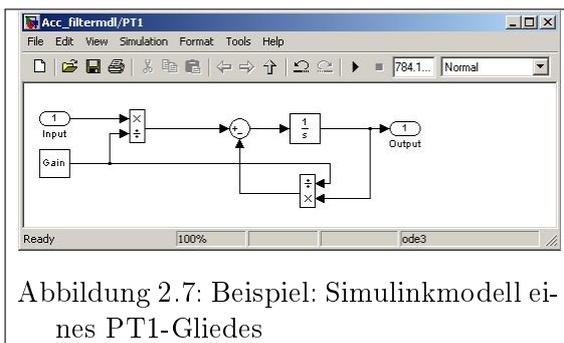


Abbildung 2.7: Beispiel: Simulinkmodell eines PT1-Gliedes

Diese Simulinkmodelle lassen sich dabei, durch zur Verfügung gestellter Schnittstellen, um eigene, in C, C++ oder Fortran geschriebene Komponenten, in sogenannten S-Function, erweitern.

Dies spielt für diese Arbeit eine besondere Rolle, da sich auf diese Weise der CarSim Solver in eine Simulink Simulation einbetten lässt.

KAPITEL 3

Theoretische Grundlagen

3.1 Maximum Likelihood Schätzung

Für das Verfassen des nachfolgenden Kapitels haben sich [FCHL05], [KN98] und [PM84] als sehr nützlich erwiesen.

Das zu parametrierende System ist durch ein mathematisches Modell der Form

$$\begin{aligned}\dot{\underline{x}} &= F(\underline{x}(t), \underline{u}(t), \underline{\Gamma}) \\ \underline{y} &= F(\underline{x}(t), \underline{u}(t), \underline{\Gamma})\end{aligned}\quad (3.1)$$

gegeben. Dieses System besitzt m Ausgangsgrößen $\underline{y}(t) = [y_1(t), \dots, y_m(t)]$. Für diese Ausgangsgröße werden nun $k \in \mathbb{N}^*$ Messungen zu den diskreten Zeitpunkten t_1, \dots, t_k vorgenommen. Sie bilden die Messmatrix

$$\underline{z} = [\underline{y}(t_1) + \underline{v}(t_1), \dots, \underline{y}(t_k) + \underline{v}(t_k)], \quad (3.2)$$

wobei sich die Messgröße $\underline{z}(t_k)$ aus der Ausgangsgröße des Systems $\underline{y}(t_k)$ und einem Messfehler $\underline{v}(t_k)$ zusammen setzt.

$$\underline{z}(t) = \underline{y}(t) + \underline{v}(t)$$

Der Messfehler für die Messgröße z_i hat dabei die Wahrscheinlichkeitsverteilung f_i , $i = 1, \dots, m$ und wird hier durch ein stationäres gaußsches weißes Rauschen mit dem Mittelwert null modelliert.

$$f_i(\sigma_i^2, \mu_i) = \frac{1}{\sigma_i \sqrt{2\pi}} e^{-\frac{1}{2} \left(\frac{z_i - \mu_i}{\sigma_i} \right)^2}, \quad (3.3)$$

$$E[v_i] = 0, \quad (3.4)$$

$$\text{Var}(v_i) = \sigma_i^2. \quad (3.5)$$

Hierbei ist $E[X]$ der Erwartungswert einer Zufallsgröße und $\text{Var}(X)$ deren Varianz.

Nimmt man an, dass die Messfehler der einzelnen Messgrößen stochastisch unabhängig sind, so gilt für die Kovarianzmatrix \underline{R} mit dem Kronecker-Delta δ_{ij} ,

$$\begin{aligned}\text{Cov}(v_i, v_j) &= \delta_{ij} \text{Var}(v_i) \\ &= \underline{R}\end{aligned}$$

was für $i = j$, bzw $X = Y$ direkt aus der Definition der Varianz und Kovarianz (siehe z.B. [FCHL05]) folgt:

$$\begin{aligned}\text{Var}(X) &= E[(X - E[X])^2] \\ \text{Cov}(X, Y) &= E[(X - E[X])(Y - E[Y])] \\ &= E[(X - E[X])^2] \text{ für } X = Y\end{aligned}$$

Da der Messfehler als Mittelwertfrei angenommen wird gilt für den Erwartungswert der Messgröße z

$$E[z_i] = \mu_i = y_i(t). \quad (3.6)$$

Aus der Differenz in Gl. (3.3) wird dann

$$z_i - \mu_i = z_i(t_k) - y_i(t_k) = v_i(t_k). \quad (3.7)$$

Die multivariante Verteilung des Messfehlervektors $\underline{v}(t) = [v_1(t), \dots, v_m(t)]$ ergibt sich, da die einzelnen Messfehler stochastisch unabhängig sind, mit dem Pro-

dukt der einzelnen Messfehlerverteilungen $f_i(\sigma_i^2, \mu_i)$.

$$\begin{aligned} f(\underline{v}(t)) &= f(v_1(t), \dots, v_m(t)) \\ &= \prod f_i(v_i(t)) \\ &= \frac{1}{\det |\underline{R}|^{\frac{1}{2}} (2\pi)^{\frac{m}{2}}} e^{-\frac{1}{2}[\underline{v}(t)]^T \underline{R}^{-1} [\underline{v}(t)]} \end{aligned}$$

Die Verteilungsfunktion der Fehlervektoren über das Messungsintervall (t_1, t_k) für die k Messungen ist für eine zeitinvariante Kovarianzmatrix gegeben mit Gl. (3.8)

$$\begin{aligned} f &= f(\underline{v}(t_1), \dots, \underline{v}(t_k)) = \prod_{i=1}^k \frac{1}{\det |\underline{R}|^{\frac{1}{2}} (2\pi)^{\frac{m}{2}}} e^{-\frac{1}{2}[\underline{v}(t_i)]^T \underline{R}^{-1} [\underline{v}(t_i)]} = \\ p(\underline{z}|\underline{\Gamma}) &= \frac{1}{\det |\underline{R}|^{\frac{k}{2}} (2\pi)^{\frac{mk}{2}}} \exp \left(-\frac{1}{2} \sum_{i=0}^k ([z(t_i) - y(t_i)]^T \underline{R}^{-1} [z(t_i) - y(t_i)]) \right), \quad (3.8) \end{aligned}$$

wenn man annimmt, das die Messfehler für verschiedene Zeitpunkte stochastisch unabhängig sind, das Rauschen also weiß ist.

$p(\underline{z}|\underline{\Gamma})$ in Gl. (3.8) gibt die Wahrscheinlichkeit an, eine gewisse Messung \underline{z} in Abhängigkeit der Parameter $\underline{\Gamma}$ zu machen.

Auf das Problem der Systemidentifikation bezogen, liegt die Messmatrix jedoch bereits vor und wir interessieren uns für den Parametervektor $\underline{\Gamma}$ des dazu gehörenden Systems.

Es liegt also ein inverses Problem vor: Die Messung ist bekannt und es sind die Parameter gesucht, die die Wahrscheinlichkeit

p , eben jene Messung zu machen, maximal werden lassen.

$$L(\underline{\Gamma}|\underline{z}) = p(\underline{z}|\underline{\Gamma}) \quad (3.9)$$

$$L(\underline{\Gamma}|\underline{z}) \stackrel{!}{=} \max \quad (3.10)$$

Notwendige und hinreichende Bedingung hierfür ist:

$$\frac{\partial L'(\underline{\Gamma}|\underline{z})}{\partial \underline{\Gamma}} = 0 \quad (3.11)$$

$$\frac{\partial L''(\underline{\Gamma}|\underline{z})}{\partial \underline{\Gamma}} < 0 \quad (3.12)$$

Die Funktion L nennt man Likelihood-Funktion und das Verfahren der Systemidentifikation, über das Maximieren von L , Maximum Likelihood-Schätzung (engl.: Maximum Likelihood Estimation, kurz: MLE).

Eine Äquivalente Aufgabenstellung ist aufgrund der Monotonie der Logarithmusfunktion (siehe [KN98]), die Minimierung der sogenannten Log-Likelihood-Funktion Gl. (3.15) mit der Bedingung:

$$\frac{\partial \ell'(\underline{\Gamma}|\underline{z})}{\partial \underline{\Gamma}} = 0 \quad (3.13)$$

$$\frac{\partial \ell''(\underline{\Gamma}|\underline{z})}{\partial \underline{\Gamma}} > 0. \quad (3.14)$$

Bei der Logarithmierung von Gl. (3.8) anfallende konstante Terme können hierbei vernachlässigt werden, da sie für die Minimierungsaufgabe irrelevant sind. Unbekannt und zu bestimmen ist, neben dem Parametervektor $\underline{\Gamma}$, Gl. (3.15) zu folge, auch die Kovarianzmatrix \underline{R} .

$$\ell(\underline{\Gamma}|\underline{z}) = \log L(\underline{\Gamma}|\underline{z}) = -\frac{k}{2} \log(\det(\underline{R})) - \frac{1}{2} \sum_{i=0}^k ([z(t_i) - y(t_i)]^T \underline{R}^{-1} [z(t_i) - y(t_i)]) \quad (3.15)$$

3.2 Die Kostenfunktion

Für die quantitative Bewertung von Simulationsergebnissen benötigt man eine Kostenfunktion,

$$\text{Cost} = f(\underline{y}_{real}, \underline{y}_{sim}),$$

welche einen Zahlenwert dafür liefert, wie gut die Ausgangsgrößen der Simulation mit den Ausgangsgrößen des Realsystems über ein stimmen, oder wie groß der bei der Simulation gemachte Fehler ist.

Ziel des Optimierungsprozess ist es dann, die Güte der Übereinstimmung zu maximieren oder die Höhe des gemachten Fehlers zu minimieren.

Die Grundlage für die Kostenfunktion, welche in FitLab ihre Anwendung findet und in [PM84] hergeleitet wird, ist die Log-Likelihood-Funktion aus Abschnitt (3.1).

Diese setzt sich aus zwei Termen zusammen. In [PM84] wird die Log-Likelihood-Funktion dabei auf den Logarithmus der Determinante von \underline{R} reduziert. Da die Kovarianzmatrix als Diagonalmatrix angenommen wird ist der Wert der Determinante von \underline{R}

$$\text{Cost} \approx \sum_{i=1}^m \sigma^2(z_i - y_i), \quad (3.16)$$

wobei für $\sigma^2(z_i - y_i)$

$$\sigma^2(z_i - y_i) = \frac{1}{N} \sum_{k=0}^{N-1} (z_i(t_k) - y_i(t_k))^2 \quad (3.17)$$

angesetzt wird.

m ist hierbei die Anzahl der Messgrößen oder Ausgänge des zu identifizierenden Systems und N die Anzahl der diskreten Messpunkte die für jeden Ausgang aufgenommen wurden.

3.3 Optimierungsverfahren

Besitzt man ein mathematisches Modell des zu identifizierenden Systems und eine Kostenfunktion welche eine quantitative Aussage dafür liefert, wie gut die aktuellen Parameter $\underline{\Gamma}$ des Systemmodells das zu identifizierende System widerspiegeln, beziehungsweise wie gut die Ausgänge des mit $\underline{\Gamma}$ parametrisierten Systems mit denen des Realsystems über ein stimmen, so benötigt man noch ein Optimierungsverfahren welches die Übereinstimmung beider Systeme, bewertet durch die Kostenfunktion, maximal werden lässt.

$$\begin{aligned} L(\underline{z}|\underline{\Gamma}) &\stackrel{!}{=} \max \\ \ell(\underline{z}|\underline{\Gamma}) &\stackrel{!}{=} \min \end{aligned}$$

Nachfolgend sollen die in FitLab verwendeten Methoden zur Lösung des Optimierungsproblems vorgestellt und erläutert werden.

3.3.1 Newton-Raphson Verfahren

Das in [Qui01] oder [PM84] beschriebene Newton-Raphson-Verfahren verfolgt bei der Optimierung den folgenden Ansatz.

Die Taylor'sche Formel ist beispielsweise in [Bus01] gegeben mit:

Ist $D \in \mathbb{R}$ ein echtes Intervall, $f \in C^{n+1}(D)$ (f also $n + 1$ -mal stetig dif-

ferenzierbar), $a \in D$: Dann gilt für alle $x \in D$

$$\begin{aligned} f(x) &= f(a) \\ &+ \frac{f'(a)}{1!}(x-a) \\ &+ \frac{f''(a)}{2!}(x-a)^2 \\ &+ \dots \\ &+ \frac{f^{(n)}(a)}{n!}(x-a)^n \\ &+ R_{n+1} \end{aligned} \tag{3.18}$$

mit

$$R_{n+1} = \frac{1}{(n+1)!} \int_a^x (x-t)^{n+1} f^{(n+2)}(t) dt \tag{3.19}$$

also $f(x) = T_{n,a}(f)(x) + R_{n+1}(x)$.

Danach lässt sich eine Funktion f durch ein Taylor-Polynom n -ter Ordnung approximieren, sofern f n -mal stetig differenzierbar ist.

Bricht man die Reihenentwicklung Gl. (3.18) nach dem zweiten nicht trivialen Glied ($n=2$) ab, so spricht man von dem Näherungspolynom 2. Ordnung. Für die Taylorentwicklung um den Entwicklungspunkt x erhält man dann

$$f(x+h) \approx f(x) + f'(x)h + \frac{1}{2}f''(x)h^2, \tag{3.20}$$

beziehungsweise mit $a = f(x)$, $b = f'(x)$ und $c = f''(x)$

$$f(x+h) \approx a + bh + \frac{1}{2}ch^2. \tag{3.21}$$

Für dieses quadratische Polynom ist die Bestimmung des Maximums besonders

einfach. Gesucht ist nun der Wert $h = \hat{h}$ und D^2 die Hesse-Matrix der $f(x + h)$ maximal werden lässt. Für das maximum muss gelten

$$\frac{df(x + \hat{h})}{dx} = 0 = b + c\hat{h} \quad (3.22)$$

also

$$\hat{h} = -\frac{b}{c}. \quad (3.23)$$

Das Maximum der Näherungsfunktion liegt damit bei

$$max = x + \hat{h} = x - \frac{b}{c} = x - \frac{f'(x)}{f''(x)}. \quad (3.24)$$

Mit Gl. (3.24) als Grundlage ist der Newton Raphson Algorithmus im eindimensionalen Fall gegeben mit:

$$x_i = x_{i-1} - \frac{f'(x_{i-1})}{f''(x_{i-1})} \quad (3.25)$$

Wenn f nicht nur von einer Variable, sondern von mehreren abhängig ist,

$$f = f(x_1, \dots, x_n),$$

ändert sich die Taylorapproximation 2. Ordnung zu

$$f(\underline{x} + \underline{h}) = f(\underline{x}) + \nabla f(\underline{x})^T \underline{h} + \frac{1}{2} \underline{h}^T D^2 f(\underline{x}) \underline{h}. \quad (3.26)$$

∇ ist hierbei der Nabla-Operator

$$\nabla f(\underline{x}) = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{pmatrix}$$

$$D^2 f(\underline{x}) = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1 \partial x_1} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n \partial x_n} \end{pmatrix}$$

Analog zum eindimensionalen Fall ergibt sich mit $\underline{a} = f(\underline{x})$, $\underline{b} = \nabla f(\underline{x})$ und $\underline{c} = D^2 f(\underline{x})$

$$f(\underline{x} + \underline{h}) = \underline{a} + \underline{b}^T \underline{h} + \frac{1}{2} \underline{h}^T \underline{c} \underline{h}. \quad (3.27)$$

Macht $\underline{h} = \hat{\underline{h}}$, $f(\underline{x} + \hat{\underline{h}})$ zum Maximum, gilt

$$\frac{df(\underline{x} + \hat{\underline{h}})}{d\underline{x}} = \underline{0} = \underline{b} + \underline{c} \hat{\underline{h}}, \quad (3.28)$$

also

$$\hat{\underline{h}} = -\underline{c}^{-1} \underline{b}. \quad (3.29)$$

Das Maximum von Gl. (3.27) liegt dann bei

$$\underline{x} + \hat{\underline{h}} = \underline{x} - \underline{c}^{-1} \underline{b}, \quad (3.30)$$

beziehungsweise

$$\underline{x} + \hat{\underline{h}} = \underline{x} - (D^2 f(\underline{x}))^{-1} \nabla f(\underline{x}) \quad (3.31)$$

Mit Gl. (3.31) als Grundlage ist der Newton Raphson Algorithmus für mehrere Variablen gegeben mit:

$$x_i = \underline{x}_{i-1} - (D^2 f(\underline{x}_{i-1}))^{-1} \nabla f(\underline{x}_{i-1}) \quad (3.32)$$

Ist die Funktion des zu optimierenden Systems unbekannt, wie es zum Beispiel

in dieser Arbeit der Fall ist, da die Systemdynamik in einer kommerziellen Software implementiert ist, müssen die partiellen Ableitungen für $D^2 f(\underline{x})$ und $\nabla f(\underline{x})$ numerisch berechnet werden, was für Systeme von vielen Parametern in einem erheblichen Rechenaufwand resultiert.

Eine weitere Problematik ergibt sich, wenn die Hesse-Matrix D^2 singularär wird und sich nicht invertieren lässt.

Der Algorithmus lässt sich vereinfachen, indem in Gl. (3.20) und Gl. (3.26) nicht durch eine quadratische Funktion approximiert wird, sondern durch eine Lineare.

$$f(x+h) \approx f(x) + f'(x)h \quad (3.33)$$

und

$$f(\underline{x} + \underline{h}) = f(\underline{x}) + \nabla f(\underline{x})^T \underline{h}. \quad (3.34)$$

Damit vereinfachen sich die Algorithmen Gl. (3.25) und Gl. (3.32) zu

$$x_i = x_{i-1} - f'(x_{i-1}) \quad (3.35)$$

und

$$x_i = \underline{x}_{i-1} - \nabla f(\underline{x}_{i-1}) \quad (3.36)$$

Schlussendlich wird noch eine Abbruchbedingung benötigt, welche bestimmt, wann wir unsere Berechnung der Lösung \underline{x} beenden können.

Geeignete ist hier neben einer Begrenzung der Anzahl der iterativen Schritte eine Überprüfung der relativen Änderung der Lösung

$$\Delta \underline{x} = |\underline{x}_{i+1} - \underline{x}_i| < \xi_{Abbr}, \quad (3.37)$$

so das wir die Berechnung beenden, wenn sich die Lösung von Rechenschritt zu Rechenschritt nur noch um sehr kleine Beträge ändert.

Einfaches Beispiel für das Newton-Raphson-Verfahren

Nachfolgend soll der Algorithmus Gl. (3.36) für mehrere Variablen an einem einfachen Beispiel demonstriert werden.

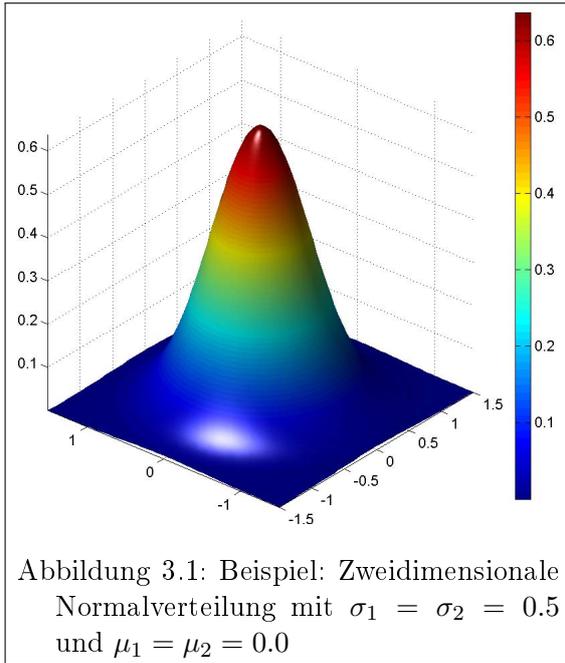
Als Zielfunktion wird die Normalverteilung für zwei stochastisch unabhängige Variablen $x_i, i = 1, 2$ gewählt.

$$f_i(\sigma_i^2, \mu_i, x_i) = \frac{1}{\sigma_i \sqrt{2\pi}} e^{-\frac{1}{2} \left(\frac{x_i - \mu_i}{\sigma_i} \right)^2}, \quad (3.38)$$

$$f(x_1, x_2) = f_1(\sigma_1^2, \mu_1, x_1) f_2(\sigma_2^2, \mu_2, x_2) \quad (3.39)$$

Für Gl. (3.36) werde die partiellen Ableitungen

$$\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}$$



benötigt.

$f(x_1, x_2)$ ist gegeben mit

$$f(x_1, x_2) = \frac{1}{\sigma_1 \sigma_2 2\pi} e^{-\frac{1}{2} \left(\frac{x_1 - \mu_1}{\sigma_1} \right)^2 - \frac{1}{2} \left(\frac{x_2 - \mu_2}{\sigma_2} \right)^2} \quad (3.40)$$

Die partiellen Ableitungen ergeben sich mit

$$\frac{\partial f}{\partial x_1} = f(x_1, x_2) \left(-1 \frac{1}{\sigma_1^2} (x_1 - \mu_1) \right) \quad (3.41)$$

$$\frac{\partial f}{\partial x_2} = f(x_1, x_2) \left(-1 \frac{1}{\sigma_2^2} (x_2 - \mu_2) \right), \quad (3.42)$$

so das wir für die iterative Berechnungsvorschrift des Maximums von $f(x_1, x_2)$ erhalten

$$\underline{x}_{i+1} = \underline{x}_i - \begin{pmatrix} f(x_1, x_2) \left(-1 \frac{1}{\sigma_1^2} (x_1 - \mu_1) \right) \\ f(x_1, x_2) \left(-1 \frac{1}{\sigma_2^2} (x_2 - \mu_2) \right) \end{pmatrix} \quad (3.43)$$

Die Zielfunktion wird parametrisiert mit:

$$\begin{aligned} \sigma_1 &= 1.00 \\ \sigma_2 &= 0.75 \\ \mu_1 &= 5.00 \\ \mu_2 &= 5.00 \end{aligned} \quad (3.44)$$

Als Startwert wird

$$x_1 = [6.0, 6.3] \quad (3.45)$$

gewählt.

Der Grenzwert für die minimale Änderung der Lösung bei der der Algorithmus abgebrochen werden soll ist mit $\xi = 0.001$ festgelegt und die Anzahl der maximalen Iterationsschritte wird mit $max\ i = 50$ gewählt.

Die Zielfunktion, zusammen mit der gegen Maximum konvergierenden Lösung \underline{x} ist in Abb. (3.2).

Die Berechnung wird nach $i = 33$ Schritten abgebrochen, da der Grenzwert ξ erreicht wurde. Als Lösung liefert die Berechnung:

$$f(x_1, x_2) = \max \quad (3.46)$$

für

$$x_1 = 5.00302 \quad (3.47)$$

$$x_2 = 5.00001. \quad (3.48)$$

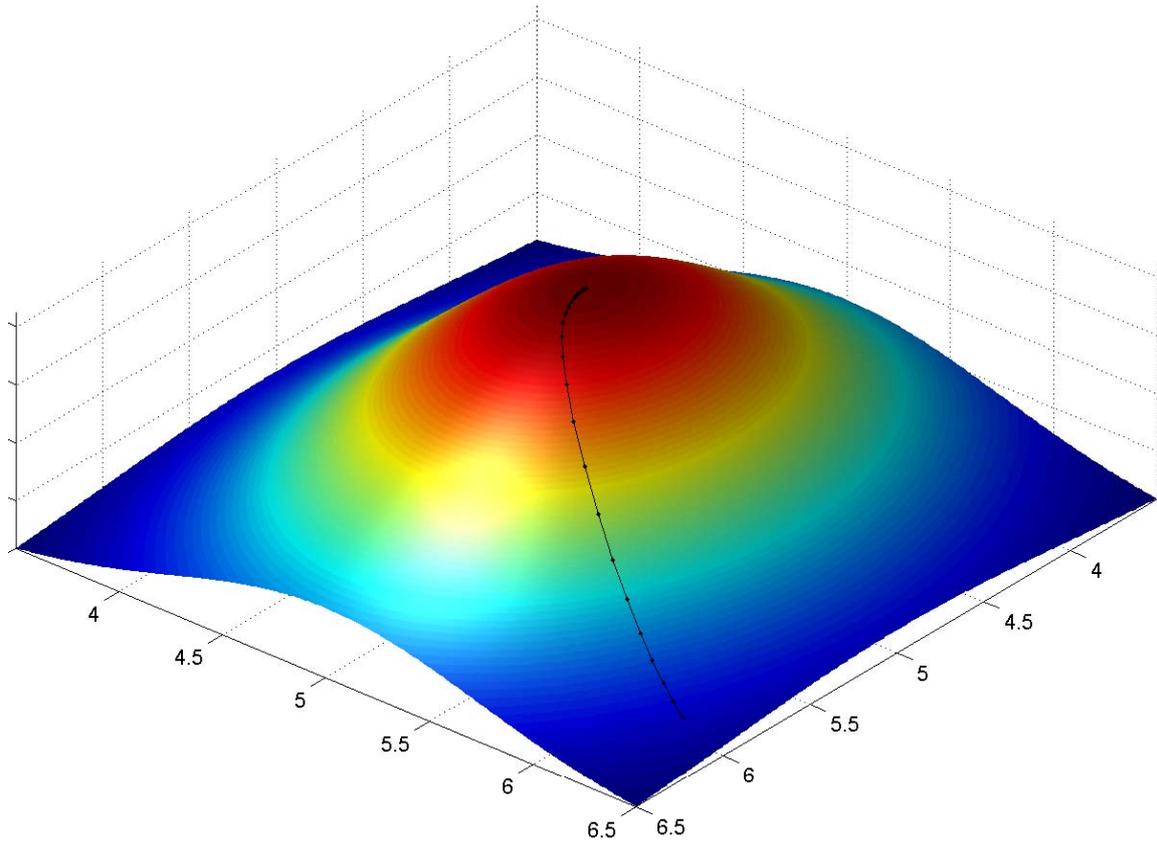


Abbildung 3.2: Fläche der Lösungen von $f(x_1, x_2)$ mit gegen Maximum konvergierenden Parametern $\underline{x}_i = [x_1, x_2]$. Startpunkt bei $x_1 = [6.0, 6.3]$

KAPITEL 4

Softwareinteraktion und Schnittstellen

4.1 Das Gesamtkonzept

In Kapitel 2 wurden die für die Systemidentifikation verwendeten Programme vorgestellt.

Nachfolgend soll die Art der Interaktion zwischen den verschiedenen Programmen

erläutert und auf die Details der verschiedenen Schnittstellen eingegangen werden.

Die Grundlage des Gesamtsystems bildet Matlab. In Matlab integriert oder auf Matlab aufgesetzt sind die Toolbox FitLab und die Simulationssoftware Simulink. In FitLab sind die Algorithmen der Systemidentifikation implementiert.

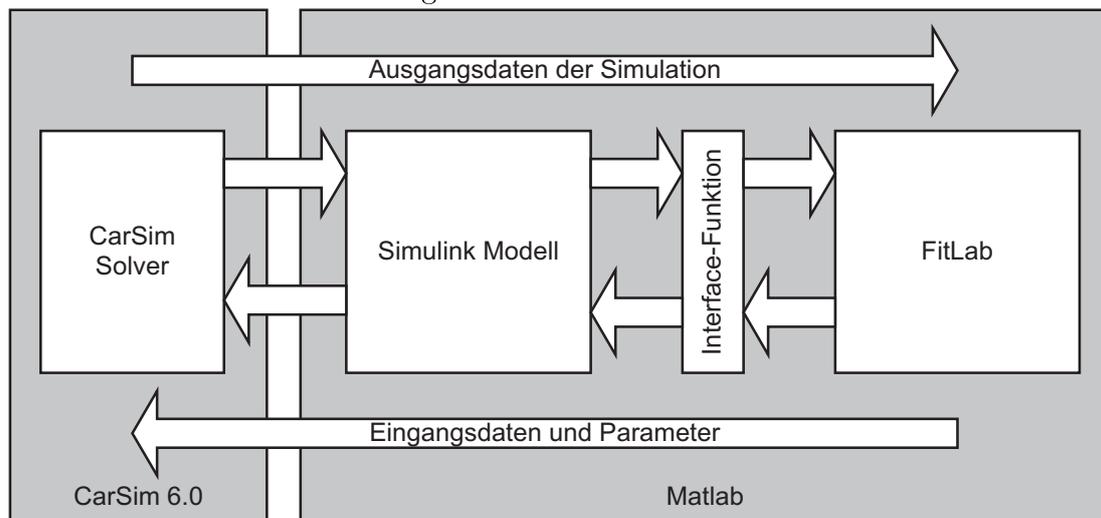


Abbildung 4.1: Struktur des Gesamtsystems. Im unteren Datenpfad werden Parameter und Eingangsverläufe bis zu CarSim durchgereicht. Der CarSim-Solver führt die Berechnungen durch. Anschliessend werden im oberen, rückläufigen Datenpfad die Ausgangsdaten zu FitLab zurückgeführt.

Der Benutzer muss FitLab dabei eine Matlab-Funktion zur Verfügung stellen, welche für die einzelnen Simulationen die Ein- und Ausgangsdaten, sowie Parameter entgegen nimmt. Über diese Matlab-Funktion wird ein Simulink-Modell angesprochen, welches wiederum den CarSim-Solver als S-Function einbindet. Eine ausführliche Beschreibung folgt in Kapitel 4.2.1.

Der CarSim-Solver führt die Berechnung der Fahrzeugdynamik durch und gibt die Ausgangsverläufe an Simulink zurück. Simulink reicht die Ausgangsdaten an die Interfacefunktion weiter. Hier findet eine kurze Nachbearbeitung der Daten statt bevor sie an FitLab zurück gegeben werden.

FitLab führt dann den Vergleich der Simulationsergebnisse mit den Realdaten durch und berechnet mit den in Kapitel 3 eingeführten Methoden, in welche Art und Weise die Parametrierung des Systems zu verändern ist.

In Abb. (4.1) ist die Gesamtstruktur des Identifikationsprozesses grafisch dargestellt.

4.2 Die Schnittstellen im Detail

4.2.1 Matlab Interface-Funktion

Die erste zu implementierende Schnittstelle ist die Matlab-Funktion welche FitLab anspricht, wenn es eine Simulation startet. Diese Funktion muss in der Matlab Skriptsprache geschrieben werden und die folgenden Parameter annehmen.

Der Parametervektor (PAR) In diesem Vektor sind alle innerhalb von FitLab definierten Parameter enthalten. Mit diesen muss die Simulation parametrisiert werden, bevor sie gestartet wird. Dies sind die, in dieser Arbeit allgemein mit $\underline{\Gamma}$ bezeichneten Parameter.

Der Zeitvektor (T) Dieser Vektor enthält alle diskreten Zeitpunkte für die FitLab einen Simulationswert erwartet. Da innerhalb von FitLab das zu simulierende Intervall beliebig gewählt werden kann, muss der Zeitvektor nicht zwangsläufig bei $T = 0$ beginnen.

Die Eingangsdaten (U) Diese Matrix besitzt je eine Spalte für jeden Eingangskanal, der der Simulation zur Verfügung gestellt wird. Die Länge jedes Eingangskanals ist gleich der Länge des Zeitvektors.

Der Offsetvektor (BIAS) Dieser Vektor enthält Offsetwerte für die ver-

schiedenen Aus- und Eingangskanäle. Damit lässt sich für Offsetbehaftete Realdaten die Höhe des Offsets zum Teil des Identifikationsprozesses machen.

Konkret muss die Matlab-Funktion die Form

$$[Y] = fName(PAR, BIAS, T, U), \quad (4.1)$$

mit der Matrix der Simulations-Ausgangsdaten Y und einem beliebigen Funktionsnamen $fName$ haben.

Anders als die Verläufe der Eingangsdaten, werden die Parameter nicht an Simulink weiter gereicht, sondern direkt dem CarSim-Solver zur Verfügung gestellt.

Wie vorher erwähnt, liest der Solver bei Simulationsstart eine Parameterdatei ein (siehe Kap. 2), in welcher die komplette Parametrierung für das zu simulieren-

de Fahrzeug zu finden ist. Die ursprünglich durch CarSim erstellte Parameterdatei soll im folgenden als `default.par` bezeichnet werden, da sie die Standardwerte für alle Parameter enthält.

