

Masterarbeit

# Entwicklung einer Emissionsrollen- Simulationsumgebung mit Human Driver Interface

Verfasser

Hans Peter Gigerl, BSc.

Eingereicht am Institut für Verbrennungskraftmaschinen und  
Thermodynamik der Technischen Universität Graz

Vorstand:

Univ.-Prof. Dipl.-Ing. Dr. techn. Helmut Eichlseder

Betreuer:

Ao.Univ.-Prof. Dipl.-Ing. Dr. techn. Stefan Hausberger

Externer Betreuer:

Dipl.-Ing. (FH) Georg Seewald

Graz, 24. Jänner 2013



Institut für Verbrennungskraftmaschinen und Thermodynamik  
Vorstand: Univ.-Prof. Dipl.-Ing. Dr. techn. Helmut Eichlseder





In Auftrag von:

**AVL List GmbH**





## Eidesstattliche Erklärung

Deutsche Fassung:

Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10. 11. 2008

Genehmigung des Senates am 1. 12. 2008

### EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Graz, am .....

.....

(Unterschrift)

Englische Fassung:

### STATUTORY DECLARATION

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

.....

date

.....

(signature)

## Vorwort

Im Zuge der Problemstellung immer höher werdender Auslastung der Emissionsrollenprüfstände in der AVL List GmbH entstand im Fachteam Vehicle Testing (DFV) die Idee eines Simulationsaufbaus zur Nachahmung des Betriebs am Rollenprüfstand. Die Anwendung umfasst dabei sowohl das Training des Bedienpersonals als auch das Recruiting neuer Mitarbeiter abseits des produktiven Betriebs am Rollenprüfstand.

Diese Masterarbeit beschäftigt sich im Rahmen des Masterstudiums Maschinenbau mit den Vertiefungen Motor- und Antriebstechnik und Fahrzeugtechnik und -sicherheit mit dem Aufbau des Simulators sowie mit der Recherche und Integration von Hard- und Software, die für die Simulation notwendig sind.

An dieser Stelle möchte ich mich bei dem Fachteamleiter der Fachabteilung DFV, Dipl.-Ing. (FH) Rainer Vögl bedanken, der mir schon vor der Gründung dieser Abteilung die Möglichkeit gab, im Bereich Methodik (DFM) als freier Mitarbeiter Einblicke in die Methodikentwicklung zu erlangen.

Ein weiteres Dankeschön geht an Dipl.-Ing. (FH) Georg Seewald, der diese Masterarbeit stets mit hohem Einsatz betreut hat und auch außerhalb der Arbeitszeit immer für Fragen offen war.

Mein Dank gebührt ebenfalls Ao.Univ.-Prof. Dipl.-Ing. Dr. techn. Stefan Hausberger, welcher die Betreuung vonseiten der Technischen Universität Graz am Institut für Thermodynamik und Verbrennungskraftmaschinen ohne zu zögern übernahm und mir immer zeitnahe Termine und rasche Hilfe ermöglichte.

Auch möchte ich mich bei Dipl.-Ing. (FH) Thomas Wimmeler, Dejan Ciglar, BSc., und Ing. Mario Lengheimer bedanken, die mir stets schnellen Support für die verwendete Soft- und Hardware bereitgestellt haben.

Der Lead Engineer für den Bereich Fahrzeugversuch, Dipl.-Ing. Tom Dein Dias Terra, ermöglichte mir immer eine sehr flexible Einteilung meiner Arbeitszeit, die mir bei der Fertigstellung dieser Arbeit sehr zugute kam, wofür ich mich recht herzlich bedanken will.

Schlussendlich möchte ich mich bei meiner Familie, ganz besonders bei meinen Eltern Irmgard Gigerl und Franz-Josef Trattner für die umfassende Unterstützung während meiner gesamten Studienzzeit bedanken.

## Zusammenfassung

Eine hohe Produktivität am Rollenprüfstand erfordert routiniertes Bedienpersonal, welches Testzyklen mit hoher Qualität reproduzierbar durchführt. Ein ungültiger Test verursacht hohe Kosten und eine zeitnahe Wiederholung ist oft nicht möglich. Zur Minimierung von ungültigen Tests soll eine Trainingsmöglichkeit abseits des produktiven Rollenbetriebs vorhanden sein.

Daraus abgeleitet beschäftigt sich diese Masterarbeit mit der Entwicklung einer Simulation des Rollenprüfstandbetriebs für Lern- und Trainingsaufgaben. Diese umfasst sowohl die Methodik zur schnellen Abbildung verschiedener Fahrzeugtypen und -klassen wie auch die Einbindung der Fahrzeugmodelle in eine geeignete Simulationsumgebung. Der Einsatzbereich erstreckt sich dabei über die Auswahl und Einarbeitung neuer Prüfstandsfahrer bis hin zum Erlernen neuer Testzyklen und Fahrzeugtypen sowie auf das Training von kundenspezifischen Fahrweisen.

In der zu entwickelnden Simulation werden jene Prüfläufe des Rollenprüfstandbetriebs nachgebildet, welche die Ausführung toleranzbehafteter Ereignisse verlangen. Der Prüfstandsfahrer muss diese Ereignisse ohne Verletzung der tolerierten Abweichung ausführen. Das Folgen einer Sollgeschwindigkeit zählt dabei ebenso wie Motorstart und -stopp, Gangwechsel und Kupplungsbetätigung zu diesen Ereignissen.

Zur Vermeidung einer Fragmentierung auf Software-Ebene soll die Simulation durch Koppelung firmeninterner Soft- und Hardware entwickelt werden. Dazu wird eine Evaluierung der in der AVL angewandten Systeme durchgeführt und mittels einer Bewertungsmatrix klassifiziert. Nach Festlegung des Simulationssystems sind die Komponenten zur Nachbildung der Fahrgastzelle zu bestimmen. Diese umfassen einen Fahrersitz, Pedalerie und Gangschaltung sowie ein Gestell zur Aufnahme des gesamten Simulationssystems.

Nach dem Aufbau des Simulators wird ein gewähltes Fahrzeug mit der entwickelten Methodik modelliert und durch Vergleich mit realen Messdaten verifiziert. Danach wird das Zusammenspiel der gewählten Soft- und Hardware von Prüfstandsfahrern getestet und subjektiv bewertet. Die gewonnenen Erfahrungen werden analysiert und stellen die Basis für zukünftige Verbesserungen dar.

Bei diesem Konzept wird stets auf Erweiterbarkeit geachtet. So soll der Fahrer zukünftig gezielt auf gewünschte Aspekte wie Streckenmaximierung, minimalen Kraftstoffverbrauch und weitere Kriterien bewertet werden. Zusätzlich zu den beschriebenen Fähigkeiten wird es möglich sein, Fahrroboter zu testen, da das Fahrzeugmodell über eine CAN (Control area network)-Schnittstelle alle wichtigen Informationen über den Betriebszustand zugänglich macht.

## Abstract

A high productivity on chassis dynamometers requires a highly-skilled operating personal which is able to run test cycles with a reproducible high quality. An invalid test causes a high amount of additional costs and often cannot be repeated in time. To minimize invalid tests a possibility to train test runs in parallel to the productive operations on the chassis dynamometer has to be created.

This master thesis contains the development of a chassis dynamometer simulator to execute training tasks. Thereby the main tasks are the creation of a methodology to perform rapid modeling of various vehicle types and classes, as well as the integration of these vehicle models in a proper simulation environment. The application area not only includes the selection and training of newly-recruited staff, but also the training of new test cycles, new vehicle types and customer specific operation modes.

These test cycles of the chassis dynamometer operation, which require the execution of tolerance-based events have to be integrated. The chassis dynamometer engineers must execute these events without leaving the defined tolerance limit. The correct compliance of the demand speed as well as the execution of engine start and stop, gear shifting, and clutch actuation belongs to these events.

To avoid a fragmentation of software the simulation is done with an interaction of AVL internal soft- and hardware. For this step, AVL internal used systems are evaluated and classified with a rating matrix. When the simulation system is defined, all components to realize the vehicle cockpit have to be determined. These components comprise the acceleration-, brake- and clutch pedal, the gear lever as well as a frame, which is used to create a compact assembly of all components.

After finishing the assembly of the simulator, the developed methodology is used to build up a model of a selected vehicle. Afterwards this model is verified with measurement data of the real vehicle. When the verification is done, the interaction between the selected soft- and hardware will be subjectively tested by chassis dynamometer engineers. The analysis of the test's output offers experiences, which are used as a base for further improvements.

Expandability is an important part of this concept. In the future the driver can be trained to different aspects like maximizing driving distance, minimizing fuel consumption and further aspects. In addition it will be possible to test and validate driving robots. For these and further tests of external methodical elements the simulator offers a CAN (control area network) interface, which sends all important model data like vehicle speed, engine speed, engine torque etc. freely available and configurable.



# INHALTSVERZEICHNIS

<b>EIDESSTATTLICHE ERKLÄRUNG .....</b>	<b>I</b>
<b>VORWORT .....</b>	<b>II</b>
<b>ZUSAMMENFASSUNG .....</b>	<b>III</b>
<b>ABSTRACT .....</b>	<b>IV</b>
<b>INHALTSVERZEICHNIS .....</b>	<b>V</b>
<b>ABBILDUNGSVERZEICHNIS .....</b>	<b>VII</b>
<b>FORMELZEICHEN UND ABKÜRZUNGEN .....</b>	<b>IX</b>
<b>TABELLENVERZEICHNIS .....</b>	<b>XII</b>
<b>1 EINLEITUNG .....</b>	<b>1</b>
<b>2 THEORETISCHE GRUNDLAGEN.....</b>	<b>2</b>
2.1 Aufbau und Funktion eines Rollenprüfstandes .....	2
2.1.1 Abbildung der Straße am Rollenprüfstand .....	2
2.1.2 Koppelungsarten.....	3
2.2 Grundlagen zur Straßensimulation .....	4
2.3 Modellbildung .....	7
2.3.1 Theoretische Modellierung .....	7
2.3.2 Experimentelle Modellierung .....	9
2.3.3 Modellierung eines Fahrzeuges .....	10
2.4 Echtzeitsysteme .....	13
2.5 Bewertungsmethodik .....	14
<b>3 RECHERCHE .....</b>	<b>15</b>
3.1 Ausgangssituation.....	15
3.2 Hard- und Software der Echtzeitumgebung.....	16
3.2.1 Bernecker & Rainer Industrie Elektronik.....	17
3.2.2 AVL PUMA-PC mit AVL ARTE Echtzeitumgebung .....	19
3.2.3 Windows-PC mit AVL COBRA Echtzeitumgebung .....	21
3.2.4 COBRA-Echtzeitsystem mit Bernecker & Rainer als I/O-System .....	21
3.2.5 Auswahl der Echtzeitumgebung .....	22
3.3 Fahrzeugmodellbildung.....	24
3.3.1 MATLAB/Simulink.....	25
3.3.2 AVL CRUISE .....	25

---

3.3.3	MoBEO – Model Based Engine Optimization .....	26
3.3.4	Auswahl des Fahrzeugmodells .....	26
3.4	Fahrerleitgerät .....	28
3.5	Nachbildung der Fahrgastzelle .....	30
3.6	Pedale und Gangschaltung .....	30
3.7	Überblick des gewählten Konzeptes .....	32
<b>4</b>	<b>AUFBAU DES SIMULATORS.....</b>	<b>33</b>
4.1	Aufbau der Fahrgastzellennachbildung.....	33
4.2	Integration der Komponenten.....	34
4.3	Einbindung der Pedale und des Ganghebels .....	35
4.4	Simulatoraufbau .....	37
4.5	Einbindung des Fahrerleitgerätes in PUMA Open .....	37
4.6	Einbindung von Fahrzyklen .....	38
4.7	Weitere Schritte .....	38
<b>5</b>	<b>MODELLBILDUNG .....</b>	<b>39</b>
5.1	Fahrzeugmodellierung in CRUISE .....	39
5.1.1	Module in CRUISE .....	39
5.1.2	Abbildung des Fahrzeugkonzeptes in CRUISE .....	43
5.2	Echtzeitberechnung in MATLAB/Simulink.....	54
5.2.1	Signalkonditionierung.....	56
5.2.2	Kompilervorgang in MATLAB/Simulink .....	61
<b>6</b>	<b>EXPERIMENTELLE UNTERSUCHUNGEN.....</b>	<b>62</b>
6.1	Vorgehensweisen zum Modellabgleich mit vorhandenen Messdaten .....	62
6.2	Erstellung des Simulationsaufbaus .....	63
6.3	Vergleich von Fahrzeugmodell und Messdaten .....	68
<b>7</b>	<b>ERGEBNISSE.....</b>	<b>72</b>
7.1	Ergebnis des Modellabgleichs mit realem Fahrzeug .....	72
7.2	Eindrücke von Prüfstandstechnikern.....	77
<b>8</b>	<b>SCHLUSSFOLGERUNGEN UND AUSBLICK .....</b>	<b>78</b>
	<b>LITERATUR .....</b>	<b>79</b>

## Abbildungsverzeichnis

Abbildung 2-1: Fahrwiderstände eines Kraftfahrzeuges .....	4
Abbildung 2-2: Geschwindigkeitsverlauf beim Ausrollversuch .....	6
Abbildung 2-3: Fahrwiderstand in Abhängigkeit von der Fahrzeuggeschwindigkeit.....	7
Abbildung 2-4: Vereinfachung von partiellen Differentialgleichungen .....	9
Abbildung 2-5: Zusammenstellung der Komponenten von Fahrzeugmodellen .....	10
Abbildung 2-6: Ein- und Zweispurmodell.....	10
Abbildung 2-7: Aufbau eines Gesamtfahrzeugmodells.....	11
Abbildung 2-8: Modellstruktur von Bremse und Antriebsstrang .....	12
Abbildung 2-9: Aufbau einer Paarvergleichsmatrix.....	14
Abbildung 3-1: X20CP148x Zentraleinheit .....	17
Abbildung 3-2: PPxxx Power Panel.....	18
Abbildung 3-3: X20 I/O System.....	18
Abbildung 3-4: Speedmaster V2.0 .....	30
Abbildung 3-5: Logitech-G27-Spielsteuerung.....	31
Abbildung 3-6: Überblick des gewählten Konzeptes .....	32
Abbildung 4-1: Rahmen zur Aufnahme der Hardware-Komponenten.....	33
Abbildung 4-2: PUMA-PC .....	34
Abbildung 4-3: AVL-F-FEM-AIN-Analog-Digital-Wandler .....	34
Abbildung 4-4: Platzierung der Komponenten im Rahmen .....	35
Abbildung 4-5: Funktion der Pedalerie .....	36
Abbildung 4-6: Messung der Position des Ganghebels .....	36
Abbildung 4-7: Simulatoreaufbau .....	37
Abbildung 5-1: Energieflusslinie und Data-Bus in CRUISE .....	40
Abbildung 5-2: spezifischer Bremsfaktor verschiedener Bremsbauweisen.....	42
Abbildung 5-3: Ausrollkurve des Fahrzeuges.....	45
Abbildung 5-4: Skizze eines Antriebsstranges .....	46
Abbildung 5-5: Zuordnung der Messpunkte zur Ermittlung der Getriebeübersetzung.....	49
Abbildung 5-6: Parametrierung des Übersetzungsverhältnisses in CRUISE .....	50

---

Abbildung 5-7: Volllastkurve des parametrisierten Fahrzeuges .....	50
Abbildung 5-8: Signalfluss in der Fahrzeugmodellierung.....	51
Abbildung 5-9: Kompilierreihenfolge des Fahrzeugmodells.....	52
Abbildung 5-10: Kompilervorgang des Fahrzeugmodells in CRUISE.....	53
Abbildung 5-11: Festlegung des Zielsystems und des Online-Parameter-Zugriffs.....	53
Abbildung 5-12: Aufbau des Simulink-Modells .....	55
Abbildung 5-13: Spannungsverlauf der Pedalpotentiometer.....	56
Abbildung 5-14: Beispiel eines „Lookup Tables“ in Simulink .....	56
Abbildung 5-15: Berechnung der Pedalwerte in Simulink .....	57
Abbildung 5-16: Koordinaten der Gänge 1–6 .....	58
Abbildung 5-17: Berechnung des eingelegten Ganges .....	59
Abbildung 5-18: Simulink-Funktion für Motorstart und -stopp .....	60
Abbildung 5-19: Parameterdefinition in Simulink .....	61
Abbildung 6-1: Anpassung des Fahrzeugmodells zur Simulation in Simulink .....	64
Abbildung 6-2: Festlegung des Kupplungsverlaufs.....	65
Abbildung 6-3: Fehlerfortpflanzung bei falsch parametrierter Bremskraft .....	66
Abbildung 6-4: Dimensionierung der Bremse .....	67
Abbildung 6-5: Geschwindigkeitsverlauf ohne Pedalkennlinie .....	68
Abbildung 6-6: Für die Integration des Pedalkennfeldes verwendeter Abschnitt .....	69
Abbildung 6-7: Simulierte Geschwindigkeit nach Integration einer Fahrpedalkennlinie .....	70
Abbildung 6-8: Fahrpedalkennlinien .....	70
Abbildung 6-9: Simulierte Geschwindigkeit nach Anpassung der Fahrpedalkennlinie .....	71
Abbildung 7-1: Vergleich der Geschwindigkeitsverläufe bei verschiedenen Kennlinien.....	72
Abbildung 7-2: Analyse der Ansprechzeit.....	73
Abbildung 7-3: Fahrpedalwert und Fahrzeuggeschwindigkeit beim Training am Simulator ..	75
Abbildung 7-4: Vergleich der Messdaten am realen Fahrzeug mit einem Simulatortraining .	76