Um nun eine Modifikation einiger dieser Werte zu bewirken, erstellt die Matlab-Funktion eine eigene Parameterdatei, `modification.par`, in welchem alle veränderten Parameter hinterlegt werden. Damit diese modifizierten Werte vom Solver berücksichtigt werden, wird am Ende der `default.par` über eine 'Include'-Anweisung die `modification.par` mit eingebunden. Das entsprechende Schlüsselwort hierfür heisst 'PARSFILE'.

```
#Originalinhalt der Datei
.
.
PARSFILE modification.par
END
```

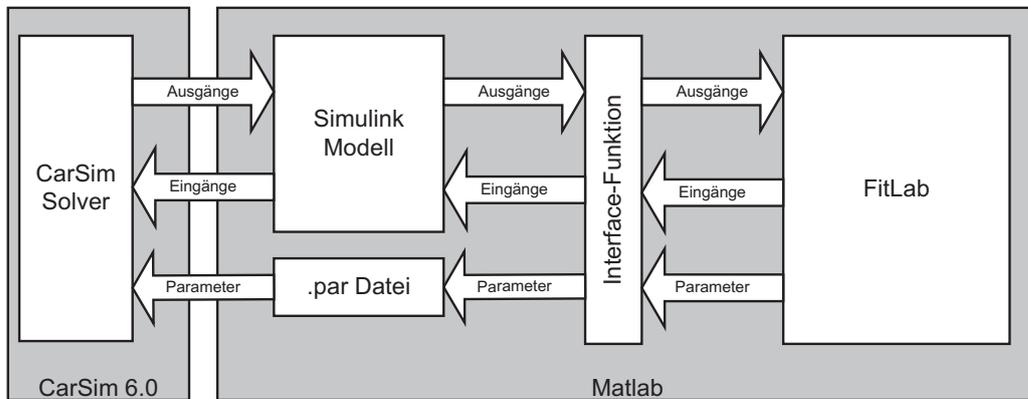


Abbildung 4.2: Kommunikationspfade innerhalb des Gesamtsystems. Die Matlab Interface-Funktion schreibt die Parameter in die .par Datei und leitet die Eingangsdaten der Simulation an Simulink weiter. Simulink ruft den CarSim-Solver auf und stellt diesem die Eingangsdaten zur Verfügung. Der Solver liest dabei die Parameter aus der .par Konfigurationsdatei.

Beim Einlesen der `.par` Datei findet der Solver die modifizierten Parameter dann zweimal, einmal in der original Konfiguration und einmal in der `modification.par`. Da letztere aber zeitlich nach den Originaldaten eingelesen wird, werden die ursprünglichen Werte aus der `default.par` überschrieben. Die gewünschte Modifikation wurde herbei geführt.

Worauf insbesondere bei den Ausgangsverläufen der Simulation geachtet werden muss, ist eine Kontrolle der Länge der Ausgangsvektoren.

Bei ungeschickter Parametrierung von CarSim passiert es des Öfteren, das aufgrund von numerischen Instabilitäten die Simulation abbricht. In diesem Fall gibt Simulink alle Ausgangsdaten bis zum Abbruch der Simulation zurück. Werden an FitLab diese zu kurzen Ausgangsvektoren zurück gegeben, bricht der Identifikationsprozess mit einer Fehlermeldung ab.

Lösung ist in Fall zu kurzer Ausgangsvektoren ein Ersetzen dieser, durch künstlich generierte Vektoren der richtigen Länge. Der Wert der Ausgänge wird hierbei auf einen beliebig hohen Wert gesetzt, der ausreichend größer sein muss, als mögliche Simulationsergebnisse. Auf diese Weise wird die Kostenfunktion in FitLab registrieren, das die fehlerhafte Parametrierung zu einer massiven Verschlechterung der Simulationsergebnisse im Vergleich zu den Realdaten geführt hat. Ein weiterer Optimierungsversuch in diese Richtung des Lösungsraumes wird damit verhindert.

Eine zweite, auf jeden Fall zu beachtende Besonderheit des CarSim-Solvers in die-

sem Zusammenhang ist, dass der Solver die `.par` Datei scheinbar nur vor Beginn jedes Simulationsdurchlaufes neu einliest, wenn das Simulink-Modell über den Aufruf

```
set_param(gcs ,  
          'SimulationCommand' ,  
          'Start' ) ,
```

gestartet wird. `gcs` steht für Get Current System und liefert ein Handle auf das aktuelle aktive Simulink-Modell.

Wird die Simulation beispielsweise über den Befehl `sim(...)` gestartet, wird die `.par` Datei einmalig eingelesen und eventuelle Änderungen an der Parametrierung in der `.par` Datei, zwischen zwei Simulationsdurchläufen, werden nicht wirksam.

Die Details der Implementierung der Interface-Funktion sind in Anhang B.1, Listing (B.1) zu finden.

4.2.2 Simulink Modell

Aufgabe des Simulinkmodells ist es, den CarSim-Solver anzusprechen. Dies geschieht, indem der Solver als S-Function eingebunden wird.

S-Functions sind in Matlab Skriptsprache, C/C++ oder Fortran geschriebene, eigenständige Module, welche über einen vordefinierten Satz an Funktionen verfügen müssen.

Simulink ruft während der Simulation diese S-Function auf, und lässt sie für

bestimmte Eingangswerte, Ausgangswerte berechnen.

Auf diese Weise lassen sich in S-Functions eigene Systeme und Komponenten implementieren, mit allen Möglichkeiten, die einem eine Hochsprache wie C oder C++ bietet.

In diesem Fall ist der CarSim-Solver als S-Function in einer Dll implementiert. Der Solver liest, wenn er initialisiert wird (Was vor Beginn einer jeden Simulation geschieht), eine .par Datei mit den Fahrzeug- und Simulationsparametern ein.

In der .par Datei ist unter anderem auch definiert, welche Eingangs und Ausgangsgrößen das Modell besitzt. Diese Eingangsdaten müssen der S-Function zur Laufzeit zur Verfügung gestellt und die Ausgangsdaten entgegengenommen werden.

Neben dem reinen Umleiten der Ein- und Ausgangsdaten sind in dem Simulink-Modell auch einige zur Berechnung bestimmter Eingangsdaten für die S-Function implementiert. Auf die Details der Berechnungen wird in Kapitel 6 eingegangen.

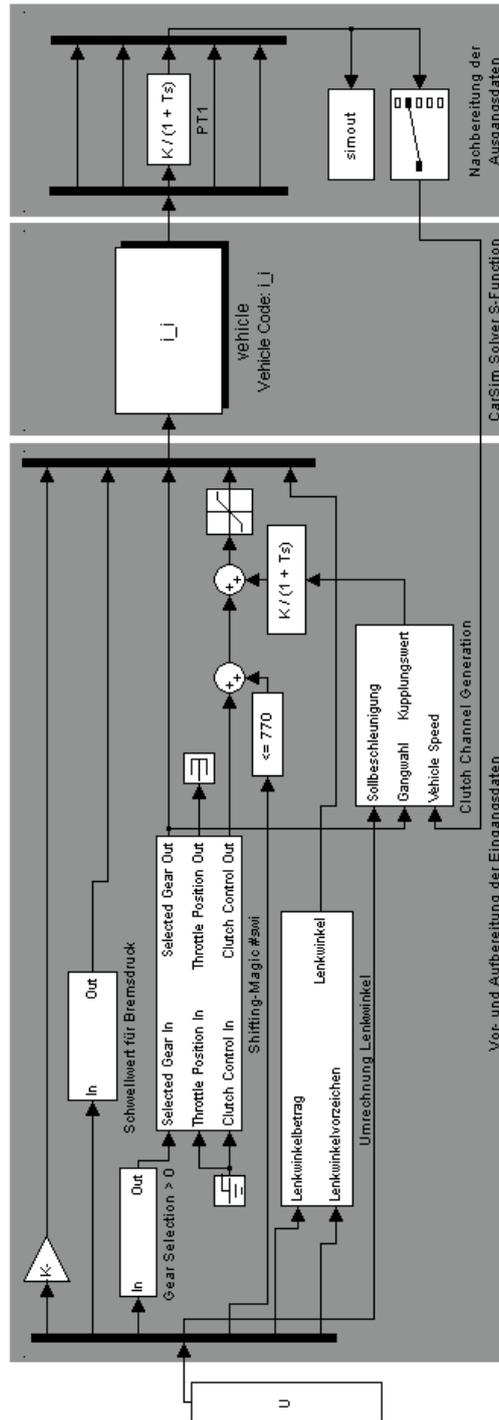


Abbildung 4.3: Das Simulink-Modell. Es setzt sich aus drei Teilbereichen zusammen. Der Datenvorbereitung, der Berechnung der Ausgangsdaten und der Nachbereitung der Ausgangsdaten.

KAPITEL 5

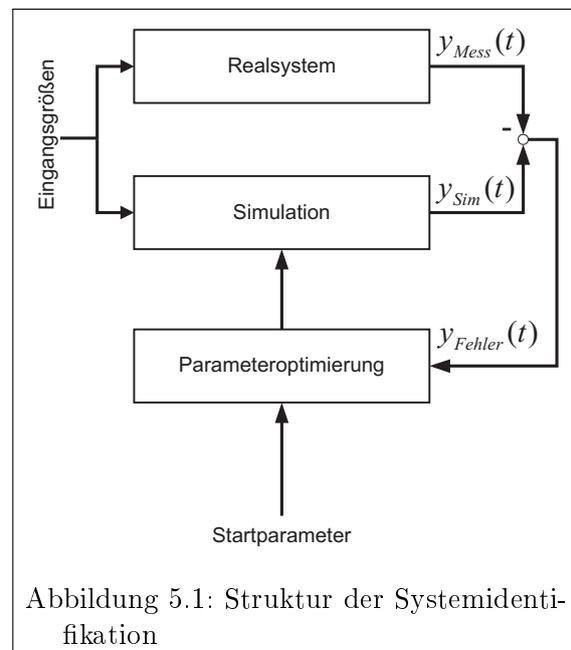
Sensibilitätsanalyse Fahrdynamik

5.1 Einleitung

Nach Kapitel 1.2 besteht die Grundidee des Systemidentifikationsprozesses in der sequentiellen Optimierung der beteiligten Parameter. Es liegt jedoch nahe, dass das Ergebnis der Systemidentifikation signifikant von der Reihenfolge der Optimierung der Parameter abhängt, wenn man annimmt, dass die verschiedenen Parameter einen verschieden starken Einfluss auf die Ausgangswerte des Systemmodells haben. Es ist also vor der eigentlichen Systemidentifikation eine Analyse der beteiligten Parameter vorzunehmen, mit dem Ziel, ein Gefühl dafür zu bekommen, wie sensibel der Ausgang des Systemmodells auf eine Variation der Parameter reagiert. Diese Erkenntnisse kann man später nutzen, um das Systemmodell als erstes nach jenen Parametern zu optimieren, welche einen besonders großen Einfluss auf das Ausgangssignal haben und welche dazu geeignet sind, den Signalfehler zu minimieren. Die verbleibenden Parameter sind dann dazu geeignet, eine Feineinstellung der Ausgangssignale vor zu nehmen.

Dies soll nachfolgend behandelt werden.

Anzumerken ist, dass Parameternamen, welche direkt aus CarSim übernommen worden sind, im folgenden Kapitel mit ihrem englischen Originalnamen benannt werden, um Unklarheiten zu vermeiden.



5.2 Parameterauswahl

5.2.1 Bekannte Parameter

Ein CarSim-Modell verfügt über 100 bis 150 verschiedene Parameter. Der mit einer Optimierung aller Parameter verbundene Rechenaufwand, wäre nicht zu handhaben. Daher liegt es nahe, nur eine Auswahl an Parametern zu behandeln und die verbleibenden Parameter auf fahrzeugtypische Werte festzulegen.

Es können einige Parameter von vorne herein für die Optimierung ausgeschlossen werden, da im Vorfeld wichtige Kenngrößen, durch Messungen am Fahrzeug und aus Herstellerangaben, ermittelt worden sind. Eine Liste dieser Größen ist in Tab. (5.1) zu finden.

Parameter	Wert
Radstand	2767.5 mm
Position des Schwerpunkt	x=1285 mm y=37 mm
Fahrzeugmasse	1889.5 kg
Übersetzungsverhältnis des Getriebes:	
1. Gang	3.500
2. Gang	1.944
3. Gang	1.360
4. Gang	1.034
5. Gang	0.844
Radgeometrie	215/55 R 16 Y
Stirnfläche	2.2 m ²

Tabelle 5.1: Bekannte Parameter. Die Werte der Parameter sind Anhang C entnommen worden.

Diese Größen sind nicht Teil des Identifikations-Prozesses und werden für die Sensibilitätsanalyse konstant gehalten.

5.2.2 Unbekannte Parameter

Die Auswahl der für die Optimierung vorgesehen Parametern ist in Tab. (5.2) zu finden.

Bei den mit einem '*' gekennzeichneten Größen, liegen keine Einzelparameter als Eingang für CarSim vor, sondern Kennlinien oder Kennlinienfelder. Der in diesem Fall verfolgte Ansatz ist die Erstellung einer normierten Kennlinie und eine Neuskalierung dieser mithilfe der Größen aus Tab. (5.2). Es werden also nicht die Formen der Kennlinien variiert, sondern die Wertebereiche über die sich die Kennlinien erstrecken.

Die Parameter aus Tab. (5.2) werden im Anschluss noch nach ihrer Zugehörigkeit zu den Bereichen

1. Längsdynamik
2. Querdynamik

aufgeteilt.

Dies erlaubt uns später durch eine geschickte Wahl der Fahrmanöver für die Sensibilitätsanalyse gezielt diese Teilbereiche der Fahrzeugdynamik anzusprechen und den Einfluss der zugeordneten Parameter zu ermitteln, ohne dass die

Kategorie	Komponente	Parameter
Front Wheel Drive	Engine	Motor Torque*
		Maximum speed*
Brake System	Transmission	Idle speed
	Front Wheel Torque	Efficiencies
	Rear Wheel Torque	Brake Torque vs. Brake Cylinder Pressure*
Steering System	Steer Compliance	Brake Torque vs. Brake Cylinder Pressure*
	Steering kinematics	Steer angle of wheel at ground vs. kingpin moment*
Front/Rear Compliance	Spring characteristics	Gearbox out vs. Steering at ground*
Tire	Lateral Tire-Force	Spring force vs. compression*
	Aligning Moment	Tire force vs. Slip angle*
		Aligning moment vs. Slip angle*

Tabelle 5.2: Parameterliste Systemidentifikation (*: Diese Größen sind in CarSim durch nichtlineare Kennlinien oder Kennlinienfelder hinterlegt)

Menge der Beteiligten Größen unübersichtlich groß wird.

die Systemparameter werden vereinfachend für die nachfolgende Betrachtung zu der Menge der Systemparameter Φ

$$\Phi = \{\Gamma_1, \dots, \Gamma_m\}$$

zusammengefasst.

5.3 Methode der Parametervariation

Die Sensibilitätsanalyse der einzelnen Parameter von Φ verläuft hierbei wie folgt:

Wie genau bei der Sensibilitätsanalyse vorgegangen werden soll, wird nachfolgend behandelt.

- (i) Festlegung der Variationschritte $i \in \mathbb{N}^*$, also der Anzahl an Veränderungen den die Systemparameter pro Betrachtung unterworfen werden sollen.

Die Ausgangsgrößen des Systemmodells sind abhängig von der Zeit t , dem Eingangsvektor \underline{u} und den Systemparametern $\Gamma_1, \dots, \Gamma_m \in \mathbb{R}$, $m \in \mathbb{N}^*$

- (ii) Auswahl von $1 \leq k \leq m$, $k \in \mathbb{N}^*$ Systemparametern $\Gamma_{\Delta k} \in \Phi$, welche bei der aktuellen Betrachtung variiert werden sollen.

$$y = F(t, \underline{u}, \Gamma_1, \dots, \Gamma_m)$$

$$\Phi_{\Delta} = \{\Gamma_{\Delta 1}, \dots, \Gamma_{\Delta k}\}$$

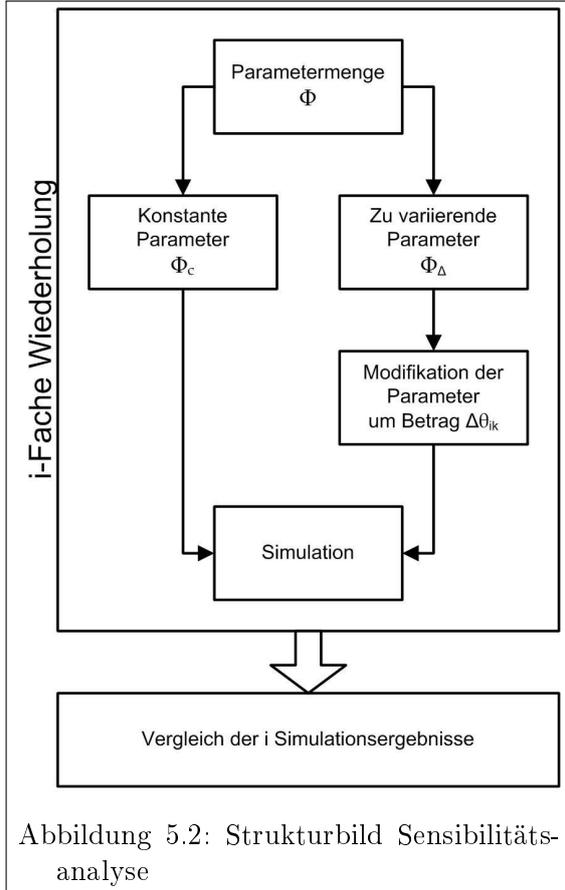


Abbildung 5.2: Strukturbild Sensibilitätsanalyse

Es verbleiben $j = m - k, j \in \mathbb{N}^*$ konstant gehaltene Systemparameter

$$\Phi_c = \{\Gamma_{c1}, \dots, \Gamma_{cj}\} = \Phi \setminus \Phi_\Delta$$

- (iii) i -fache Variation der Elemente von Φ_Δ .

$$\Phi_\Delta^1 = \{\Gamma_{\Delta 1} + \Delta\Gamma_{11}, \dots, \Gamma_{\Delta k} + \Delta\Gamma_{k1}\}$$

⋮

$$\Phi_\Delta^i = \{\Gamma_{\Delta 1} + \Delta\Gamma_{1i}, \dots, \Gamma_{\Delta k} + \Delta\Gamma_{ki}\}$$

$\Delta\Gamma_{ki} \in \mathbb{R}$ sind hierbei beliebig wählbare Werte.

- (iv) Mit den konstant gehaltenen Systemparametern Φ_c und den i Sätzen

an modifizierten Parametern Φ_Δ^i ergeben sich i Gesamtparametersätze

$$\Phi_1 = \Phi_c \cup \Phi_\Delta^1$$

⋮

$$\Phi_i = \Phi_c \cup \Phi_\Delta^i$$

- (v) Für diese i Parametersätze können nun i verschiedene Zeitverläufe von

$$F_i(t, \underline{u}, \Gamma_{c1}, \dots, \Gamma_{cj}, \Gamma_{\Delta 1}, \dots, \Gamma_{\Delta k})$$

berechnet und mit einander verglichen werden. Über den Vergleich gewinnt man Rückschlüsse über den Einfluss der in (ii) gewählten und in (iii) variierten Parametern.

Die Gewichtung der verschiedenen Parametern geschieht hierbei nicht über ein Formalismus, da es schwer ist durch einen solchen, einzelne Effekte zu bewerten, die durchaus in der Festlegung der Reihenfolge der Parameteridentifikationen eine Rolle spielen können.

Statt dessen wird eine individuelle Betrachtung der Einzelkurven vorgenommen und nach Stärke des Einflusses des betroffenen Parameters und eventueller Besonderheiten bei bestimmten Phasen des Fahrmanövers sortiert.

5.4 Längsdynamik

5.4.1 Das Fahrmanöver

Für die Sensibilitätsanalyse der Längsdynamik ist als erstes ein geeignetes Fahrmanöver zu wählen, welches nach Möglichkeit nur Komponenten der Fahrzeuglängsdynamik anregt und die Horizontaldynamik unberührt lässt. Es bietet sich dem zu Folge ein reiner Beschleunigungs- und Verzögerungsversuch an.

Hier wird bei konstanten Lenkwinkel ($\delta = 0$) unter Volllast beschleunigt. Der selektierte Gang wird nach und nach erhöht, wobei mit jeder Änderung des aktuellen Ganges das Gaspedal kurzfristig gegen Null moduliert wird.

Die genauen zeitlichen Verläufe der Eingangsgößen für das gewählte Fahrmanöver sind nachfolgend in Abb. (5.3) bis Abb. (5.5) zu sehen. Diese sind für die Sensibilitätsanalyse künstlich generiert worden, sie sind also kein Teil der im Vorfeld aufgenommenen Realdaten.

5.4.2 Die Parameter

Die für die Längsdynamik relevanten Komponenten und ihre Parameter nach Tab. (5.2) sind in Tab. (5.3) aufgeführt und im Folgenden kurz beschrieben:

Motor Torque Das maximale Moment, das der Motor abzugeben in der Lage ist. Das Motormoment ist eine Funktion der Drosselklappenstellung und

Komponente	Parameter
Engine	Motor Torque Maximum speed Idle speed
Transmission	Efficiencies
Front Wheel Torque	Brake Torque vs. Brake Cylinder Pressure
Rear Wheel Torque	Brake Torque vs. Brake Cylinder Pressure

Tabelle 5.3: Komponenten der Längsdynamik

der aktuellen Drehzahl. Vorgegeben wird der Momentenverlauf über diese beiden Parameter als Kennlinienfeld.

Maximum speed Die maximale Drehzahl des Motors.

Idle speed Die Leerlaufdrehzahl des Motors

Transmission Efficiencies Der Wirkungsgrad der Getriebeübersetzung. Dieser Wert ist für alle gewählten Gänge gleich.

Front Brake Torque Das Bremsmoment pro Bremsdruck an den vorderen Reifen.

Rear Brake Torque Das Bremsmoment pro Bremsdruck an den hinteren Reifen.

Bei den Motor-Charakteristika ergibt sich prinzipiell das Problem, dass der Motor nicht durch ein frei parametrierbares Modell zur Verfügung steht, sondern durch

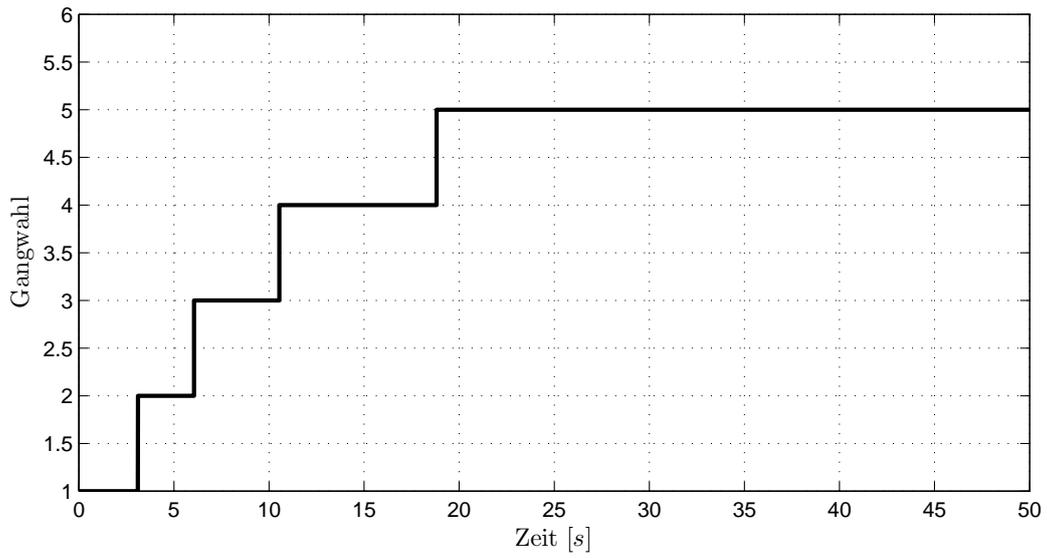


Abbildung 5.3: Zeitverlauf Gangwahl

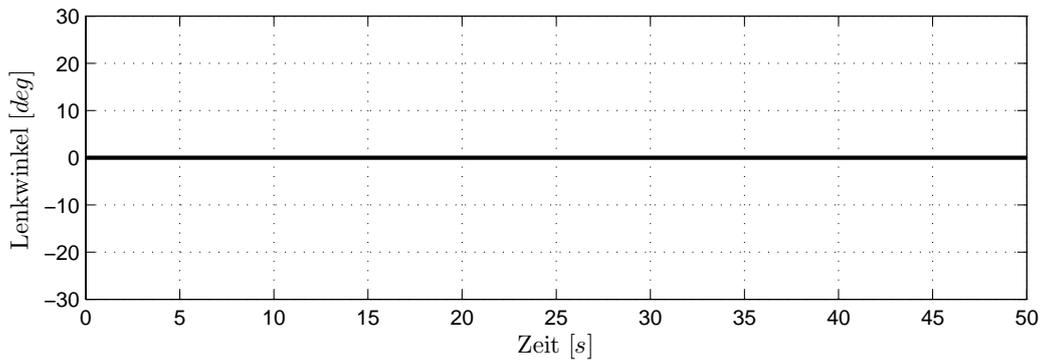


Abbildung 5.4: Zeitverlauf Lenkwinkel

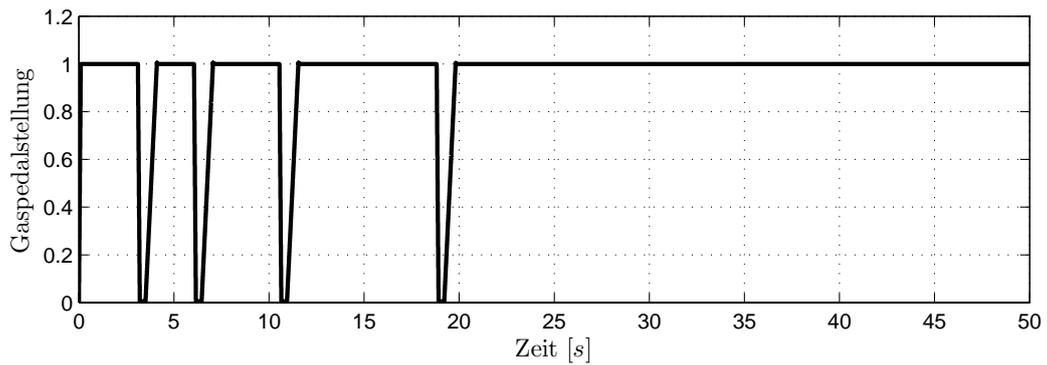


Abbildung 5.5: Zeitverlauf Gaspedal

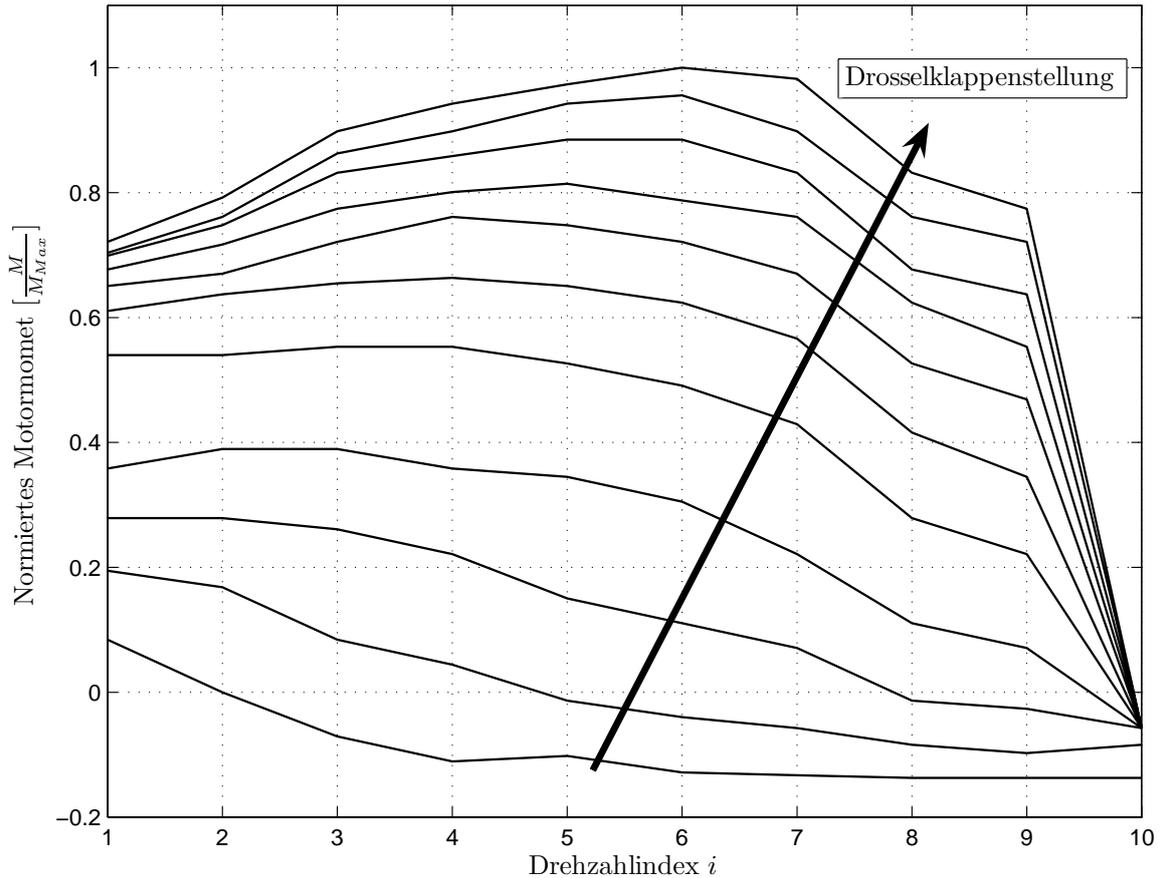


Abbildung 5.6: Normiertes Motorkennlinienfeld. Berechnung der Drehzahlen für die x-Achse erfolgt mit Hilfe des Drehzahlindex i über Gl. (5.2). Quelle für Form des Kennlinienfeldes: CarSim 6.0

ein Kennlinienfeld, in welchem für verschiedene Drosselklappenstellungen und Drehzahlen die aktuell gelieferten Drehmomente des Motors aufgetragen sind. Dieses Kennlinienfeld kann je nach Auflösung mehrere hundert Einzeldaten enthalten, deren gezielte und systematische Variation einen nicht zu realisierenden Aufwand darstellt.

Um hier trotzdem eine Variation zu ermöglichen, wurde ein normiertes Kennlinienfeld \underline{M}_N vorbereitet, das über die Wahl eines maximalen Drehmoment M_{Max} , einer maximalen Drehzahl n_{max}

und einer Leerlaufdrehzahl n_0 skaliert und verwendbar gemacht wird.

Die Berechnung des neu skalierten Kennlinienfeldes des Motor \underline{M} und dessen Formatierung in ein von CarSim lesbares Format, erfolgt hierbei nach

$$\underline{M} = \begin{bmatrix} 0 & \underline{N} \\ \underline{D} & \underline{M}_N \cdot M_{Max} \end{bmatrix} \quad (5.1)$$

Hier ist \underline{M}_N eine $(m \times k)$ Matrix, die Drehzahlmatrix \underline{N} ist $(1 \times k)$ und die Matrix der Drosselklappenstellungen \underline{D} ist $(m \times 1)$.

Die Elemente der Drehzahlmatrix werden hierbei für den Matrixindex i (siehe auch Tab. (5.6)) bestimmt nach

$$\underline{N} = (n_i) = (n_0 + (i - 1) \cdot \Delta s) \quad (5.2)$$

$$\Delta s = \frac{n_{max} - n_0}{k - 1}. \quad (5.3)$$

Dies ist eine einfache Diskretisierung Abbildung Intervalls (n_0, n_{max}) , auf die Matrix \underline{N}

Die Matrix der Drosselklappenstellung ist konstant und gegeben mit

$$\underline{D} = (d_i) = ((i - 1) \cdot \frac{1}{k}), \quad (5.4)$$

was eine diskrete Abbildung des Intervalls $(0,1)$ auf die Matrix \underline{D} ist.