## Formelzeichen und Abkürzungen

### Lateinische Formelzeichen

$a_i$	$m/s^2$	Beschleunigung bei Messpunkt $i$
$a_x$	$m/s^2$	Beschleunigung in Fahrzeuginnenrichtung
$A_B$	$m^2$	Fläche des Bremskolbens
$c_B$	-	spezifischer Bremsfaktor
$c_w$	-	Luftwiderstandsbeiwert
$f_R$	-	Rollwiderstandsbeiwert
$g$	$m/s^2$	Erdbeschleunigung
$i$	-	Getriebeübersetzung allgemein
$i_i$	-	Getriebeübersetzung für Gang $i$
$i_H$	-	Hinterachsübersetzung
$I_{ges}$	$kgm^2$	auf Rad bezogenes gesamtes Trägheitsmoment
$F$	N	Kraft allgemein
$F_A$	N	Antriebskraft
$F_B$	N	Kraft zur Beschleunigung des Fahrzeuges
$F_L$	N	Luftwiderstand
$F_N$	N	Normalkraft
$F_{r0}$	N	Rollwiderstandskoeffizient 0. Ordnung
$F_{r1}$	N	Rollwiderstandskoeffizient 1. Ordnung
$F_{r4}$	N	Rollwiderstandskoeffizient 4. Ordnung
$F_R$	N	Rollwiderstand
$F_{RH}$	N	Rollwiderstand an den Hinterrädern
$F_{RV}$	N	Rollwiderstand an den Vorderrädern
$F_S$	N	Steigungswiderstand
$m$	kg	Fahrzeugmasse
$m_{rot}$	kg	translatorische Ersatzmasse
$M$	Nm	Moment allgemein
$M_B$	Nm	Bremsmoment
$M_{rot}$	Nm	Beschleunigungsmoment der rotatorischen Massen
$n_{GO}$	1/s	Getriebeausgangsdrehzahl
$n_{mot}$	1/s	Motordrehzahl
$n_r$	1/s	Raddrehzahl
$p_B$	Pa	Bremsdruck
$P_{Antrieb}$	W	Antriebsleistung

$P_{\text{Beschleunigung}}$	W	Beschleunigungsleistung
$P_{\text{Getriebe}}$	W	Verlustleistung des Getriebes
$P_{\text{Luftwiderstand}}$	W	Verlustleistung aufgrund des Luftwiderstands
$P_{\text{Nebenaggregate}}$	W	Verlustleistung der Nebenaggregate
$P_{\text{Rollwiderstand}}$	W	Verlustleistung aufgrund des Rollwiderstands
$P_{\text{Schlupf}}$	W	Verlustleistung aufgrund des Reifenschlupfs
$P_{\text{Steigung}}$	W	Leistung zur Überwindung der Straßensteigung
$r_B$	m	Mittlerer Reibradius der Bremse
$r_{\text{dyn}}$	m	dynamischer Reifenradius
$R_0$	N	Koeffizient der Ausrollkurve 0.Ordnung
$R_1$	N/(m/s)	Koeffizient der Ausrollkurve 1.Ordnung
$R_2$	N/(m/s) <sup>2</sup>	Koeffizient der Ausrollkurve 2.Ordnung
$t_i$	s	Zeitpunkt des Messpunktes i
$v$	m/s	Geschwindigkeit allgemein
$v_i$	m/s	Geschwindigkeit zum Messpunkt i
$v_{\text{FZ}}$	m/s	Fahrzeuggeschwindigkeit

### Griechische Formelzeichen

$\beta$	°	Straßensteigungswinkel
$\rho_L$	kg/m <sup>3</sup>	Luftdichte
$\eta_B$	-	Bremswirkungsgrad
$\mu_B$	-	Reibkoeffizient Bremsbelag – Bremsscheibe
$\omega$	1/s	Winkelgeschwindigkeit allgemein
$\dot{\omega}$	1/s	Winkelbeschleunigung des Rades
$\omega_r$	1/s	Winkelgeschwindigkeit des Rades

### Operatoren und Bezeichnungen

$\Sigma$	Summe
$\Delta$	Differenz zweier Größen

### Weitere Abkürzungen

A/D	Analog/Digital
AK	Bezeichnung eines Übertragungsprotokolls für Daten
ARTE	AVL Real Time Environment
ARTE.Lab	AVL Real Time Environment Laboratory

---

AVL	Anstalt für Verbrennungskraftmaschinen List
AVL AST	AVL Advanced Simulation Technology
B&R	Bernecker & Rainer
C	Bezeichnung einer Programmiersprache
CAN	Control Area Network
CMC	Common Model Compiler
COBRA	Weiterentwicklung von ARTE
CPU	Central Processing Unit
CRUISE	Tool zur Modellierung von Fahrzeugen
csv	Comma-separated Values, Dateieindung von Textdateien
dll	Dynamic Link Library, Dateieindung von Windows-Bibliotheken
DriCon	Driver Controller
exe	Executable, Dateieindung von Windows-Anwendungen
F-FEM	Fast Front End Module
F-FEM-AIN	F-FEM Analog Input
GUI	Graphical User Interface
HiL	Hardware In The Loop
INtime	Bezeichnung eines Echtzeitbetriebssystems
I/O	Input/Output
IEEE 1394	Bezeichnung der FireWire-Kommunikationsschnittstelle
MoBEO	Model Based Engine Optimization
PC	Personal Computer
PC104	Bezeichnung eines Stecksystems für Computerkomponenten
PT100	Platin Widerstandstemperatursensor mit 100 $\Omega$ Nennwiderstand bei $T = 0^\circ$
PT1000	Platin Widerstandstemperatursensor mit 1000 $\Omega$ Nennwiderstand bei $T = 0^\circ$
PUMA	Prüfstands- und Messtechnikautomatisierung
PUMA-PC	Computer Hardware zum Betrieb von PUMA
RS232	Standard für eine serielle Schnittstelle bei Computern
rta	Real Time Application, Dateieindung von INtime-Anwendungen
SAM	Bezeichnung einer Betriebsart von CRUISE
USB	Universal Serial Bus
VNC	Virtual Networking Computing

## Tabellenverzeichnis

Tabelle 3-1: Preisevaluierung des B&R Systems .....	19
Tabelle 3-2: Preisevaluierung des Windows-PCs mit COBRA.....	22
Tabelle 3-3: Paarvergleich der Kriterien zur Recherche der Echtzeitumgebung .....	23
Tabelle 3-4: Nutzwertanalyse der Echtzeitumgebungsvarianten .....	24
Tabelle 3-5: Paarvergleich der Kriterien zur Recherche der Fahrzeugmodellbildung .....	26
Tabelle 3-6: Nutzwertanalyse der Fahrzeugmodellbildungsvarianten.....	27



# 1 Einleitung

Weltweit existieren für die Zulassung von Kraftfahrzeugen unterschiedliche Normen und Grenzwerte für den Schadstoffausstoß. Die Zertifizierung von Personenkraftwagen findet auf Emissions-Rollenprüfständen statt. Die dort herrschenden konditionierten Bedingungen stellen für alle Fahrzeuge die gleichen Voraussetzungen dar und liefern mithilfe aufwendiger Emissionsmesssysteme in Verbindung mit gut geschultem Personal stets reproduzierbare Messungen. Das Training dieses Personals kann aus Zeit- und Kostengründen nicht immer am Rollenprüfstand erfolgen.

Als Alternative zum Training am Prüfstand soll ein Simulator entwickelt werden, welcher Tests mit verschiedenen Fahrzeugen simulieren soll. Dies erfordert eine Nachbildung einer Fahrgastzelle samt Pedalen und Gangschaltung, welche mithilfe einer Fahrzeugsimulation den Betrieb realitätsgetreu nachstellt. Die stets steigende Zahl verschiedener Testzyklen und die hohe Vielfalt an Fahrzeugkonzepten und -klassen erfordert eine Methodik rascher Modellbildung, die mit wenigen Validierungsmessungen vorstättengeht.

Das Ziel dieser Masterarbeit ist die Entwicklung einer Simulation des Rollenprüfstandbetriebs für Lern- und Trainingsaufgaben. Dabei soll in der AVL eine Recherche geeigneter Fahrzeugmodelle bzw. Modellierungstools samt Hardware für die Simulation und den Nachbau einer Fahrgastzelle durchgeführt werden. Anschließend werden die recherchierten Varianten mit einer geeigneten Bewertungsmethodik klassifiziert. Nach Festlegung des Konzeptes soll das Cockpit aufgebaut, sämtliche Hardware integriert und die Methodik einer raschen Modellbildung entwickelt und anhand eines Beispiels erläutert werden.

Kapitel 2 beschäftigt sich mit den theoretischen Grundlagen von Rollenprüfständen und der Simulation des Fahrwiderstands wie auch mit der Modellbildung und der Anwendung von Echtzeitsystemen.

In Kapitel 3 wird die Recherche und Auswahl der gesamten Hard- und Software sowie der Fahrzeugmodelle und der Modellierungstools erläutert.

Kapitel 4 zeigt den mechanischen Aufbau des Simulators, während in Kapitel 5 näher auf die Fahrzeugmodellbildung mit dem Tool „AVL CRUISE“ eingegangen wird.

In Kapitel 6 wird gezeigt, wie anhand einer experimentellen Untersuchung das Fahrzeugmodell mithilfe von Messdaten validiert wird.

In Kapitel 7 werden schlussendlich die Ergebnisse der Modellvalidierung mithilfe eines Vergleichs von realen Messdaten eines Abgastests am Rollenprüfstand und am Simulator dargestellt.

## 2 Theoretische Grundlagen

Für die Entwicklung einer Simulation des Rollenprüfstandbetriebs werden zunächst die nötigen Grundlagen erarbeitet. Dies umfasst die Kenntnis der Funktion und der Betriebsarten eines Rollenprüfstands, die zur Bestimmung der zu modellierenden Komponenten notwendig ist. Danach erfolgt die Ausarbeitung der Grundlagen zur Modellbildung, um die Evaluierung geeigneter Modelle durchzuführen. Für den Betrieb am Simulator wird ein echtzeitfähiges System aufgebaut, das eine zyklische Berechnung der gesamten Modellierung ermöglicht. Dazu werden die Begriffe „Echtzeit“, „Echtzeitsystem“ und „Echtzeitbetriebssystem“ erläutert.

### 2.1 Aufbau und Funktion eines Rollenprüfstandes [1]

Zur Steigerung der Produktivität werden Fahrversuche auf Rollenprüfstände verlegt und die real wirkenden Bedingungen simuliert. Die Übertragbarkeit der einzelnen Prüfumgebungen (Offline-Simulation, Hardware in the loop (HiL), Komponentenprüfstand, Motorprüfstand, Rollenprüfstand, Straße etc.) muss dabei stets gewährleistet sein. Durch Klima- und Höhenkammern besteht die Möglichkeit, unterschiedlichste Betriebszustände am Prüfstand durchzuführen.

#### 2.1.1 Abbildung der Straße am Rollenprüfstand

Durch Simulation von Zugkräften und Geschwindigkeiten stellt der Rollenprüfstand das Verhalten des Fahrzeuges auf der Straße nach. Dies lässt sich durch drei Grundregelungsarten erreichen.

##### **Zugkraftregelung**

Bei der Zugkraftregelung kann jedem Rad eine geschwindigkeitsunabhängige Zugkraft aufgeprägt werden.

##### **Geschwindigkeitsregelung**

Diese Regelart regelt kraftunabhängig an jedem Rad eine bestimmte Geschwindigkeit ein. Durch Variation der Raddrehzahlen werden beispielsweise Allradssysteme untersucht.

## **Straßensimulation**

Die Straßensimulation ist eine Erweiterung der Zugkraftregelung. Dabei wird eine bestimmte Zugkraft an den Rädern aufgebracht, die jedoch von der Fahrzeugmasse und -geschwindigkeit, der momentanen Beschleunigung sowie der simulierten Straßensteigung abhängt.

### **2.1.2 Koppelungsarten**

Rollenprüfstände sind sowohl als Ein- als auch als Mehrachsensysteme ausgeführt. Das einfachste System ist somit ein Einachsprüfstand mit einer Belastungsmaschine. Wie zuvor beschrieben, gibt es Prüfstände, die jedes Rad unabhängig beaufschlagen. Sollen die Räder dann dennoch auf die gleiche Kraft oder Geschwindigkeit geregelt sein, müssen die Rollen gekoppelt werden.

#### **Mechanische Koppelung**

Hier werden die Rollen starr miteinander verbunden (z.B. über ein Getriebe oder einen Zahnriemen).

#### **Elektrische Koppelung**

Hier wird der Drehwinkel der Rollen durch Drehwinkelmessung untereinander synchronisiert.

#### **Koppelungsart nachgeführte Geschwindigkeit**

Diese Koppelungsart regelt bei den nachgeführten Rollen die Drehzahl der führenden Rolle ein. Dabei werden bei der nachgeführten Rolle keine Zugkräfte übertragen, die führende Rolle muss diese Kräfte mitaufbringen. Ist der Prüfling z.B. ein Fahrzeug mit Vorderradantrieb, so ist es physikalisch gesehen nicht nötig, die Hinterräder auf einer Rolle zu positionieren. Durch den Stillstand der Hinterräder ist aber dessen Anteil am Fahrwiderstand von der Rolle an den Vorderrädern zu simulieren. Verfügt das Fahrzeug aber beispielsweise über ein nicht deaktivierbares Fahrstabilitätssystem, so würde dies bei stehenden Hinterrädern das Fahrzeug unerwünscht beeinflussen. Aus diesem Grund werden die Hinterräder daher mit der Drehzahl der Vorderräder nachgeführt. Dabei ist die Regelung so ausgelegt, dass auftretende Regelabweichungen klein genug bleiben, damit bei Nachführung der Rolle die Fahrstabilitätssysteme nicht eingreifen.

### Betrieb mit blockierter Achse

Bei einem Mehrachsprüfstand werden nicht benötigte Achsen blockiert, beispielsweise kann bei einem Allradprüfstand bei Prüfung eines Fahrzeugs mit Vorderradantrieb die Hinterachse stillstehen.

### Ungekoppelter Betrieb

Hier werden alle Rollen unabhängig voneinander angesteuert.

## 2.2 Grundlagen zur Straßensimulation

Bei der Straßensimulation erfolgt eine Nachbildung der im realen Betrieb des Fahrzeuges wirkenden Fahrwiderstände von der Rolle. Dazu stehen zwei Möglichkeiten zur Verfügung. Bei der konventionellen Methode wird auf die Fahrwiderstandsgleichung zurückgegriffen. Da diese aber einige Effekte wie z.B. automatische Spoiler vernachlässigt, wurde ein zweiter Ansatz gebildet. Bei dieser „erweiterten Straßensimulation“ kommt ein echtzeitfähiges Mehrkörpermodell des Fahrzeuges zum Einsatz. In Abhängigkeit der gemessenen Zugkraft am Rollenprüfstand wird aus diesem Modell die Fahrzeuggeschwindigkeit errechnet und am Prüfstand eingestellt. [1]

Die Methode der konventionellen Straßensimulation ist bei vielen gesetzlichen Zertifizierungen vorgeschrieben. Aus diesem Grund wird diese folgend weiter beschrieben.

### Komponenten des Fahrwiderstands [1]

Bei einer Fahrt auf einer steigenden Straße muss der Motor mit der auf die Fahrbahn wirkenden Antriebskraft  $F_A$  den Steigungswiderstand  $F_S$ , den Rollwiderstand  $F_R$  und den Luftwiderstand  $F_L$  überwinden sowie die Kraft zur Beschleunigung des Fahrzeuges  $F_B$  aufbringen. Dies ist in Abbildung 2-1 schematisch dargestellt.

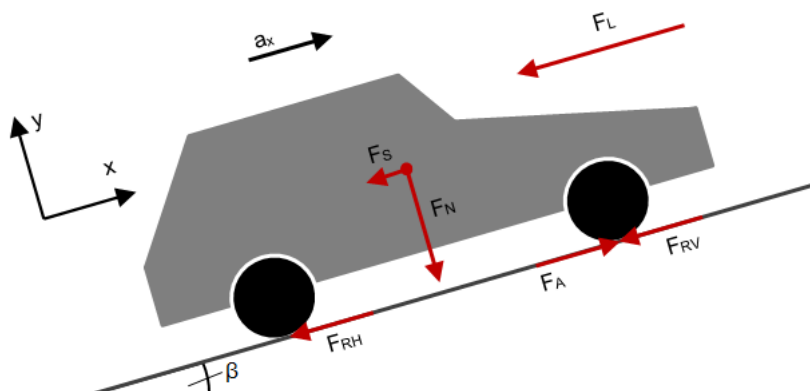


Abbildung 2-1: Fahrwiderstände eines Kraftfahrzeuges

Bildet man die Summe aller am Fahrzeug wirkenden Kräfte, so erhält man:

$$\sum F_x = m * a_x: \quad m * a_x = F_A - F_R - F_L - F_S \quad (2-1)$$

Die jeweiligen Anteile der Fahrwiderstände können wie folgt ermittelt werden:

- Rollwiderstand  $F_R$

$$\circ F_R = f_R * m * g = (F_{r0} + F_{r1} * v + F_{r4} * v^4) * m * g \quad (2-2)$$

- Luftwiderstand  $F_L$

$$\circ F_L = c_w * A * \frac{\rho_L}{2} * v^2 \quad (2-3)$$

- Steigungswiderstand  $F_S$

$$\circ F_S = m * g * \sin \beta \quad (2-4)$$

## Die Straßensimulation [2]

Die konventionelle Straßensimulation basiert auf der Fahrwiderstandsgleichung. Allgemein setzt sich die nötige Antriebsleistung des Motors aus folgenden Teilleistungen zusammen:

$$P_{Antrieb} = P_{Rollwiderstand} + P_{Luftwiderstand} + P_{Beschleunigung} + P_{Steigung} + P_{Schlupf} + P_{Getriebe} + P_{Nebenaggregate} \quad (2-5)$$

$P_{Schlupf}$ ,  $P_{Getriebe}$ ,  $P_{Nebenaggregate}$  treten sowohl am Rollenprüfstand als auch auf der Straße auf und werden daher nicht simuliert. Man muss jedoch beachten, dass sich der Schlupf aufgrund der unterschiedlichen Kontaktverhältnisse von Reifen und Straße bzw. Reifen und Rolle unterscheidet. Auf der Rolle sind nun  $P_{Rollwiderstand}$ ,  $P_{Luftwiderstand}$ ,  $P_{Beschleunigung}$  und  $P_{Steigung}$  so nachzubilden, dass diese für den gesamten Geschwindigkeitsbereich idente Werte für Rolle und Straße annehmen.

Für eine exakte Bestimmung der zu beschleunigenden Fahrzeugmasse ist auch eine Berücksichtigung der Trägheitsmomente des Antriebsstranges wie Räder, Wellen, Getriebe etc. nötig. Dazu sind die rotatorischen Anteile in eine äquivalente translatorisch zu beschleunigende Masse umzurechnen. Darauf Einfluss haben nur jene rotatorischen Anteile, die nicht durch das Fahrzeug angetrieben werden.

Laut Gleichung (2-2) setzt sich der Rollwiderstand aus einem konstanten Anteil und einer Funktion der Fahrzeuggeschwindigkeit in 1. und 4. Potenz zusammen. Die Abhängigkeit zur 4. Potenz tritt bei hoher Geschwindigkeit auf, diese Abhängigkeit muss aufgrund der starken Wärmeentwicklung außerhalb des Betriebsbereiches liegen [3]. Aus diesem Grund wird diese bei folgender Betrachtung vernachlässigt. Der Luftwiderstand ist eine Funktion der Geschwindigkeit in 2. Potenz. Demnach lassen sich Roll- und Luftwiderstand mit einem Polynom 2. Ordnung darstellen. Der Steigungswiderstand ist nur vom Steigungswinkel  $\beta$  abhängig und ist bei unveränderlicher Steigung konstant. Die Antriebskraft bleibt hier unbeachtet. Auf ebener Straße oder bei positiver Steigung ohne Antriebsleistung verzögert das Fahrzeug, die Beschleunigung nimmt also einen negativen Wert an. Durch die Multiplikation von Gleichung (2-1) mit „-1“ lässt sich der Fahrwiderstand besser mit einem Diagramm darstellen.

Nach erfolgter Multiplikation wird nun ein Polynom 2. Ordnung zur Abbildung von Roll- und Luftwiderstand verwendet.

$$-m \cdot a_x = F_R + F_L + F_S = R_0 + R_1 \cdot v + R_2 \cdot v^2 + F_S \quad (2-6)$$

Um am Rollenprüfstand Emissions-, Kennfeld- oder Fahrbarkeitsmessungen durchzuführen, ist es zunächst nötig, einen Ausrollversuch durchzuführen. Dabei wird das Fahrzeug unter vorgeschriebenen Bedingungen mehrmals auf eine vorgegebene Geschwindigkeit beschleunigt, danach im ausgekuppelten Zustand auf der Straße ausgerollt und der Geschwindigkeits-Zeit-Verlauf ermittelt. Anschließend wird die Kurve in Geschwindigkeitsdifferenzen  $\Delta v_i$ , bzw. in Zeitdifferenzen  $\Delta t_i$  eingeteilt.

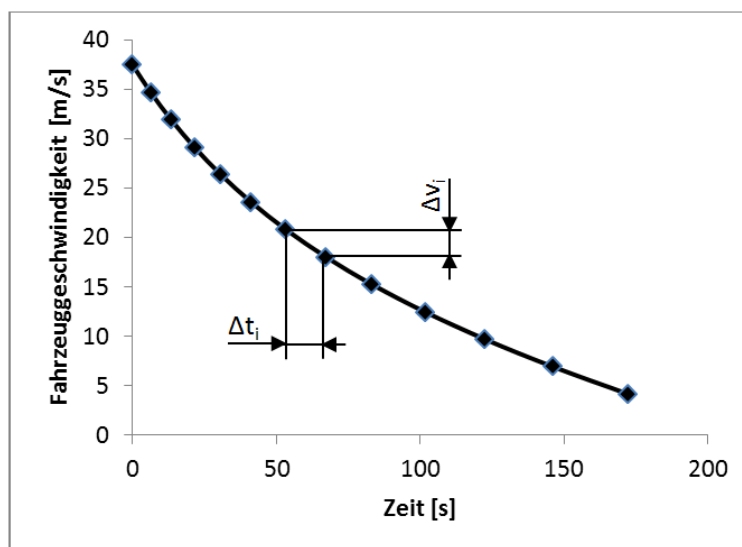


Abbildung 2-2: Geschwindigkeitsverlauf beim Ausrollversuch

Über die Beziehung

$$a_i = \frac{\Delta v_i}{\Delta t_i} \quad (2-7)$$

erfolgt nun die Zuordnung der Beschleunigungswerte des Fahrzeuges auf die Fahrzeuggeschwindigkeit. Über die Fahrzeugmasse wird so der Fahrwiderstand berechnet.

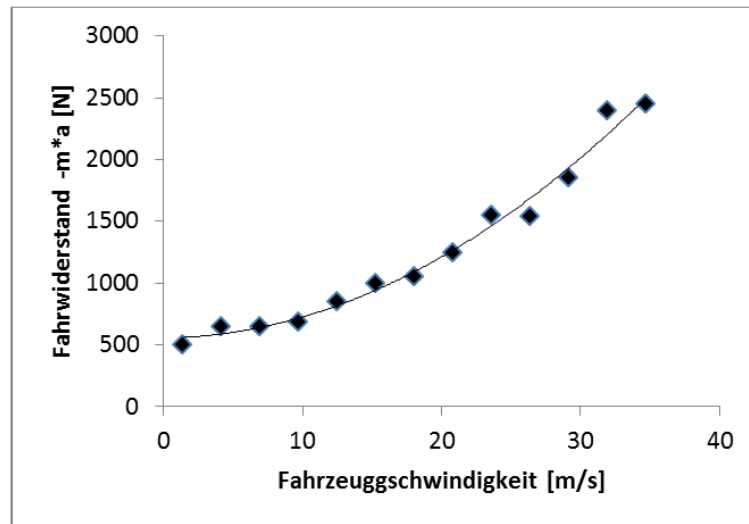


Abbildung 2-3: Fahrwiderstand in Abhängigkeit von der Fahrzeuggeschwindigkeit

Der Kontakt Rolle-Reifen weist im Rollentest bereits einen Rollwiderstand auf. Auch die Reibungsverluste des Rollenprüfstands verursachen einen Fahrwiderstand für das Fahrzeug. Diese Verluste werden ebenfalls mit einem Polynom 2. Ordnung abgebildet. Der Rollenprüfstand muss nun die Differenz der zuvor ermittelten Ausrollkurve und der Eigenverluste der Rolle aufbringen.

## 2.3 Modellbildung

Der Simulator erfordert Fahrzeugmodelle von unterschiedlichen Fahrzeugtypen und -klassen. Dieses Kapitel befasst sich mit den Grundlagen der Modellbildung von physikalischen Systemen, im Speziellen der Fahrzeugmodellbildung.

### 2.3.1 Theoretische Modellierung [4]

Mathematisch formulierte Naturgesetze bilden die Grundlage der theoretischen Modellbildung. Die betrachteten Prozesselemente werden dann zu Teil- und Gesamtprozessen zusammengefügt. Zunächst ist eine Vereinfachung des Prozesses nötig, um die Berechnung in erträglichem Aufwand zu halten bzw. um sie überhaupt zu ermöglichen.

Grundlagen bilden:

1. Bilanzgleichungen für Masse, Energie und Impulse
2. Physikalisch-chemische Zustandsgleichungen
3. Phänomenologische Gleichungen (z.B. Wärmeleitung, Diffusion)
4. Entropiebilanzgleichungen
5. Schaltungsgleichungen (Beschreibung der Verschaltung der Prozesselemente)

Die Abhängigkeit von Ort und Zeit führt zu partiellen Differentialgleichungen. Wird die Ortsabhängigkeit vernachlässigt, so betrachtet man die Systeme als Systeme mit „konzentrierten Parametern“, die durch gewöhnliche Differentialgleichungen beschrieben werden. Häufig sind Systeme partieller und gewöhnlicher Differentialgleichungen nicht analytisch lösbar und müssen durch Linearisierung, Reduktion der Modellordnung oder Approximation der Systeme vereinfacht werden. Auch wird die zuvor erwähnte Annahme von konzentrierten (ortsunabhängigen) Parametern angewendet.

Man nennt die durch diese Methode entstandenen Modelle auch „White-Box“-Modelle, da der physikalische Zusammenhang im Modell eingebettet ist und sich somit das Verhalten zwischen Ein- und Ausgang aus diesen mathematisch abgebildeten physikalischen Gesetzen zusammensetzt.



Abbildung 2-4 zeigt den Prozess der Vereinfachung partieller Differentialgleichungen. Durch Linearisierung werden nichtlineare Differentialgleichungen in lineare Differentialgleichungen überführt. Die Approximation der Differentialgleichungen auf Systeme mit konzentrierten Parametern reduziert partielle Differentialgleichungen auf gewöhnliche Differentialgleichungen.

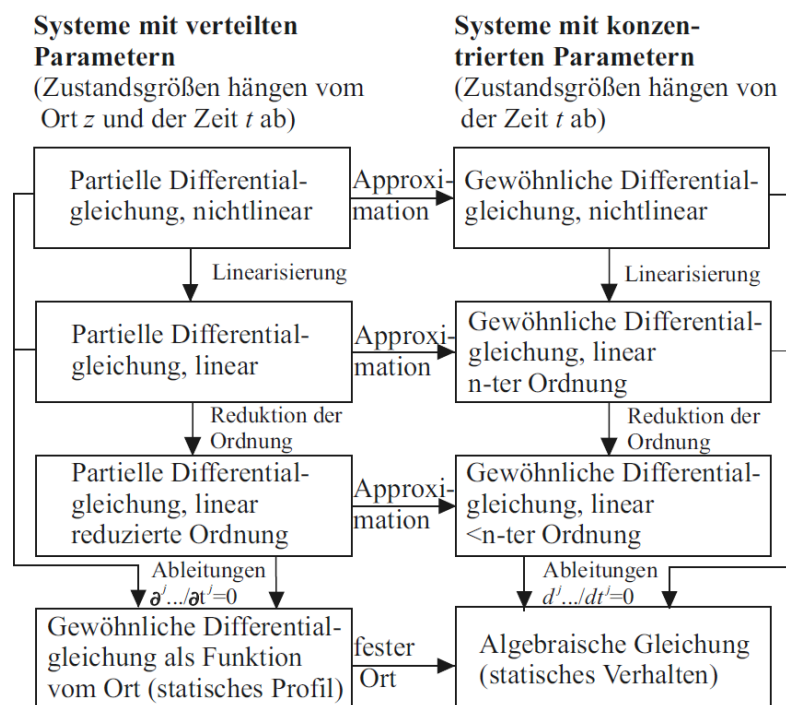


Abbildung 2-4: Vereinfachung von partiellen Differentialgleichungen [4]

### 2.3.2 Experimentelle Modellierung [4]

Bei der experimentellen Modellbildung geht man von Vorkenntnissen aus, die aus einer theoretischen Analyse oder aus Messungen stammen. Die Ergebnisse der Messung von Ein- und Ausgangsgrößen werden mittels einer Identifikationsmethode so ausgewertet, dass deren Zusammenhang in einem mathematischen Modell beschrieben werden kann. Die Eingangssignale können reale, aber auch künstlich eingeführte Testsignale sein. Lassen sich die experimentelle sowie die theoretische Modellbildung durchführen, so besteht die Möglichkeit, die Ergebnisse zu vergleichen und durch Analyse der Differenzen die Modelle zu korrigieren. Die Parameter eines experimentellen Modells sind nur Zahlenwerte, deren physikalischer Zusammenhang mit den Grunddaten des Prozesses nicht bekannt ist. Trotzdem liefern experimentelle Modelle oft bessere Ergebnisse als theoretische Modelle bei der Simulation des realen Prozesses und werden meist mit geringerem Aufwand ermittelt. Man nennt diese Modelle „Black-Box“-Modelle, da diese nur aus Kenntnis der gemessenen Ein- und Ausgangssignale entstehen und die physikalischen Zusammenhänge nicht in die Modellbildung eingehen.

### 2.3.3 Modellierung eines Fahrzeuges [5]

Jeder Körper hat drei translatorische und drei rotatorische Freiheitsgrade. Die translatorischen Freiheitsgrade beschreiben die Bewegung des Fahrzeuges in den drei Hauptachsen, die rotatorischen Freiheitsgrade umfassen die Drehung um die drei Hauptachsen. Bei der Modellierung eines Gesamtfahrzeuges ist es nun nötig, die erforderliche Modelltiefe zu definieren. Dazu muss bestimmt werden, welche Freiheitsgrade beim zu entwickelnden Modell von Interesse sind. Für ein Längsdynamikmodell kann beispielsweise die Modellierung des translatorischen Freiheitsgrades in Fahrzeuginnenrichtung ausreichend sein.

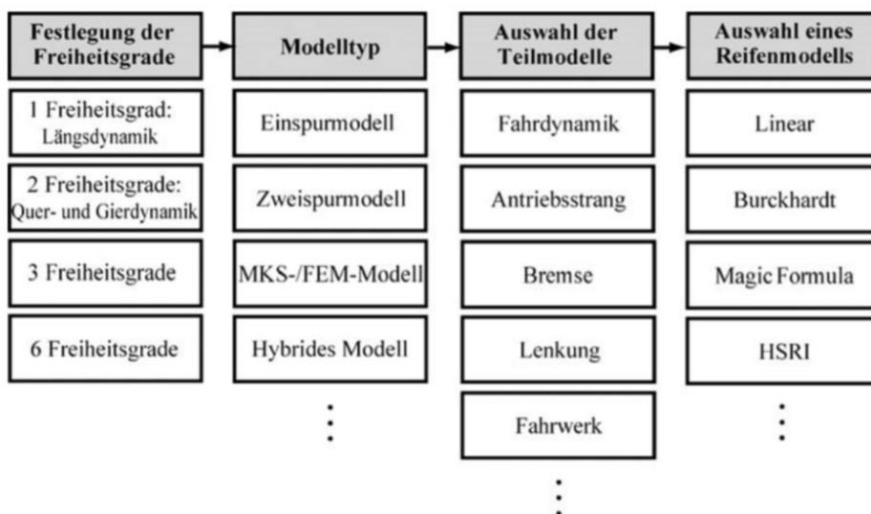


Abbildung 2-5: Zusammenstellung der Komponenten von Fahrzeugmodellen [5]

Bekannte Beispiele der Längs- und Querdynamik sind die Ein- und Zweispurmodelle. Weitere Modelltypen sind Mehrkörper- und Hybridmodelle, bei denen unterschiedliche Modelltypen miteinander gekoppelt werden, um die Nachteile der einzelnen Modelltypen auszugleichen.

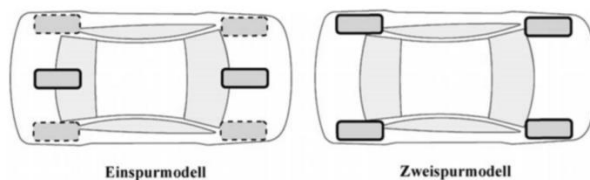


Abbildung 2-6: Ein- und Zweispurmodell [5]

Für alle Fahrzeugkomponenten werden folglich Modelle entwickelt, die gekoppelt das Gesamtfahrzeugmodell bilden. Dieses setzt sich aus den Teilmodellen Fahrdynamik, Karosserie, Fahrwerk, Antriebsstrang, Bremse und Lenkung zusammen.

Abbildung 2-7 zeigt die Einteilung der nötigen Modelle für Längs-, Vertikal- und Querdynamik, die direkt über Momente  $M$ , Kräfte  $F$ , Winkelgeschwindigkeiten  $\omega$  und Geschwindigkeiten  $v$  miteinander gekoppelt sind.

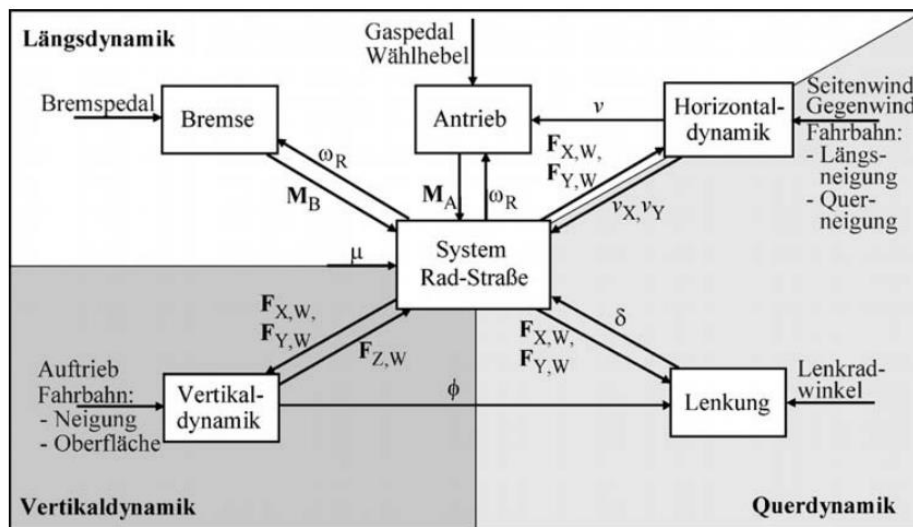


Abbildung 2-7: Aufbau eines Gesamtfahrzeugmodells [5]

Für die modellhafte Abbildung des Fahrzeugbetriebs auf der Rolle ist ein längsdynamisches Modell ausreichend.

### Modellierung des Antriebsstranges und der Bremsanlage

Die Modellierung des Antriebsstranges und der Bremsanlage wird aus der beispielhaften Modellierung eines Fahrzeuges mit Automatikgetriebe [5] abgeleitet. Für ein Fahrzeug mit Dieselmotor und manuellem Getriebe ergibt sich die Modellierung des Antriebsstranges aus den Komponenten:

- Einspritzsystem
- Verbrennungsmotor
- Kupplung
- Getriebe
- Differential
- System Rad-Straße
- Bremse

Eingangsgrößen sind dabei:

- Gaspedalstellung
- Bremspedalstellung
- Kupplungspedalstellung
- Eingelegter Gang

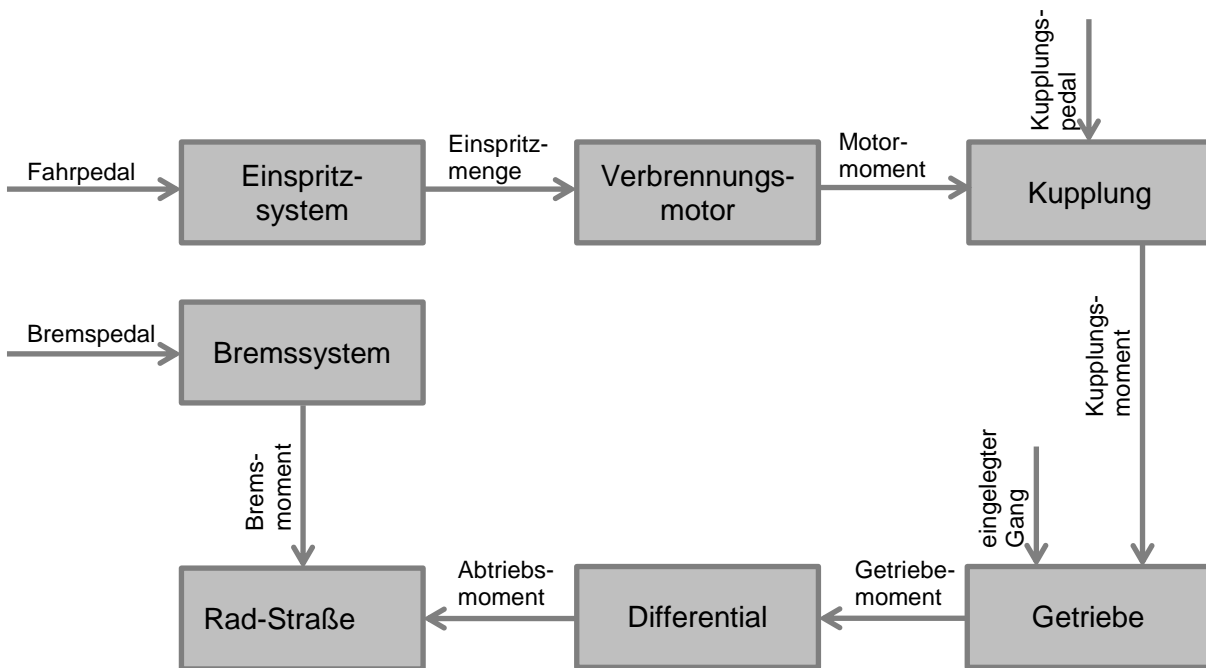


Abbildung 2-8: Modellstruktur von Bremse und Antriebsstrang

### Modellierung des Reifens

Sogenannte Reifenlatsche übertragen die Kräfte zwischen Reifen und Straße. Die komplexe anisotrope Bauweise eines Reifens stellt besondere Ansprüche an die Modellierung. Für Längsdynamik ist die Berechnung des Reifenschlupfs und des Rollwiderstandes ausschlaggebend. Hierfür stehen Modelle mit unterschiedlichen Modelltiefen zur Verfügung (siehe Kapitel 2.3.3).