Das für die Sensibilitätsanalyse verwendete, normierte Motorkennlinienfeld ist in Abb. (5.6) dargestellt. Die Form des Kennlinienfelder ist hierbei den mitgelieferten Motormodellen von CarSim entnommen.

Die verbleibenden Parameter können direkt an CarSim übergeben werden.

5.4.3 Die Parametervariation

Mit den Parametern für die Längsdynamik aus Tab. (5.3) ergeben sich eine Vielzahl von Kombinationsmöglichkeiten. Da die Simulationen im einzelnen recht rechenaufwändig sind, werden nicht alle Kombinationen für die Untersuchungen tatsächlich betrachtet, sondern nur

eine sinnvolle Auswahl. Die Kombinationsmöglichkeiten, welche für die Sensibilitätsanalyse heran gezogen werden sind in Tab. (5.4) zu finden. Jede Zeile steht hierbei für eine Serie von Simulationsdurchläufen, in welchen die in den Spalten gekennzeichneten Größen variiert werden.

Prinzipiell verläuft der Vorgang der Parametervariation nach dem in Kapitel (5.3) beschriebenen Schema.

Die Variation der Parametervektoren erfolgt $i = 4$ mal. Die $k \cdot i$ Skalare aus Unterkapitel (5.3), Punkt (iii), $\Delta\Gamma_{ki}$ wurde so gewählt, dass

$$\Delta\Gamma_{k1} \leq \dots \leq \Delta\Gamma_{ki},$$

gilt. Sie werden also zyklisch erhöht.

Die Werte für die Parameter wurden aus CarSim entnommen. Auf diese Weise ist sicher gestellt, das fahrzeugtypische Werte verwendet werden. Des Weiteren wurden die Parameter für jeden Variationsschritt wie folgt modifiziert:

1. Variation: 60% des Startwert
2. Variation: 80% des Startwert
3. Variation: 100% des Startwert
4. Variation: 120% des Startwert

Folgende Werte wurden hierbei verwendet:

Nr.	max. Torq.	max. Speed	idle Speed	Eff.	Fr. Br. Torq.	Re. Br. Torq.
01	X					
02	X	X				
03	X	X		X		
04			X			
05		X		X		
06				X		
07					X	
08						X
09					X	X

Tabelle 5.4: Variationskombinationen Längsdynamik

Parameter	Startwert
Max. Engine Torque	300Nm
Max. Engine Speed	$6200 \frac{1}{min}$
Engine Idle Speed	$600 \frac{1}{min}$
Gear Efficiency	0.85
Front Brake Torque	150Nm
Rear Brake Torque	70Nm

Tabelle 5.5: Die Startwerte

Max. Motor Speed: Erhöhung des Parameters bewirkt eine Erhöhung der Fahrzeugendgeschwindigkeit.

Motor Idle Speed: Vernachlässigbarer Einfluss für das dynamische Verhalten des Systems. Lediglich im Leerlauf von Bedeutung.

Transm. Eff.: Erhöhung bewirkt eine Parallelverschiebung der Beschleunigungskurve in positive Richtung.

Front Brk. Torq.: Parallelverschiebung der Beschleunigungskurve im Bereich des Bremsmanövers.

Rear Brk. Torq.: Parallelverschiebung der Beschleunigungskurve im Bereich des Bremsmanöver. Für ungünstige Verhältnisse zu 'Front Brk. Torq.' kann das Fahrzeug während des Bremsmanövers instabil werden. Hier ist später Vorsicht geboten.

5.4.4 Die Ergebnisse

Sämtliche, bei der Sensibilitätsanalyse der Längsdynamik angefallenen Zeitverläufe, sind in Anhang (A.1) zu finden.

Prinzipiell findet man anhand der Abbildungen (A.1) bis (A.10) folgende Eigenschaften für die Parameter aus Tab. (5.3).

Max. Motor Torque: Erhöhung bewirkt eine schnellere Beschleunigung im Drehzahl-Anfangsbereich. Fahrzeug strebt schneller gegen Endgeschwindigkeit.

Die Festlegung der Reihenfolge der Parameteridentifikation erfolgt nun nach folgenden Kriterien:

1. Stärke des Einflusses
2. Besondere Effekte oder Eigenheiten

Mit Hilfe dieser Kriterien und der Betrachtung der Ausgangsverläufe (A.1) bis (A.10) gelangt man so zu folgender Reihenfolge für die Parameteroptimierung.

1. **Max. Motor Speed:** Identifikation des Drehzahlbereiches, welcher für die Nachbildung der Dynamik relevant ist.
2. **Max. Motor Torque** Skalierung des Drehmoment des Motors.
3. **Transm. Eff.:** Aufgrund des recht geringen Einflusses dieses Parameters wird dieser für Feineinstellungen verwendet.
4. **Motor Idle Speed:** Aufgrund des recht geringen Einflusses dieses Parameters wird dieser für Feineinstellungen verwendet.
5. **Front Brk. Torq.:** Dieser Parameter lässt sich getrennt von den anderen Parametern in einem Bremsmanöver identifizieren.
6. **Rear Brk. Torq.:** Dieser Parameter lässt sich getrennt von den anderen Parametern in einem Bremsmanöver identifizieren.

5.5 Querdynamik

5.5.1 Das Fahrmanöver

Das Fahrmanöver für die Sensibilitätsanalyse der Querdynamik hat prinzipiell die Anforderung zu erfüllen, dass die Elemente der Fahrzeug-Querdynamik möglichst stark und die Elemente der Fahrzeug-Längsdynamik möglichst gering angeregt werden.

Letzteres wird durch Halten einer konstanten Geschwindigkeit mit Hilfe des internen CarSim-Geschwindigkeitsreglers erreicht. Konkret wurde für alle Manöver eine Richtgeschwindigkeit von $60 \frac{km}{h}$ gewählt.

Die Anregung der Querdynamik erfolgt als erstes mit einem Lenkwinkelsprung auf $\delta = 60^\circ$ bei $t = 10s$ und einen Rücksprung auf $\delta = 0^\circ$ bei $t = 20s$. Ab $t = 30s$ erfolgt eine sinusförmige Anregung des Lenkrad mit einer Amplitude von $\delta = 90^\circ$ und einer Frequenz von $f = 0.25Hz$.

Sämtliche Eingangsgrößen und ihre Zeitverläufe sind noch einmal in Abb. (5.7) bis (5.9) zu finden. Der Verlauf von Abb. (5.9) ist hierbei das Ergebnis der vorher angesprochenen Geschwindigkeitsregelung.

5.5.2 Die Parameter

Die für die Längsdynamik relevanten Komponenten und ihre Parameter nach Tab. (5.2) sind in Tab. (5.6) zusammengefasst und im Folgenden erläutert.

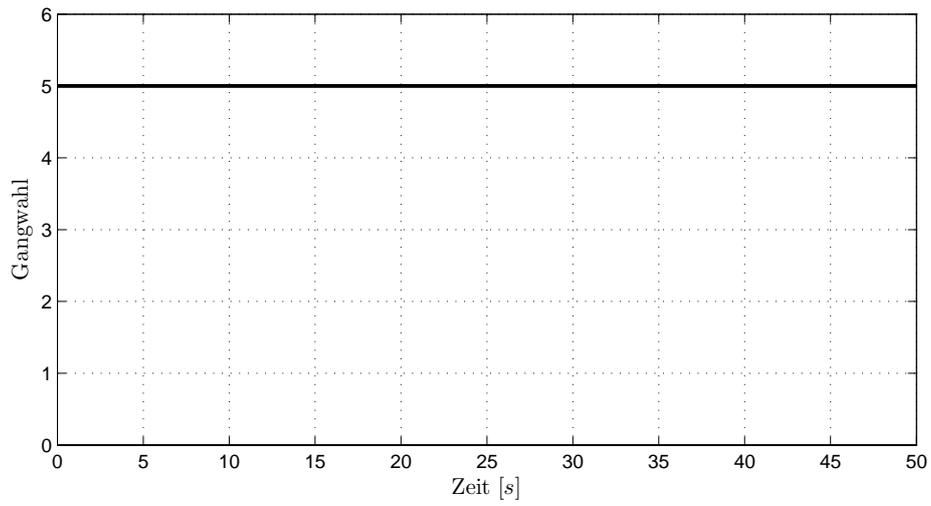


Abbildung 5.7: Zeitverlauf Gangwahl

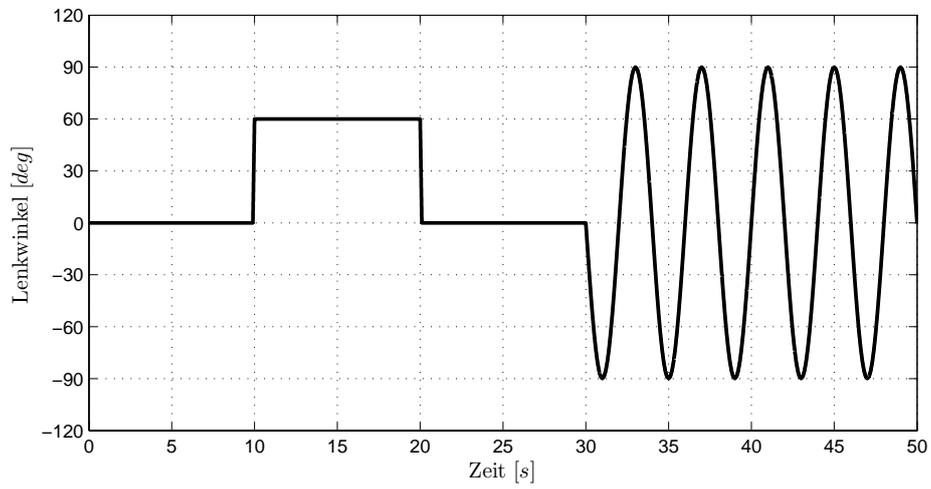


Abbildung 5.8: Zeitverlauf Lenkwinkel

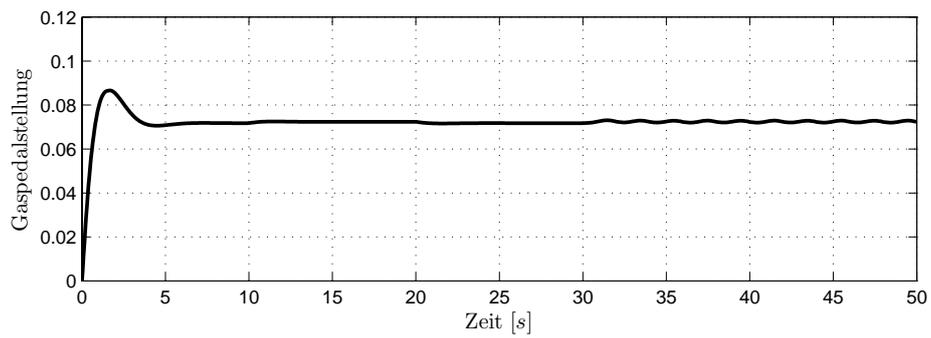


Abbildung 5.9: Zeitverlauf Gaspedal

Komponente	Parameter
Steering System	Steer Compliance Steering Kinematics
Front/Rear Compliance	Spring Characteristics
Tire	Lateral Tire-Force Aligning moment

Tabelle 5.6: Komponenten der Querdy-
namik

der Simulation erst neu skaliert werden müssen.

Die Originalkennlinien stammen hierbei von den mitgelieferten Komponenten aus CarSim 6.0. Diese wurden normiert und werden vor Gebrauch in der Sensibilitätsanalyse neu skaliert.

Diese haben eine Form nach Abb. (5.10) bis Abb. (5.12)

Es folgt eine Beschreibung der für die Variation gewählten Parameter:

Steering Kinematics Der Lenkwinkel des Reifen auf der Fahrbahn bei einem Lenkgetriebeausgang von 23° .

Steer Compliance Nachgiebigkeit der Lenkung unter Einwirkung eines Momentes an der Lenkachse.

Spring Characteristics Die Rückstellkraft der Federn der Radaufhängung in Abhängigkeit des Federweges. Dies ist eine nicht-lineare Kennlinie.

Lateral Tire-Force Reifen-Seitenkraft in Abhängigkeit vom Schwimmwinkel. Dies ist eine nicht-lineare Kennlinie.

Aligning Moment Rückstellmoment in Abhängigkeit vom Schwimmwinkel. Dies ist eine nicht-lineare Kurve Kennlinie.

Die letzten drei Parameter dieser Kurzbeschreibung sind hierbei nicht-lineare Kennlinien, welche in normierter Form vorgegeben werden und vor Gebrauch in

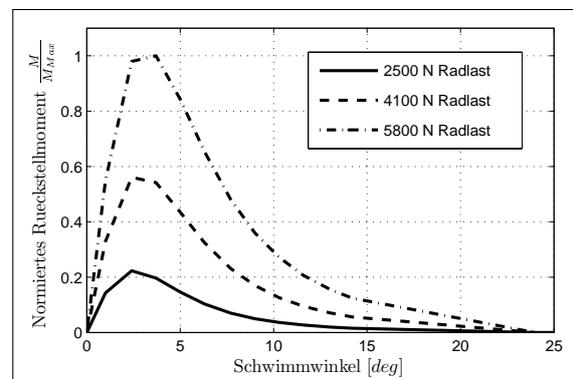


Abbildung 5.10: Normierte Kurve der Reifenrückstellkraft. Quelle für Form der Kennlinie: CarSim 6.0

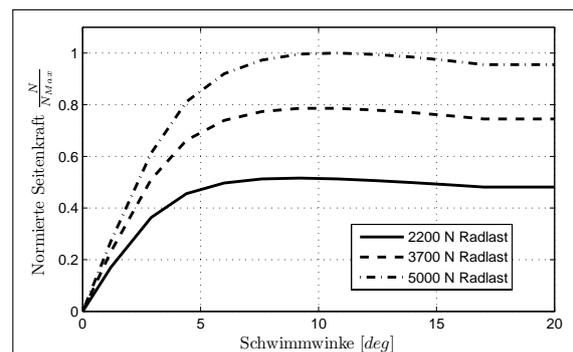


Abbildung 5.11: Normierte Kurve der Reifenseitenkraft. Quelle für Form der Kennlinie: CarSim 6.0

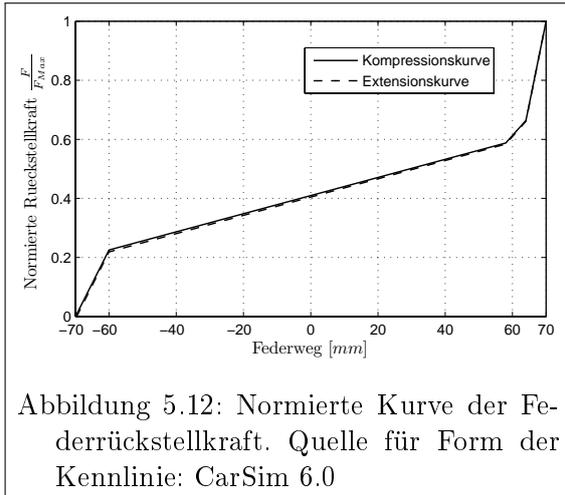


Abbildung 5.12: Normierte Kurve der Federrückstellkraft. Quelle für Form der Kennlinie: CarSim 6.0

5.5.3 Die Parametervariation

Die Parameterkombinationen, die während der Sensibilitätsanalyse variiert wurden sind in Tab. (5.8) zu finden. Jede Zeile steht hierbei für eine Serie von Simulationsdurchläufen, in welchen die in den Spalten gekennzeichneten Größen variiert wurden.

Prinzipiell wird bei der Parametervariation, wie in Kapitel (5.3) beschrieben, vorgegangen.

Die Variation der Parametervektoren erfolgt $i = 4$ mal. Für die $k \cdot i$ Skalare aus Unterkapitel (5.3), Punkt (iii), $\Delta\Gamma_{ki}$ wurde gewählt

$$\Delta\Gamma_{k1} \leq \dots \leq \Delta\Gamma_{ki},$$

sie werden also auch hier zyklisch erhöht.

Die Kennlinie der Federrückstellkraft der Radaufhängung ('Spring Characteristics') setzt sich aufgrund ihres Hystereseverhaltens aus zwei Einzelkennlinien zusammen,

die getrennt parametrisiert werden können. Beide Einzelkennlinien werden dabei über die Wahl einer maximalen Kraft skaliert und der Simulation zur Verfügung gestellt. Die beiden Einzelparameter sind 'Spring Force Extension' für die sich ausdehnende Feder und 'Spring Force Compression' für die sich stauchende Feder.

Für diese Parameter muss darauf geachtet werden, das gilt

$$f_{Ext}(x) < f_{Comp}(x) \quad \forall x \in \mathbb{R}, \quad (5.5)$$

da CarSim ansonsten einen Fehler meldet und abbricht.

Da diese beiden Parameter prinzipiell zusammen gehören werden sie bei der Sensibilitätsanalyse stets gleichzeitig und im selben Verhältnis variiert.

Die Werte für die Parameter wurden aus CarSim entnommen. Auf diese Weise ist sicher gestellt, dass fahrzeugtypische Werte verwendet werden. Des Weiteren wurden die Parameter für jeden Variationschritt, analog zur Längsdynamik, wie folgt modifiziert:

1. Variation: 60% des Startwert
2. Variation: 80% des Startwert
3. Variation: 100% des Startwert
4. Variation: 120% des Startwert

Folgende Werte wurden hierbei verwendet:

Parameter	Startwert
Steer at Ground	26.0°
Steer Compliance	0.01
Max. Tire Alig. Moment	97.51
Max. Tire Lat. Force	4498.1
Susp. Ext. max Force	9691.0
Susp. Compr. max Force	9771

Tabelle 5.7: Die Startwerte

5.5.4 Die Ergebnisse

Sämtliche, bei der Sensibilitätsanalyse der Querdynamik angefallenen Zeitverläufe, sind in Anhang (A.2) zu finden.

Mit Hilfe der Ausgangsverläufe (A.11) bis (A.21) gelangt man so zu nachfolgender Reihenfolge für die Parameteroptimierung. Dabei wurden die Parameter ausschließlich nach der Reihenfolge der Stärke ihres Einflusses auf die beiden Ausgangssignale, dem Roll- und Gierwinkel, sortiert.

1. **Steer at Ground**
2. **Steer Compliance**
3. **Lateral Tire Force**
4. **Aligning Moment**
5. **Spring Force Extension + Compression**

- A** Steering at Ground bei 23° Gearbox output [deg]
- B** Steer Compliance bei ± 1 Nm Kingpin Moment [deg]
- C** Max. Aligning Moment (Skalierung einer Kennlinie) [Nm]
- D** Max. Lateral Tire Force (Skalierung einer Kennlinie) [N]
- E** Max. Spring Force Extension (Skalierung einer Kennlinie) [N]
- F** Max. Spring Force Compression (Skalierung einer Kennlinie) [N]

Nr.	A	B	C	D	E	F
01	X					
02	X	X				
03	X	X	X	X	X	X
04		X				
05		X	X	X		
06		X			X	X
07			X	X		
08			X	X	X	X
09					X	X
10			X			
11				X		

Tabelle 5.8: Variationskombinationen Querdynamik

KAPITEL 6

Die Systemidentifikation

6.1 Das Fahrzeug

Gegenstand der Systemidentifikation ist das ViewCar. Ein Messfahrzeug des Instituts für Verkehrssystemtechnik des Deutschen Zentrums für Luft und Raumfahrt in Braunschweig. Dieses Fahrzeug dient der Analyse von Wahrnehmungsprozessen und Verhalten von Fahrern im Straßenverkehr.



Abbildung 6.1: Das ViewCar des Instituts für Verkehrssystemtechnik. Quelle: [Sun07b]

Die Eckdaten lauten wie folgt:

Modell: Audi 4B (A6)

Motor: 3.0L (Otto) V6

Reifen: Continental ContiWinter Contact

Reifengröße: Vorne 205/55 R16; Hinten 205/55 R16

Antriebsart: Vorderradantrieb

Max. Motorleistung: 162 kW (220 PS)
bei 6300 $\frac{1}{min}$

Max. Geschwindigkeit: 241 $\frac{km}{h}$



Abbildung 6.2: Das ViewCar des Instituts für Verkehrssystemtechnik. Quelle: [Sun07b]

6.2 Die Fahrmanöver

Im Rahmen der Versuche einer Systemidentifikation durch einen automatisierten Optimierungsprozess durch X. Sun in seiner Arbeit [Sun07b], wurde eine Auswahl von Fahrmanövern getroffen, für welche anschließend eine Reihe von Messdaten aufgenommen wurden.

Diese Messdaten werden in dieser Arbeit als Grundlage verwendet.

Die Versuche wurden auf dem VW-Testgelände in Ehra durchgeführt. Siehe Abb. (6.3).

Die genauen Manöverspezifikationen sind in Anhang C zu finden.



Abbildung 6.3: VW-Testgelände in Ehra.
Quelle: [Sun07b]

Folgende Fahrmanöver wurden dabei durchgeführt:

1. Stationäre Kreisfahrt nach ISO 4138 (Open Loop)
2. Doppelter Spurwechsel nach ISO 3888 (Teil 2)

3. Sinusförmige Lenkradanregung nach ISO 7401

4. Bremsen in der Kurve nach ISO 7595

Konkret wurde für folgende Fahrten Messungen gemacht (Quelle: [Sun07a]):

Stationäre Kreisfahrt: $R = 30.48\text{m}$

Richtung	Geschwindigkeit
Rechts	15 km/h
	25 km/h
	35 km/h
	45 km/h
Links	25 km/h
	50 km/h

Tabelle 6.1

Bremsen in der Kurve: $R = 30.48\text{m}$

Richtung	Geschw. und Bremsung
Rechts	47 km/h, leichte Bremsung
	47 km/h, mittlere Bremsung
	47 km/h, härtere Bremsung
	50 km/h, harte Bremsung
Links	25 km/h, leichte Bremsung
	50 km/h, mittlere Bremsung

Tabelle 6.2

Stationäre Kreisfahrt: $R = 68\text{m}$

Richtung	Geschwindigkeit
Links	30 km/h
	50 km/h
	60 km/h
	70 km/h
Rechts	30 km/h
	70 km/h
Tabelle 6.3	

Lenkwinkelsprung: 50 km/h

Richtung	Art
Links	22.5°
	45.0°
	67.5°
	90.0°
	112.5°
Rechts	22.5°
	45°
	67.5°
	90.0°
	112.5°
Tabelle 6.6	

Bremsen in der Kurve: $R = 68\text{m}$

Richtung	Geschwindigkeit
Links	70 km/h, leichte Bremsung
	70 km/h, mittlere Bremsung
	70 km/h, stärker als mittlere Bremsung
	harte Bremsung
Rechts	70 km/h, leichte Bremsung
	70 km/h, mittlere Bremsung
	70 km/h, harte Bremsung
Tabelle 6.4	

Sinuswedelfahrt : 80 km/h

Parameter	
Lenkwinkelamplitude so, dass $a_{Quer} = 4 \frac{m}{s^2}$	
Frequenz: 0.5 oder 1 Hz	
1. Einschlag nach rechts / links	
Gaspedalstellung = const	
Dauer: jeweils mindestens 3 x	
Tabelle 6.7	

Beschleunigen in der Kurve: $R = 68\text{m}$,
50 → 80 km/h

Richtung	Art
Links	langsam beschleunigen
	schneller beschleunigen
Tabelle 6.5	

Sinuswedelfahrt : 70 km/h

Parameter
Lenkwinkelamplitude: 150° ($a_{Quer} \approx 2 \frac{m}{s^2}$)
Frequenz: 0.2 Hz
1. Einschlag nach li
Gaspedalstellung = const
Dauer: jeweils mindestens 3 x
Pylonen ca. alle 50m ($20m/s * 5s / 2$)
Tabelle 6.8

Konstante Geschwindigkeit auf grader Strecke

Parameter
Geschwindigkeit: 90 km/h
Gangwahl: 4
Dauer: > 5s
Tabelle 6.9

Verzögerung auf grader Strecke

Parameter
Geschwindigkeit: 90 km/h \rightarrow 60 km/h
Gangwahl: 4
Bremspedal: 0 %
Tabelle 6.10

Beschleunigung auf grader Strecke

Parameter
Geschwindigkeit: 90 km/h \rightarrow 130 km/h
Gangwahl: 4
Gaspedal: 60 %
Tabelle 6.11

Bremsen auf grader Strecke

Parameter
Geschwindigkeit: 130 km/h \rightarrow 90 km/h
Gangwahl: 4
Bremspedal: 35 %
Tabelle 6.12

Verzögern auf grader Strecke

Parameter
Geschwindigkeit: 130 km/h \rightarrow 70 km/h
Gangwahl: 4
Gaspedal: 0 %
Bremspedal: 0 %
Tabelle 6.13

Verzögern auf grader Strecke

Parameter
Geschwindigkeit: 60 km/h → 20 km/h
Gangwahl: 4
Gaspedal: 0 %
Bremspedal: 0 %
Tabelle 6.14

Doppelter Spurwechsel nach ISO-3888

Parameter
Geschwindigkeiten: 50,55, 60, 65, kmh
2m vor Anfang Kupplung: 100 %
Dauer: 3 x bei v _{max} , sonst 1x
Tabelle 6.15

6.3 Die Rohdaten

Die im Rahmen der Arbeit von [Sun07b] aufgenommenen Messdaten entstammen zu einem Teil, direkt der standardgemäß installierten Fahrzeugelektronik und wurde über das Can-Bus System ausgelesen, zum anderen wurde für den Lenkwinkel und die Fahrzeugortung auf die zusätzlich installierte ViewCar Hardware zurück gegriffen.

Die durch die Messung zur Verfügung stehenden Eingangsgrößen sind:

1. **Lenkwinkel** in Grad
2. **Gaspedalstellung** in %

3. **Bremspedalstellung** in %

An relevanten Ausgangsdaten wurde aufgenommen:

1. **Motordrehzahl** [$\frac{1}{min}$]
2. **Geschwindigkeit** in X-, Y-, und Z-Richtung [$\frac{m}{s}$]
3. **Beschleunigung** in X-, Y-, und Z-Richtung [$\frac{m}{s^2}$]
4. **Winkel** um die X-, Y-, und Z-Achse [rad]
5. **Winkelgeschwindigkeit** um die X-, Y-, und Z-Achse [$\frac{rad}{s}$]

6.4 Generierung der Eingangssignale

In Kap. 6.3 wurden die während der Messung aufgenommenen Eingangssignale aufgelistet.

1. **Lenkwinkel** in Grad
2. **Gaspedalstellung** in %
3. **Bremspedalstellung** in %

Die vollständige Beschreibung der Eingangsdaten, beinhaltet für ein Fahrzeug mit manueller Schaltung jedoch sowohl die aktuelle Gangwahl, als auch die Betätigung der Kupplung.

Diese Daten sind nicht mit aufgenommen worden, lassen sich jedoch annäherungsweise durch gegebenen Verläufe rekonstruieren. Dies soll Thema dieses Unterkapitels sein.

Ist die Kupplung grade in einem schlupffreien Zustand, drehen also Ein- und Ausgangsschaft synchron, so ergibt sich die aktuelle Gangwahl eindeutig über das Übersetzungsverhältnis zwischen Motordrehzahl und Reifendrehzahl.

Die Motordrehzahl ist als Ausgangsgröße bekannt, die Reifendrehzahl ergibt sich mit der Kenntnis über den Reifenumfang, aus der aktuellen Fahrzeuggeschwindigkeit, welche ebenso als Messgröße vorliegt.

Der Reifenumfang ist durch die Reifenkennung gegeben (siehe Kap. 6.1).

Diese lautet für die im ViewCar verwendeten Reifen:

205/55 R16

Die Bedeutung der einzelnen Größen ist in Tab. (6.16) zu finden.

Der Reifendurchmesser gibt sich demnach in cm mit:

$$r_{Reifen} = \frac{l_{Breite} \cdot i_{HzB}}{10} + l_{innne} \cdot \frac{2.54cm}{Zoll} \quad (6.1)$$

Aus der Winkeldefinition in Rad

$$\varphi = \frac{s}{r} \quad (6.2)$$

Größe	Bedeutung
205/55 R16	die Reifenbreite (l_{Breite}) [mm]
205/ 55 R16	das Verhältnis Reifenhöhe zu Reifenbreite in % (i_{HzB})
205/55 R16	die Bauart
205/55 R16	der Reifeninnendurchmesser [Zoll] (l_{innen})

Tabelle 6.16: Beschreibung der Reifenkennungen

mit dem aufgespannten Kreisumfang s und dem Kreisradius r , ergibt sich nach einmaligen Ableiten nach der Zeit

$$\dot{\varphi} = \omega = \frac{\dot{s}}{r} = \frac{v}{r}, \quad (6.3)$$

mit der Umlaufgeschwindigkeit v .

Die Winkelgeschwindigkeit des Reifen während der Fahrt ist demnach, unter Vernachlässigung von dynamischen Reifendeformationen,

$$\omega_{Reifen} = \frac{v_{Fahrzeug}}{r_{Reifen}}. \quad (6.4)$$

Das aktuelle Übersetzungsverhältnis des Getriebes ergibt sich dann mit:

$$\begin{aligned} i_{Getriebe}(t) &= \frac{\omega_{Motor}(t)}{\omega_{Reifen}(t)} \\ &= \frac{\omega_{Motor}(t) \cdot r_{Reifen}(t)}{v_{Fahrzeug}(t)}. \end{aligned} \quad (6.5)$$

Nach Kap. 5.1, Tab. (5.1), sind die Übersetzungsverhältnisse des Getriebes aus den zum Fahrzeug gehörenden Datenblättern bereits bekannt.

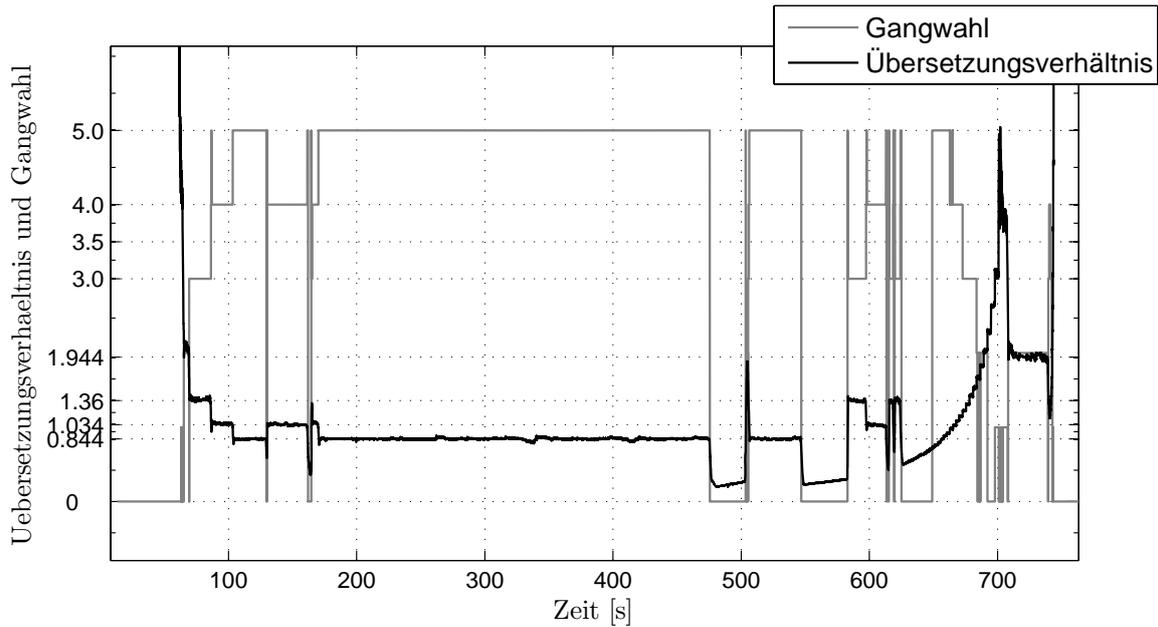


Abbildung 6.4: Beispielplot, Verlauf des Motor-, Reifendrehzahlverhältnisses und Zuordnung der Gänge für einen Längsdynamikversuch.