Eingangsgrößen für das Gesamtfahrzeugmodell sind die vom Fahrer kommenden Signale für Fahr-, Kupplungs- und Bremspedalstellung und die Information des eingelegten Ganges. Als Eingangsgrößen der Umgebung zählen beispielsweise Straßensteigung, Windverhältnisse und Kenngrößen des Straßenbelags. Wenn notwendig erfolgt auch die Einbindung eines Antiblockiersystems und von Fahrerassistenzsystemen.

## 2.4 Echtzeitsysteme [6]

Da für Echtzeitsysteme und Echtzeitbetriebssysteme mehrere Definitionen bzw. Betrachtungsweisen existieren, werden die Definitionen aus [6] folgend angeführt:

*Def.: Echtzeitbetriebssystem (DIN 44300): Unter Echtzeit versteht man den Betrieb eines Rechensystems, bei dem Programme zur Verarbeitung anfallender Daten ständig betriebsbereit sind, derart, dass die Verarbeitungsergebnisse innerhalb einer vorgegebenen Zeitspanne verfügbar sind. Die Daten können je nach Anwendungsfall nach einer zeitlich zufälligen Verteilung oder zu vorherbestimmten Zeitpunkten anfallen.*

*Def.: Echtzeitsystem (Oxford Dictionary of Computing): Ein Echtzeitsystem ist ein System, bei dem der Zeitpunkt, zu dem Ausgaben erzeugt werden, bedeutend ist. Das liegt für gewöhnlich daran, dass die Eingabe mit einigen Änderungen der physikalischen Welt korrespondiert und die Ausgabe sich auf diese Änderungen beziehen muss. Die Verzögerung zwischen der Zeit der Eingabe und der Zeit der Ausgabe muss ausreichend klein für eine akzeptable „Rechtzeitigkeit“ sein.*

Echtzeitsysteme müssen also korrekte Reaktionen in einer definierten Zeitspanne liefern. Bei Überschreitung dieser Zeitspanne gibt es Fehlfunktionen. Daraus folgt, dass die Richtigkeit der Echtzeitberechnung sowohl vom Resultat der Berechnung der einzelnen Bearbeitungsschritte als auch von der Einhaltung der zulässigen Bearbeitungszeit abhängig ist.

### Harte Echtzeitsysteme

Bei einem harten Echtzeitsystem muss die Berechnung in der zulässigen Zeitspanne erfolgen, ein Überschreiten ist nicht zulässig. Harte Echtzeitsysteme kommen dort zum Einsatz, wo auch nur gelegentliche Überschreitungen der zulässigen Zeitspanne erhebliche Folgen haben. Dies ist z.B. für Anwendungen in Kernkraftwerken der Fall.

### Weiche Echtzeitsysteme

Bei weichen Echtzeitsystemen ist eine Überschreitung der Zeitspanne erlaubt, die Einhaltung selbiger muss jedoch zum Großteil gewährleistet sein. Ein Beispiel einer weichen Echtzeitbedingung ist die Einhaltung einer Bildwiederholungsrate bei einem TV-Gerät.

## Zeitschranken

Die Abarbeitung einer Echtzeitaufgabe kann grundsätzlich durch eine obere oder eine untere Zeitschranke begrenzt sein. Auch eine Begrenzung von beiden Seiten ist möglich. Die untere Zeitschranke bezeichnet somit die minimale Dauer einer Abarbeitung.

Da die obere Zeitschranke gewissermaßen die Mindestgeschwindigkeit des Systems darstellt, die untere Zeitschranke jedoch immer durch „Warten“ erfüllt wird, wirkt die obere Zeitschranke für das System bestimmend.

## 2.5 Bewertungsmethodik [7]

Um die zu recherchierenden Möglichkeiten (siehe Kapitel 3) zu bewerten, wird eine Nutzwertanalyse durchgeführt. Die Ermittlung der Gewichtung erfolgt durch einen Paarvergleich der Bewertungskriterien. Durch die Verwendung eines Paarvergleichs wird die Gewichtung objektiver erstellt, da diese Methode immer nur zwei Kriterien miteinander vergleicht. Dies lässt nur eine Entscheidung zu, welches der beiden Kriterien wichtiger ist. Eine Skala zur Verteilung der Wichtigkeit ist nicht vorgesehen.

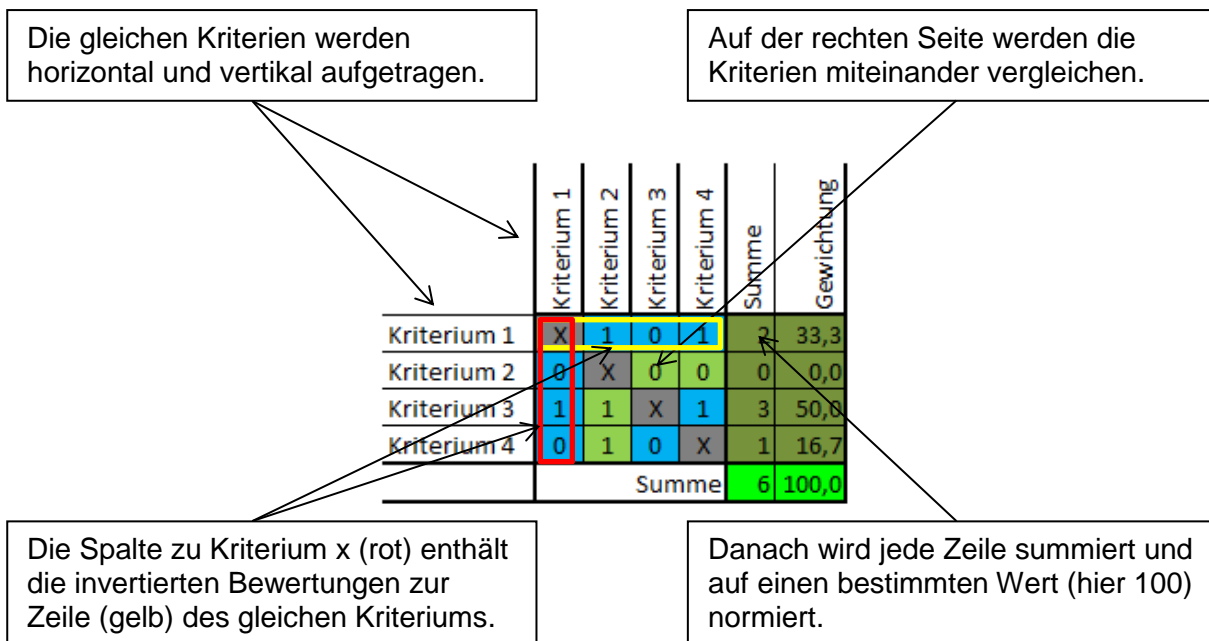


Abbildung 2-9: Aufbau einer Paarvergleichsmatrix

Mit den ermittelten Gewichtungen wird eine Nutzwertanalyse durchgeführt und dadurch die günstigste Variante ausgewählt.

## **3 Recherche**

### **3.1 Ausgangssituation**

Um die Aufgaben eines Prüfstandstechnikers korrekt nachzubilden, ist eine korrekte Simulation der Prüfstands Umgebung nötig. Aus diesem Grund entsteht die Anforderung, eine Fahrgastzelle nachzubilden, welche eine Pedalerie und eine Gangschaltung aufweist. Die Sitzposition soll dabei verstellbar sein. Weiters muss ein Fahrerleitgerät vorhanden sein, welches die Sollwerte von Fahrzeugeschwindigkeit und einzulegendem Gang anzeigt. Für die Echtzeitberechnung eines Fahrzeugmodells kommt ein Echtzeitrechner zum Einsatz, der in Verbindung mit einer geeigneten Schnittstelle die Signale der Pedalerie und der Gangschaltung auswertet und dem Fahrzeugmodell zugänglich macht. Dieses berechnet in Abhängigkeit dieser Eingangsgrößen die Motordrehzahl, Motorlast und die Fahrzeugeschwindigkeit. Dabei soll auf qualitativ richtiges Motorverhalten geachtet werden, das anhand vorhandener Messdaten validiert wird und auf neue Fahrzeuge angepasst werden kann. Folgende Komponenten werden evaluiert:

#### **Hard- und Software der Echtzeitumgebung**

Das Echtzeitsystem muss in der Lage sein, das Fahrzeugmodell in festgelegter Frequenz zu berechnen. Dabei ist darauf zu achten, dass es mit den Tools für die Modellbildung kompatibel ist und geeignete Eingänge hat, um die Signale der Pedalerie und des Ganghebels zu verarbeiten.

#### **Fahrzeugmodellbildung**

Die Längsdynamik des modellierten Fahrzeuges muss qualitativ das gleiche Beschleunigungsverhalten aufweisen. Das bedeutet, dass Drehmomentenaufbau und Ansprechzeit mit dem realen Fahrzeug übereinstimmen. Für diese Kalibrierung müssen alle notwendigen Parameter im Modell zugänglich sein. Es ist darauf zu achten, dass das Modell am gewählten Echtzeitsystem lauffähig ist.

#### **Fahrerleitgerät**

Das Fahrerleitgerät soll ein ähnliches User Interface besitzen, wie es auch am Rollenprüfstand zur Anwendung kommt. Es muss eine Kommunikationsschnittstelle zum gewählten Echtzeitsystem aufweisen, um die nötigen Größen (Sollwerte aus dem Testzyklus und Istwerte aus dem Fahrzeugmodell) zu verarbeiten und korrekt anzuzeigen.

## **Fahrgastzelle**

Die Nachbildung der Fahrgastzelle stellt das Trägergerüst des Simulators dar und hat die Aufgabe, sämtliche Hardware aufzunehmen. Dazu zählen die Pedalerie, die Gangschaltung, Fahrerleitgerät mit Bildschirm sowie die gesamte Hardware des Echtzeitsystems. Der Fahrersitz wird verstellbar ausgeführt. Für das Recruiting neuer Mitarbeiter und zur Vorführung auf Messen muss der Simulator eine ausreichende Mobilität gewährleisten, um ihn ohne großen Aufwand in einem Lieferwagen zu verladen.

## **Pedalerie und Gangschaltung**

Der Simulator muss ein Fahr-, ein Brems- und ein Kupplungspedal aufweisen, welche über geeignete Signale in das Echtzeitsystem eingelesen werden. Auch eine im Echtzeitsystem auswertbare Gangschaltung für ein manuelles Getriebe ist vorgesehen.

## **3.2 Hard- und Software der Echtzeitumgebung**

Gemäß den beschriebenen Anforderungen wird eine Recherche durchgeführt, um eine Liste der firmenintern zum Einsatz kommenden Echtzeitsysteme zu erstellen. Der Begriff „Echtzeitsystem“ bezeichnet hier die Kombination aus Hardware, Ein- und Ausgängen und Echtzeitbetriebssystem. Der in folgender Recherche verwendete Begriff „I/O“ beschreibt alle Hardware-Komponenten, die eine Signalein- und -ausgabe an das System ermöglichen. Dazu zählen Analog-Digital-Wandlerschnittstellen für die Messung und Ausgabe von Spannungen, Auswertungsschnittstellen für Digitalsignale, BUS-Schnittstellen etc. In die Recherche gehen nur Echtzeitsysteme ein, die ein auf die Hardware abgestimmtes Echtzeitbetriebssystem aufweisen.

Folgende Systeme stehen zur Auswahl:

- Bernecker & Rainer Industrie Elektronik
- AVL PUMA-PC mit AVL ARTE Echtzeitumgebung
- Windows PC mit AVL COBRA Echtzeitumgebung
- COBRA Echtzeitsystem mit Bernecker & Rainer als I/O System



### 3.2.1 Bernecker & Rainer Industrie Elektronik

Bernecker & Rainer, auch als B&R bekannt, ist eine österreichische Automatisierungsfirma, die sehr modulare Steuerungssysteme anbietet. In der AVL werden diese Systeme häufig für Prototypen und Konzeptstudien verwendet. Die Recherche ergibt zwei unterschiedliche Grundtypen, die sich durch die Visualisierung unterscheiden. Bei beiden Typen kommt als Echtzeitbetriebssystem „Automation Runtime“ zum Einsatz. Dies ist ein von B&R entwickeltes Betriebssystem. Über die Software „B&R Automation Studio“ wird nun über mehrere Programmiersprachen und einem Editor für ein User Interface das gewünschte Programm erstellt. Dies wird auf eine Compact-Flash-Speicherkarte übertragen, die als Hauptspeicher dient. Alle Schnittstellen sind über Programmfunktionen frei programmierbar. Dies ermöglicht einen hohen Grad an Kompatibilität zu anderen Systemen, der jedoch teilweise mit hohem Programmieraufwand verbunden ist.

#### X20CP148x Zentraleinheiten [8]

X20 CPUs basieren auf Intel Celeron Prozessoren und decken mit ihrer Modularität ein weites Anwendungsspektrum ab. Das „x“ am Ende der Bezeichnung ist dabei eine Kennung für die Rechenleistung des Prozessors. Das System verfügt über USB und Ethernet und bietet einen variablen Steckplatz, der beispielsweise für eine CAN-Schnittstelle verwendet wird. Diese CPUs verfügen nicht über einen Bildschirmausgang. Die Visualisierung des User Interfaces erfolgt über Ethernet via einer VNC (Virtual Network Computing)-Verbindung.



Abbildung 3-1: X20CP148x Zentraleinheit [8]

### B&R Power Panel PPxxx [9]

Analog zu den X20CP148x Zentraleinheiten besitzen auch diese Panels USB- und Ethernet-Schnittstellen und einen variablen Steckplatz. Zum Unterschied zu X20CP148x Einheiten geht die Bedienung mit dem integrierten Touchscreen Display oder ebenfalls über eine VNC-Verbindung vonstatten.



Abbildung 3-2: PPxxx Power Panel [9]

### Das X20 I/O System [10]

Beide der oben beschriebenen Systeme verfügen über das X20 I/O System. Hierbei handelt es sich um ein scheibenbasierendes System, welches eine flexible und universelle Lösung für die Anpassung an verschiedenste Anwendungsfälle bietet. Dieses System umfasst z.B. analoge und digitale Ein- und Ausgänge sowie Temperaturmessmodule für PT100- und PT1000-Sensoren. Die Komponenten werden in beliebiger Reihenfolge an die Zentraleinheit angesteckt.



Abbildung 3-3: X20 I/O System[10]

## Preisevaluierung

Zur Preisevaluierung wird eine Zentralrecheneinheit mit guter Rechenleistung gewählt. Für Pedalerie und Gangschaltung sind fünf analoge Eingänge erforderlich (Potentiometer für Fahr-, Brems- und Kupplungspedal, Koordinaten des Ganghebels). Die Berechnung erfolgt mit Daten aus [11]. In dieser überschlägigen Berechnung sind Preise für nötiges Zubehör, wie Netzteil, Verkabelung und Hutschiene vernachlässigt.

	Bezeichnung	Anzahl	Einzelpreis [€]	Gesamtpreis [€]
Zentraleinheit	X20CP1485-1	1	1330	1330
Analoger Eingang (4 fach)	X20AI4632	2	215	430
CAN Schnittstelle (2 fach)	X20IF2772	1	196	196
Summe				1956

Tabelle 3-1: Preisevaluierung des B&R Systems

## Simulationsumgebung

Wie zuvor erwähnt, werden die B&R Zentraleinheiten und Power Panels über die Software „Automation Studio“ programmiert. Die Programmierung ist sowohl grafisch als auch textbasierend und unterstützt mehrere proprietäre Programmiersprachen und die universelle Programmiersprache „C“. Das Automation Studio bietet auch eine Schnittstelle zu MATLAB/Simulink (siehe Kapitel 3.3.1). Über ein Installationspaket lassen sich die Ein- und Ausgänge von Simulink-Modellen so kompilieren, dass der generierte Code im Automation Studio verwendbar ist. Die Eingänge der analogen Karte werden somit direkt im Simulink-Modell ausgewertet.

### 3.2.2 AVL PUMA-PC mit AVL ARTE Echtzeitumgebung

#### PUMA Open [12]

PUMA Open ist ein Automatisierungssystem zur Entwicklung von Motoren, Getrieben und Antriebssträngen. Auch Fahrzeugprüfstände nutzen eine angepasste Version dieses Automatisierungssystems. Es bietet eine offene Plattform, welche standardisierte Schnittstellen für Datenerfassung und Kommunikation bietet. Die darauf ausgelegte Hardware ist, ebenso wie die Software, modular aufgebaut. Diese modulare Struktur basiert auf mehreren Layern. Der erste Layer ist das parallel zu Windows laufende Echtzeitbetriebssystem INtime. Der nächste Layer ist ARTE (AVL Real-Time Environment), darüber befindet sich der Layer mit den Teilsystemen von PUMA. ARTE kann als Hülle von INtime betrachtet werden, welches die Entwicklung von Echtzeitkomponenten erleichtert.

## **PUMA-PC**

Grundsätzlich ist PUMA Open eine Windows-Anwendung, welche auf jedem Rechner mit einem Microsoft-Windows-Betriebssystem installiert werden kann. Das I/O System und die dafür nötigen Treiber von PUMA Open sind jedoch auf bestimmte Hardware ausgelegt, die sich PUMA PC nennt. Ohne diese Hardware ist der Betrieb von Schnittstellen, wie z.B. CAN und analogen I/Os nicht möglich. Aus diesem Grund bietet die AVL ein Hardwaresystem an, das für den Betrieb von PUMA Open zertifiziert ist. Zur Versionierung nennt sich diese Hardware „PUMA-PC“ mit der Jahreszahl der Veröffentlichung, z.B. „PUMA-PC 2012“.

### **Das I/O System**

Der PUMA-PC bietet eine frei belegbare CAN-Karte. Für analoge sowie digitale Ein- und Ausgänge gibt es 19“ Module, die sich „AVL F-FEM“ nennen. Für das Einlesen einer Spannung ist hier das Analog-Input Modul „AVL F-FEM-AIN“ notwendig. Die Verbindung mit dem PUMA-PC erfolgt über die IEEE-1394-FireWire-Schnittstelle.

### **Simulationsumgebung [13]**

Dem ARTE-Echtzeitsystem können über das AVL ARTE.Lab Studio auch Simulink-Modelle hinzugefügt werden. ARTE.Lab bietet die Möglichkeit, Simulink-Modelle zu interpretieren, um sie für ein Echtzeitzeitsystem zu kompilieren, damit diese auf PUMA Open lauffähig sind.

Alle Komponenten von PUMA Open sind mit Variablen („Systemnamen und Normnamen“) verbunden. Diese lassen sich auf die analogen Eingänge und die Eingänge des kompilierten Fahrzeugmodells zuweisen. Somit kann eine Verbindung der analogen Spannung der Potentiometer für Pedale und Gangschaltung zum Fahrzeugmodell erstellt werden.

### **Preisevaluierung**

Puma-PCs werden bei allen AVL-Prüfständen verwendet, daher gibt es in der AVL zahlreiche durch neue Versionen ersetzte Rechner, die frei zur Verfügung stehen. Aus diesem Grund ist es möglich, diese PCs kostenlos zu nutzen.

### **3.2.3 Windows-PC mit AVL COBRA Echtzeitumgebung**

#### **Hardware und I/O-System**

AVL COBRA ist auf jedem Windows-PC lauffähig. COBRA hat eine umfassende Stand-Alone-Unterstützung, das bedeutet, es muss kein Automatisierungssystem vorhanden sein, welches die Treiber der nötigen I/O-Karten bereitstellt. Die Treiber werden von COBRA selbst bereitgestellt. AVL COBRA ist ein noch sehr junges System, welches sich noch bis vor Kurzem in der Testphase befand. Aus diesem Grund gibt es zurzeit nur Treiber für Hardware aus der Testphase. Diese umfassen I/O-Karten auf PC104-Basis für analoge sowie digitale Ein- und Ausgänge und eine CAN-Karte.

#### **Simulationsumgebung**

COBRA basiert auf einem Komponentenkonzept, d.h. alle in Echtzeit berechneten Komponenten können unabhängig voneinander zur Laufzeit hinzugefügt, gelöscht oder verändert werden. Die Verbindung der Komponenten erfolgt mithilfe sogenannter „Channels“. Durch die Analyse dieser Channels erkennt COBRA die Abhängigkeiten der Komponenten und berücksichtigt dies bei der Abarbeitungsreihenfolge. Da COBRA auf ARTE basiert, werden auch hier Simulink-Modelle mittels ARTE.Lab eingebunden.

#### **Preisevaluierung**

Eine Preisrecherche beim Einkauf der AVL ergibt einen Kaufpreis von 1500 Euro für ein System, bestehend aus Recheneinheit, CAN-Karte und Analog- und Digitalkarte.

### **3.2.4 COBRA-Echtzeitsystem mit Bernecker & Rainer als I/O-System**

In dem unter Kapitel 0 beschriebenen System ist Hardware für einen industriellen Temperaturbereich verbaut. Daraus resultiert ein hoher Preis, der sich durch Verwendung von Standardkomponenten senken lässt. Da jedoch keine Treiberunterstützung für andere Hardware vorhanden ist, ist eine Alternativlösung für das I/O-System zu finden. Die Recherche ergibt, dass ein Treiber für eine CAN-Karte existiert, welche in einem normalen Windows-Rechner eingebunden wird. Über diese Schnittstelle werden I/Os einer B&R-Hardware mit einer kostengünstigen Zentraleinheit verwendet. Zur Auswertung der I/Os ist eine kostengünstigere Zentraleinheit mit weniger Rechenleistung ausreichend. Ein handelsüblicher Desktop PC mit eingebauter CAN-Schnittstelle kann nun als Echtzeitsystem verwendet werden. Dieses System ist einfach erweiterbar und bietet ein frei konfigurierbares User Interface. Dabei besteht durch die Verwendung von „AVL COBRA“ Kompatibilität zu zahlreicher AVL-Software.

## Simulationsumgebung

Die Simulationsumgebung entspricht jener von Kapitel 3.2.3.

## Preisevaluierung

Die Berechnung für B&R-Komponenten erfolgt mit Daten aus [11].

	Bezeichnung	Anzahl	Einzelpreis [€]	Gesamtpreis [€]
Zentraleinheit	X20CP1483	1	1330	1330
Analoger Eingang (4 fach)	X20AI4632	2	215	430
CAN-Schnittstelle (2 fach)	X20IF2772	1	196	196
Desktop PC		1	500	500*
CAN-Karte Desktop PC		1	100	100*
Summe				2556

Tabelle 3-2: Preisevaluierung des Windows-PCs mit COBRA

\*geschätzter Wert

### 3.2.5 Auswahl der Echtzeitumgebung

Zur Klassifizierung der möglichen Echtzeitsysteme fließen auch die Ergebnisse der Fahrzeugmodellbildungsvarianten (siehe Kapitel 3.3.4) ein. Daraus abgeleitet erfolgt die Gewichtung der Kriterien mittels eines Paarvergleichs (siehe Kapitel 2.5) zur Durchführung einer Nutzwertanalyse. Aus der Modellrecherche (Kapitel 3.3) ist ersichtlich, dass die in der AVL verwendeten Fahrzeugmodelle auf Simulink oder CRUISE beruhen. CRUISE-Modelle können zwar genauso in Simulink exportiert werden, sie benötigen aber eine Lizenz, um auch ausführbar zu sein. Das gilt genauso für die Erstellung der Echtzeitanwendung aus MATLAB/Simulink. Es ist möglich, das CRUISE-Modell auf ARTE-Basis zu kompilieren, die Lizenzierung von COBRA und Bernecker & Rainer ist hingegen nicht erlaubt.

## Paarvergleich

Die Gewichtung für die Nutzwertanalyse ergibt sich aus dem Einsatz eines Paarvergleichs. Bei der Wahl der Bewertungskriterien werden die Kompatibilität zu den Modellierungsplattformen, der Preis, verfügbare Schnittstellen, die Visualisierungsmöglichkeiten und die Rechenleistung beachtet.

Kriterien	Matlab Fähigkeit	Cruise Fähigkeit	Preis	CAN Schnittstelle	Ethernet Schnittstelle	GUI Editor	Rechenleistung	Größe und Gewicht	Simulation des Gesamtprüfstandes	Fahrerleitgerät integrierbar?	Summe	Gewichtung	
Matlab Fähigkeit	X	1	1	0	1	1	0	1	1	1	7	16	
Cruise Fähigkeit	0	X	1	0	1	1	0	1	1	1	6	13	
Preis	0	0	X	0	0	0	0	0	0	1	1	2,2	
CAN Schnittstelle	1	1	1	X	1	1	0	1	1	1	8	18	
Ethernet Schnittstelle	0	0	1	0	X	0	0	1	0	0	2	4,4	
GUI Editor	0	0	1	0	1	X	0	1	0	0	3	6,7	
Rechenleistung	1	1	1	1	1	1	X	1	1	1	9	20	
Größe und Gewicht	0	0	1	0	0	0	0	X	0	0	1	2,2	
Simulation des Gesamtprüfstandes	0	0	1	0	1	1	0	1	X	1	5	11	
Fahrerleitgerät integrierbar?	0	0	0	0	1	1	0	1	0	X	3	6,7	
											Summe	45	100

Tabelle 3-3: Paarvergleich der Kriterien zur Recherche der Echtzeitumgebung

## Nutzwertanalyse

Nach Ermittlung der Gewichtung wird anhand der festgelegten Kriterien die Nutzwertanalyse mit einer Notenskala für „0“ bis „10“ durchgeführt, wobei die Note „10“ den günstigsten Fall darstellt. Jene Kriterien, die Entscheidungsfragen sind, erhalten die Note „10“ für „ja“ und „0“ für „nein“. Die Bewertung des Preises erfolgt, indem der Höchstpreis aller Komponenten auf die Note „0“ normiert wird, um mittels Interpolation die Noten der günstigeren Varianten zu bestimmen. Ein Preis von 0 € entspricht folglich der Note „10“. Da die exakte Benotung der Rechenleistung nur durch Vergleichsmessungen möglich ist, erfolgt diese daher subjektiv unter Miteinbeziehung der Taktrate und Bauweise der CPU sowie des eingesetzten Betriebssystems. Das Kriterium „GUI Editor“ beschreibt die Möglichkeit, ein User Interface zu erstellen. Die Note „0“ charakterisiert „Erstellung einer GUI nicht möglich“, „5“ beschreibt die Erstellungsmöglichkeit über ein separates Programm, „10“ bedeutet die Existenz eines GUI Editors. Eine echtzeitfähige CAN- bzw. Ethernet-Schnittstelle ergibt die Note „10“. Eine nicht echtzeitfähige Schnittstelle wird mit der Note „5“ bewertet.

	Gewicht G		Bernecker&Rainer		AVL PUMA-PC mit AVL ARTE Echtzeitumgebung		Windows-PC mit AVL COBRA Echtzeitumgebung		COBRA Echtzeitsystem mit Bernecker & Rainer als I/O System	
	Note	Note x G	Note	Note x G	Note	Note x G	Note	Note x G	Note	Note x G
Matlab Fähigkeit (ja/nein)	16	10	156	10	156	10	156	10	156	
Cruise Fähigkeit (ja/nein)	13	0	0	10	133	0	0	0	0	
Preis	2	3	7	10	22	5	11	0	0	
CAN Schnittstelle	18	10	178	10	178	10	178	10	178	
Ethernet Schnittstelle	4	10	44	10	44	5	22	10	44	
GUI Editor	7	10	67	0	0	5	33	10	67	
Rechenleistung	20	5	100	8	160	10	200	10	200	
Größe und Gewicht	2	10	22	1	2	10	22	7	16	
Simulation des Gesamtprüfstandes (ja/nein)	11	0	0	10	111	0	0	0	0	
Fahrerleitgerät integrierbar? (ja/nein)	7	0	0	10	67	0	0	0	0	
			573		873		622		660	

Tabelle 3-4: Nutzwertanalyse der Echtzeitumgebungsvarianten

Das Ergebnis zeigt eine eindeutige Dominanz des AVL PUMA-PC-Systems mit PUMA Open als Echtzeitplattform. Diese Variante ist für AVL-interne Zwecke kostenlos und bietet eine hohe Modularität. Das Fahrerleitgerät kann einfach integriert werden, ein neues User Interface ist nicht nötig. Nachteilig sind der hohe Platzbedarf und das hohe Gewicht.

### 3.3 Fahrzeugmodellbildung

Zur modellhaften Abbildung des Fahrzeuges wird eine Recherche der in der AVL existierenden Modellbildungsvarianten durchgeführt. Anschließend erfolgt eine Klassifizierung analog zu Kapitel 3.2.5. Als Grundanforderungen sind hier Echtzeitfähigkeit sowie Anpassungsfähigkeit an unterschiedliche Fahrzeugtypen zu nennen.

Folgende Fahrzeugmodellbildungsvarianten werden evaluiert:

- MATLAB/Simulink
- AVL CRUISE
- MoBEO



### 3.3.1 MATLAB/Simulink [14]

MATLAB war ursprünglich eine Benutzeroberfläche für Standardroutinen der numerischen Algebra, daher der Name MATrix LABoratory. Dieses Tool hat sich zu einer eigenständigen Software entwickelt, das zur numerischen Berechnung und Visualisierung von technischen Problemen verwendet wird. Eine weitere Säule unter MATLAB ist Simulink und der darin verfügbare Real-Time-Workshop.

Simulink ist eine grafische Entwicklungsumgebung für lineare und nichtlineare dynamische Systeme. Der Aufbau von Simulink-Modellen ist blockorientiert, d.h. das Modell wird über Funktionsblöcke aufgebaut, welche über Signalfusslinien miteinander verbunden sind.

Simulink kommt sehr oft für die Lösung technischer Probleme zum Einsatz, da es die Möglichkeit bietet, mathematische Probleme grafisch abzubilden.

### 3.3.2 AVL CRUISE [15]

CRUISE ist ein Produkt von AVL AST (AVL Advanced Simulation Methods) zur Simulation und Optimierung von Fahrbarkeit (Beschleunigungsverhalten und Elastizität), Kraftstoffverbrauch, Emissionen, Getriebeübersetzungen und Bremsperformance. Der modulare Aufbau ermöglicht die Modellierung aller existierenden und zukünftigen Fahrzeugkonzepte. Durch die Verwendung unterschiedlicher Module, wie z.B. Verbrennungsmotor, Getriebe, Kupplung und Elektromotor, lassen sich sowohl exotische Konzepte als auch Hybridfahrzeuge modellieren und evaluieren. CRUISE bietet die Möglichkeit, Fahrermodelle mit individuellem Fahrverhalten, variabler Gangwechselzeit und weiteren Variabilitäten einzusetzen. Auch ein Temperaturmodell des Motorverhaltens für Kaltstartuntersuchungen ist vorhanden.

Um die Anzahl der möglichen Applikationen weiter zu erhöhen steht ein MATLAB/Simulink Interface zur Verfügung, welches das in CRUISE modellierte Fahrzeugmodell in ein Simulink-Modell exportiert und zuvor festgelegte Signale als Ein- und Ausgang verwendet. Mittels des Common Model Compilers (CMC) wird eine Matlab-dll (S-Function), also ein in MATLAB ausführbarer Programmcode, erstellt. Die Ein- und Ausgänge können nun in Simulink verwendet werden. Durch Verwendung des in Kapitel 3.2.2 beschriebenen ARTE.Lab-Studios besteht die Möglichkeit zur Kompilierung dieser CRUISE-Simulink-Modelle für die Echtzeitumgebung ARTE. Die eingebaute Lizenzierung lässt jedoch keine Verwendung von AVL COBRA und Bernecker & Rainer zu.

### 3.3.3 MoBEO – Model Based Engine Optimization [16]

Für die Optimierung von Motoren kommt in der AVL MoBEO zum Einsatz. Diese Motormodelle sind so optimiert, dass eine Simulation in Echtzeit und schneller möglich ist. Dazu werden physikalische Modelle verwendet, die zur Beschleunigung der Simulationsgeschwindigkeit mit empirisch ermittelten Komponenten vereinfacht sind. Die Modelle von MoBEO basieren auf Simulink. Aus diesem Grund werden unkompilierte MoBEO-Modelle nicht weitergegeben, da man so Einblick in den Aufbau des Modells hätte. Die Parametrierung erfolgt ausschließlich durch das Team von MoBEO.

### 3.3.4 Auswahl des Fahrzeugmodells

Bei der Festlegung der Kriterien wird darauf geachtet, dass die Modelle bereits in der Entwicklungsumgebung simuliert werden können. Das erspart Zeit bei der Modellkalibrierung und erlaubt Versuche am Entwicklungsrechner ohne Verwendung der Echtzeit-Hardware. Für diese Simulationsaufgaben ist sowohl MATLAB als auch CRUISE geeignet. Werden also keine CRUISE-Modelle verwendet, muss das gewählte Modell MATLAB-fähig sein.

### Paarvergleich

Anhand eines Paarvergleiches wird die Gewichtung der Kriterien ermittelt, die sich auf die Kompatibilität von Modellierungstools auf Echtzeitplattformen beziehen. Weitere Kriterien lassen sich den Themen Veränderbarkeit und Anpassungsfähigkeit zuordnen.

Kriterien	Kenntnis des Modellaufbaus	ARTE Fähigkeit	MATLAB Fähigkeit	Modellumfang	Online Parameter	Grafische Darstellung	Anpassungsfähigkeit	Preis	Summe	Gewichtung
Kenntnis des Modellaufbaus	X	1	1	0	1	1	0	1	5	18
ARTE Fähigkeit	0	X	0	1	1	1	0	1	4	14
MATLAB Fähigkeit	0	1	X	1	1	1	0	1	5	18
Modellumfang	1	0	0	X	1	1	0	1	4	14
Online Parameter	0	0	0	0	X	0	0	1	1	3,6
Grafische Darstellung	0	0	0	0	1	X	0	0	1	3,6
Anpassungsfähigkeit	1	1	1	1	1	1	X	1	7	25
Preis	0	0	0	0	0	1	0	X	1	3,6
								Summe	28	100

Tabelle 3-5: Paarvergleich der Kriterien zur Recherche der Fahrzeugmodellbildung

## Nutzwertanalyse

Analog zu Kapitel 3.2.5 werden Entscheidungsfragen mit „10“ oder „0“ für „ja“ und „nein“ benotet.

Simulink-Modelle können bereits einen kompilierten Teil enthalten oder Kennfelder verwenden, die nicht eindeutig zuordenbar sind. Aufgrund der oft fehlenden Kommentare ist der Aufbau oft schwer nachzuvollziehen, daher die Note „8“. CRUISE hingegen bietet ein modulares System, wo alle Module zusammenwirken und dessen Funktionen im Handbuch genau erläutert sind. MoBEO Modelle werden nur als kompilierte Dateien ausgegeben und können nicht verändert werden. Modelleinsicht ist ebenfalls nicht möglich.

Die Benotung des Modellumfangs bezieht sich auf die Tatsache, dass zwar sehr viele MATLAB/Simulink-Modelle existieren, diese aber aufgrund verschiedener Modellierungsstile nicht uneingeschränkt miteinander vereinbar sind. Kompatibilität unter den Teilmodellen ist daher nicht immer gewährleistet (Note „7“). MoBEO beschreibt nur Motormodelle, diese jedoch mit hohem Umfang. Der Rest der Antriebsstrangmodellierung muss durch Modelle aus anderen Quellen ergänzt werden (Note „5“).

Alle Parameter in einem Simulink-Modell sind verstellbar, wenn das gewählte Echtzeitsystem dies erlaubt (Note „10“). CRUISE unterstützt nur die Verstellung von skalaren Größen (Note „5“).

Die Modellbildung in Simulink und CRUISE beruht auf der Basis von Funktionsblöcken. Im Gegensatz zu Simulink erfolgt in CRUISE die Darstellung von Kennlinien und Kennfeldern in übersichtlichen Diagrammen (Note „10“). In Simulink werden diese nur qualitativ in den Funktionsblöcken angezeigt (Note „5“).