Gangwahl	Übersetzungsverhältnis
1	3.500
2	1.944
3	1.360
4	1.034
5	0.844

Tabelle 6.17: Übersetzungsverhältnisse des ViewCar-Getriebes. Quelle: [Aud04]

Mit diesen Übersetzungsverhältnissen, Tab. (6.17), lässt sich dann jedem $i_{\text{Getriebe}}(t)$ einen Gang zu ordnen.

Während der dynamischen Übergänge zwischen zwei Gängen, lässt sich die aktuelle Gangwahl nicht eindeutig bestimmen, da die Kupplung mit Schlupf oder komplett offen arbeitet.

Während des Übergangs zwischen zwei Gängen ist anzunehmen, dass der Fahrer erst die Kupplung betätigt, das Getriebe also vom Antriebsstrang trennt, dann das Übersetzungsverhältnis ändert und Anschließend die Kupplung langsam wieder kommen lässt.

Dies wurde bei der Abbildung $i_{\text{Getriebe}} \rightarrow \text{Gangwahl}$ so berücksichtigt, dass der Gang noch eine Weile gehalten wird, auch wenn das berechnete Übersetzungsverhältnis von dem zum Gang gehörenden Verhältnis abweicht. Beträgt die Abweichung mehr als 15%, wird der Gang auf null gesetzt. Nähert sich das aktuelle Übersetzungsverhältnis auf 15% an ein neues, zu einem Gang gehörendes Verhältnis an, so wird dieser Gang als der Aktuelle betrachtet.

Die Berechnung des Zeitsignals Gangwahl(t) wurde hierbei offline, vor der Simulation getätigt. Zu diesem Zweck wurde ein kleines Programm geschrieben, welche die eben erläuterten Zusammenhänge realisiert.

Nach [BS07] zeichnet sich die trockene, mechanische Kupplung dadurch aus, dass gilt:

$$M_{An} = M_{Ab} \quad (6.6)$$

$$n_{An} \neq n_{Ab}, \quad (6.7)$$

mit dem Moment M , der Drehzahl n sowie den Indizes An für die Antriebsseite und Ab für die Abtriebsseite.

In CarSim ist das, für einen Kupplungswert s maximal übertragbare Moment M_{AbMax} der Kupplung durch

$$M_{AbMax} = M_{Max}(1 - s) \quad (6.8)$$

implementiert, wobei M_{Max} das maximal übertragbare Moment für $s = 0$ ist.

Mit der Momentengleichung

$$J \frac{d\omega}{dt} = \sum_{i=1}^k M_i, \quad (6.9)$$

einmaligen Ableiten von Gl. (6.3) nach der Zeit

$$\frac{d}{dt} \omega = \frac{d}{dt} \frac{v}{r} = \frac{a}{r}, \quad (6.10)$$

und einsetzen von Gl. (6.10) in Gl. (6.9) erhält man

$$J \frac{a}{r} = M_{Ab}. \quad (6.11)$$

Das Trägheitsmoment J ergibt sich aus der Massenträgheit des Fahrzeugs, welche

als Punktmasse am Reifenradius r wirkt. Dieses ergibt sich nach [HRW01] mit

$$J = \int r^2 dm, \quad (6.12)$$

was sich für eine Punktmasse auf

$$J = mr^2 \quad (6.13)$$

reduziert. Setzt man Gl. (6.13) in Gl. (6.11) ein, erhält man:

$$a \cdot m \cdot r = M_{Ab}. \quad (6.14)$$

Mit der Beschleunigung a des Fahrzeugs entlang der X-Achse, der Fahrzeugmasse m und dem Reifenradius r nach Gl. (6.1).

Mit Gl. (6.8) lässt sich dann für jeden Zeitpunkt genau der Kupplungswert berechnen, der die Kupplung genau das Moment

$$M_{AbMax} = M_{Ab} \quad (6.15)$$

übertragen lässt, welches für die aktuelle Fahrzeugbeschleunigung notwendig ist.

Nun befindet sich die Kupplung im Antriebsstrang jedoch nach dem Motor und vor dem Getriebe. Das für die Beschleunigung tatsächlich wirksame Moment ist also noch um das, im Getriebe gewählte Übersetzungsverhältnis zu erhöhen.

$$a \cdot m \cdot r \cdot i_{Getr} = M_{Ab}. \quad (6.16)$$

Stellt man Gl. (6.8) nach s um, so erhält man

$$s = 1 - \frac{M_{AbMax}}{M_{Max}} = 1 - \frac{a \cdot m \cdot r \cdot i}{M_{Max}}. \quad (6.17)$$

6.5 Vorbereitung und Einrichtung der Software

6.5.1 Auswahl der CarSim Fahrzeugkomponenten

Im folgenden Kapitel soll ein erstes CarSim Modell ausgewählt werden, welches als Grundlage für die Systemidentifikation dient.

Dabei ist es erstrebenswert durch geschickte Auswahl der Komponenten, das Modell möglichst nahe an das Zielfahrzeug heranzubringen, damit der Optimierungsbedarf und damit der Zeitaufwand durch die Systemidentifikation minimal wird und um nach Möglichkeit bei der Optimierung die globalen Maxima heraus zu finden, was bei einem Startwert in der Nähe des voraussichtlichen Maximums wahrscheinlich ist (siehe Kapitel 1.2).

Wie schon während der Sensibilitätsanalyse besprochen, sind bereits einige Parameter des Fahrzeugs bekannt. Diese fließen an dieser Stelle in den Modellierungsprozess mit ein.

Es folgt eine Liste der Fahrzeugkomponenten.

Sprung Mass Diese Teilkomponente enthält die geometrischen Abmessungen, die Masse und die Trägheitsmomente des Fahrzeugs bezüglich der drei räumlichen Achsen. Vom Trägheitsmoment abgesehen, sind al-

le Größen aus vorangegangenen Messungen bekannt (siehe Anhang C). Das Trägheitsmoment wurde von einem in CarSim mitgeliefert Fahrzeug ähnlicher Bauart übernommen.

Aerodynamics Hier findet die Konfiguration der aerodynamischen Eigenschaften des Fahrzeugs statt. Gewählt wurde der Datensatz *Big SUV*, da dieser dem zu identifizierenden System am nächsten kommt. Angepasst wurde der Parameter *Frontal Area A*, welche auf den in Anhang C zu findenden Wert gesetzt wurde.

Animator Shape Diese Komponente hat nur Einfluss auf das Aussehen des Fahrzeugs im CarSim Animator und kann beliebig konfiguriert werden.

Powertrain Die Art des Antriebs. Hier ist die Option *Front Wheel Drive* zu wählen.

Engine Hier ist die Art des Motors zu wählen. Aufgrund der Ähnlichkeit mit dem zu identifizierenden System, wurde der Datensatz *4.0L-238kW Engine* gewählt.

Toque Transfer: Clutch Hier ist eine mechanische Kupplung zu wählen. Konkret wurde der Datensatz *Maximum 1200 Nm (Linear)* gewählt.

Transmission Hier sind die Getriebeübersetzungen nach Anhang C einzutragen.

Front Differential Hier wurde der Datensatz *Free - Gear Ratio 4.4* gewählt.

Als nächstes sind die Rohdaten zu laden. Unter dem Menüpunkt **Data -> Load Time Sections from File** gelangt man zu dem in Abb. (6.6) dargestellten Dialogfenster.

Dort kann man über **Open File**, die Rohdaten der Fahrversuche laden. Anschliessend lässt sich über die Eingabefelder **Start** und **End** auswählen, welcher Zeitbereich aus den aufgenommenen Daten verwendet werden soll. Gehen die Messdaten über eine Versuchsdauer von 10 Minuten, möchte man aber nur über die ersten 100 Sekunden optimieren, so ist dies hier einzustellen.

Weiterhin lassen sich hier auch mehrere Versuchsdaten laden. Ist dies geschehen, so erfolgt die Optimierung der Parameter über alle mit **On** markierten Datensätze. Im Beispiel von Abb. (6.6) wurden 2 Datensätze geladen, wobei nur der untere aktiv ist.

Nachdem die Rohdaten geladen sind, kann der Toolbox mitgeteilt werden, welche der geladenen Zeitverläufe Eingangs-, welche Ausgangsgrößen sind und welche Ausgänge in die Kostenfunktion mit einfließen sollen.

Zu dem entsprechenden Dialog gelangt man über **Model -> Channels**.

Hier steht einem eine Liste der gemessenen Größen zur Verfügung, sowie die Reiter **Output**, **Input** und **State**. Für uns interessant sind die ersten beiden.

Unter Input werden die Eingangsgrößen des Modells eingetragen, die da wären:

1. Gaspedalwert
2. Bremspedalwert
3. Lenkwinkel Betrag
4. Lenkwinkel Vorzeichen
5. Gangwahl
6. Motordrehzahl
7. Längsbeschleunigung

Die letzten beiden Größen dienen hierbei der Online-Generierung des Kupplungssignals.

Die Diskussion und Wahl der Ausgangsdaten erfolgt in den nachfolgenden Kapiteln.

Abschließend kann die Einrichtung der zu optimierenden Parameter erfolgen. Zum entsprechenden Dialog gelangt man über **Model -> Parameters**. Siehe Abb. (6.8).

Dort müssen alle Parameter angelegt werden, welche die Simulation erwartet. Gleichzeitig lassen sich für alle Parameter Grenzen definieren, innerhalb derer sie sich bewegen dürfen. Weiterhin lässt sich über die Option **Identify** einstellen, ob der Parameter identifiziert werden soll, oder ob konstant gehalten wird.

Die Diskussion und Wahl der Parameter erfolgt in den nachfolgenden Kapiteln.

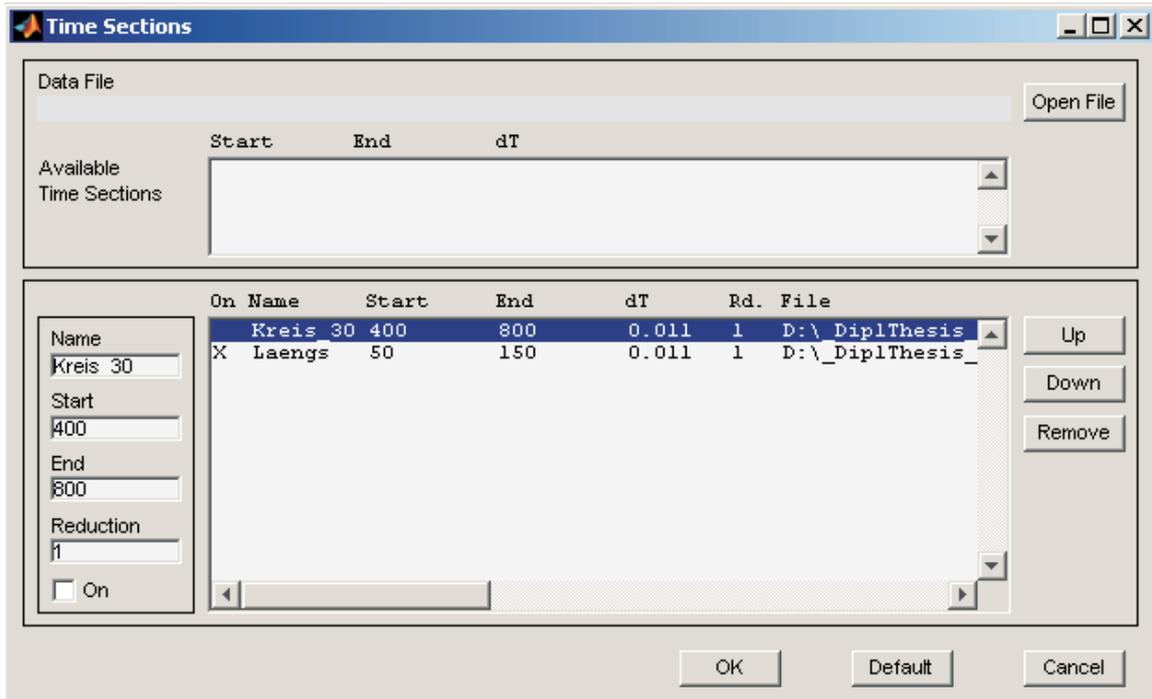


Abbildung 6.6: Dialogfenster: Laden der Rohdaten

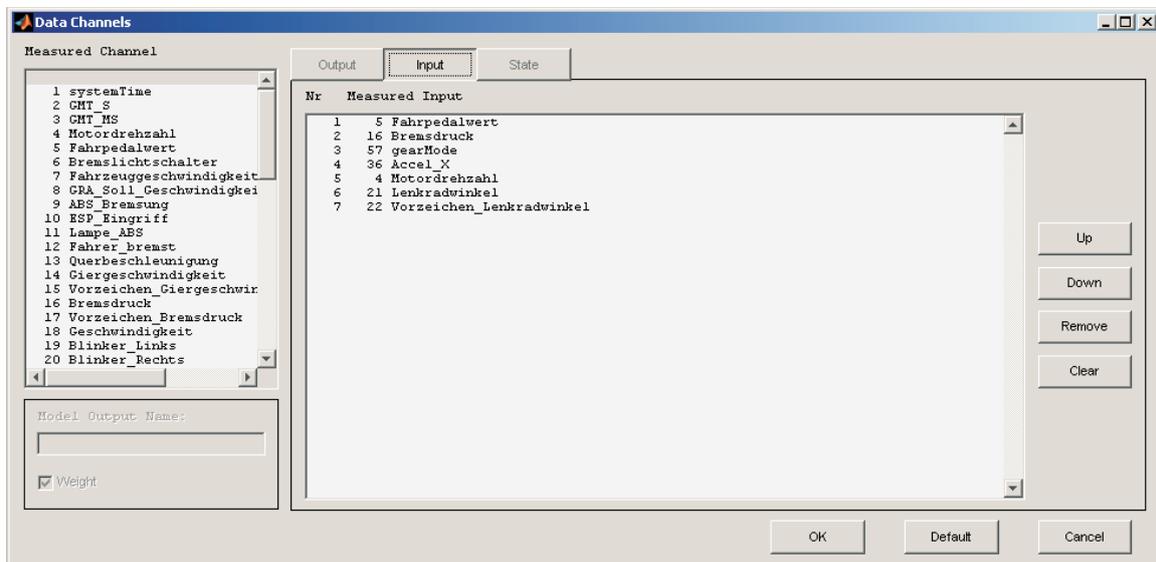


Abbildung 6.7: Dialogfenster, Wahl der Ein- und Ausgangskanäle.

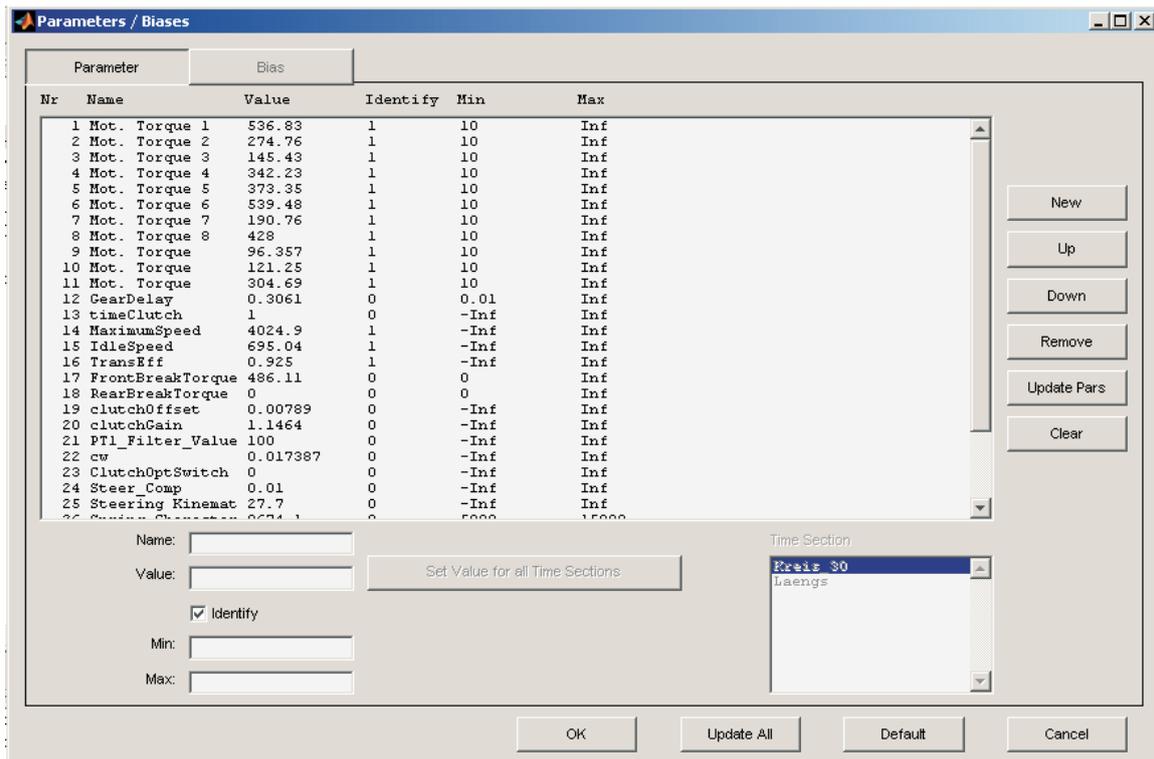


Abbildung 6.8: Dialogfenster, Wahl der Parameter.

6.6 Die Parameteridentifikation

Die Parameteridentifikation wird in zwei Schritten durchgeführt.

Es erfolgt eine thematische Trennung der Parameter, analog zu Kap. 5.2.2, nach den Punkten:

1. Längsdynamik
2. Querdynamik

Die Identifikation findet dann in genau dieser Reihenfolge statt. Die ist erforderlich, da die Längsdynamik ohne kor-

rekte Querdynamik identifiziert werden kann, die Querdynamik nicht aber ohne korrekte Längsdynamik. Dies ist einfach der Tatsache geschuldet, dass das Fahrzeug für die Querdynamikmanöver erst einmal auf eine gewisse, von den Messdaten vorgegebene Zielgeschwindigkeit gebracht werden muss, was die Längsdynamik zu leisten hat. Wäre die Längsdynamik nicht korrekt identifiziert, liefert die Simulation falsche Geschwindigkeitswerte für die vorgegebenen Eingangsverläufe, was zwangsläufig zu fehlerhaften Werten für die simulationsergebnisse der Querdynamik führt.

6.6.1 Identifikation der Querdy- namik

Die Optimierung der Längsdynamik er- folgt nach den in Kap. 5.5 gewählten Pa- rametern.

Komponente	Parameter
Steering System	Steer Compliance Steering Kinematics
Front/Rear Compliance	Spring Characteri- stics
Tire	Lateral Tire-Force Aligning moment

Tabelle 6.18: Komponenten der Querdy- namik

Eine zu beachtende Besonderheit, stellt die Kennlinie der Federrückstellkraft Abb. (6.9) dar.

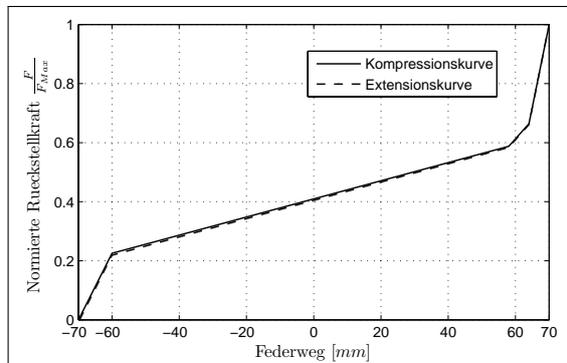


Abbildung 6.9: Normierte Kurve der Fe- derrückstellkraft. Quelle für Form der Kennlinie: CarSim 6.0

In Kap. 5.5 wurde bereits festgestellt, dass Gl. (6.18) gelten muss, damit CarSim die Simulation überhaupt erst startet.

$$f_{Ext}(x) < f_{Comp}(x) \quad \forall x \in \mathbb{R}, \quad (6.18)$$

Nun hat sich während der Systemidenti- fikation gezeigt, dass CarSim zur Überprü- fung der Bedingung Gl. (6.18) die beiden Kennlinien aus Abb. (6.9) über die an- gegebenen Werte hinaus extrapoliert und bei einer Verletzung von Gl. (6.18) im ex- trapolierten Bereich ebenso mit einer Feh- lermeldung abbricht. Weiterhin kann es aufgrund von numerischen Ungenauigkei- ten passieren, dass nach einer Neuskalier- ung der Kennlinien mit einem Faktor x oder y , $x, y \in \mathbb{R}$, $x \neq y$

$$x \cdot f_{Ext}(x) < x \cdot f_{Comp}(x) \quad \forall x \in \mathbb{R} \quad (6.19)$$

$$y \cdot f_{Ext}(x) < y \cdot f_{Comp}(x) \quad \forall x \in \mathbb{R} \quad (6.20)$$

für eine Skalierung mit x eine Verletzung von Gl. (6.18) detektiert wird, nicht aber für y , obwohl sich an der Natur der Kenn- linie nichts geändert hat.

Bei der aus CarSim entnommenen Kenn- linie, ist genau dies der Fall. Diese muss vor der Systemidentifikation so verändert werden, dass der Abstand zwischen den beiden Kurven f_{Ext} und f_{Comp} im Ex- trapolationsbereich größer wird und sie nicht parallel verlaufen.

Interessierende Ausgangsgröße, wieder in Hinblick auf das Anwendungsgebiet des resultierenden Modells, dem Fahrsimula- tor, ist bei der Querdyamik die Gierwin- kelgeschwindigkeit und die Querbeschleu- nigung.

Die Reihenfolge der Parameteridentifi- kation erfolgt nach der in Kap. 5.5 festge- legten Reihenfolge.

1. Steer at Ground

- 2. Steer Compliance
- 3. Lateral Tire Force
- 4. Aligning Moment
- 5. Spring Force Extension + Compression

Dort wurde das Kennlinienfeld im Ganzen in seinem Wertebereich variiert.

$$\underline{M}_N \cdot M_{Max} \tag{6.21}$$

\underline{M}_N ist das normierte Kennlinienfeld. Die Spaltenvektoren \underline{x}_i von \underline{M}_N beschreiben das Motormoment in Abhängigkeit der Drosselklappenstellung für eine bestimmte Drehzahl n_i .

$$\underline{M}_N = [\underline{x}_1, \underline{x}_2, \dots, \underline{x}_n] \tag{6.22}$$

Die Zeilenvektoren \underline{y}_i von \underline{M}_N beschreiben das Motormoment in Abhängigkeit der Drehzahl für eine bestimmte Drosselklappenstellung d_i .

$$\underline{M}_N = \begin{bmatrix} \underline{y}_1 \\ \underline{y}_2 \\ \vdots \\ \underline{y}_m \end{bmatrix} \tag{6.23}$$

6.6.2 Identifikation der Längsdynamik

Die Optimierung der Längsdynamik erfolgt nach den in Kap. 5.4 gewählten Parametern. Diese sind noch einmal in Tab. (6.19) aufgeführt.

Komponente	Parameter
Engine	Motor Torque Maximum speed Idle speed
Transmission	Efficiencies
Front Wheel Torque	Brake Torque vs. Brake Cylinder Pressure
Rear Wheel Torque	Brake Torque vs. Brake Cylinder Pressure

Tabelle 6.19: Komponenten der Längsdynamik

Zufriedenstellende Ergebnisse während der Systemidentifikation haben sich dabei eingestellt, wenn man nicht das Kennlinienfeld \underline{M}_N im ganzen über den zu identifizierenden Parameter M_{Max} skaliert, sondern für jeden Spaltenvektor \underline{x}_i über einen Parameter M_{xi} .

$$\underline{M}_N = [\underline{x}_1 \cdot M_{x1}, \underline{x}_2 \cdot M_{x2}, \dots, \underline{x}_n \cdot M_{xn}] \tag{6.24}$$

Die Elemente des vektorielle Parameter

$$\underline{M}_x = [M_{x1}, M_{x2}, M_{xn}], \tag{6.25}$$

sind dann Gegenstand der Systemidentifikation.

Dabei hat sich jedoch im Rahmen der Ausarbeitung der Systemidentifikation gezeigt, dass eine Skalierung des Motorkennlinienfeld Abb. (5.6), in der Art wie es in der Systemidentifikation in Kap. 5.4 vorweg geschehen ist, nicht ausreicht.

Abb. (6.10) veranschaulicht diese Zusammenhänge.

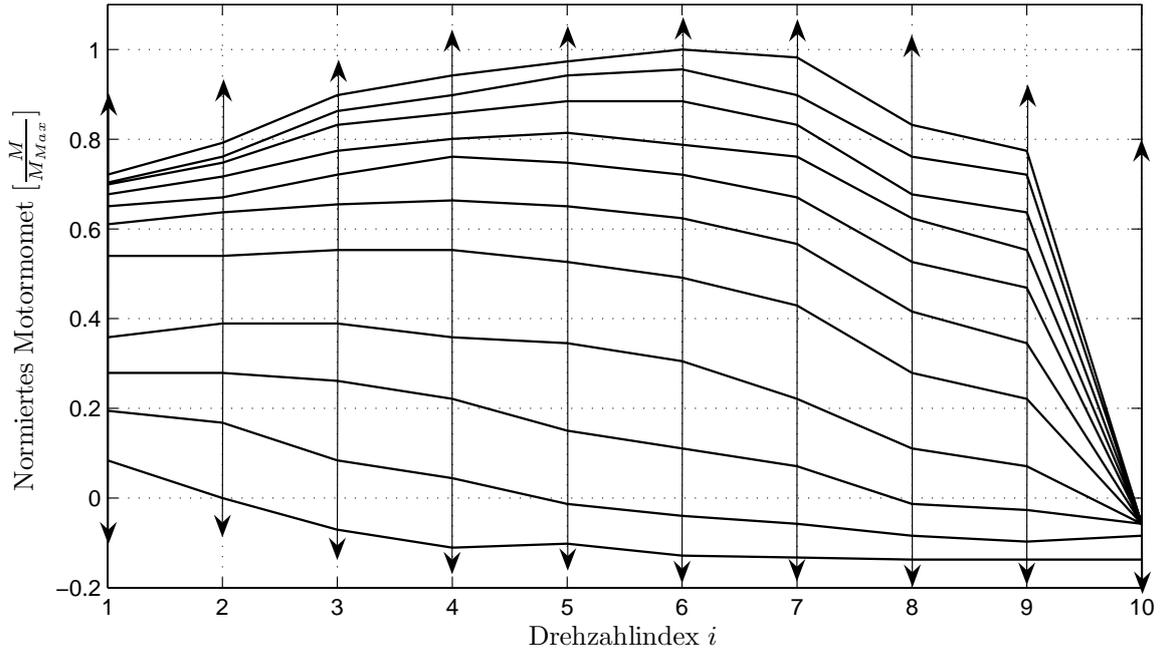


Abbildung 6.10: Normiertes Motorkennlinienfeld. Die Identifikation des Motorkennlinienfeld erfolgt über eine Skalierung der einzelnen Spalten des Feldes (durch die Doppelpfeile angedeutet). Quelle für Form des Kennlinienfeld: CarSim 6.0

Praktisch ist das Kennlinienfeld eine (10×11) Matrix, wodurch sich die Anzahl der zu identifizierenden Parametern von den Grenzen des Drehzahlintervalls (n_0, n_{max}) und dem maximalen Moment M_{Max} auf 11 Parameter für das Kennlinienfeld und dem Drehzahlintervall erhöht. Statt drei sind nunmehr 12 Parameter zu identifizieren, was sich deutlich in der Rechenzeit niederschlägt.

Die verbleibenden Parameter aus Tab. (6.19) bleiben unverändert.

Nach der Festlegung der zu identifizierenden Parameter wie voran gegangen geschehen und der Eingänge des Systems nach Kap. (6.4), ist nun noch eine Auswahl an Ausgängen zu treffen, nach denen das System optimiert werden soll.

Die Auswahl geschieht dabei unter Berücksichtigung des späteren Einsatzgebietes des gewonnenen CarSim Modells, dem dynamischen Fahr Simulator des Instituts für Verkehrssystemtechnik.

Dort wird zum einen der kinematische Zustand des Fahrzeugs für die Ansteuerung des Simulator benötigt, als auch die Motordrehzahl, welche dem Fahrer des Simulator zur Verfügung gestellt wird.

Als Ausgangsgrößen wird demnach gewählt

1. **Motordrehzahl**
2. **Fahrzeuggeschwindigkeit**
3. **Längsbeschleunigung**

Bei der Aufnahme der Realdaten wurden bezüglich der Längsdynamikmanöver alle Messungen direkt hintereinander durchgeführt und sämtliche Manöverdaten sind in einem Datensatz zu finden.

Bei der Systemidentifikation wurde dann so vorgegangen, das erstmal nur nach einem Ausschnitt der Messdaten optimiert wurde. Konkret wurde mit den ersten 100 Sekunden begonnen und Schrittweise um 100 Sekunden erhöht.

Auf diese Weise wurden die Parameter, bei minimierten Rechenaufwand, da nur für 100 Sekunden Fahrdaten optimiert wurde, schon einmal in den richtigen Bereich des Lösungsraum gebracht.

Mit der späteren Optimierung über längere Fahrdatenausschnitte erfolgt nur noch eine Feineinstellung.

Eine Ausnahme bildet das Motorkennlinienfeld.

Die Verläufe der Eingangsgrößen Motordrehzahl n und Drosselklappenstellung d

$$\underline{x}(t) = [n(t), d(t)] \quad (6.26)$$

beschreiben im Motorkennlinienfeld M eine Trajektorie $\underline{x}(t)$. Die Abbildung eines jeden Punktes des Motorkennlinienfeld auf das aktuelle Motormoment

$$f : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$$

ist dabei Gegenstand der Systemidentifikation. Dabei kann die Optimierung von f jedoch nur die Bereiche des Motorkennlinienfeld berücksichtigen, die von $\underline{x}(t)$

auch während des Optimierungsprozess durchlaufen wurden.

Fließen neue Messdaten in den Optimierungsprozess mit ein, ist das Motorkennlinienfeld neu zu identifizieren.

Anhang C ist zu entnehmen, dass für den Motor des zu identifizierenden Systems das maximale Moment bei $3200 \frac{1}{\text{min}}$ liegt. Das Kennlinienfeld Abb. (6.10) wird nun so skaliert, dass das maximale Moment nahe eben dieser Drehzahl liegt. Das auf diese Weise konstruierte Kennlinienfeld wird als \underline{M}_{Skal} bezeichnet. Es ist in diskretisierter Form gegeben und setzt sich aus 110 einzelnen Werten für Drosselklappen-, Motordrehzahlkombinationen zusammen.

Nimmt man nun den Eingangsvektor

$$\underline{x}(t) = [n(t), d(t)] \quad (6.27)$$

und diskretisiert $n(t)$ und $d(t)$ mit der selben Auflösung, wie das Motorkennlinienfeld \underline{M}_{Skal} , so lässt sich jedes $x(t)$ einem Datenpunkt im Kennlinienfeld zuordnen.

Dieser Datenpunkt bestimmt dann signifikant welches Moment der Motor für $x(t)$ erzeugt.

Zählt man nun, wie viele der zeitdiskreten Messdaten $x(t)$ auf jeden Datenpunkt liegen, bekommt man eine Aussage dafür, wie validiert jeder Punkt des Motorkennlinienfeld \underline{M}_{Skal} ist. Wird ein Punkt nur ein einziges mal oder überhaupt nicht angesteuert, so ist es unwahrscheinlich, das dieser Punkt korrekt identifiziert wird.