	Gewicht G	Matlab/Simulink		AVL CRUISE		MoBEO	
		Note	Note x G	Note	Note x G	Note	Note x G
Kenntnis des Modellaufbaus	18	8	143	10	179	0	0
ARTE Fähigkeit (ja/nein)	14	10	143	10	143	0	0
MATLAB Fähigkeit (ja/nein)	18	10	179	10	179	10	179
Modellumfang	14	7	100	10	143	5	71
Online Parameter	4	10	36	5	18	0	0
Grafische Darstellung	4	5	18	10	36	0	0
Anpassungsfähigkeit	25	10	250	10	250	0	0
Preis	4	0	0	0	0	10	36
			868		946		286

Tabelle 3-6: Nutzwertanalyse der Fahrzeugmodellbildungsvarianten

Das Ergebnis zeigt, dass sowohl CRUISE als auch MATLAB/Simulink gut für die Fahrzeugmodellierung geeignet sind. Der modulare Aufbau eines Gesamtfahrzeugs mit allen nötigen Modulen in einem Tool frei parametrierbar zusammengefasst ermöglicht eine rasche und nachvollziehbare Modellbildung. Als Nachteil von CRUISE ist die nur teilweise mögliche Online-Änderung von Parametern zu nennen.

### 3.4 Fahrerleitgerät

Das Fahrerleitgerät hat die Aufgabe, dem Fahrer alle nötigen Informationen für die Durchführung eines Testlaufes zu geben. Für die Auswahl wird von der Verwendung von PUMA-PC mit CRUISE ausgegangen. Da nur „Bernecker & Rainer“-Hardware die Möglichkeit einer selbst programmierbaren Oberfläche bietet, muss auf vorhandene Fahrerleitgeräte zurückgegriffen werden.

Für das Fahrerleitgerät ergeben sich folgende Anforderungen:

- Darstellung der Sollgeschwindigkeit samt Toleranzband
- Darstellung des Zeitpunktes für Motorstart und -stopp
- Darstellung des Gangwechsels und Kupplungsstatus bei manuellen Getrieben
- Darstellung der Wählhebelposition bei Automatikgetrieben

Für AVL-Prüfstände mit „AVL PUMA Open“-Automatisierungssystemen sind drei Arten von Fahrerleitgeräten verfügbar.

#### **Pierburg DAS4000**

Das Fahrerleitgerät Pierburg DAS4000 war bis Sommer 2012 auf den Rollenprüfständen in der AVL im Einsatz. Es verfügt über einen eigenen Rechner, der über RS232 mittels AK Protokoll angesteuert wird. Alle Prüfzyklen sind auf diesem Rechner abgelegt (Auswahl über PUMA Open). Über eine analoge Spannung wird die momentane Fahrzeuggeschwindigkeit von der Rollensteuerung übergeben.

## **SADA XP**

Das Fahrerleitgerät SADA XP löst das DAS4000 ab. Die Ermittlung der Ist-Geschwindigkeit erfolgt auch hier über einen analogen Eingang. Der Sollwertverlauf wird nicht lokal gespeichert, sondern von PUMA Open an das Fahrerleitgerät gesendet.

## **iDriversAid**

Dieses Fahrerleitgerät entstand im Zuge der Entwicklung von PUMA Vehicle. PUMA Open ist ein System, das grundsätzlich für den Betrieb von Motorprüfständen konzipiert ist. Durch den modularen Aufbau kann es auch für Fahrzeugprüfstände genutzt werden. Der Nachteil dabei ist, dass zusätzliche Hardware zur Steuerung des Rollenprüfstandes sowie für das Fahrerleitgerät notwendig ist. Aus diesem Grund startete die Entwicklung von PUMA Vehicle, um ein System zu erschaffen, welches Fahrerleitgerät, Rollensteuerung und Automatisierung auf einem Rechner lauffähig macht. Dazu ist eine Softwarelösung für ein Fahrerleitgerät entwickelt worden, das mit PUMA Open kompatibel ist.

## **Auswahl des Fahrerleitgeräts**

Durch die Umstellung von DAS4000 auf SADA XP stehen mehrere DAS4000 zur Verfügung. Auch die Softwarelösung von PUMA Vehicle „iDriversAid“ ist für Anwendungen in der AVL verfügbar. Da die Softwarelösung direkt am PUMA-PC installiert wird und somit kein Platz für zusätzliche Hardware notwendig ist, kommt im Simulator die Softwarelösung „iDriversAid“ zum Einsatz.

### 3.5 Nachbildung der Fahrgastzelle

Zur Auswahl des Sitzes bzw. des Rahmens für den Simulator gibt es die Möglichkeiten, diese Komponenten selbst zu konstruieren oder auf Fahrersitze zurückzugreifen, die im Computerspielsektor zum Einsatz kommen. Eine Internetrecherche erbringt die Information, dass mögliche Fahrersitze in einer Preisklasse von unter 500 Euro liegen. Aus diesem Grund fällt die Wahl auf den „Speedmaster V2.0“ von speedmaster2.de. Dieser Aufbau bietet bereits vorhandene Montiermöglichkeiten für Gangschaltung und Pedalerie und besitzt einen verstellbaren Sitz.



Abbildung 3-4: Speedmaster V2.0 [17]

### 3.6 Pedale und Gangschaltung

Um das Fahrzeug für den Prüfstandstechniker möglichst realitätsnah abzubilden, müssen Kupplungs-, Brems- und Fahrpedal an den gleichen Positionen wie bei einem realen Fahrzeug vorhanden sein. Der Simulator muss auch einen Schalthebel aufweisen. Analog zur Nachbildung der Fahrgastzelle ist es möglich, die Pedalerie selbst zu konstruieren und zu fertigen. Zur Überprüfung der Eignung von zugekaufter Pedalerie wird eine Spielsteuerung subjektiv getestet. Zur Auswahl steht nur jene, die aus allen drei Pedalen besteht. Die Auswertung der Signale muss durch analoge Spannungsmessung möglich sein.

#### Eigenkonstruktion

Bei Spielsteuerungen befinden sich die Drehachsen der Pedale an der Unterseite, die Pedale sind also stehend montiert. Am realen Fahrzeug sind Kupplungs- und Bremspedal jedoch hängend ausgeführt. Das Kraft-Weg Verhältnis eines Kupplungspedales entspricht der Federkennlinie der Kupplung. Beim Bremspedal ist zusätzlich eine Dämpfung vorhanden. Der Vorteil der Eigenkonstruktion liegt in der Möglichkeit, die Geometrie der Pedale eines realen Fahrzeuges exakt umzusetzen. Weiters können die Kraft-Weg Verhältnisse durch Einsatz von Feder- und Dämpferelementen nachgebildet werden.

### **Logitech-G27-Gaming-Lenkrad**

Da eine Spielsteuerung des Typs Logitech G27 leihweise verfügbar ist, wird diese auf Erfüllung der nötigen Anforderungen getestet. Diese Spielsteuerung besteht aus den drei Komponenten Lenkrad, Pedalerie und Gangschaltung. Da das Lenkrad nicht von Interesse ist, werden nur die Gehäuse von Pedalerie und Ganghebel demontiert. Es stellt sich heraus, dass beide Komponenten auf Potentiometer, also regelbare Widerstände, basieren. Im Lenkrad befindet sich ein Analog/Digital-Wandler, der die analogen Spannungen der Pedale und der Gangschaltung auswertet. Dieser Umstand macht es möglich, die Position von Pedalen und Ganghebel über das Messen der Potentiometerspannungen zu ermitteln. Somit ist der Einsatz am PUMA-PC möglich. Der Kraft-Weg-Verlauf der Pedale ist den Verhältnissen eines realen Fahrzeuges angeglichen, so ist das Fahrpedal sehr leichtgängig, das Kupplungspedal schwergängiger und das Bremspedal hat am meisten Widerstand. Da dies über Druckfedern geschieht, ist jedoch der dämpfende Teil am Bremspedal nicht vorhanden.

### **Auswahl der Pedale und Gangschaltung**

Trotz der nicht realitätsgetreuen Kraft-Weg-Verhältnisse der Pedale fällt die Wahl auf das Logitech-G27-Gaming-Lenkrad, da die Federkraft der Pedale ausreicht, um bei Betätigung eine ausreichende Rückmeldung des betätigten Pedalweges zu bekommen. Die Schwierigkeit bei der Durchführung von Abgastests und somit das Hauptaugenmerk des Simulators liegen an den Beschleunigungs- und Konstantfahrten. Die nötige Bremskraft ist bei den Verzögerungsteilen gering, somit wird die Bremse meist nur mit geringem Gradienten betätigt. Daher ist der dämpfende Anteil des Bremssystems nicht ausschlaggebend und die Federkraft kann die Verhältnisse akzeptabel wiedergeben. Da über die Potentiometer schon ein System zur Wegmessung vorhanden ist, kann dieses direkt weiterverwendet werden. Dabei werden die Spannungen ausgewertet und in Pedalwege von 0-100 % bzw. in den aktuellen Gang umgerechnet. Als Nachteil lässt sich die stehende Anordnung der Pedale nennen, die sowohl einen kleineren Pedalweg als auch eine andere Bewegungsrichtung zur Folge hat.



Abbildung 3-5: Logitech-G27-Spielsteuerung [18]

### 3.7 Überblick des gewählten Konzeptes

Zusammenfassend besteht das Komplettsystem aus folgenden Komponenten:

- AVL PUMA-PC mit AVL ARTE Echtzeitumgebung als Echtzeitplattform
- AVL CRUISE zur Fahrzeugmodellbildung
- PUMA Vehicle Softwarelösung als Fahrerleitgerät
- Speedmaster V2.0 als Cockpitaufbau
- Logitech G27 Gaming Lenkrad als Pedalerie und Gangschaltung

#### Kurzbeschreibung der Komponenten

Die Nachbildung der Fahrgastzelle bietet das Grundgerüst des Simulators und soll alle Komponenten aufnehmen. Die gewählte Echtzeitumgebung AVL PUMA Open basiert auf der PUMA-PC-Hardware und ist in einem 19-Zoll-Einschubgehäuse verbaut. Der analoge Eingang „AVL F-FEM-AIN“ ist ebenfalls in einem 19-Zoll-Gehäuse untergebracht. Die Visualisierung erfolgt über einen am Simulator angebrachten 24-Zoll-Bildschirm.

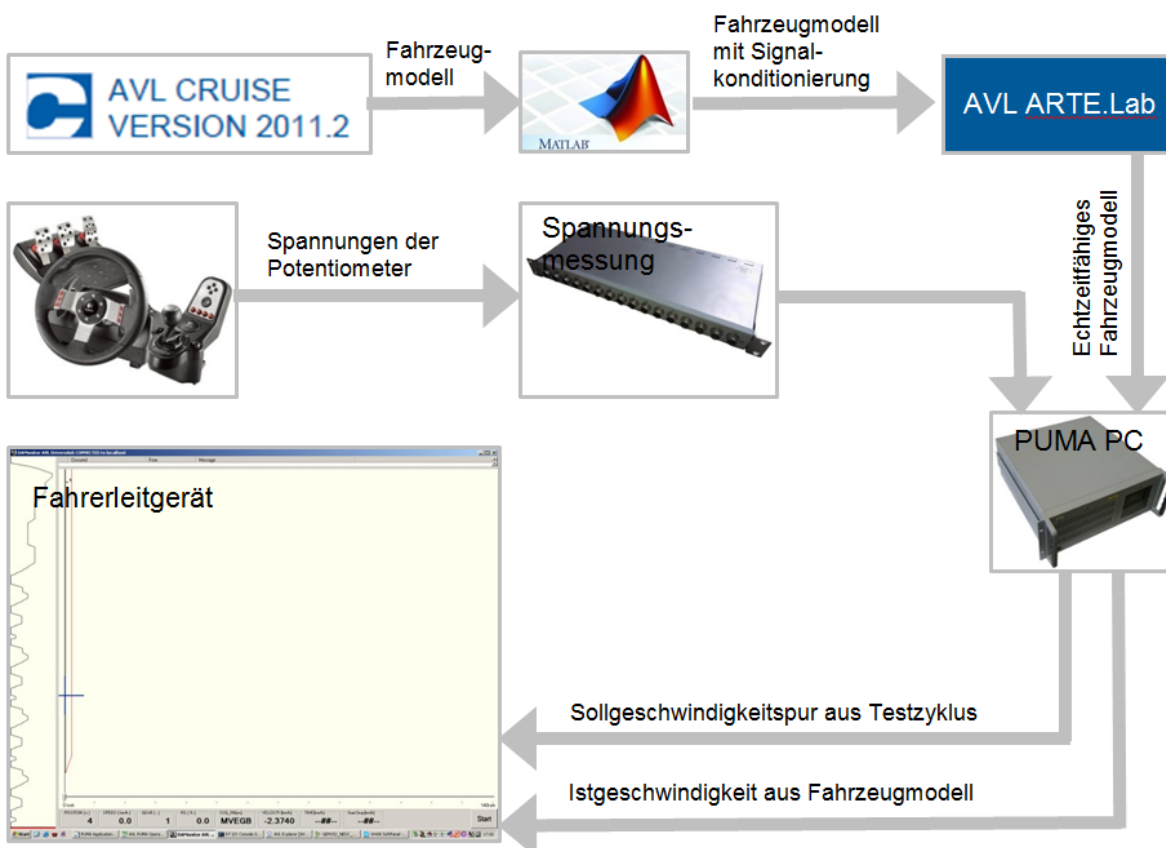


Abbildung 3-6: Überblick des gewählten Konzeptes



## 4 Aufbau des Simulators

Der Simulator muss ohne Demontage transportfähig und an beliebigen Orten einsatzfähig sein. Alle Komponenten, einschließlich Netzteilen und Bildschirm, müssen somit im Simulator untergebracht werden. Dabei wird stets auf einfache Austauschbarkeit aller Komponenten geachtet.

### 4.1 Aufbau der Fahrgastzellennachbildung

Im 3D-CAD-Programm Autodesk Inventor erfolgt eine konzeptionelle Modellierung der Komponenten zur Durchführung einer Positionsanalyse. Die Analyse ergibt, dass der Aufbau ohne weitere Anpassungen nicht in der Lage ist, alle Komponenten aufzunehmen. Aus diesem Grund wird ein Rahmen konstruiert, der unterhalb des Fahrersitzes positioniert wird und alle Komponenten aufnimmt.

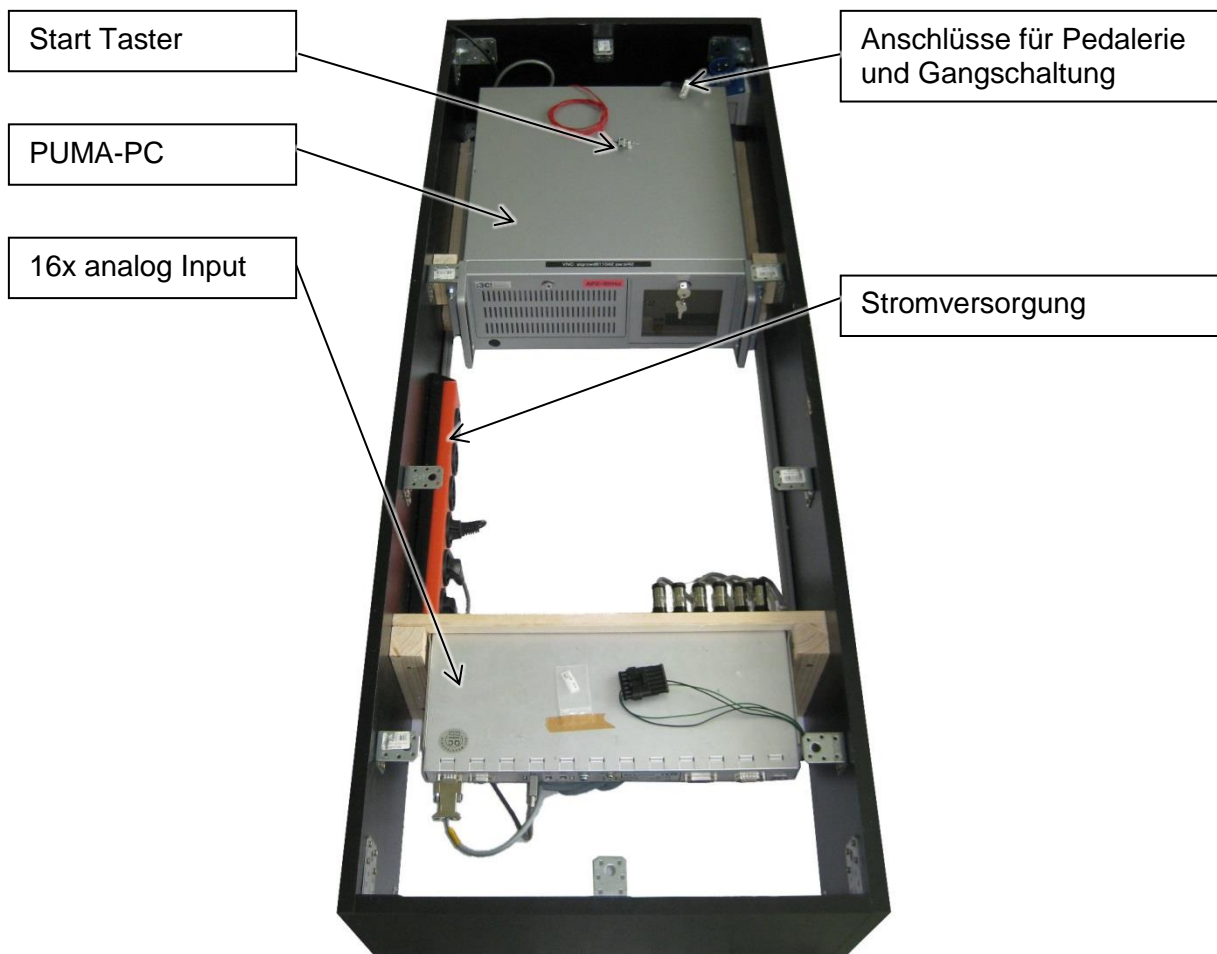


Abbildung 4-1: Rahmen zur Aufnahme der Hardware-Komponenten

## 4.2 Integration der Komponenten

### PUMA-PC

Durch die Einschubtechnik des PUMA-PCs stehen zwei seitliche Schienen zur Montage zur Verfügung. An Distanzleisten aus Holz werden zwei Aluminium L-Profile am Rahmen angebracht, die in die Nut der Schienen eingreifen. Durch die vorgesehene Verschraubung erfolgt die Fixierung des Gehäuses am Rahmen.