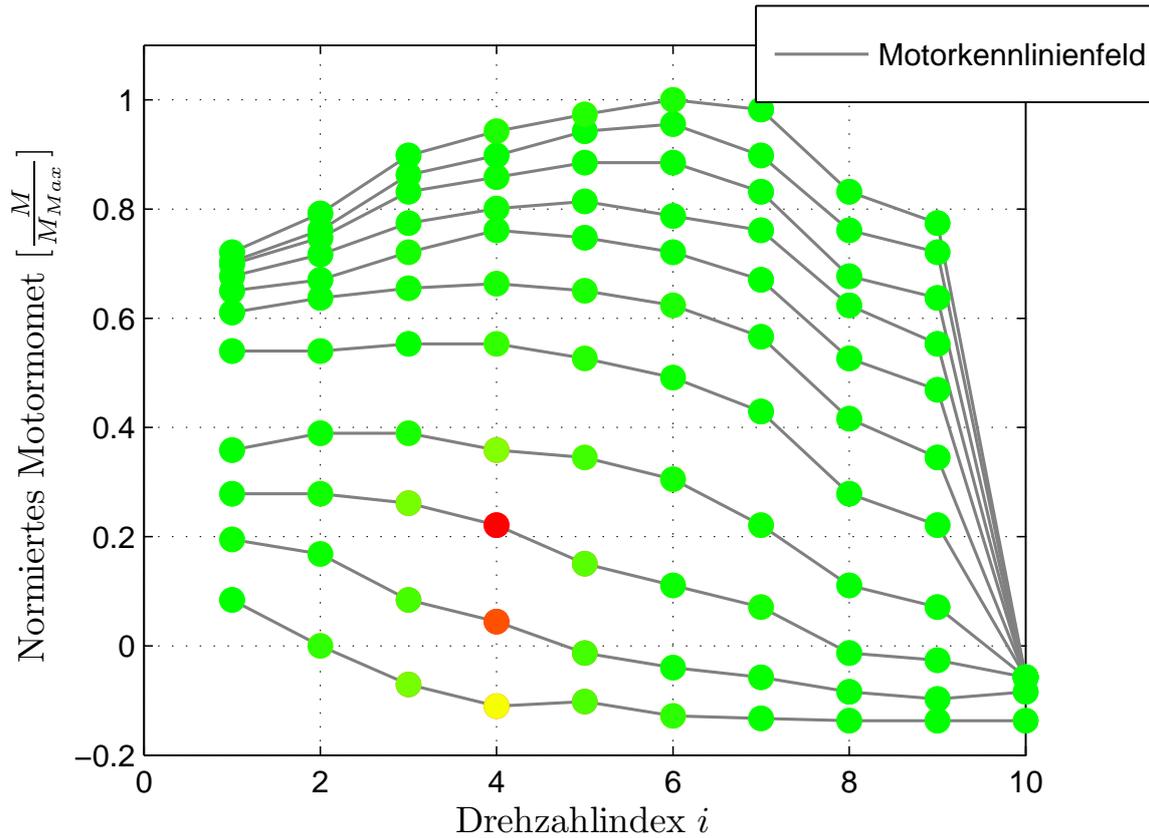


Abbildung 6.11: Das Motorkennlinienfeld. Jeder Punkt präsentiert ein Element des diskreten Kennlinienfeld. Die Farbe des Punktes gibt an, wie oft dieses Element von den Eingangsdaten $x(t)$ angesteuert wurde (grün=es liegen wenige $x(t)$ auf diesem Element, rot=es liegen viele $x(t)$ auf diesem Element).

Ordnet man nun der Anzahl der Zeitpunkte n , die $x(t)$ auf einen jeden Datenpunkt des Kennlinienfeldes liegen, einen Farbwert zwischen Grün→Rot zu (grün=wenige Zeitpunkte, rot=viele Zeitpunkte), so bekommt man eine visuelle Aussage dafür, wie oft bestimmte Bereiche des Kennlinienfeld von den Eingangsverläufen durchlaufen werden.

Die resultierende Grafik ist in Abb. (6.11) dargestellt.

Wie sich zeigt, konzentrieren sich sehr viele $x(t)$ auf einen sehr kleinen Bereich des Kennlinienfeld, so dass die anderen Datenpunkte farblich nicht mehr ins Gewicht fallen.

Um eine bessere Aussage dafür zu bekommen, welche Bereiche überhaupt Abgedeckt werden, wurde in Abb. (6.12) der Farbwert für jeden Datenpunkt über $\sqrt[4]{n}$ bestimmt.

Ganz allgemein folgt aus dieser Problematik, dass die identifizierte Längsdynamik

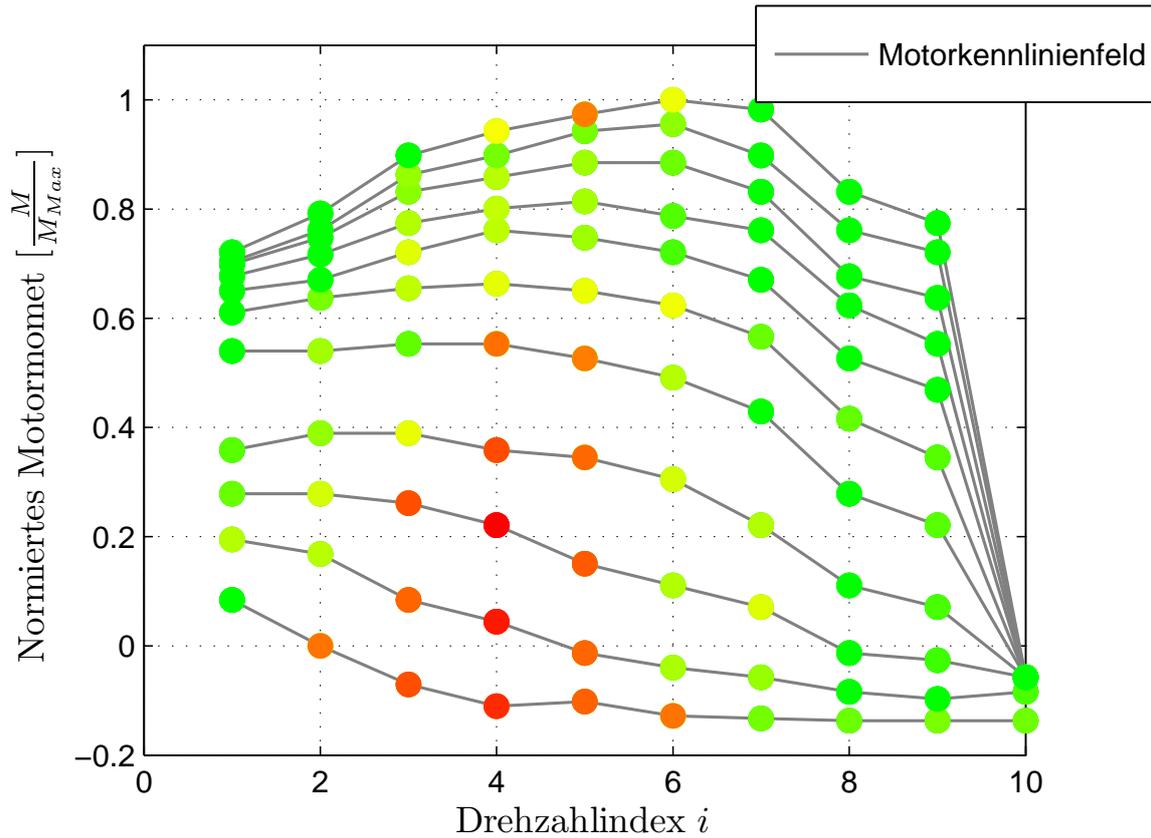


Abbildung 6.12: Das Motorkennlinienfeld. Jeder Punkt präsentiert ein Element des diskreten Kennlinienfeld. Die Farbe des Punktes gibt an, wie oft dieses Element von den Eingangsdaten $x(t)$ angesteuert wurde (grün=es liegen wenige $x(t)$ auf diesem Element, rot=es liegen viele $x(t)$ auf diesem Element). Vor der Zuordnung zu einem Farbwert wurde aus der Anzahl der Zeitpunkte die auf dem jeweiligen Element liegen, die vierte Wurzel gezogen.

zwangsläufig nur in den Bereichen Gültigkeit hat, die bezüglich der Verläufe von $\underline{x}(t)$ auch Teil der Messdaten waren.

Die Reihenfolge der Parameteridentifikation erfolgt dann nach Kap. 5.4.

1. **Max. Motor Speed:**

2. **Max. Motor Torque**

3. **Transm. Eff.**

4. **Motor Idle Speed**

5. **Front Brk. Torq.**

6. **Rear Brk. Torq.**

6.7 Ergebnisse und Ergebnisvalidierung

Die durch den Identifikationsprozess gewonnenen Parameter sind in Tab. (6.20) und Tab. (6.21) aufgeführt.

ID	Parameter	Wert
1	Eng. Torque 01	482.85 [Nm]
2	Eng. Torque 02	165.29 [Nm]
3	Eng. Torque 03	133.56 [Nm]
4	Eng. Torque 04	425.24 [Nm]
5	Eng. Torque 05	556.02 [Nm]
6	Eng. Torque 06	636.64 [Nm]
7	Eng. Torque 07	161.87 [Nm]
8	Eng. Torque 08	414.29 [Nm]
9	Eng. Torque 09	231.43 [Nm]
10	Eng. Torque 10	010.00 [Nm]
11	Eng. Torque 11	414.30 [Nm]
14	Max. Eng. Speed	4024.9 $[\frac{1}{min}]$
15	IdleSpeed	695.04 $[\frac{1}{min}]$
16	TransEff	0.925 [1]
17	Fr. Br. Torque	486.11 [Nm]
18	Re. Br. Torque	0 [Nm]

Tabelle 6.20: Längsdynamik: Die durch den Identifikationsprozess gewonnenen Parameter

Die Systemidentifikation über einen Optimierungsprozess ist ein mathematisches Problem. Macht ein Parametersatz $\underline{\Gamma}$ die Kostenfunktion aus Kapitel 3.2 minimal und spiegeln die Ausgangsverläufe das zu identifizierende System best möglich wieder, so ist das Problem gelöst.

Eine interessante Betrachtung ist dann, in wie weit die identifizierten Parameter physikalisch und technisch Sinnvoll sind.

ID	Parameter	Wert
24	Steer Comp.	0.01 $[\frac{\circ}{Nm}]$
25	Steer Kinema.	27.7 $[\frac{\circ}{\circ}]$
26	Spring Character	9674.1 $[N/mm]$
27	Lat. Tire-Force	4496.8 $[N/\circ]$
28	Align. Moment	-0.14 $[Nm/\circ]$
29	Shock Absorber	1804 $[N/(mm/s)]$
30	Auxiliary Moment	1000 $[Nm/\circ]$
31	Align. Mom. Sc.	14.11 $[\circ]$

Tabelle 6.21: Querdynamik: Die durch den Identifikationsprozess gewonnenen Parameter

Es ist denkbar, dass für eine Größe eine negative Länge identifiziert wurde, was physikalisch unsinnig ist, vom mathematischen Standpunkt aus das Optimierungsproblem aber zu lösen vermag.

Es folgt eine Betrachtung der identifizierten Parameter unter diesem Gesichtspunkt:

Das Motorkennlinienfeld Bei Betrachtung der Parameter 1 bis 11 aus Tab. (6.20) zeigt sich, das ein sehr unregelmäßiges Kennlinienfeld identifiziert wurde. Es ist unwahrscheinlich, dass dies der realen Motor-Charakteristik entspricht. Die Ursache wird bei der starken Reduzierung des Kennlinienfeld auf einige wenige Parameter und der Art der Messdaten (siehe Kap. 6.6.2) zu finden sein.

Max. Motordrehzahl Ein kurzer Quervergleich mit Anhang C zeigt, das die identifizierte maximale Drehzahl des Motors weit unter dem tatsächlichen

Wert liegt (der Motor hat seine maximale Leistung bei $6300 \frac{1}{\text{min}}$). Hier wurde eindeutig ein falscher Wert identifiziert. Die Ursache liegt in der in Kap. 6.6.2 besprochenen Problematik.

Die Bremsmomente Wie sich bei Betrachtung dieser Parameter zeigt, liefert die Optimierung einen großen Wert für das Bremsmoment der Vorderreifen und kein Bremsmoment für die Hinterreifen, was eindeutig nicht den Gegebenheiten bei einem realen Fahrzeug entspricht.

Reifen-Rückstellmoment Tab. (6.21) zeigt, dass ein negatives Reifen-Rückstellmoment für positive Schwimmwinkel identifiziert wurde, was der Natur der eines Rückstellmomentes zuwider läuft.

Ein Vergleich der Ausgangsverläufe des Realsystems und der Simulation wird in Abb. (6.13) bis Abb. (6.19) dargestellt.

Die größte Auffälligkeit ist das starke Rauschen in einigen Bereichen des Beschleunigungssignals.

Zurückzuführen ist dies zu einem Teil auf numerische Instabilitäten im CarSim-Solver.

Problematisch an dieser Stelle ist auch die Generierung des Kupplungssignals, welches aus dem Beschleunigungssignal abgeleitet wurde. Das Beschleunigungssignal ist jedoch mit einem starken Messrauschen überlagert, so dass das Kupplungssignal ebenso stark verrauscht ist.

Es liegt an dieser Stelle nahe, die fehlerbehafteten Ausgänge der Simulation sowie das Kupplungssignal zu filtern um das kritische Rauschen zu entfernen.

Hier gilt es einen Kompromiss zu finden, zwischen der Dämpfung unerwünschter Frequenzanteile und der eingehandelten Phasenverzögerung der Ausgänge.

Gelöst wird das Problem, indem die Filterparameterierung zum Teil des Optimierungsproblems gemacht wird.

Als Filter dient ein einfaches PT1-Glied

$$G(s) = \frac{K}{1 + Ts}. \quad (6.28)$$

T ist der zu identifizierende Parameter, für den Verstärkungsfaktor wird $K = 1$ gewählt.

Unbefriedigend sind die Verläufe der Motordrehzahl Abb. (6.19). Es hat sich gezeigt, dass dies verbessert werden kann, indem das Motorkennlinienfeld auf andere Weise modifiziert und optimiert wird (z.B. Variation der Zeilen des Kennlinienfeld und nicht der Spalten). Dies geht jedoch auf Kosten der Qualität der anderen Ausgangsverläufe. Da diese jedoch für die Bewegungsansteuerung des Fahrsimulators benötigt werden, wurden die Abweichungen in der Motordrehzahl in Kauf genommen.

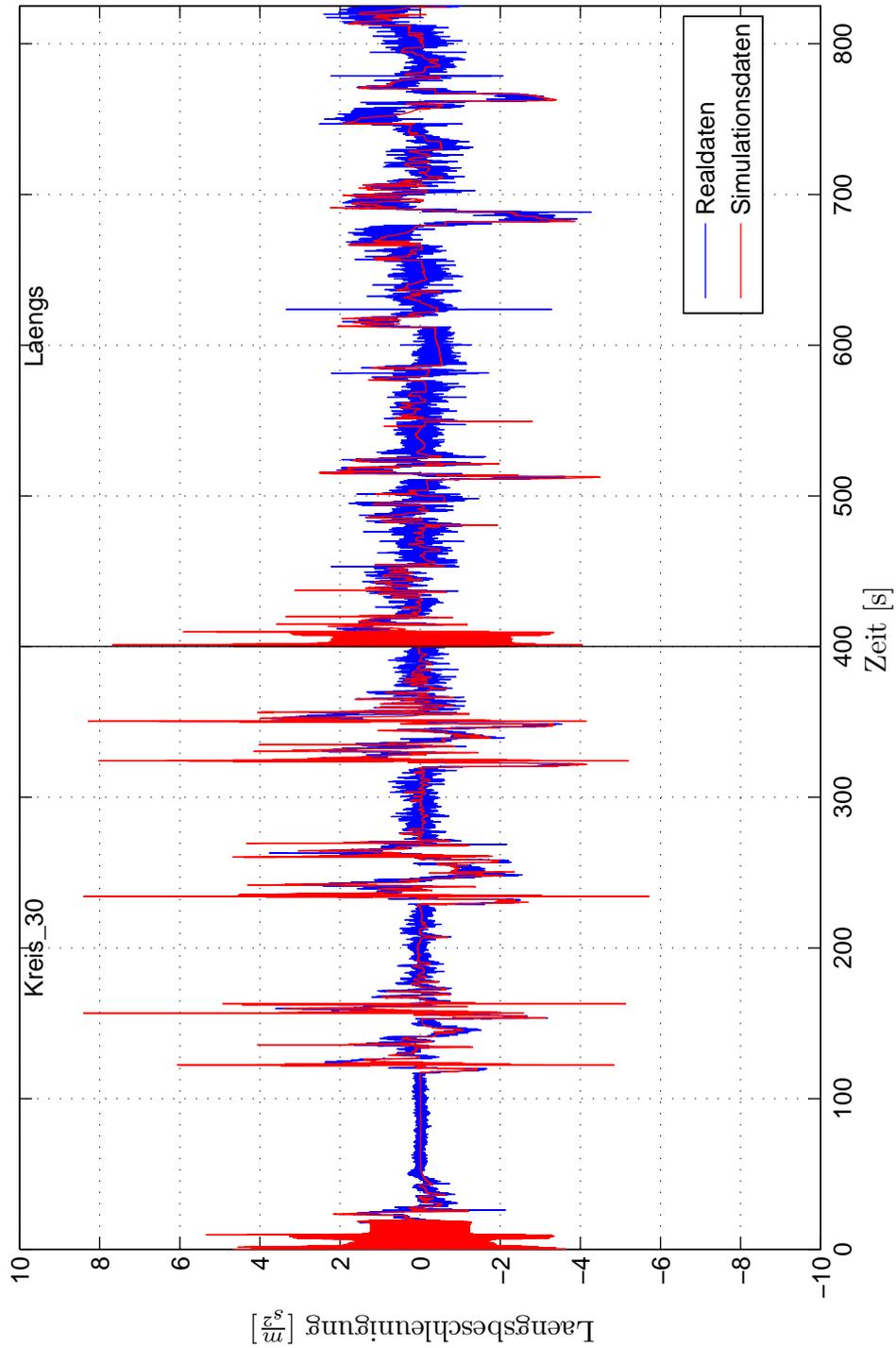


Abbildung 6.13: Die Längsbeschleunigung des Simulationsmodell im Vergleich mit den Realdaten. Es wurde für jeweils einen Längs- und Querdynamikversuch Simuliert. Die Ausgänge sind ungefiltert.

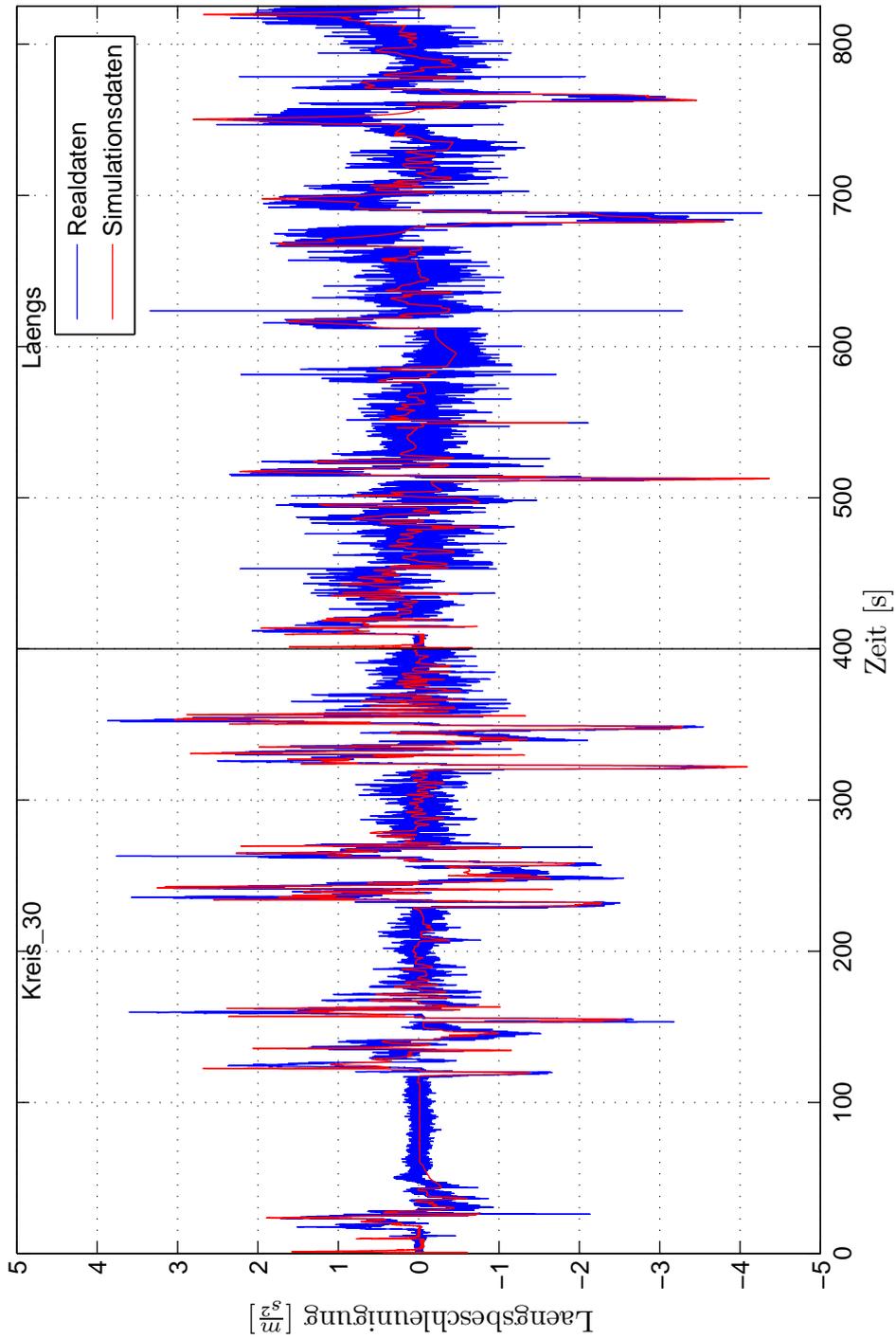


Abbildung 6.14: Die Längsbeschleunigung des Simulationsmodell im Vergleich mit den Realdaten. Es wurde für jeweils einen Längs- und Querdynamikversuch Simuliert. Die Ausgänge sind gefiltert.

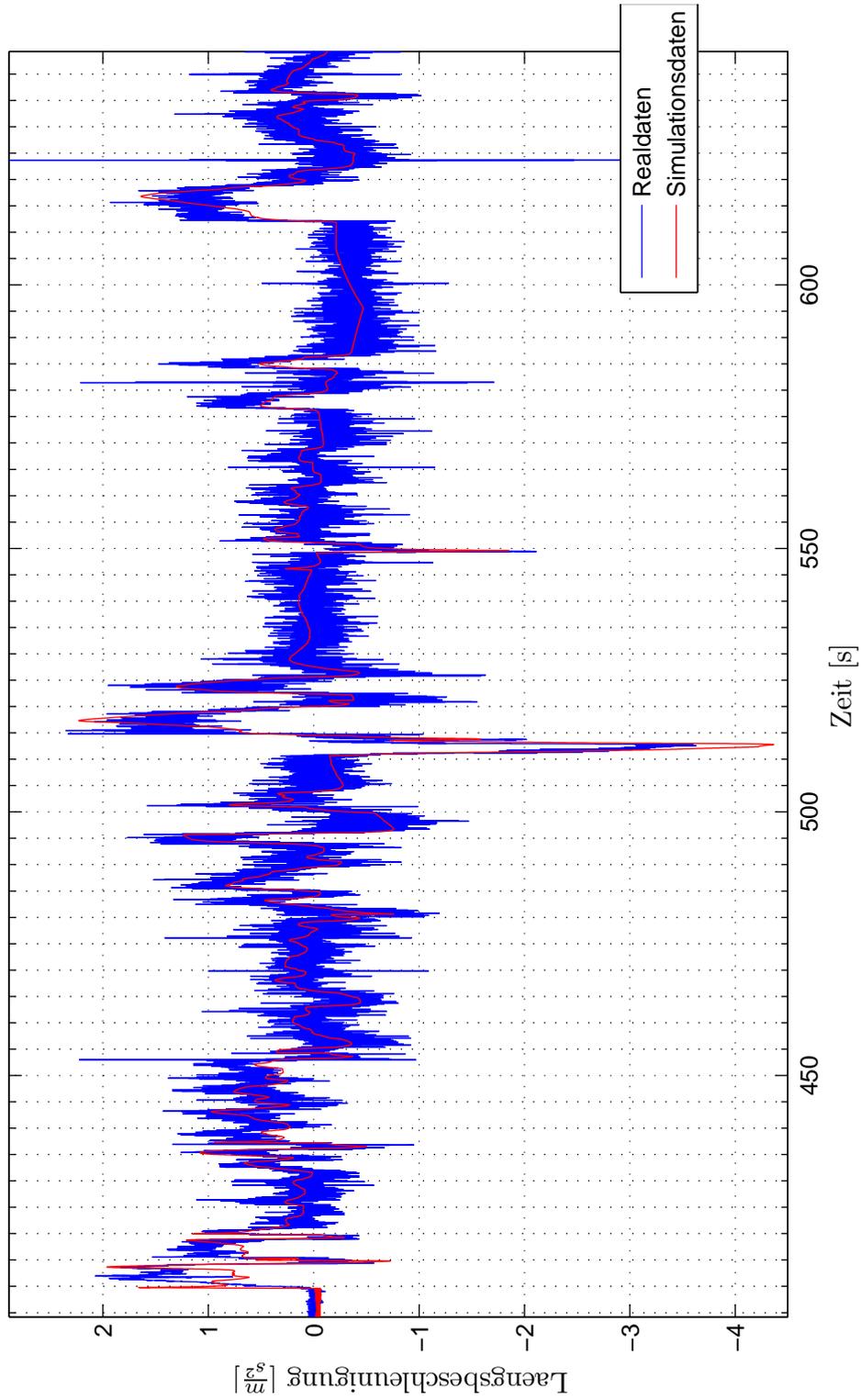


Abbildung 6.15: Ausschnitt der Längsbeschleunigung des Simulationsmodell im Vergleich mit den Realdaten. Die Ausgänge sind gefiltert.

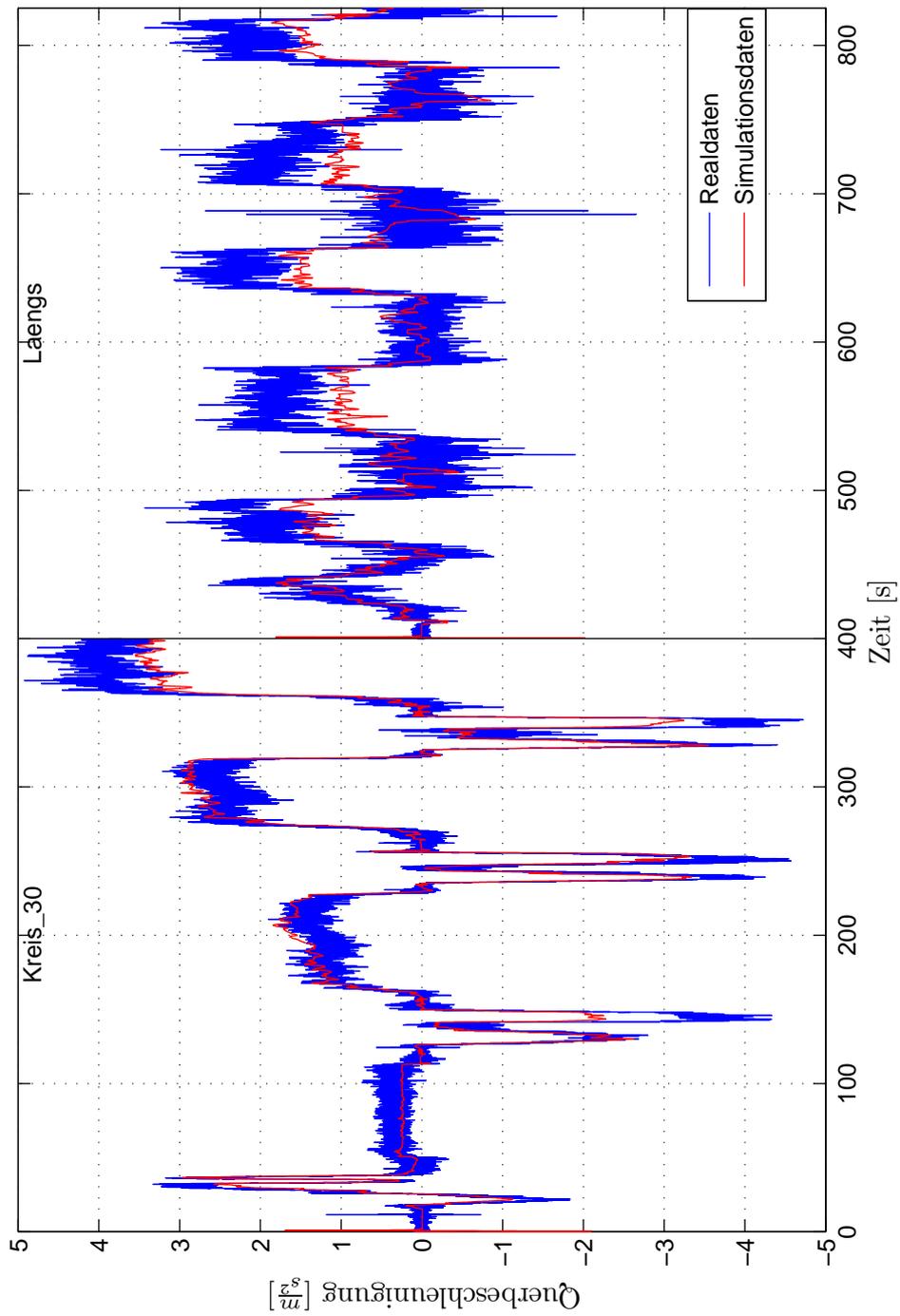


Abbildung 6.16: Die Querbeschleunigung des Simulationsmodell im Vergleich mit den Realdaten. Es wurde für jeweils einen Längs- und Querdynamikversuch Simuliert. Die Ausgänge sind ungefiltert.

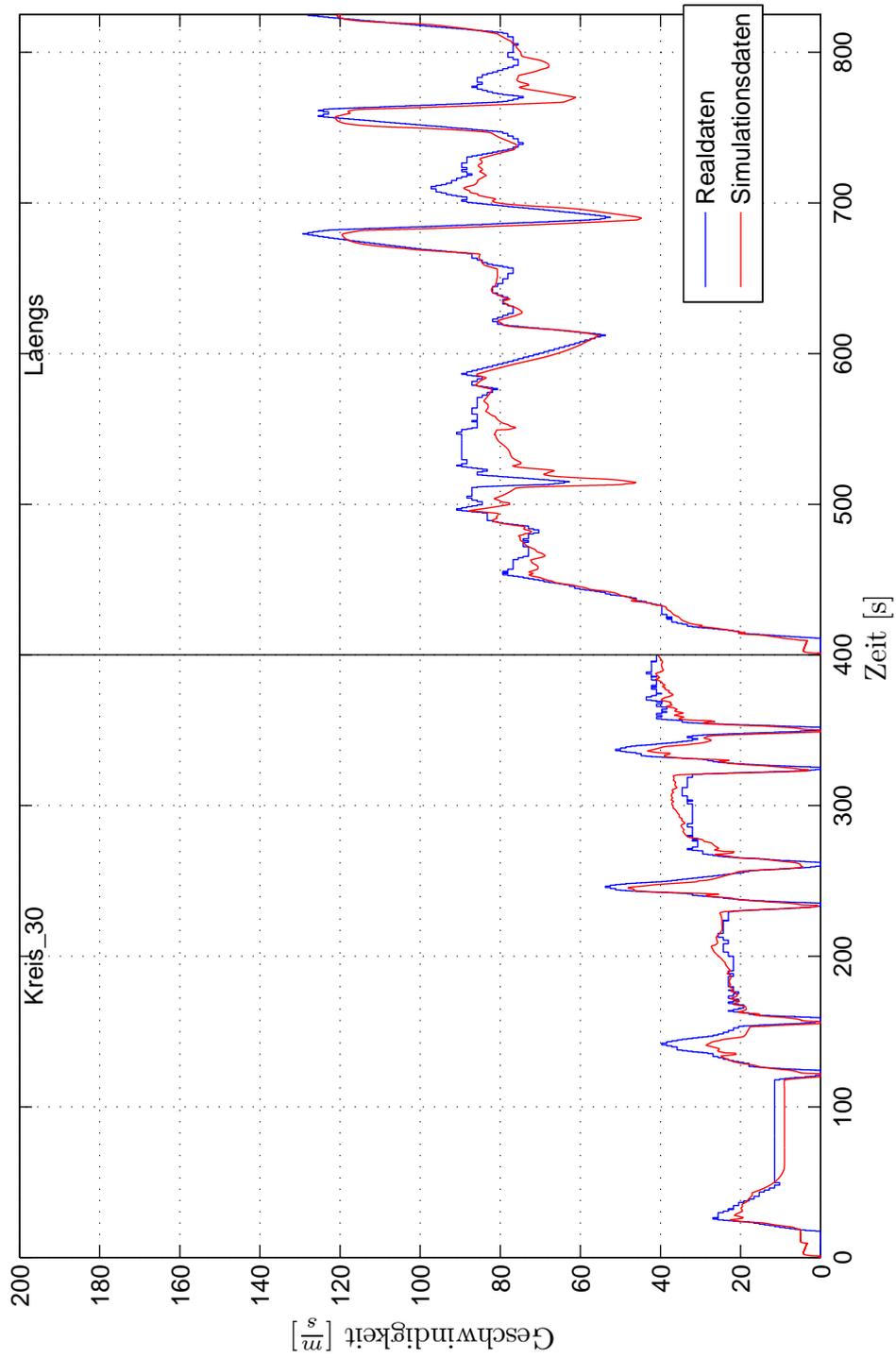


Abbildung 6.17: Die Geschwindigkeit des Simulationsmodell im Vergleich mit den Realdaten. Es wurde für jeweils einen Längs- und Querdynamikversuch Simuliert.

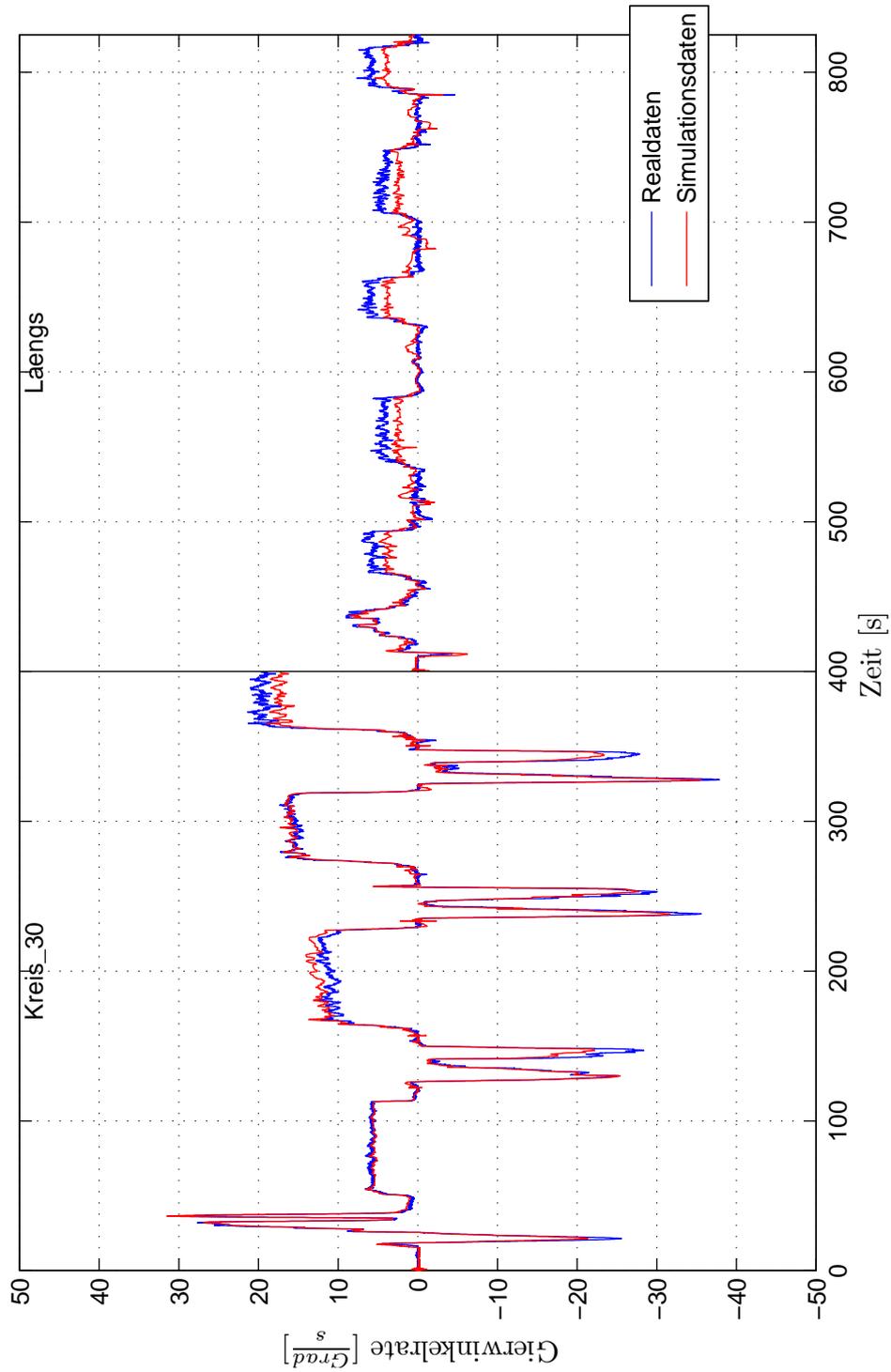


Abbildung 6.18: Die Gierrate des Simulationsmodell im Vergleich mit den Realdaten. Es wurde für jeweils einen Längs- und Querdynamikversuch Simuliert.

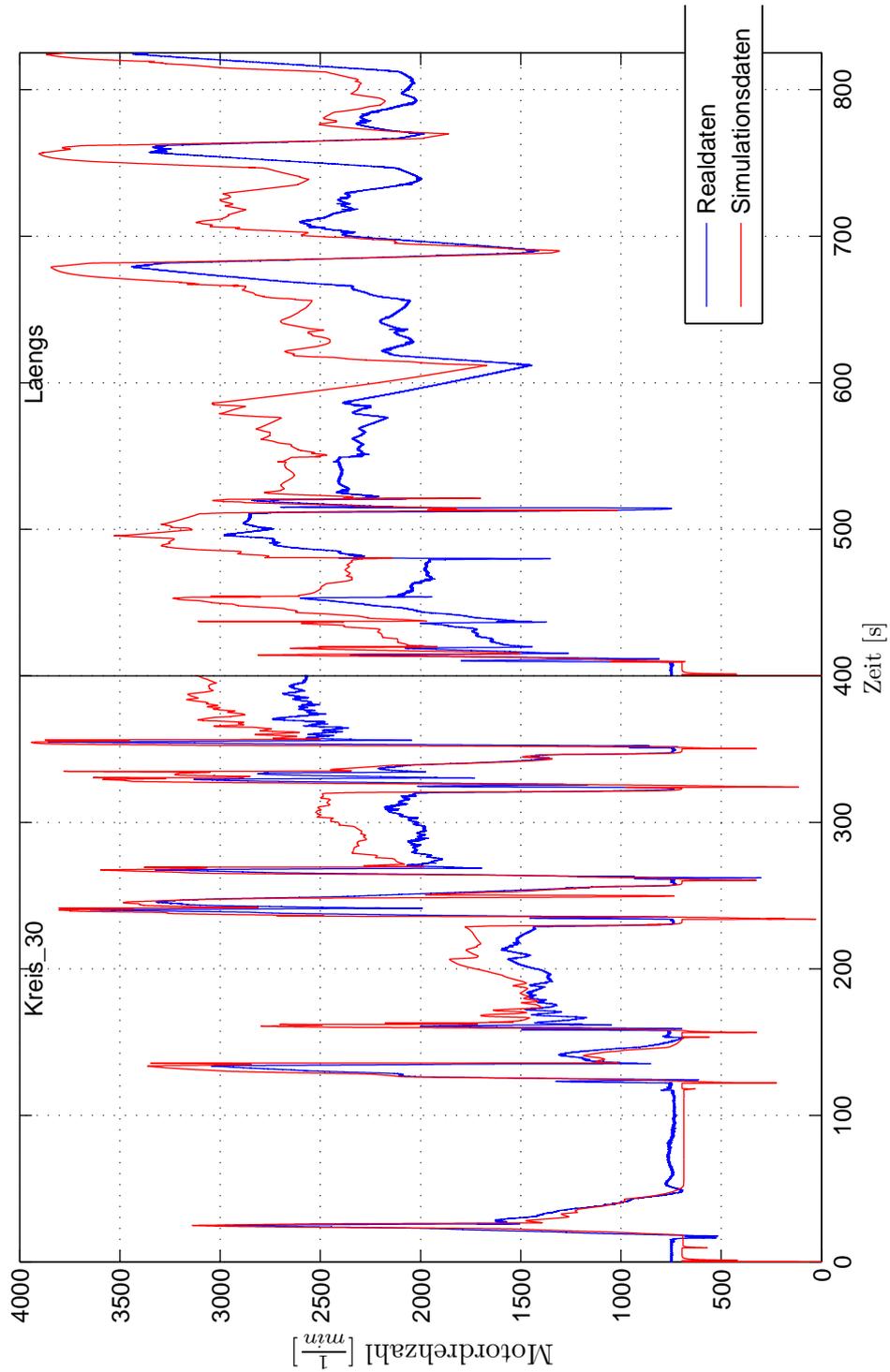


Abbildung 6.19: Die Motordrehzahl des Simulationsmodell im Vergleich mit den Realdaten. Es wurde für jeweils einen Längs- und Querdynamikversuch Simuliert.

6.8 Zusammenfassung und Ausblick

Zusammenfassend lässt sich sagen, dass das Verfahren der Parameteridentifikation über einen Optimierungsprozess durchaus funktioniert. Die Übereinstimmung zwischen simulierten und gemessenen Ausgängen ist unter dem Gesichtspunkt der relativ wenigen Parameter, die angepasst wurden, zufriedenstellend.

Insgesamt wurden 12 Parameter durch den Optimierungsprozess modifiziert. 14 weitere Parameter waren durch Messungen oder Herstellerangaben bekannt. Nimmt man bei der Angabe des Entwicklers von CarSim, dass die Software über 100-150 Parameter verfügt, die 100 als Grundlage, so ergibt sich, dass im Modell 25% der Parameter an das View-Car angepasst wurden. Aufgrund der starken Vereinfachung bei den vielen Kennlinien und Kennlinienfelder ist anzunehmen, dass sich auch diese noch wesentlich besser identifizieren lassen, wenn man sie über eine größere Anzahl von Parametern beeinflusst.

Von den 9 zur Verfügung stehenden Fahrmanövern wurden zwei für die Optimierung heran gezogen. Auch hier ist Potential zur Verbesserung des Parameteridentifikation vorhanden. Die beiden verwendeten Manöver hatten eine Gesamtlänge von ca. 850 Sekunden.

Die Optimierung des Modells nach den 12 CarSim-Parametern über die 850 Sekunden Versuchsdaten hat auf einen 2 GHz Dual Core Computer knappe 20 Stun-

den gedauert, wobei jedoch daran erinnert werden muss, dass beispielsweise der Parameter *Motormoment* durch 12 einzeln zu identifizierende Werte repräsentiert wurde. Die Anzahl der für FitLab sichtbaren Parameter erhöht sich damit insgesamt auf 24.

Als Ausblick ist das Potential bei der Identifikation der Kennlinien zu nennen. Gelingt es einen diese beispielsweise durch Funktion zu approximieren, lässt sich hier eventuell die Anzahl der benötigten Größen zur korrekten Beschreibung der Kennlinie reduzieren und eine besseren Beeinflussung der Kennlinien erreichen.

Verfügt man über die nötige Zeit wäre eine Optimierung über alle aufgenommenen Realdaten erstrebenswert.

Eine weitere, in dieser Arbeit nicht durchleuchtete Problematik ist die Abhängigkeit der Endergebnisse bezüglich der Anfangswerte. Als Startwert haben Vorgaben aus CarSim gedient. Zu prüfen wäre hier, in wie weit das Endergebnis durch eine Variation der Startparameter beeinflusst und eventuell verbessert werden kann.

Literaturverzeichnis

- [Aud04] AUDI: *Technische Daten Audi A6 Avant 3.0.* -, -, 2004.
- [BS07] BRAESS und SEIFFERT: *Handbuch Kraftfahrzeugtechnik.* Vieweg, 2007.
- [Bus01] BUSAM: *Analysis I-II.* -, -, 2001.
- [FCHL05] F.M.DEKKING, C.KRAAIKAMP, H.P.LOPUHAA und L.E.MEESTER: *A Modern Introduction to Probability and Statistics - Understanding Why and How.* Springer-Verlag London Limited, 2005.
- [HRW01] HALLIDAY, RESNICK und WALKER: *Physik.* Wiley-VCH, 2001.
- [KN98] KRAMER, U. und M. NECULAU: *Simulationstechnik.* Carl Hanser Verlag München Wien, 1998.
- [PM84] PLAETSCHKE, E. und D.B. MACKIE: *Maximum-Likelihood-Schätzung von Parametern linearer Systeme aus Flugversuchsdaten - Ein FORTRAN-Programm.* -, -, 1984.
- [Qui01] QUINN, K.: *The Newton Raphson Algorithm for Function Optimization.* -, -, 2001.
- [Sun07a] SUN, X.: *Manöverprotokoll Ehra.* -, -, 2007.
- [Sun07b] SUN, X.: *Parameteridentification of the ViewCar modelled in CarSim 7.0.* -, -, 2007.

ANHANG A

Ergebnisse der Sensibilitätsanalyse

A.1 Längsdynamik

Nr.	max. Torq.	max. Speed	idle Speed	Effiec.	Front Brk. Torq.	Rear Brk. Torq.
01	X					
02	X	X				
03	X	X		X		
04			X			
05		X		X		
06				X		
07					X	
08						X
09					X	X
10	X					

Tabelle A.1: Variationskombinationen Längsdynamik

Parameterbeschreibungen:

max. Torq. Maximum motor torque (Skalierung einer Kennlinie) [Nm]

max. Speed Maximum motor speed $[\frac{1}{\text{min}}]$

idle Speed Idle motor speed $[\frac{1}{\text{min}}]$

Effiec. Transmission efficiencies [1]

Front Brk. Torq. Front break torque @ 1 MPa break pressure [Nm]

Rear Brk. Torq. Rear break torque @ 1 MPa break pressure [Nm]

Nr. 001

Varierte Variablen Maximum Torque

Variationsschritte

Parameter	1	2	3	4
max. Torq. [Nm]	100	200	300	400
max. Speed [$\frac{1}{min}$]	6200	6200	6200	6200
idle Speed [$\frac{1}{min}$]	600	600	600	600
Front Brk. Torq. [Nm]	150	150	150	150
Rear Brk. Torq. [Nm]	70	70	70	70
Effiec. [1]	0.85	0.85	0.85	0.85

Tabelle A.2

Ausgangssignale

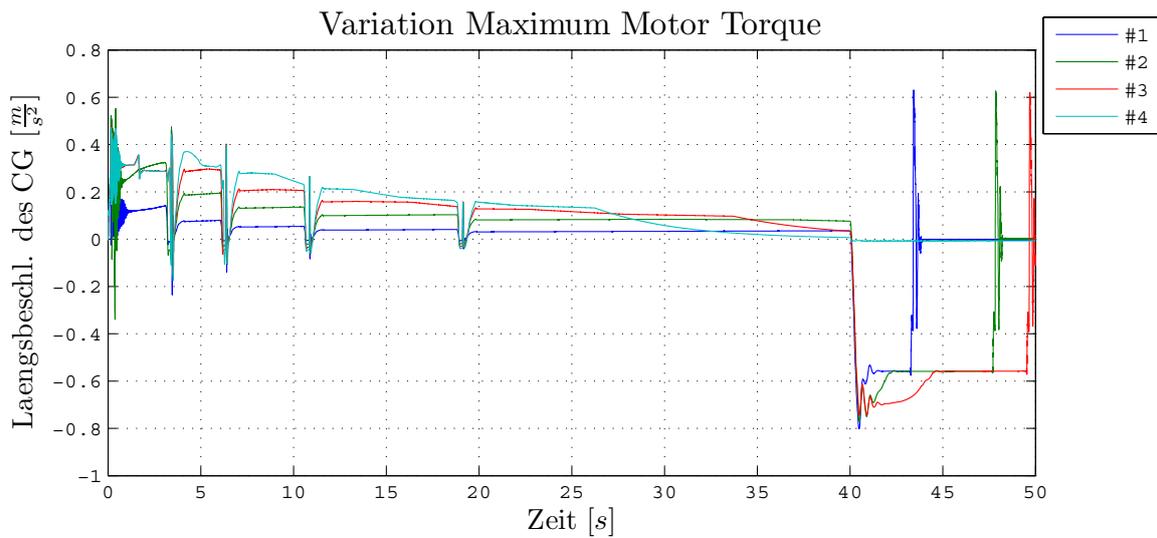


Abbildung A.1

Nr. 002

Varierte Variablen Maximum Torque, maximum Motorspeed

Variationsschritte

Parameter	1	2	3	4
max. Torq. [Nm]	100	200	300	400
max. Speed [$\frac{1}{min}$]	5000	6200	7200	8200
idle Speed [$\frac{1}{min}$]	600	600	600	600
Front Brk. Torq. [Nm]	150	150	150	150
Rear Brk. Torq. [Nm]	70	70	70	70
Effiec. [1]	0.85	0.85	0.85	0.85

Tabelle A.3

Ausgangssignale

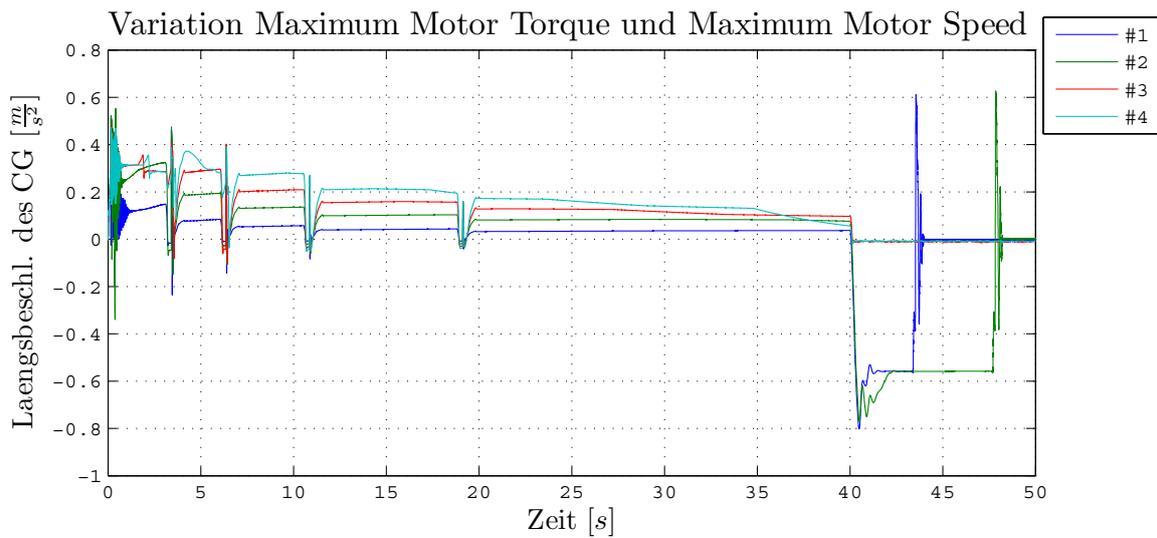


Abbildung A.2

Nr. 003

Variierte Variablen Maximum Torque, maximum Motorspeed, transmission Efficiencies

Variationsschritte

Parameter	1	2	3	4
max. Torq. [Nm]	100	200	300	400
max. Speed [$\frac{1}{min}$]	5000	6200	7200	8200
idle Speed [$\frac{1}{min}$]	600	600	600	600
Front Brk. Torq. [Nm]	150	150	150	150
Rear Brk. Torq. [Nm]	70	70	70	70
Effiec. [1]	0.80	0.85	0.90	0.95

Tabelle A.4

Ausgangssignale

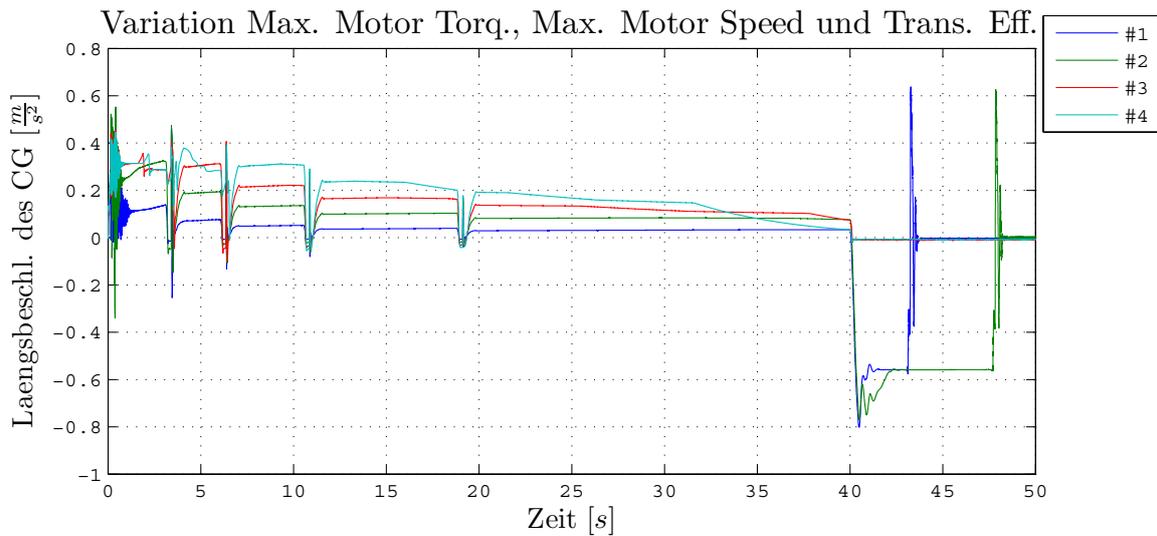


Abbildung A.3

Nr. 004

Variierte Variablen Idle Motorspeed

Variationsschritte

Parameter	1	2	3	4
max. Torq. [Nm]	300	300	300	300
max. Speed [$\frac{1}{min}$]	7200	7200	7200	7200
idle Speed [$\frac{1}{min}$]	400	600	800	1000
Front Brk. Torq. [Nm]	150	150	150	150
Rear Brk. Torq. [Nm]	70	70	70	70
Effiec. [1]	0.90	0.90	0.90	0.90

Tabelle A.5

Ausgangssignale

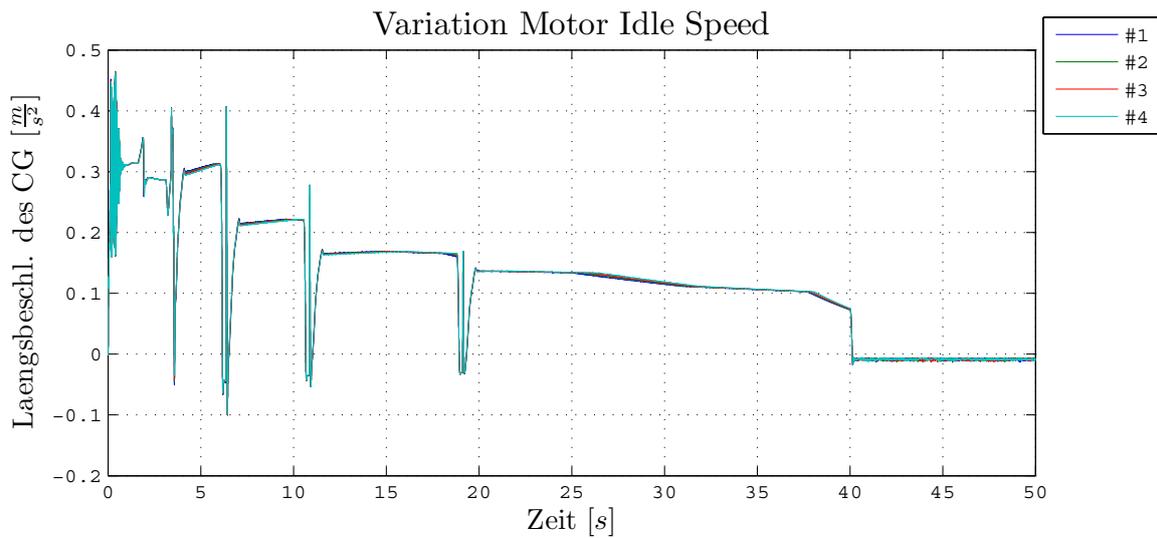


Abbildung A.4

Nr. 005

Varierte Variablen Maximum Motorspeed, transmission Efficiencies

Variationsschritte

Parameter	1	2	3	4
max. Torq. [Nm]	300	300	300	300
max. Speed [$\frac{1}{min}$]	5000	6200	7200	8200
idle Speed [$\frac{1}{min}$]	600	600	600	600
Front Brk. Torq. [Nm]	150	150	150	150
Rear Brk. Torq. [Nm]	70	70	70	70
Effiec. [1]	0.4	0.6	0.8	1.00

Tabelle A.6

Ausgangssignale

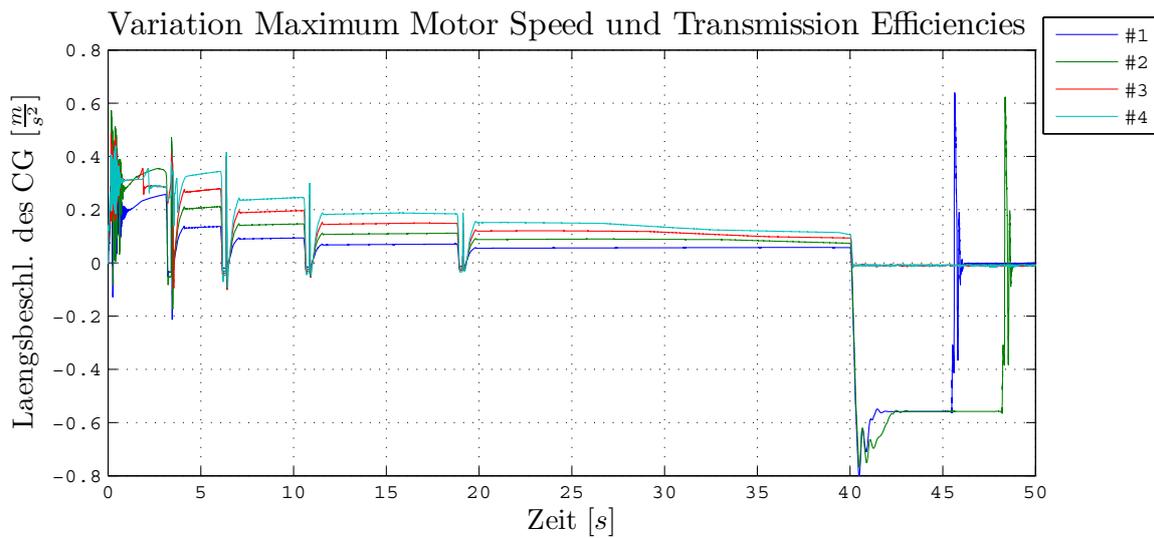


Abbildung A.5

Nr. 006

Varierte Variablen Transmission Efficiencies

Variationsschritte

Parameter	1	2	3	4
max. Torq. [Nm]	300	300	300	300
max. Speed [$\frac{1}{min}$]	6200	6200	6200	6200
idle Speed [$\frac{1}{min}$]	600	600	600	600
Front Brk. Torq. [Nm]	150	150	150	150
Rear Brk. Torq. [Nm]	70	70	70	70
Effiec. [1]	0.2	0.4	0.6	0.8

Tabelle A.7

Ausgangssignale

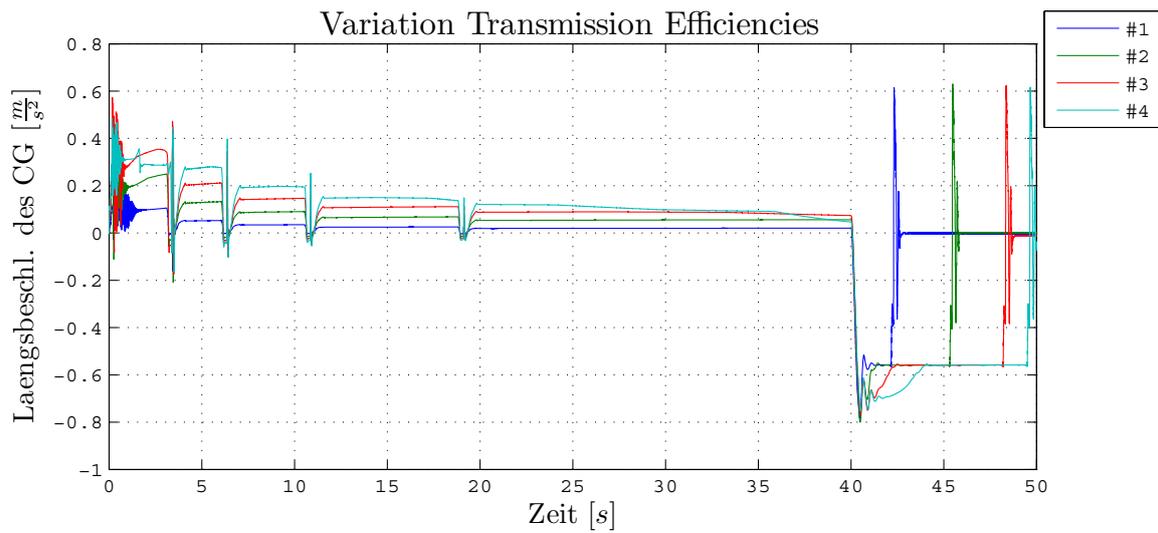


Abbildung A.6

Nr. 007

Varierte Variablen Front break torque

Variationsschritte

Parameter	1	2	3	4
max. Torq. [Nm]	250	250	250	250
max. Speed [$\frac{1}{min}$]	6200	6200	6200	6200
idle Speed [$\frac{1}{min}$]	600	600	600	600
Front Brk. Torq. [Nm]	50	100	150	250
Rear Brk. Torq. [Nm]	70	70	70	70
Effiec. [1]	0.90	0.90	0.90	0.90

Tabelle A.8

Ausgangssignale

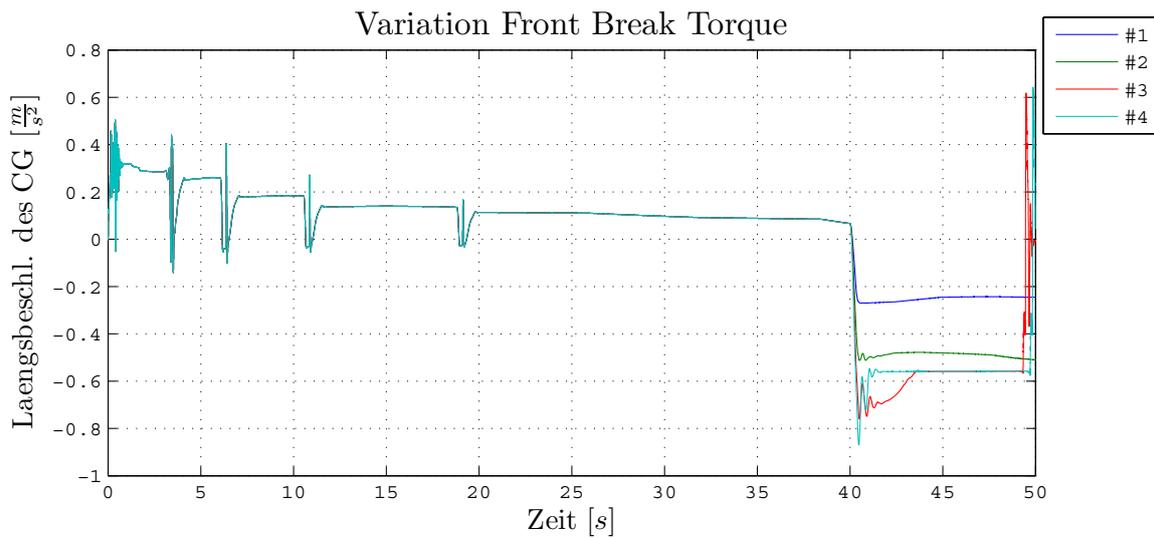


Abbildung A.7

Nr. 008

Varierte Variablen Rear break torque

Variationsschritte

Parameter	1	2	3	4
max. Torq. [Nm]	250	250	250	250
max. Speed [$\frac{1}{min}$]	6200	6200	6200	6200
idle Speed [$\frac{1}{min}$]	600	600	600	600
Front Brk. Torq. [Nm]	150	150	150	150
Rear Brk. Torq. [Nm]	50	100	150	200
Effiec. [1]	0.90	0.90	0.90	0.90