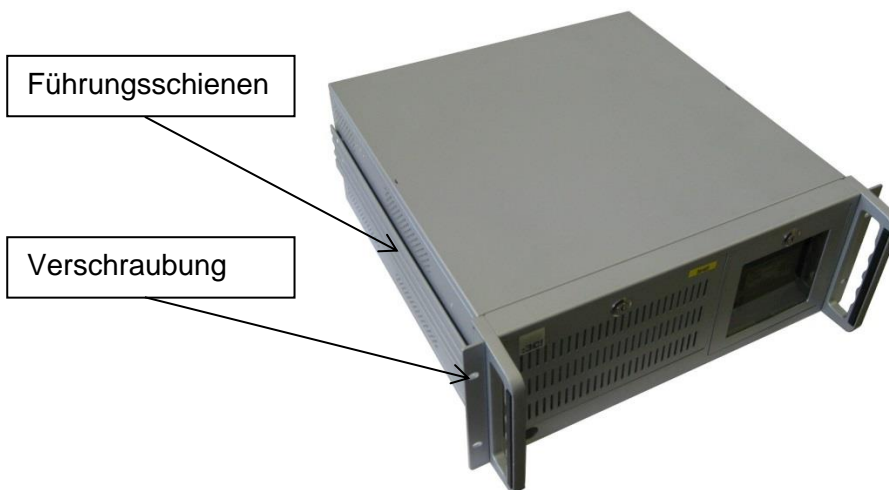


Abbildung 4-2: PUMA-PC

### Analoge Spannungsmessung F-FEM-AIN

Der F-FEM-AIN genannte Analog/Digital-Wandler der AVL-PUMA-PC-Hardware verfügt ebenso über den 19-Zoll-Standard und kann an der Vorderseite fixiert werden. Zur Befestigung des A/D-Wandlers und dessen Netzteil dient eine Holzplatte, an der die Komponenten verschraubt werden.

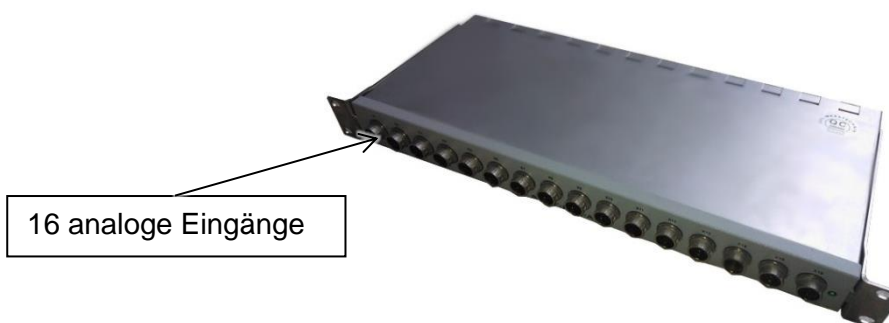


Abbildung 4-3: AVL-F-FEM-AIN-Analog-Digital-Wandler

## Spannungsversorgung und Verkabelung

Um den Monitor, den PUMA-PC und den Analog/Digital-Wandler mit Spannung zu versorgen, ist eine 230 V Versorgung notwendig. Dazu wird eine Verteilerleiste installiert. Der Simulator verfügt zur getrennten Verlegung von Signalleitung und Spannungsversorgung auf beiden Seiten über Kabelkanäle zwischen PUMA-PC und Analog/Digital-Wandler.

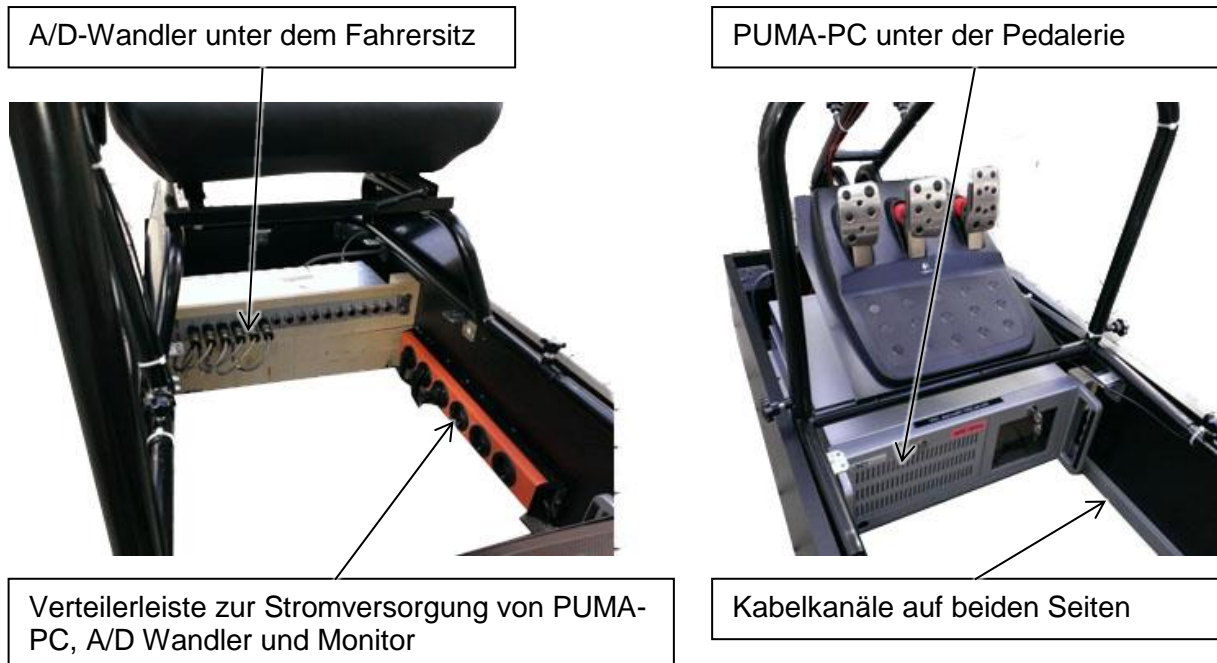


Abbildung 4-4: Platzierung der Komponenten im Rahmen

## 4.3 Einbindung der Pedale und des Ganghebels

Zur Untersuchung der Signale werden Pedalerie und Ganghebel geöffnet. Die Versorgung der Pedale erfolgt unabhängig voneinander mit 5 V. An einem regelbaren Widerstand liegt abhängig von der Pedalstellung eine Spannung zwischen 0 V und 5 V an. Diese Signale werden über eine 9-polige Steckverbindung an das Lenkrad übergeben. Über dieselbe Methode erfolgt die Positionsmessung des Ganghebels. Die Spannung ist proportional zum Verdrehwinkel um die x- bzw. die y-Achse. Im Lenkrad befindet sich ein Analog/Digital Wandler, der die Signale der Spielkonsole zur Verfügung stellt. Da der Lenkwinkel nicht von Interesse ist, werden Pedal- und Gangschaltungssignale direkt über die 9-polige Steckverbindung mit dem analogen Eingang des PUMA-PCs verbunden. Die 5 V Versorgung erfolgt ebenfalls über den PUMA-PC über eine USB-Schnittstelle.

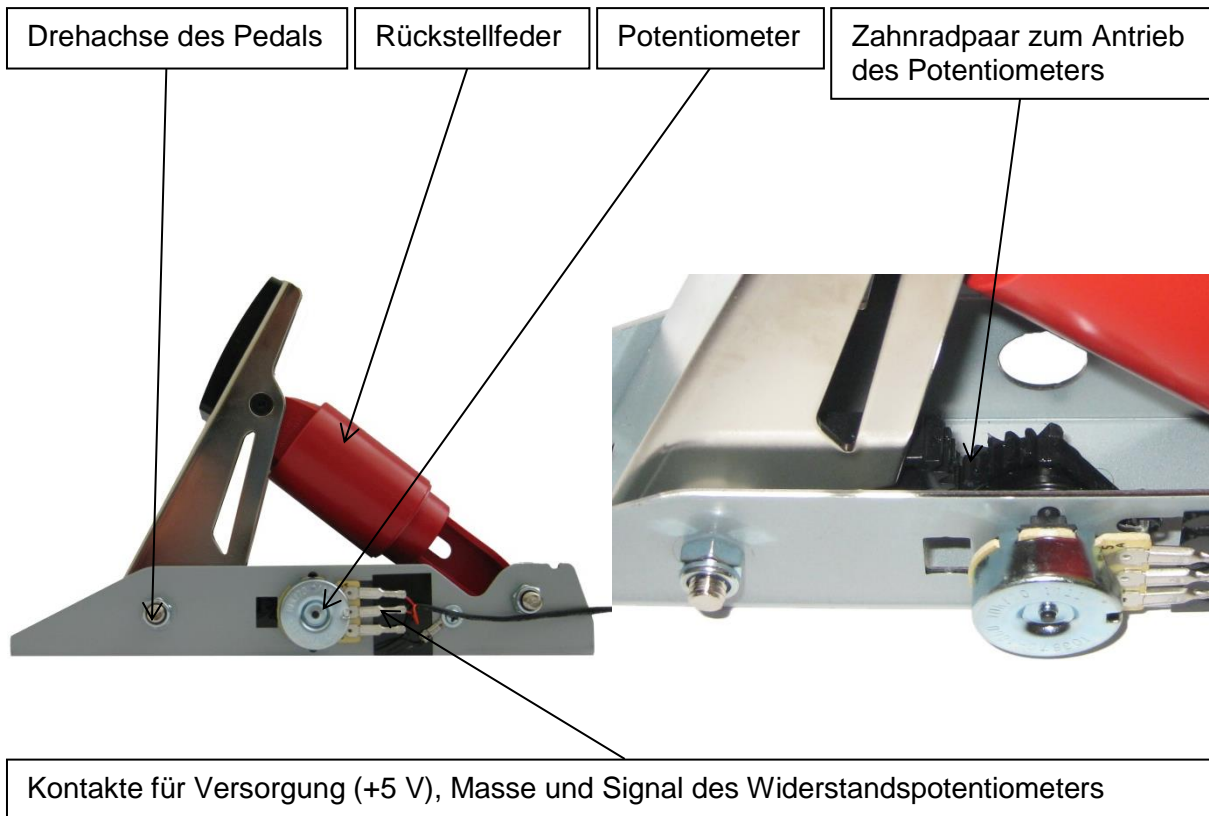


Abbildung 4-5: Funktion der Pedalerie

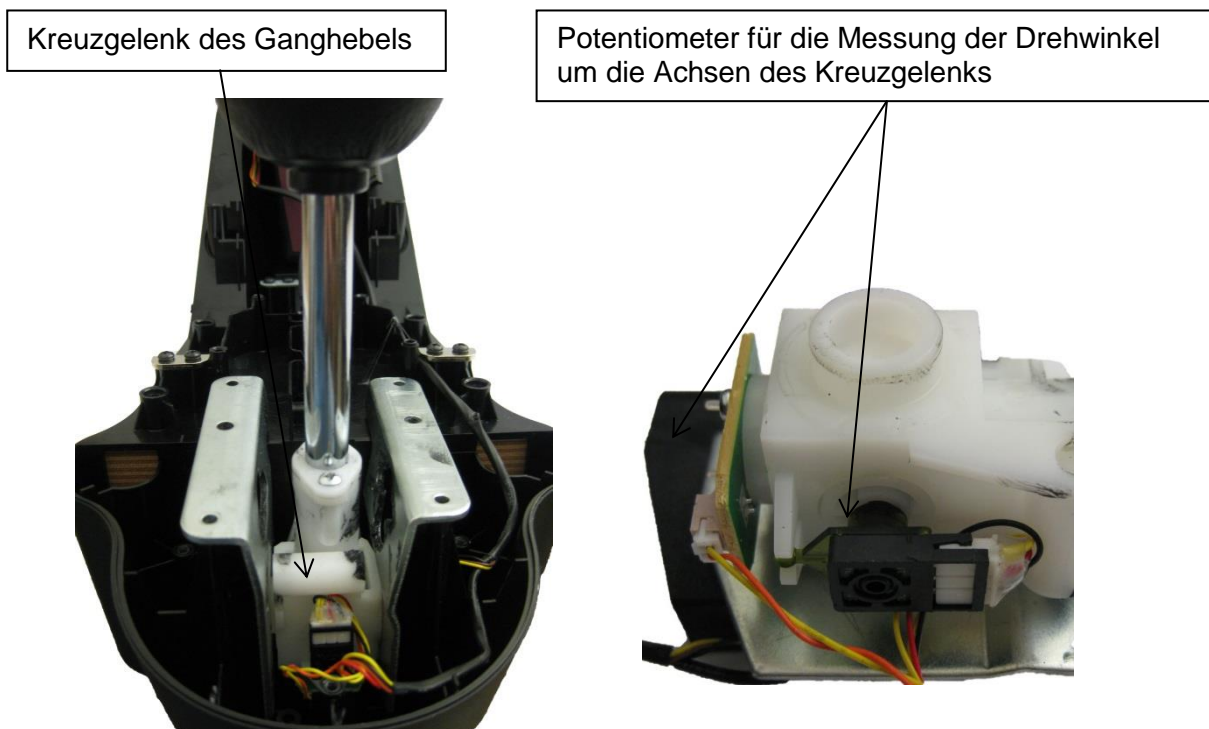


Abbildung 4-6: Messung der Position des Ganghebels

## 4.4 Simulatore Aufbau

Abbildung 4-7 zeigt den Aufbau des Simulators. Der aus beplankten Pressplatten bestehende Rahmen wird mit sechs Stahlwinkeln versehen, die den Fahrersitz samt Gestell aufnehmen. Das Gestell weist bereits vorgefertigte Aufnahmen für die Pedalerie und Gangschaltung auf. Zur Fixierung des Gestells ragt der Holzrahmen über die Bodenrohre des Gestells. Damit ist es möglich, das Gestell samt Monitor und Sitz ohne Werkzeug zu demontieren.



Abbildung 4-7: Simulatore Aufbau

## 4.5 Einbindung des Fahrerleitgerätes in PUMA Open

Das Installationspaket des Fahrerleitgeräts wird von einem Softwareentwickler von PUMA Vehicle bezogen. Durch Kopieren der Dateien in das Verzeichnis der „PUMA Open“-Installation am Simulator ist die Software bereits testfähig. Das „PUMA Open“-Automatisierungssystem verfügt über ein Tool zum Start und zur Überwachung aller Prozesse. In einer Konfigurationsdatei kann das Fahrerleitgerät als Prozess deklariert werden und wird beim Start von PUMA Open auch gestartet. Die frei verwendbaren Variablen von PUMA Open („Normnamen“ bzw. „Systemnamen“) senden die nötigen Informationen für das Fahrerleitgerät. Dieses verfügt so über die nötigen Informationen für dessen Betrieb.

## 4.6 Einbindung von Fahrzyklen

Das gewählte Fahrerleitgerät unterstützt die Sollwertdatenbanken der Abgastestzyklen des Rollenprüfstandes nicht. Es ist nötig, das Toleranzband hinzuzufügen, da es im Gegensatz zu dem Fahrerleitgerät des Rollenprüfstandes am Fahrerleitgerät iDriversAid nicht eigenständig berechnet wird.

## 4.7 Weitere Schritte

Die Integration des Fahrzeugmodells wird in Kapitel 5.2 erklärt. Nach Einbindung des Modells erfolgt die Berechnung ab dem Start des Automatisierungssystems. Alle nötigen Informationen werden über „PUMA Open“-Normnamen bzw. -Systemnamen an das Fahrerleitgerät geschickt.

PUMA Open bietet die Möglichkeit eines frei parametrierbaren Automatiklaufes. So werden dort Routinen erstellt, die vor dem Start des Testlaufs einen Systemcheck und eine Kalibrierung des Abgasmesssystems durchführen und die Datenerfassung initialisieren.

Um einen Testlauf am Simulator zu starten, wird von einem PUMA-Fachexperten ein Automatiklauf erstellt, der die Informationen der Sollwerte von Geschwindigkeit, Motorstart und -stopp und Gangwechsel an das Fahrerleitgerät sendet. Dieses gibt dem Fahrer die Information, dass der Test startbereit ist und wartet auf den Startbefehl. Beim Start des Automatiklaufs empfängt das Fahrerleitgerät fortlaufend die Sollwerte von PUMA Open. Der Ist-Wert der Geschwindigkeit wird sowohl numerisch als auch mittels Fadenkreuz angezeigt.

Ein vom Automatisierungssystem gestarteter und gestoppter Rekorder speichert alle Größen, die von Interesse sind.

## 5 Modellbildung

Das Training von Rollentests auf unterschiedliche Fahrzeugkonzepte und Fahrzeugklassen erfordert eine hohe Modularität des Fahrzeugmodells. Dem Prüfstandstechniker muss es zugleich möglich sein, die gewünschte Fahrzeugklasse auszuwählen.

Dieses Kapitel befasst sich mit der Modellbildung in AVL CRUISE und der Integration in die Echtzeitumgebung von PUMA Open.

### 5.1 Fahrzeugmodellierung in CRUISE

Wie in Abschnitt 3.3.2 beschrieben, ist AVL CRUISE ein modulares System zur Modellbildung unterschiedlichster Fahrzeuge und Antriebskonzepte. Der Simulator soll verschiedene Fahrzeugkonzepte als Modelle bereitstellen, um ein Training dieser zu ermöglichen. Dazu wird der Vorgang der Modellerstellung und Parametrierung anhand von vorhandenen Messdaten eines konventionellen Mittelklassefahrzeuges mit manuellem Getriebe und Hinterradantrieb beschrieben.

#### 5.1.1 Module in CRUISE [15]

CRUISE bietet für alle Komponenten des Antriebsstrangs Modelle an, die in CRUISE „Module“ genannt werden. Somit ist es möglich, verschiedene Kombinationen des Antriebsstrangs nachzubilden und zu simulieren. Alle Module sind sehr flexibel gestaltet und ermöglichen unterschiedliche Eingangsgrößen. Für all jene, die eine Massenträgheit besitzen, ist diese im jeweiligen Modul zu parametrieren.

Über den sogenannten „Vehicle Modeler“ werden per Drag&Drop aus einer Toolbar die Module ausgewählt und platziert. Sie sind miteinander über eine „Energieflusslinie“ verbunden, die den Energiefluss des Antriebsstrangs repräsentiert. Diese Linie enthält Signale wie Drehzahl und Drehmoment, die dann den Modulen zur Verfügung stehen. Weitere erforderliche Signale, wie z.B. Bremsdruck, werden über den „Data Bus“ vom gewählten Ursprung zum jeweiligen Modul verbunden. Dieser „Data Bus“ stellt 199 Inputs und 199 Outputs zum Transfer von Variablen zur Verfügung, die nicht in der „Energieflusslinie“ enthalten sind. Nachfolgend werden die in CRUISE verwendeten Module allgemein beschrieben. Die Beschreibung der Anwendung dieser Module zur Fahrzeugmodellierung am Simulator erfolgt in Kapitel 5.1.2.