Tabelle A.9

Ausgangssignale

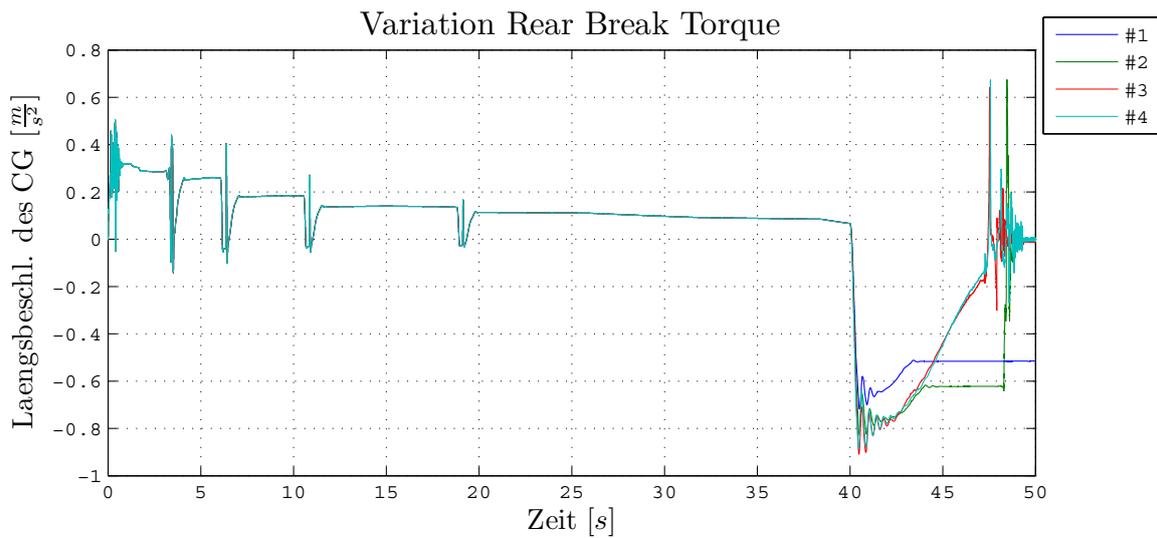


Abbildung A.8

Nr. 009

Varierte Variablen Rear break torque, front break torque

Variationsschritte

Parameter	1	2	3	4
max. Torq. [Nm]	250	250	250	250
max. Speed [$\frac{1}{min}$]	6200	6200	6200	6200
idle Speed [$\frac{1}{min}$]	600	600	600	600
Front Brk. Torq. [Nm]	50	100	150	200
Rear Brk. Torq. [Nm]	30	40	70	70
Effiec. [1]	0.90	0.90	0.90	0.90

Tabelle A.10

Ausgangssignale

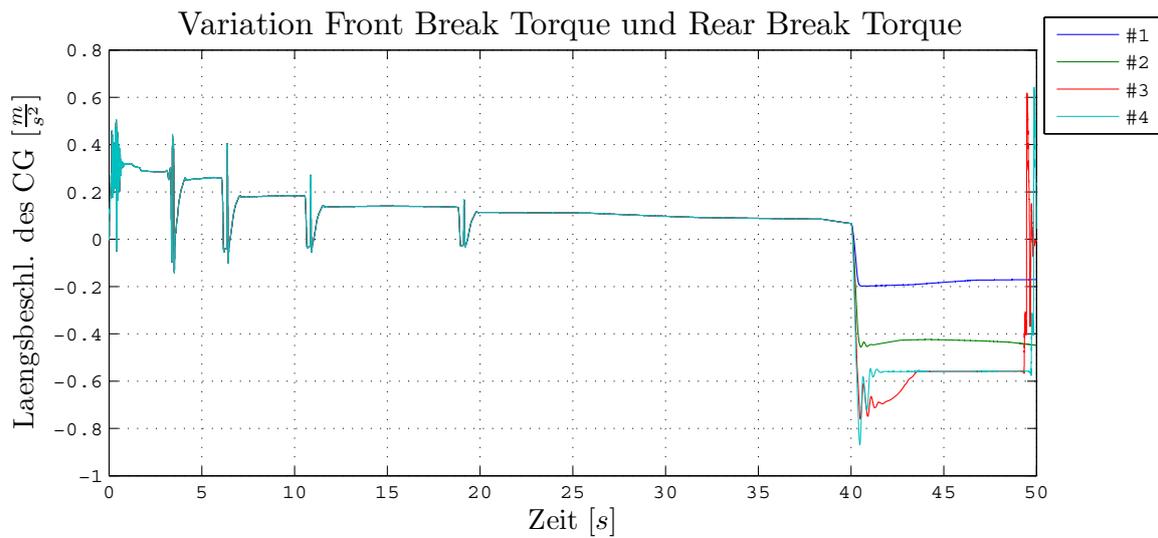


Abbildung A.9

Nr. 010

Variierte Variablen max. Motor Speed

Variationsschritte

Parameter	1	2	3	4
max. Torq. [Nm]	300	300	300	300
max. Speed [$\frac{1}{min}$]	5000	6200	7200	8200
idle Speed [$\frac{1}{min}$]	600	600	600	600
Front Brk. Torq. [Nm]	150	150	150	150
Rear Brk. Torq. [Nm]	70	70	70	70
Effiec. [1]	0.90	0.90	0.90	0.90

Tabelle A.11

Ausgangssignale

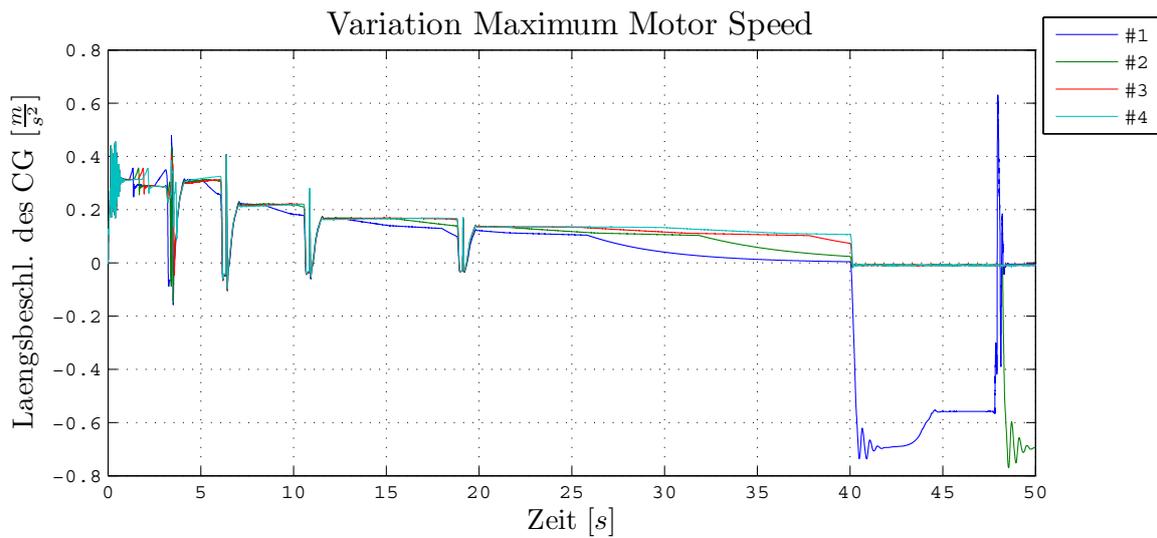


Abbildung A.10

A.2 Querdynamik

Nr.	A	B	C	D	E	F
01	X					
02	X	X				
03	X	X	X	X	X	X
04		X				
05		X	X	X		
06		X			X	X
07			X	X		
08			X	X	X	X
09					X	X
10			X			
11				X		

Tabelle A.12: Variationskombinationen Querdynamik

Parameterbeschreibungen:

- A** Steering at Ground bei 23° Gearbox output [deg] (Kurz: Steer@Gr.)
- B** Steer Compliance bei ± 1 Nm Kinpin Moment [deg] (Kurz: Steer Compl.)
- C** Max. Aligning Moment (Skalierung einer Kennlinie) [Nm] (Kurz: Al. Mom.)
- D** Max. Lateral Tire Force (Skalierung einer Kennlinie) [N] (Kurz: Lat. Tire F.)
- E** Max. Spring Force Extension (Skalierung einer Kennlinie) [N] (Kurz: Spring F. Ex.)
- F** Max. Spring Force Compression (Skalierung einer Kennlinie) [N] (Kurz: Spring F. Comp.)

Nr. 001

Variierte Variablen Steer@Gr.

Variationsschritte

Parameter	1	2	3	4
Steer@Gr. [deg]	06.0	16.0	26.0	36.0
Steer Compl. [deg]	0.01	0.01	0.01	0.01
Al. Mom. [Nm]	97.51	97.51	97.51	97.51
Lat. Tire F. [N]	4498.1	4498.1	4498.1	4498.1
Spring F. Ex. [N]	9691.0	9691.0	9691.0	9691.0
Spring F. Comp. [N]	9771	9771	9771	9771

Ausgangssignale

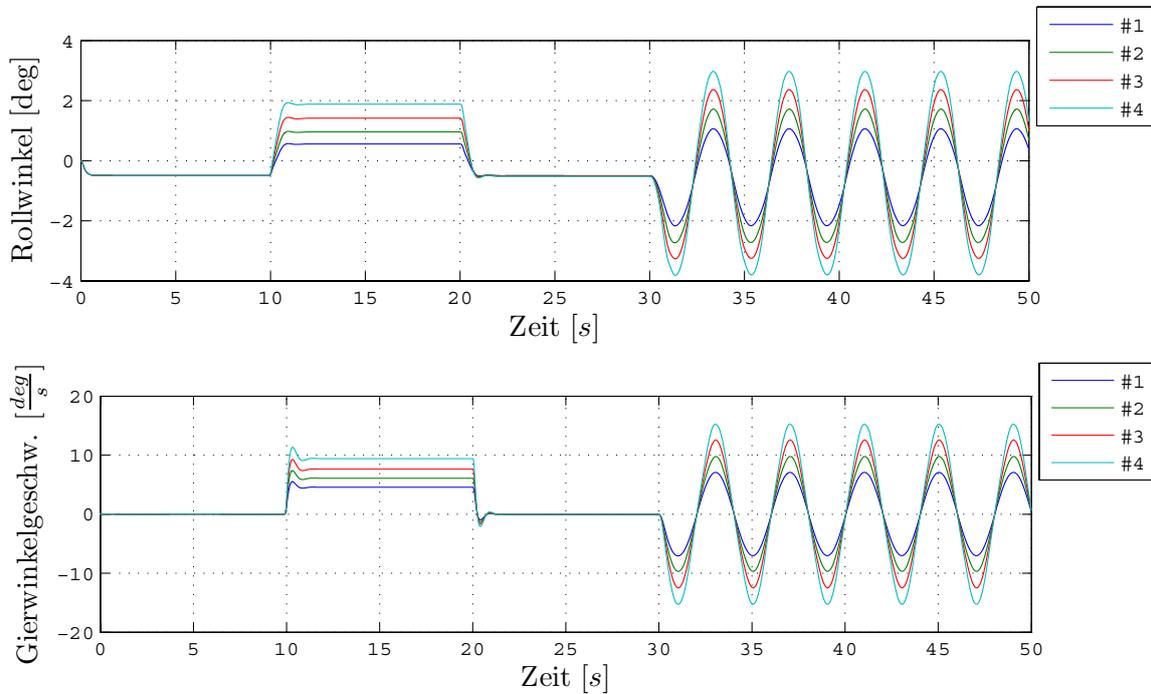


Abbildung A.11: Ausgangsverläufe Nr. 001

Nr. 002

Variierte Variablen Steer@Gr., Steer Compl.

Variationsschritte

Parameter	1	2	3	4
Steer@Gr. [deg]	06.0	16.0	26.0	36.0
Steer Compl. [deg]	0.01	0.05	0.10	0.15
Al. Mom. [Nm]	97.51	97.51	97.51	97.51
Lat. Tire F. [N]	4498.1	4498.1	4498.1	4498.1
Spring F. Ex. [N]	9691.0	9691.0	9691.0	9691.0
Spring F. Comp. [N]	9771	9771	9771	9771

Ausgangssignale

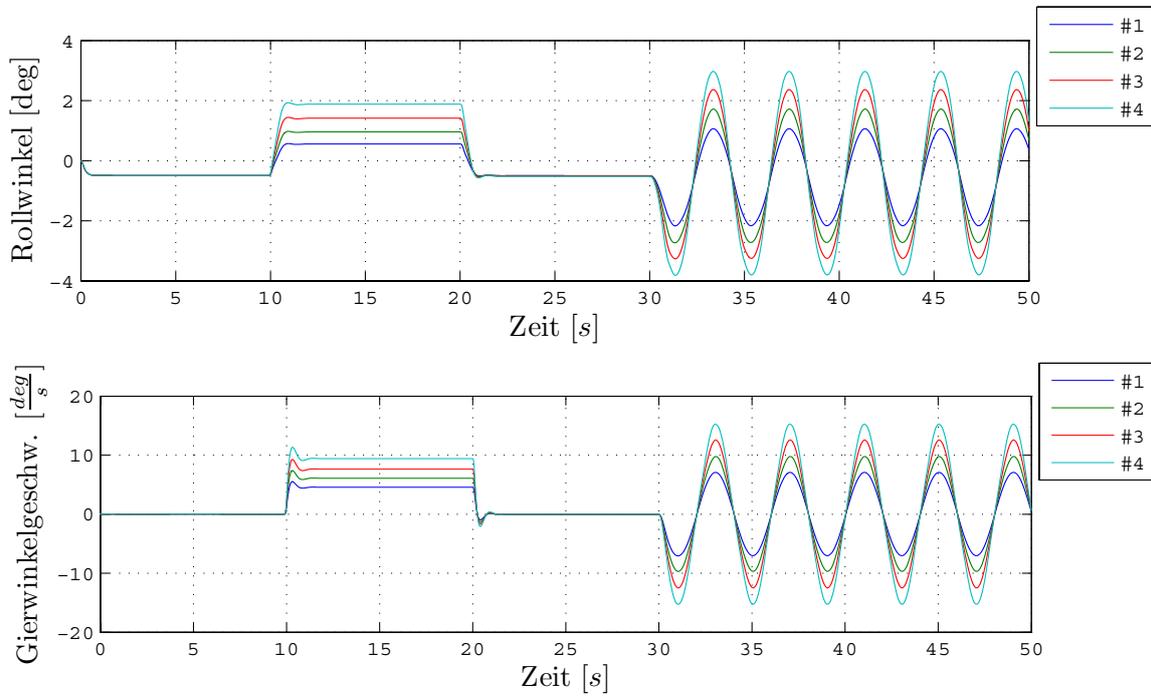


Abbildung A.12: Ausgangsverläufe Nr. 002

Nr. 003

Varierte Variablen Alle Variablen

Variationsschritte

Parameter	1	2	3	4
Steer@Gr. [deg]	06.0	16.0	26.0	36.0
Steer Compl. [deg]	0.01	0.05	0.10	0.15
Al. Mom. [Nm]	27.5123	47.5123	67.5123	97.5123
Lat. Tire F. [N]	1498.1	2498.1	3498.1	4498.1
Spring F. Ex. [N]	0.2·9691.0	0.4·9691.0	0.6·9691.0	0.8·9691.0
Spring F. Comp. [N]	0.2·9771	0.4·9771	0.6·9771	0.8·9771

Ausgangssignale

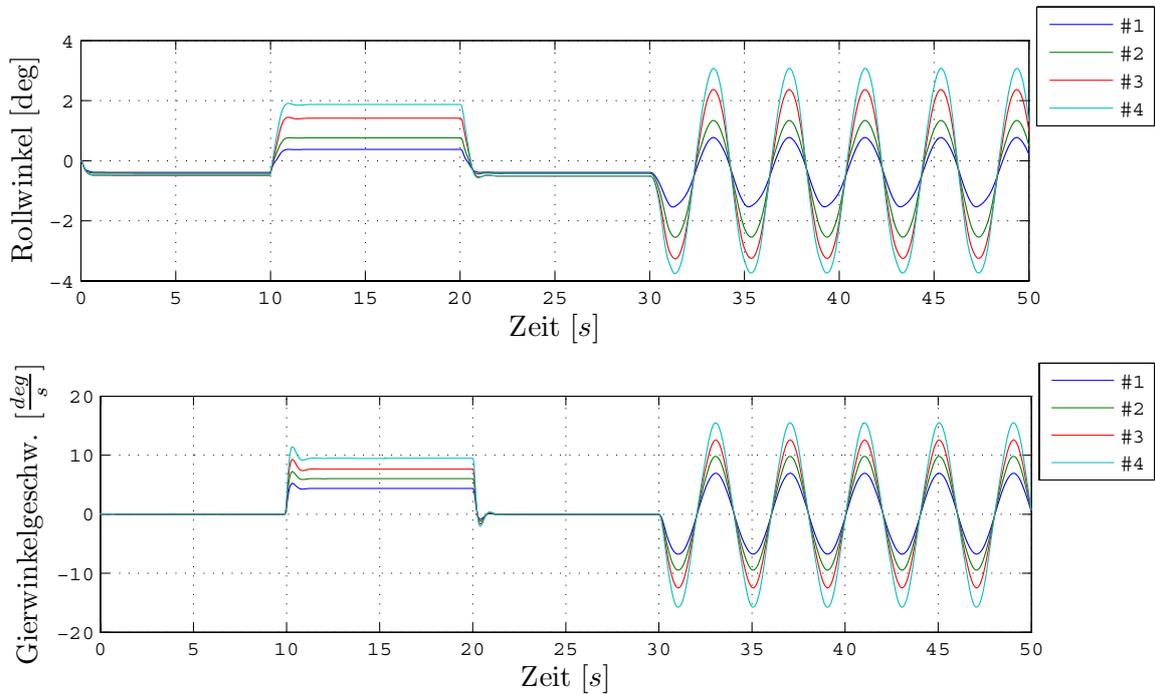


Abbildung A.13: Ausgangsverläufe Nr. 003

Nr. 004

Varierte Variablen Steer Compl.

Variationsschritte

Parameter	1	2	3	4
Steer@Gr. [deg]	26.0	26.0	26.0	26.0
Steer Compl. [deg]	0.01	0.05	0.10	0.15
Al. Mom. [Nm]	97.51	97.51	97.51	97.51
Lat. Tire F. [N]	4498.1	4498.1	4498.1	4498.1
Spring F. Ex. [N]	9691.0	9691.0	9691.0	9691.0
Spring F. Comp. [N]	9771	9771	9771	9771

Ausgangssignale

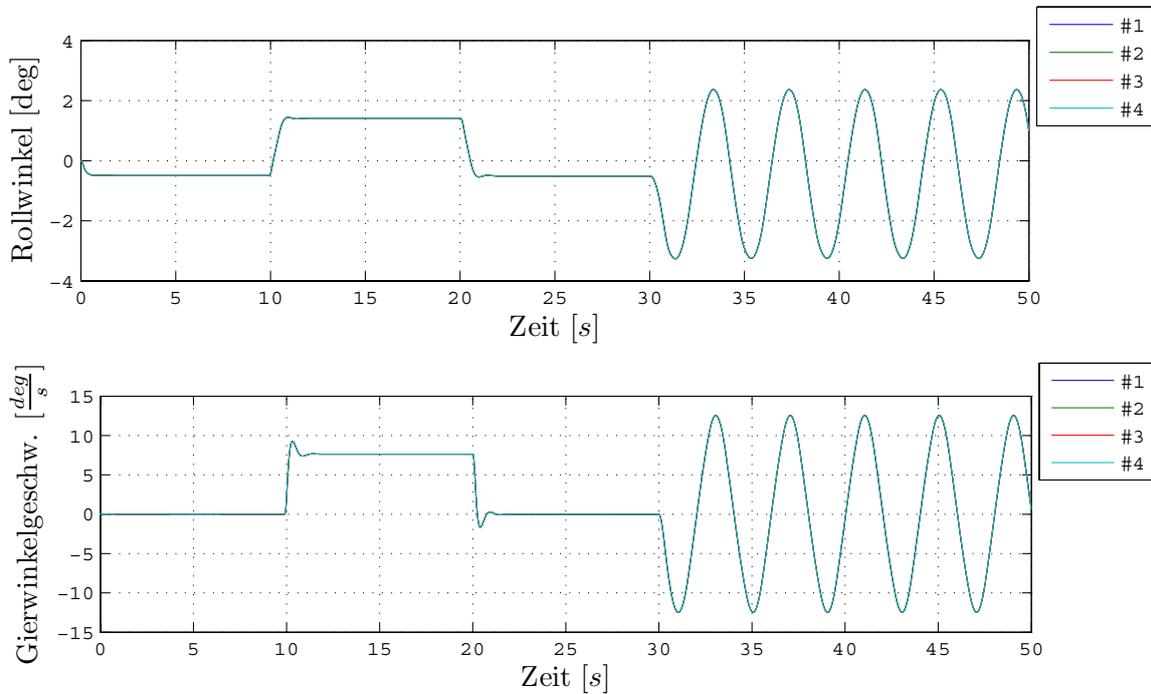


Abbildung A.14: Ausgangsverläufe Nr. 004

Nr. 005

Variierte Variablen Steer Compl., Al. Mom., Lat. Tire F.

Variationsschritte

Parameter	1	2	3	4
Steer@Gr. [deg]	26.0	26.0	26.0	26.0
Steer Compl. [deg]	0.01	0.05	0.10	0.15
Al. Mom. [Nm]	37.51	57.51	77.51	97.51
Lat. Tire F. [N]	1498.1	2498.1	3498.1	4498.1
Spring F. Ex. [N]	9691.0	9691.0	9691.0	9691.0
Spring F. Comp. [N]	9771	9771	9771	9771

Ausgangssignale

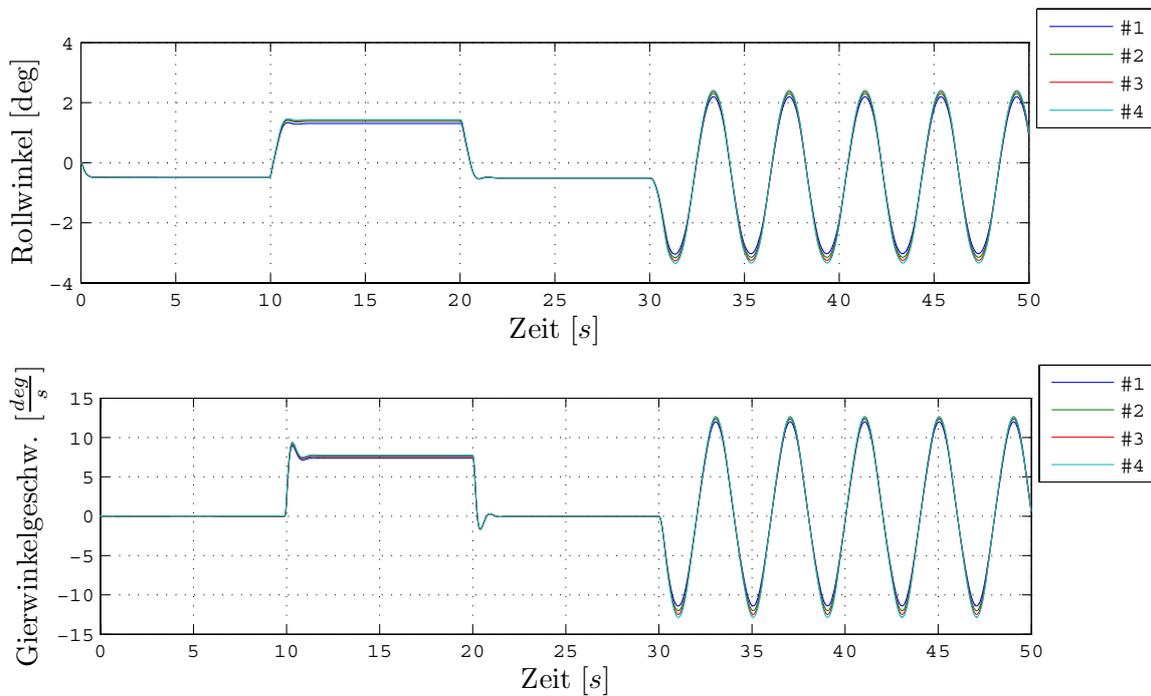


Abbildung A.15: Ausgangsverläufe Nr. 005

Nr. 006

Variierte Variablen Steer Compl., Spring F. Ex., Spring F. Comp.

Variationsschritte

Parameter	1	2	3	4
Steer@Gr. [deg]	26.0	26.0	26.0	26.0
Steer Compl. [deg]	0.01	0.05	0.10	0.15
Al. Mom. [Nm]	97.51	97.51	97.51	97.51
Lat. Tire F. [N]	4498.1	4498.1	4498.1	4498.1
Spring F. Ex. [N]	0.2·9691.0	0.4·9691.0	0.6·9691.0	0.8·9691.0
Spring F. Comp. [N]	0.2·9771	0.4·9771	0.6·9771	0.8·9771

Ausgangssignale

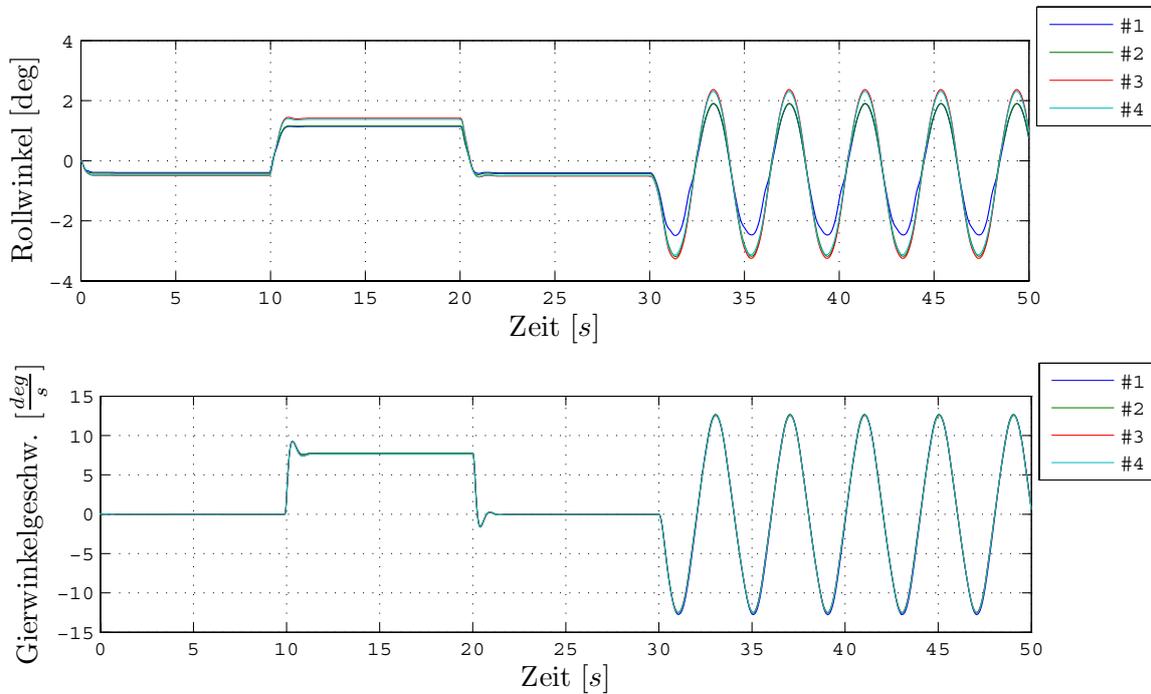


Abbildung A.16: Ausgangsverläufe Nr. 006

Nr. 007

Variierte Variablen Al. Mom., Lat. Tire F.

Variationsschritte

Parameter	1	2	3	4
Steer@Gr. [deg]	26.0	26.0	26.0	26.0
Steer Compl. [deg]	0.01	0.01	0.01	0.01
Al. Mom. [Nm]	37.51	57.51	77.51	97.51
Lat. Tire F. [N]	1498.1	2498.1	3498.1	4498.1
Spring F. Ex. [N]	9691.0	9691.0	9691.0	9691.0
Spring F. Comp. [N]	9771	9771	9771	9771

Ausgangssignale

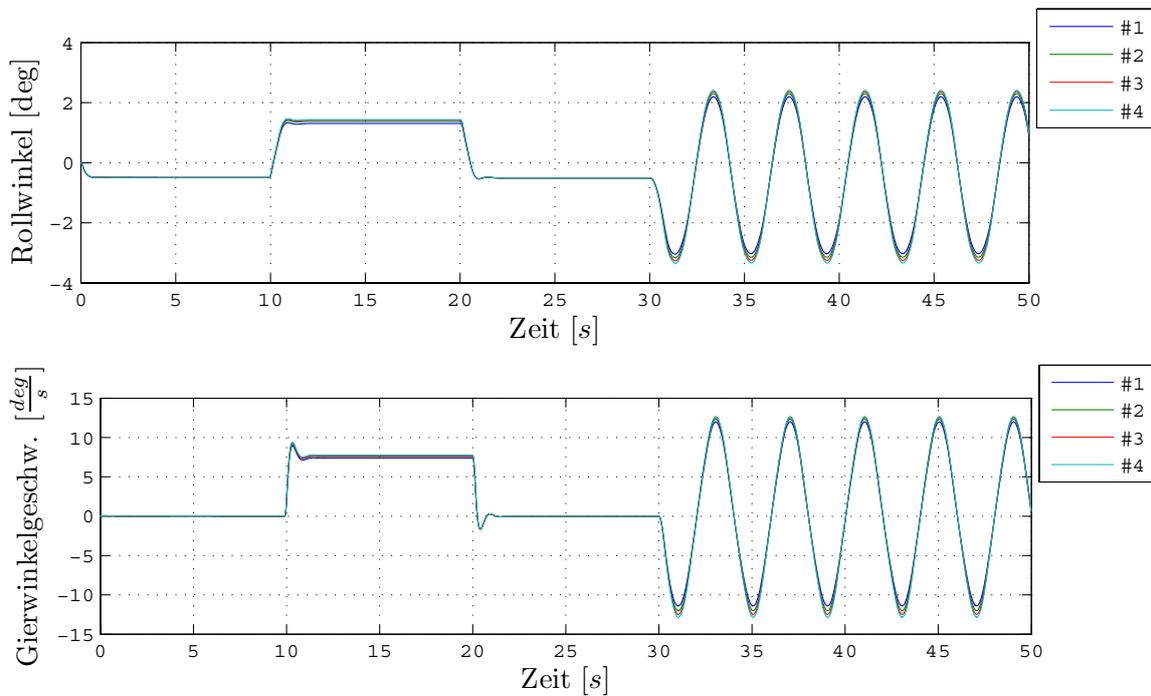


Abbildung A.17: Ausgangsverläufe Nr. 007

Nr. 008

Variierte Variablen Al. Mom., Lat. Tire F., Spring F. Ex., Spring F. Comp.

Variationsschritte

Parameter	1	2	3	4
Steer@Gr. [deg]	26.0	26.0	26.0	26.0
Steer Compl. [deg]	0.01	0.01	0.01	0.01
Al. Mom. [Nm]	37.51	57.51	77.51	97.51
Lat. Tire F. [N]	1498.1	2498.1	3498.1	4498.1
Spring F. Ex. [N]	0.2·9691.0	0.4·9691.0	0.6·9691.0	0.8·9691.0
Spring F. Comp. [N]	0.2·9771	0.4·9771	0.6·9771	0.8·9771

Ausgangssignale

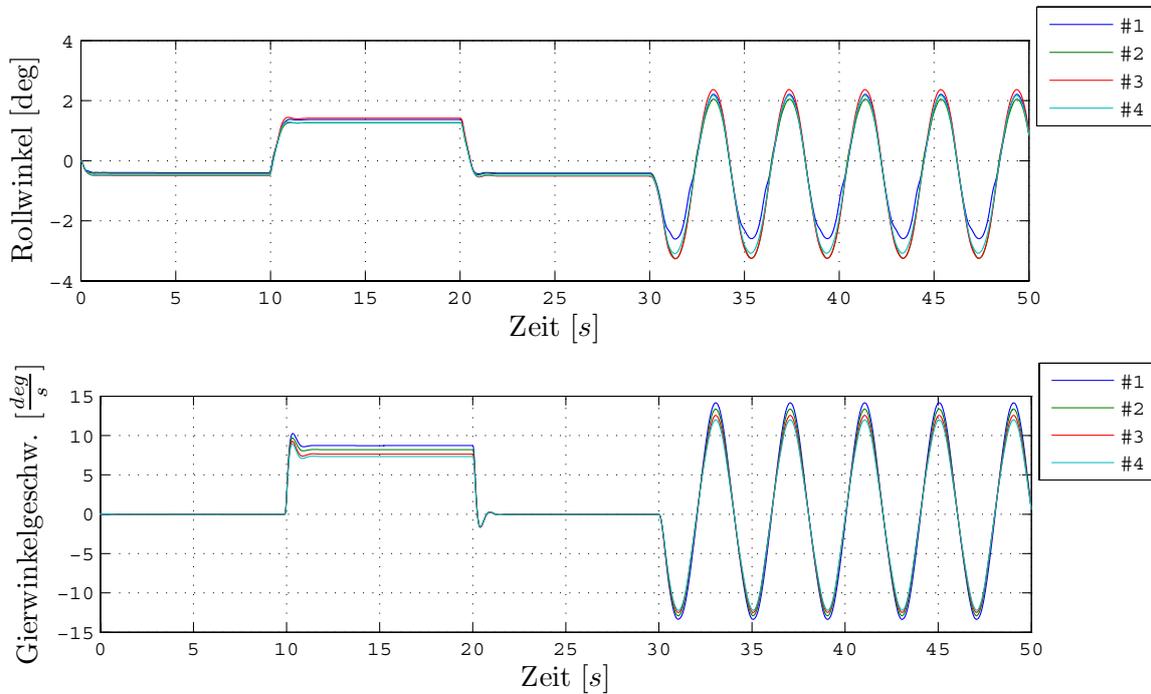


Abbildung A.18: Ausgangsverläufe Nr. 008

Nr. 009

Variierte Variablen Spring F. Ex., Spring F. Comp.

Variationsschritte

Parameter	1	2	3	4
Steer@Gr. [deg]	26.0	26.0	26.0	26.0
Steer Compl. [deg]	0.01	0.01	0.01	0.01
Al. Mom. [Nm]	97.51	97.51	97.51	97.51
Lat. Tire F. [N]	4498.1	4498.1	4498.1	4498.1
Spring F. Ex. [N]	0.2·9691.0	0.4·9691.0	0.6·9691.0	0.8·9691.0
Spring F. Comp. [N]	0.2·9771	0.4·9771	0.6·9771	0.8·9771

Ausgangssignale

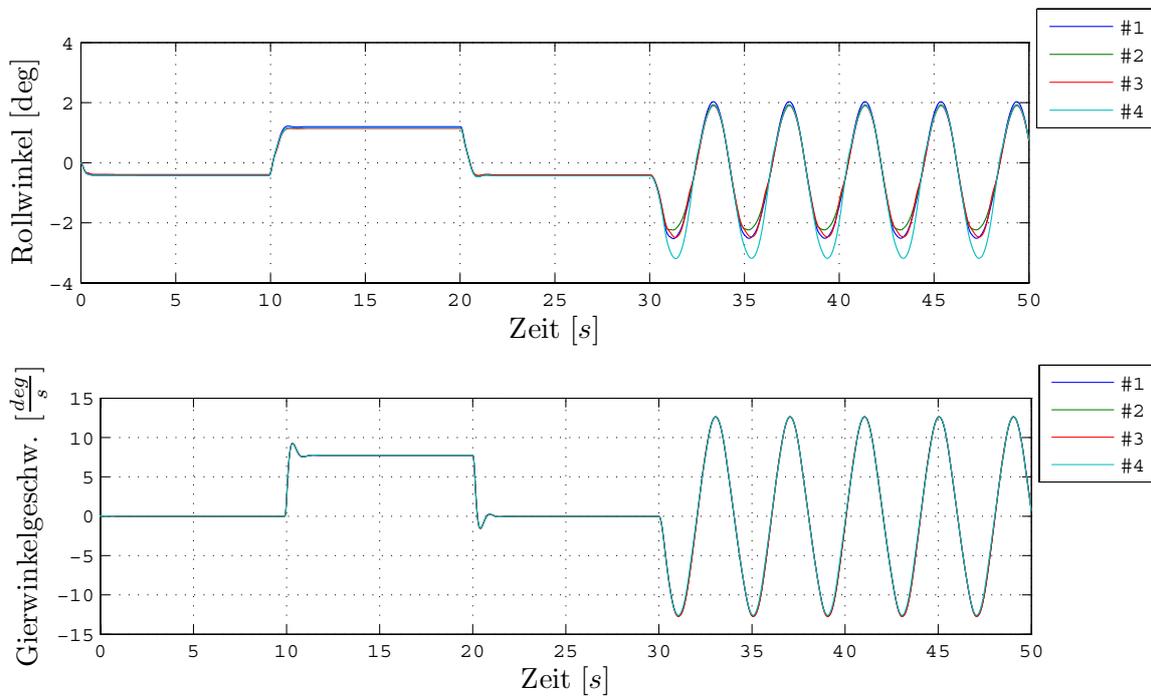


Abbildung A.19: Ausgangsverläufe Nr. 009

Nr. 010

Varierte Variablen Al. Mom.

Variationsschritte

Parameter	1	2	3	4
Steer@Gr. [deg]	26.0	26.0	26.0	26.0
Steer Compl. [deg]	0.01	0.01	0.01	0.01
Al. Mom. [Nm]	37.51	57.51	77.51	97.51
Lat. Tire F. [N]	4498.1	4498.1	4498.1	4498.1
Spring F. Ex. [N]	9691.0	9691.0	9691.0	9691.0
Spring F. Comp. [N]	9771	9771	9771	9771

Ausgangssignale

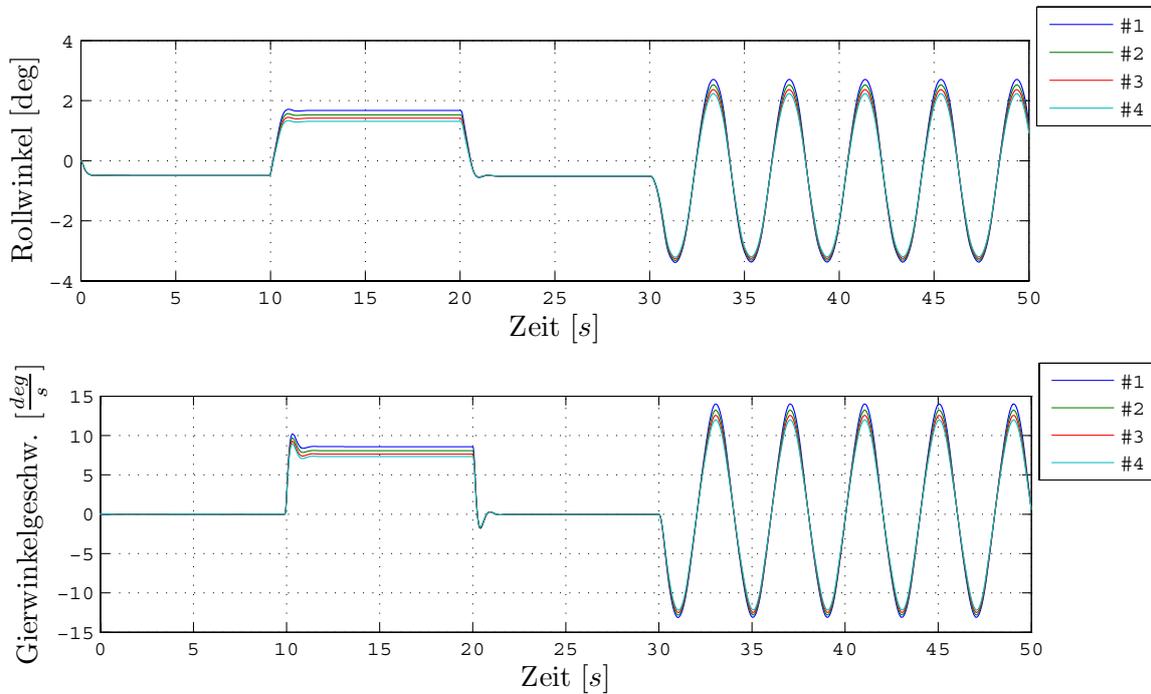


Abbildung A.20: Ausgangsverläufe Nr. 010

Nr. 011

Varierte Variablen Lat. Tire F.

Variationsschritte

Parameter	1	2	3	4
Steer@Gr. [deg]	26.0	26.0	26.0	26.0
Steer Compl. [deg]	0.01	0.01	0.01	0.01
Al. Mom. [Nm]	97.51	97.51	97.51	97.51
Lat. Tire F. [N]	1498.1	2498.1	3498.1	4498.1
Spring F. Ex. [N]	9691.0	9691.0	9691.0	9691.0
Spring F. Comp. [N]	9771	9771	9771	9771

Ausgangssignale

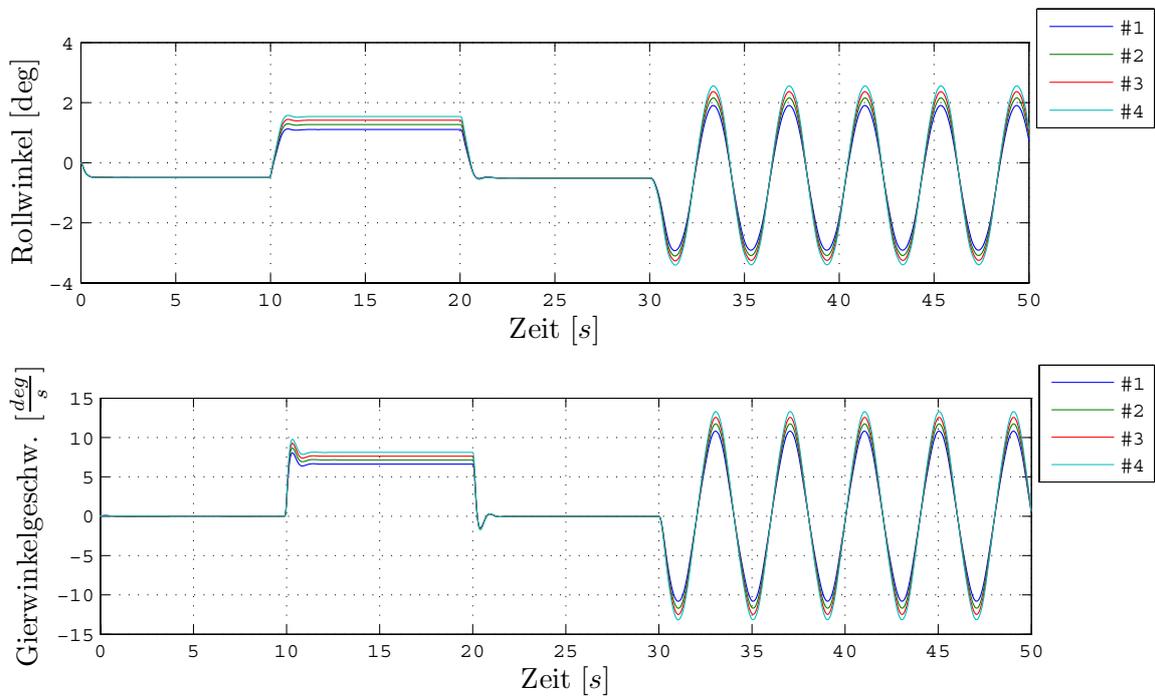


Abbildung A.21: Ausgangsverläufe Nr. 011

ANHANG B

Programme und Modelle

B.1 Schnittstellen

Listing B.1: Matlab Interface-Funktion

```

1 %
2 % PAR    vector with parameter values
3 % BIAS   vector with bias parameter values of the
4 %                               current time slice
5 % T      vector with time axis of the current time slice
6 % U      array with all inputs for the current time slice
7 %
8 % Y      calculated output for the current time slice
9
10 %=====
11
12 % unknown parameters
13 MotorTorque = PAR(1:11);
14 gearDelay = PAR(12);
15 timeClutch = PAR(13);
16 MaximumSpeed = PAR(14);
17 IdleSpeed = PAR(15);
18 TransEff = PAR(16);
19 FrontBreakTorque = PAR(17);
20 RearBreakTorque = PAR(18);
21 clutchOffset = PAR(19);
22 clutchGain = PAR(20);
23 PT1_Filter_Value = PAR(21);
24 cw = PAR(22);
25 Steer_Comp = PAR(24);
26 Steering_Kinemat = PAR(25);
27 Spring_Character = PAR(26);
28 Lateral_Tire_Force = PAR(27);
29 Aligning_Moment = PAR(28);
30 ShckAbs = PAR(29);
31 AuxMx = PAR(30);
32 AligningMaxAngle = PAR(31);
33
34 %% Timeshift of Gearchoice
35 % We need to detect changes in gear selection
36 % before the changes occur.
37 % Because of this, we shift the whole channel for
38 % a specific amount and delay it while simulation

```

```

39 % for the same amount.
40
41 GearChoice = zeros(size(U,1),1);
42
43 dT = 0.01;
44 Tv = gearDelay;
45 Tv = 0;
46
47 ns = Tv/dT;
48 ns = round(ns);
49
50 for i=1:(size(U,1)-(ns))
51     GearChoice(i) = U(i+(ns),3);
52 end
53
54 %% Assign Workspace Parameter
55 % some of the Parameters are for the Simulink-Modell,
56 % not for CarSim. They're assigned to workspace here
57
58 assignin('base','U',[[T-ones(size(T,1),1).*T(1)] U]);
59 assignin('base','T',T);
60 assignin('base','UGearChoice_timeshifted',...
61         [[T-ones(size(T,1),1).*T(1)] GearChoice]);
62 assignin('base','gearDelay', gearDelay);
63 assignin('base','timeClutch', timeClutch);
64 assignin('base','clutchOffset', clutchOffset);
65 assignin('base','clutchGain', clutchGain);
66 assignin('base','PT1_Filter_Value', PT1_Filter_Value);
67 assignin('base','cw', cw);
68
69
70 %% Write CarSim-Parameter
71
72 createParsfile('mdlFiles\params.par', ...
73     MotorTorque,...
74     MaximumSpeed,...
75     IdleSpeed,...
76     FrontBreakTorque,...
77     RearBreakTorque,...
78     TransEff);
79
80 SprRat = 9771/9691;
81

```

```

82 createParsfileQuerdyn( 'mdlFiles\paramsQuer.par' , ...
83     Steering_Kinemat , ...
84     Steer_Comp , ...
85     Aligning_Moment , ...
86     AligningMaxAngle , ...
87     Lateral_Tire_Force , ...
88     Spring_Character/SprRat , ...
89     Spring_Character , ...
90     ShckAbs , ...
91     AuxMx);
92
93
94 %% Compute CarSim Modell
95
96 cd mdlFiles
97 simout = runSimulation( 'calcSimulation' , ...
98     T(size(T,1)) - T(1), dT);
99
100 %% Write results
101
102 dT = 0.001;
103 Y = zeros(size(T,1),5);
104
105 if( int32( ((T(size(T,1))-T(1))*1/dT) + 1 ) ...
106     <= size(simout.signals.values,1) )
107     for k = 1:size(T,1)
108         Y(k,:) = ...
109             simout.signals.values ...
110             (int32((T(k)-T(1))*1/dT) + 1, :);
111     end
112 else
113     disp('Too_short_output_vector_detected!')
114 end
115
116
117 if (size(Y,1) ~= size(T,1))
118     disp('Uhm, trouble with output size ....');
119     out= ones(size(T,1),1) .* Inf;
120 end
121
122 Y(:,3) = Y(:,3) .* 9.81;    % Transform g's to m/s^2

```

ANHANG C

Daten und Spezifikationen

Test report nach ISO 10392					
1.	Vehicle identification (Audi A6, BS-VC 999, FIN: WAUZZZ4B93N030051)				
1.1	Marke:	Audi			
1.2	Model:	4B (A6)			
1.3	Type:	3.0 (Otto)			
1.4	Tyres:	Continental ContiWinter Contact			
	Tyre size, front:	205/55 R16	rear:	205/55 R16	
	Tyre pressure, front:	3,0	rear:	3,2	
1.5	Suspension setting (if adjustable)				
2.	Measurement data				
2.1	Loading conditions (description of the load, e.g. dummies, luggage, etc. and their locations in the vehicle):				
	mit Messtechnik + Fahrer + Mitfahrer hinten links <i>93 kg 85 kg</i>				
2.2	Masses (loads):				
	Front left:	534,5	kg	Rear left:	455,5 kg
	Front right:	477,5	kg	Rear right:	422,0 kg
	Front total:	1012	kg	Rear total:	877,5 kg
	Total vehicle:				1889,5 kg
2.3	Track, front:	1560	mm	rear:	1560 mm
2.4	Wheelbase, left:	2765	mm	right:	2770 mm
	Wheelbase				2767,5 mm
2.5	Static loaded radius:				
	Front left:	296	mm	Rear left:	300 mm
	Front right:	298	mm	Rear right:	300 mm
3.	Test results				
3.1	Longitudinal displacement between centre of front axle and centre of gravity:			$x =$	1285 mm
3.2	Distance between the longitudinal median plane of the vehicle and the centre of gravity (positive to the left)			$y =$	37 mm
3.3	Height of the centre of gravity above the ground				
3.3.1	Test with lifted front axle				
	lifting height:				600 mm
	lifting angle:				12,5 °
	rear axle load, left:	496	kg	right:	425,5 kg
	rear axle load total (vehicle inclined):				921,5 kg
	height of centre of gravity, front axle lifted:				$Z_1 =$ 590 mm
3.3.1	Test with lifted rear axle				
	lifting height:				600 mm
	lifting angle:				12,5 °
	front axle load, left:	561	kg	right:	495 kg
	front axle load total (vehicle inclined):				1056 kg
	height of centre of gravity, rear axle lifted:				$Z_2 =$ 587 mm

$$Z_3 = \frac{Z_2 + Z_1}{2} = 588,5$$

Es folgt die genaue Spezifikation der Fahrmanöver. Die Durchführung der Manöver geschah im Rahmen der Arbeit von [Sun07b], aus welcher auch die nachfolgenden Spezifikationen entnommen wurden.

Stationäre Kreisfahrt (DIN ISO 4138)

1. Teststrecke

Ein Radius von 30m wird als Minimum angesehen, und die Windgeschwindigkeit soll 7m/s nicht überschreiten. Bei größeren Radien mit entsprechend höheren Geschwindigkeiten ist eine niedrigere Windgeschwindigkeit wünschenswert.

2. Testverfahren

Das Fahrzeug ist mit der niedrigsten realisierbaren Geschwindigkeit auf dem ausgewählten Kurs zu fahren. Die Messsignale sind mit fixierter Lenkrad- und Drosselklappenstellung aufzuzeichnen. Danach wird das Fahrzeug mit der nächsten für die Messung relevanten Geschwindigkeit gefahren. Falls die Messgeräte zwischen den Tests neu eingestellt werden müssen, kann zu diesem Zweck angehalten werden. Die Daten sollten in Höchstschriftweiten von 0.5m/s² aufgenommen werden. Wenn sich große Änderungen in den Messergebnissen in Abhängigkeit von der Querschleunigung ergeben, kann es nützlich sein, die Messergebnisse bei kleineren Schrittweiten aufzunehmen. Zu jeder Querschleunigungsstufe sollen die Lenkrad- und Drosselklappenstellung während der Messdatenerfassung so konstant wie möglich gehalten werden. Unabhängig vom gewählten Radius ist der vorgegebene Kurs mit einem Spielraum von 0.3m zu jeder Seite einzuhalten. Die Messdaten sind bei jeder Querschleunigungsstufe für mindestens 3s konstanter Fahrt aufzuzeichnen. Es wird empfohlen, jeweils die höchste bei der Fahrgeschwindigkeit mögliche Gangstufe einzulegen. Die Messdaten sind bis zum Erreichen des Grenzbereiches aufzuzeichnen, bei dem es nicht mehr möglich ist, den stationären Zustand aufrechtzuerhalten. Die Messdaten sind für Rechts- und Linkskurven aufzunehmen. Mit der anderen Kurvenrichtung ist erst nach Abschluss der Datenaufnahme für die erste Richtung zu beginnen. Alternativ können die Messdaten auch, beginnend bei der kleineren Beschleunigungsstufe, bei jeder Stufe abwechselnd für beide Seiten gewonnen werden. Die gewählte Methode ist im Datenblatt zu vermerken.

3. Die Messung

Folgende Größen sind unbedingt zu messen:

- a) Lenkradwinkel

- b) Querschleunigung (diese Größe kann auch von anderen Parametern abgeleitet werden)
- c) Giergeschwindigkeit
- d) Längsgeschwindigkeit
- e) Lenkradmoment
- f) Wankwinkel
- g) Schwimmwinkel

DIN ISO 7401, Sinuswedeltest und Lenkwinkelsprung

1. Teststrecke

Für Versuche mit stochastischer Lenkwinkeleingabe muss die Fahrbahnoberfläche mindestens 8 m breit sein und so lang, dass zusätzlich zu den erforderlichen Anlauf- und Bremsstrecken mindestens 30s mit der Testgeschwindigkeit gefahren werden kann. Die Windgeschwindigkeit darf nicht mehr als 7m/s betragen. Für Testgeschwindigkeiten über 30m/s ist eine niedrigere Windgeschwindigkeit wünschenswert.

2. Testverfahren

2.1 Testgeschwindigkeit

Alle Tests müssen mit einer Geschwindigkeit von 80km/h gefahren werden. Wenn höhere oder niedrigere Testgeschwindigkeiten gewählt werden, muss dies in Schritten von 20 km/h erfolgen.

2.2 Lenkradwinkelamplitude

Die Amplitude des Lenkradwinkels muss bei stationärer Kreisfahrt auf einem Radius ermittelt werden, der die vorher gewählte Querschleunigung bei der erforderlichen Testgeschwindigkeit ergibt.

2.3 Sinuswedeltest

Der Fahrzeug muss mit der Testgeschwindigkeit geradeaus gefahren werden. Beginnend mit einer Giergeschwindigkeit von $(0 \pm 0,5)^\circ/s$ muss eine vollständige Periode sinusförmiger Lenkradwinkeleingabe mit einer Lenkfrequenz von $0,5Hz$

erfolgen. Wahlweise wird eine Lenkfrequenz von 1Hz empfohlen. Der zulässige Amplitudenfehler im Vergleich mit der echten Sinuswelle beträgt $\pm 5\%$ des ersten Spitzenwertes. Die erforderliche Querbeschleunigung ist $4m/s^2$. Wahlweise werden Beschleunigungen von $2m/s^2$ und $6m/s^2$ und bis zur Haftgrenze(siehe ISO Technical Report, ISO/TR8275) empfohlen. Die Stellung des Gaspedals darf nicht verändert werden, auch wenn sich die Fahrgeschwindigkeit verringert. Daten müssen sowohl für eine anfangs nach links als auch rechts beginnende Lenkradwinkelingabe ermittelt werden. Alle Daten können erst in einer Richtung, danach in der anderen Richtung aufgezeichnet werden. Alternativ können Daten nacheinander erst in einer Richtung, dann in der anderen Richtung für jede Querbeschleunigung(von der niedrigsten bis zur höchsten) aufgezeichnet werden. Alle Tests müssen mindestens dreimal durchgeführt werden, um Mittelwerte und Standardabweichungen zu erhalten.

2.4 Lenkwinkelsprung

Das Fahrzeug muss mit der Testgeschwindigkeit geradeaus gefahren werden. Beginnend mit einer Giergeschwindigkeit von $(0 \pm 0,5)^\circ/s$ muss das Lenkrad auf einen vorher bestimmten Wert so schnell wie möglich eingeschlagen werden, der einige Sekunden lang beibehalten werden muss oder so lange, bis die gemessenen Variablen der Fahrzeugbewegung einen Beharrungszustand erreicht haben. Die Stellung des Gaspedals darf nicht verändert werden, auch wenn sich die Fahrgeschwindigkeit verringert. Daten müssen sowohl für Linkskurven als auch für Rechtskurven ermittelt werden. Alle Daten können erst in einer Richtung, danach in der anderen Richtung aufgezeichnet werden. Alternativ können Daten nacheinander erst in einer Richtung, dann in der anderen Richtung für jede Beschleunigungsebene, angefangen von der niedrigsten bis zur höchsten, aufgezeichnet werden. Das angewandte Verfahren muss in den allgemeinen Daten vermerkt werden. Die Daten müssen über den gewünschten Lenkradwinkelbereich ermittelt werden. Die erforderliche Querbeschleunigungen von $2m/s^2$ und $6m/s^2$ werden empfohlen. Alle Tests müssen mindestens dreimal durchgeführt werden.

3. Die Messung

Folgende Größen müssen gemessen werden:

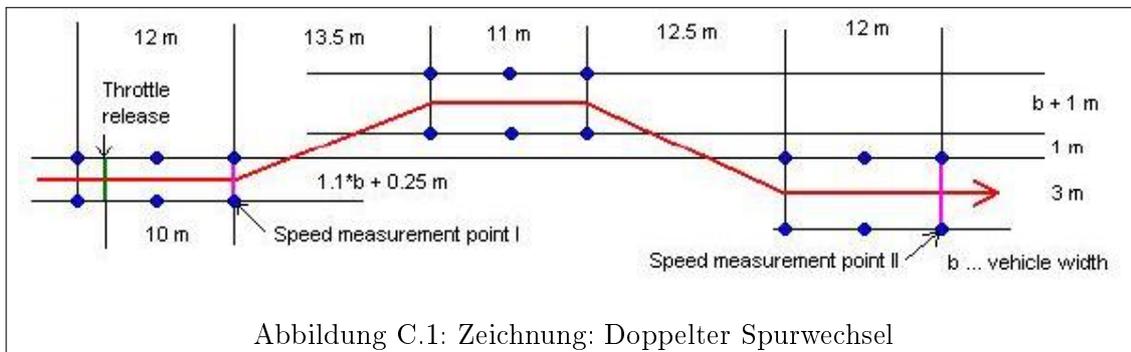
- a) Lenkradwinkel
- b) Querbeschleunigung
- c) Giergeschwindigkeit

- d) der stationäre Schwimmwinkel (der stationäre Schwimmwinkel ist nur bei den Tests mit Lenkwinkelsprungeingabe erforderlich)
- e) Längsgeschwindigkeit
- f) Wankwinkel
- g) Lenkradmoment

ISO-Spurwechsel (DIN ISO 3888)

1. Teststrecke

Die Teststrecke soll wie in den unten angegebenen Zeichnungen aufgebaut werden:



2. Testverfahren:

Der Fahrer tritt die Kupplung 2.0 m vor der Einfahrt in die Fahrgasse voll durch. Es darf während des ganzen folgenden Manövers keine Brems- oder Gaspedal Betätigung erfolgen. Die Anfahrtsgeschwindigkeit soll, anfangend mit 50 km/h, bei jedem neuen Durchlauf iterativ um 2 km/h erhöht werden. Die Iteration soll so lange fortgeführt werden bis keine gültigen Tests mehr gefahren werden können (z.B. konnte die Fahrgasse nicht mehr durchfahren werden ohne das die Verkehrshüte überfahren wurden). Der Fahrer soll drei Testläufe mit maximaler Geschwindigkeit fahren. Dies wird benötigt um Eingangs- und Ausgangs Variationen zu berechnen.

3. Die Messung

Folgende Größen sollen gemessen werden:

- a) Lenkradwinkel

- b) Querbearbeschleunigung
- c) Wankwinkel
- d) Wankwinkelgeschwindigkeit
- e) Quergeschwindigkeit
- f) Längsgeschwindigkeit
- g) Schwimmwinkel
- h) Lenkradmoment
- i) Giergeschwindigkeit

Bremsen in der Kurve (DIN ISO 7975)

Testverfahren

Das Hauptziel dieses Tests ist, die Auswirkung des Bremsens auf das Richtungsverhalten eines Fahrzeugs zu bestimmen, dessen stationäre kreisförmige Bewegung nur durch die Bremsaktion gestört wird. Die Reifendrucke können aufgezeichnet werden.

Ausgangsbedingungen:

Das Fahrzeug ist in die stationäre Ausgangsbedingung zu bringen, indem die gewünschte Kreisbahn(siehe Tabelle) in dem höchsten Gang befahren wird, der mit den Testbedingungen bei jeder Höhe der in Tabelle definierten Querbearbeschleunigung vereinbar ist. Die Stellung des Lenkrades und des Fahrpedals soll so konstant wie möglich gehalten werden. Die Ausgangsbedingungen vor der Bremsbetätigung sind in Tabelle definiert.

Brems-Verfahren:

Die Freigabe des Fahrpedals und die Bremsbetätigung sollen so schnell wie möglich erfolgen. Die Kupplung kann sofort oder am Ende des Versuchslaufs getrennt werden; welche der Möglichkeiten ausgewählt wurde, soll in den allgemeinen Daten in Anhang angegeben werden. Nach der Anstiegszeit (weniger als 0.4s) soll der Druck im Bremssystem oder die Bremspedalkraft oder der Bremspedalweg so konstant wie möglich gehalten werden(ein einstellbarer Anschlag unter dem Pedal kann dabei helfen) und das Lenkrad soll festgelegt sein, bis der Versuchslauf

beendet ist. Die Versuchsläufe für jede Kombination von Radien und Querbeschleunigungen, wie in Tabelle (oben) definiert, sind bei ansteigendem Level des Drucks, der Pedalkraft oder des Pedalwegs durchzuführen, bis eines der Räder blockiert (falls möglich). Der Test kann darüber hinaus, indem weitere Räder blockieren, weitergeführt werden, bis alle Räder blockiert sind; jedoch können sich aus dem Testen unter dieser Bedingung rasche und ausgeprägte Änderungen der Reifeneigenschaften ergeben, die eine große Variationsbreite in den Testergebnissen verursachen können. Die minimale Bremsbetätigung soll einer mittleren Längsverzögerung von 2m/s^2 entsprechen und dann in Schritten von nicht mehr als 1m/s^2 ansteigen. Wenn die Ergebnisse sehr rasch mit der Längsverzögerung variieren, kann es besser sein, kleinere Schritte zu wählen. Der Test soll in beiden Kreisrichtungen durchgeführt werden. Es ist notwendig, folgende Größen zu messen:

- a) Lenkradwinkel
- b) Druck in dem Bremskreis, der wenigstens eine der Bremsen an den Vorderrädern beaufschlagt oder Bremspedalkraft oder Bremspedalweg
- c) Querbeschleunigung
- d) Längsverzögerung
- e) Längsgeschwindigkeit
- f) Giergeschwindigkeit
- g) Bremsweg
- h) Wankwinkel
- i) Nickwinkel
- j) Schwimmwinkel
- k) Quergeschwindigkeit
- l) Seitliche Abweichung des Schwerpunkts vom Referenzweg

Verzeichnisstruktur der CD

- doc
 - Dokumentation
 - Organisation
 - Präsentationen
- Literatur
- Modelle & Scripte
 - 01-SensAnalyse
 - 03-FitLabParameterEstimation
 - 05-CarSim Modell
- Tools
 - Analyse2_070913
 - FitLab