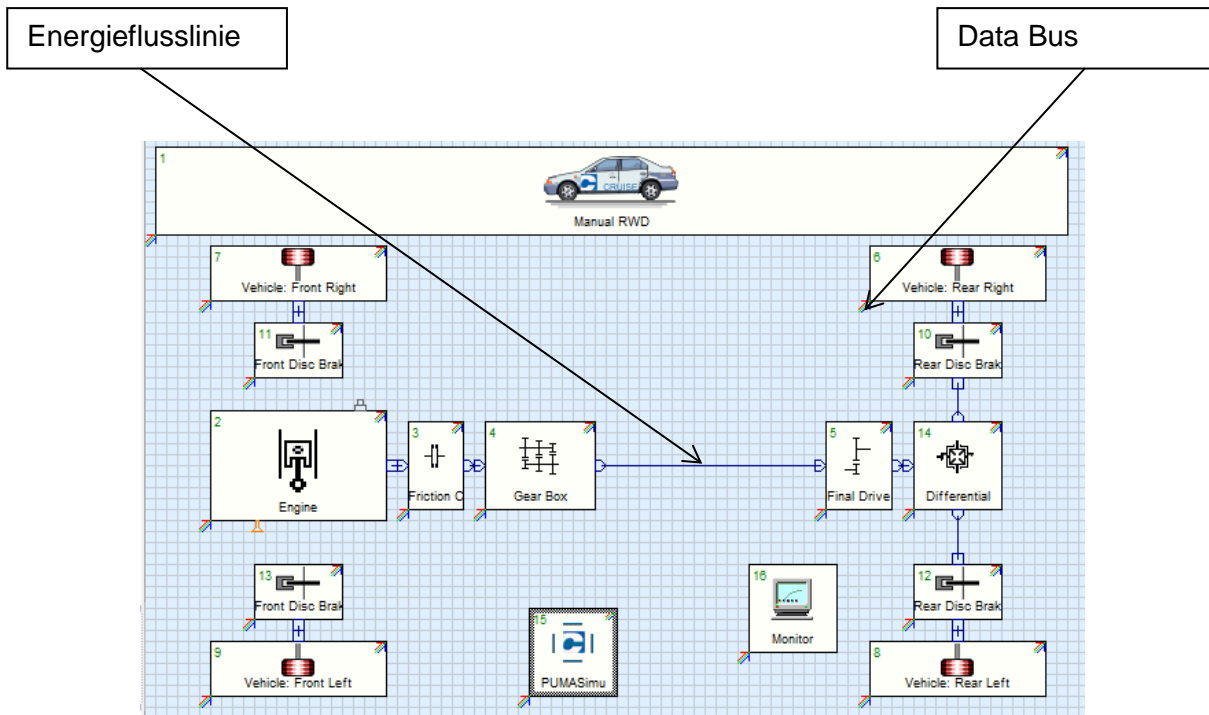


Abbildung 5-1: Energieflusslinie und Data-Bus in CRUISE

### Modul Combustion Engine

Dieses Modul beinhaltet eine parametrierbare Vollastkurve. Die Bedatung kann auf verschiedene Arten erfolgen. So sind über die Drehzahl etwa die Parametrierung von Motorleistung, Drehmoment oder Mitteldruck möglich. Der Hubraum und die Zylinderzahl sowie Leerlauf und Maximaldrehzahl sind über Konstanten zu parametrieren. Für Kraftstoffverbrauch und die Emissionen  $\text{NO}_x$ , HC, CO und Partikel stellt CRUISE Kennfelder in Abhängigkeit von Drehzahl und Motorlast zur Verfügung. Um das Kaltstartverhalten zu beachten, ist ein Temperaturmodell integriert. Der Schubetrieb des Motors ist über eine Kennlinie, abhängig von der Motordrehzahl, abgebildet. Ein Turbolader kann ebenso in das Motormodell aufgenommen werden. Die Parametrierung des Turboladers erfolgt entweder über den Ladedruck oder den Mitteldruck als Funktion der Motordrehzahl. Das Ansprechverhalten wird über eine konstante Dauer oder in Abhängigkeit von der Motordrehzahl angepasst. Der Aufbau von Ladedruck bzw. Mitteldruck startet bei einem eingestellten Startwert des Mitteldrucks. Wahlweise wird der Aufbau des Ladedrucks dann über einen PT1- oder einen PT2-Filter modelliert. Unabhängig vom Turbolader kann die vom Motor zum Aufbau der maximalen Leistung benötigte Zeitdauer eingestellt werden.



### **Modul Friction Clutch**

Das Modell der Reibkupplung unterbricht den Kraftfluss zwischen Motor und Getriebe. Es ermöglicht eine Gleitbewegung, um die unterschiedlichen Drehzahlen von Motor und Antriebsstrang zu kompensieren. Die Kupplung kann über 2 Detailstufen parametrierbar werden. Für das simple Modell erfolgt nur die Definition des maximal möglichen Drehmoments, über den Betrag des Kupplungspedalwegs wird daraus das übertragbare Drehmoment bestimmt. Bei dem detaillierten Modell wird die Geometrie der Kupplung parametrierbar. Eine Kennlinie ordnet dem Pedalweg eine Anpresskraft auf den Kupplungsbelag zu. Bei diesem Modell sind Haft- und Gleitreibungskoeffizient als Konstantwert oder abhängig von der Differenzdrehzahl einstellbar.

### **Modul Gear Box**

Das Getriebemodell enthält eine variable Anzahl von parametrierbaren Getriebeübersetzungen. Dabei werden für jeden Gang die Verluste über einen Konstantwert, abhängig von der Temperatur, sowie abhängig von Drehmoment und Motordrehzahl eingestellt.

### **Modul Final Drive**

Das Modell der Achsübersetzung errechnet Drehmoment und Drehzahl an der angetriebenen Achse über die Parametrierung der Übersetzung. Ähnlich zum Getriebemodell werden die Verluste abhängig von der Eingangsdrehzahl, von der Temperatur sowie von Drehmoment und Motordrehzahl eingestellt.

### **Modul Differential**

Dieses Modell dient zur Gleichverteilung des anliegenden Drehmoments auf beide Räder der Antriebsachse. Sind unterschiedliche Abtriebsmomente erwünscht, so kann dies über einen Drehmomentverteilungsfaktor parametrierbar werden. Bei einseitigen Variationen am Kontakt zwischen Reifen und Straße wird das übertragbare Drehmoment auf das Doppelte des übertragbaren Moments des Reifens mit dem geringen Reibwert limitiert. Um dies zu verhindern, steht eine Differentialsperre zur Verfügung. Diese kann fix aktiviert sein oder über eine externe Variable parametrierbar werden.

## Modul Rear (Front) Brake

Das Modell der Bremsanlage dient zur Modellierung von Trommel- und Scheibenbremsen. Die Bremse wird entweder über das geforderte Bremsmoment oder den anliegenden Bremsdruck kontrolliert. Über die Dimensionen von Bremszylinder, effektivem Reibradius und des Wirkungsgrades erfolgt im Falle der Kontrolle über den Bremsdruck die Berechnung des Bremsmoments. Der spezifische Bremsfaktor dient zur Unterscheidung zwischen Trommel-(Duplex- und Simplex-Bauweise) und Scheibenbremse.

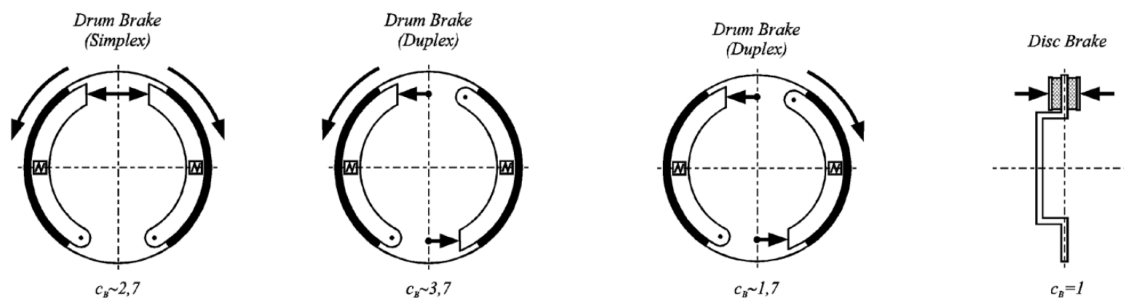


Abbildung 5-2: spezifischer Bremsfaktor verschiedener Bremsbauweisen [15]

Das Bremsmoment wird mittels folgender Gleichung berechnet:

$$M_B = 2 * p_B * A_B * \eta_B * \mu_B * r_B * c_B \quad (5-1)$$

## Modul Tire

Das Reifenmodell errechnet die Fahrzeuggeschwindigkeit über den statischen und dynamischen Rollradius. Der dynamische Rollradius wird dabei als Festwert, oder über eine Kennlinie abhängig von der Fahrzeuggeschwindigkeit parametrisiert. Für die Modellierung des Schlupfes stehen mehrere Methoden zur Verfügung, auf die hier nicht näher eingegangen wird.

Eine Kennlinie stellt den Rollwiderstand als Funktion der Fahrzeuggeschwindigkeit dar. Dabei stehen zusätzliche Parameter zur Verfügung, die den Einfluss der Radlastabhängigkeit, der Reifendruckabhängigkeit und der Temperaturabhängigkeit berücksichtigen. Auf weitere verfügbare Methoden wird hier nicht eingegangen.

Eine andere Möglichkeit zur Vorgabe des Fahrwiderstands erfolgt über die Ausrollkurve im Modul „Vehicle“. Der Rollwiderstand wird dann nicht aus der Parametrierung des Moduls „Tire“ entnommen.

## **Modul Vehicle**

Dieses Modul enthält Basisinformationen über das modellierte Fahrzeug wie Eigengewicht, Gesamtgewicht, Schwerpunktlage und Radstand. Weiters wird hier der gesamte Fahrwiderstand berechnet. Dies geschieht entweder über physikalische Parametrierung der Fahrwiderstandsgleichung (siehe Kapitel 2.2), über eine Funktion, die z.B. mit einem Referenzfahrzeug aufgenommen wird, oder über eine Kennlinie, welche über die Fahrzeuggeschwindigkeit den Fahrwiderstand beinhaltet.

## **Modul Cockpit**

Dieses Modul stellt die nötigen Modelleingangsgrößen wie Fahrerwunschmoment, Bremskraft und Sollgang zur Verfügung. Da das Fahrzeugmodell in dieser Arbeit unter Echtzeit kompiliert wird, wird das Cockpit durch das CRUISE-Interface ersetzt.

## **Modul CRUISE Interface**

Das CRUISE-Interface stellt einen bidirektionalen Link zu MATLAB bzw. einem Simulink-Modell her, d.h. es kann Signale von MATLAB empfangen und an MATLAB senden. Über den „Data Bus“ werden die gewünschten Signale verbunden. Beispielsweise besteht die Möglichkeit zur Vorgabe der Bremspedalstellung von MATLAB an CRUISE oder die Weiterverarbeitung der errechneten Fahrzeuggeschwindigkeit von CRUISE in MATLAB.

### **5.1.2 Abbildung des Fahrzeugkonzeptes in CRUISE**

Zu Erstellung von Fahrzeugmodellen unterschiedlicher Fahrzeugkonzepte und Fahrzeugklassen sind zwei Hauptaufgaben durchzuführen. Zuerst ist mit den notwendigen Modulen der Antriebsstrang nachzubilden. Dazu beinhaltet CRUISE bereits Vorlagen von front-, heck- und allradgetriebenen Fahrzeugen mit manuellem Getriebe und Handschaltgetriebe. Sind diese Vorlagen der zu modellierenden Fahrzeugklasse angepasst, so ist als zweiter Schritt die Bedatung der Module für die zu modellierende Fahrzeugklasse vorzunehmen.

Die Methodik der Modellbildung für den Betrieb am Simulator muss mit möglichst kleinem zeitlichen Aufwand durchgeführt werden können. So sollen die bei einem ungültigen Test gewonnenen Messdaten ausreichen, um daraus das geprüfte Fahrzeug realitätsnah abzubilden, damit ein Training am Simulator möglich wird. Die charakteristischen Parameter, die das gewünschte Verhalten des Fahrzeugmodells bestimmen, müssen bekannt sein.

Die Modellerstellung erfolgt anhand der Messdaten eines Prototyps einer 6-Zylinder-Diesel Mittelklasselimousine mit manuellem 6 Gang Getriebe.

Für die Parametrierung werden folgende Messdaten benötigt:

- Motordrehzahl
- Aktueller Gang
- Fahrzeuggeschwindigkeit
- Pedalwert

Für jeden an AVL-Rollenprüfständen durchgeführten Test wird eine Tabelle mit Zusatzinformationen erstellt, aus der folgende Informationen entnommen sind:

- Fahrzeuggewicht
- Reifendimension
- Parameter der Ausrollkurve

### **Bestimmung des Fahrwiderstandes**

Das Fahrzeuggewicht wird als Parameter im CRUISE-Modul „Vehicle“ gesetzt. Wie in Kapitel 2.2 beschrieben, wird die Ausrollkurve anhand folgender Formel abgebildet:

$$-m * a_x = R_0 + R_1 * v + R_2 * v^2 + F_S \quad (2-2)$$

Da die meisten Tests keine Straßensteigung beinhalten, kann diese weggelassen werden. Falls erforderlich, bietet CRUISE eine eigene Variable zur Simulation der Straßensteigung, diese muss somit nicht im Fahrwiderstand enthalten sein.

$$-m * a_x = R_0 + R_1 * v + R_2 * v^2 \quad (5-2)$$

Die Parameter  $R_0$ ,  $R_1$  und  $R_2$  sind für das verwendete Fahrzeug dokumentiert. Dies ergibt die Ausrollkurve des Fahrzeuges.

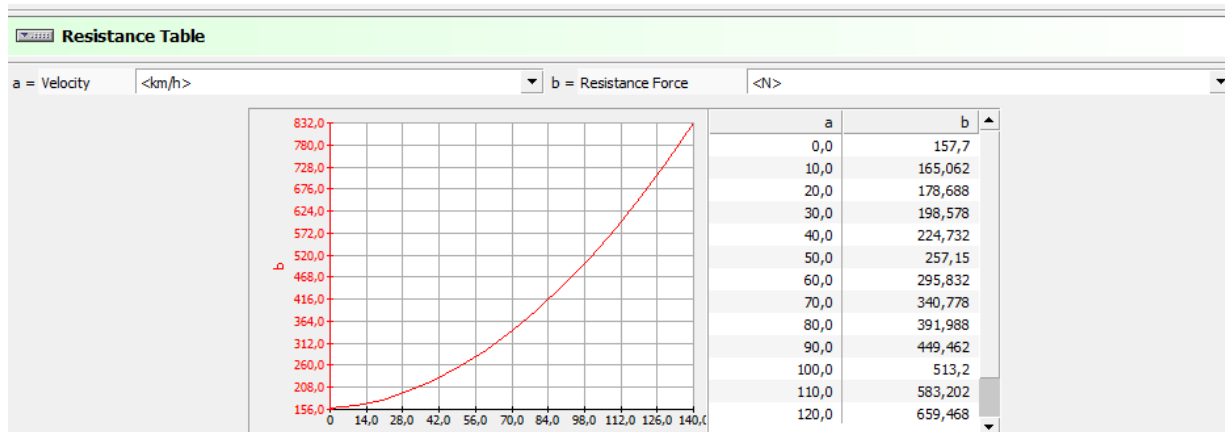


Abbildung 5-3: Ausrollkurve des Fahrzeuges

### Bestimmung der Getriebeübersetzung

Der statische und der dynamische Rollradius ergeben sich aus der Reifendimension. Dies erfolgt über ein im Internet vorhandenes Tool [19]. Der Schlupf wird bei der Modellierung vernachlässigt.

Die Achsübersetzung bei dieser Fahrzeugklasse liegt im Bereich von  $i = 2$  bis  $i = 4$ . Über die Messwerte von Motordrehzahl und Fahrzeuggeschwindigkeit ist nur die Gesamtübersetzung von Achse und Getriebe  $i_H \cdot i_i$  ermittelbar. Aus diesem Grund muss einer dieser Werte angenommen werden. Es wird eine Getriebeübersetzung im 4. Gang von  $i_4 = 1$  gewählt, da PKW-Getriebe häufig eine Getriebestufe mit der Übersetzung  $i = 1$  aufweisen. Daraus errechnet sich eine Achsübersetzung von  $i_H = 3.17$ . Grundsätzlich ist die Aufteilung der Übersetzungen in diesem Anwendungsfall aber nicht ausschlaggebend, solange die Gesamtübersetzung zusammen mit dem Rollradius die richtige Fahrzeuggeschwindigkeit ergibt. Bei genauerer Betrachtung spielt diese Aufteilung eine Rolle, weil die Übersetzungsverteilung zusammen mit den Trägheitsmomenten des Antriebsstrangs die rotatorische Massenträgheit ergibt. Damit hängt die Beschleunigung des Fahrzeuges sowohl von der Fahrzeugmasse als auch von der Trägheit der rotatorischen Massen wie Räder, Wellen, Differential, Getriebe und Motor etc. mit den dazugehörigen Winkelbeschleunigungen ab. Diese Trägheitsmomente werden für die Modellierung auf von CRUISE vorgeschlagene Richtwerte parametrisiert. Folgend wird kurz auf den Einfluss der Trägheitsmomente eingegangen.

Der Motor muss über den Antriebsstrang eine Kraft  $F_A$  am Rad aufbringen, um den Beschleunigungs-, Roll-, Luft- und Steigungswiderstand zu überwinden.

$$F_A = m * a_x + F_R + F_L + F_S \quad (5-3)$$

Zur Berücksichtigung der zu beschleunigenden rotierenden Massen im Antriebsstrang können diese in eine äquivalente translatorisch zu beschleunigende Masse  $m_{rot}$  umgerechnet werden. Das erforderliche Moment zur Beschleunigung der sich drehenden Massen in Bezug auf die Raddrehzahl ergibt sich durch:

$$M_{rot} = I_{Ges} * \dot{\omega}_r \quad (5-4)$$

$I_{Ges}$  beschreibt dabei das auf die Raddrehzahl reduzierte Trägheitsmoment aller rotierenden Massen und wird durch die Trägheitsmomente der jeweiligen Komponente sowie der Achsübersetzung  $i_H$  und der Getriebeübersetzung  $i_i$  ermittelt:

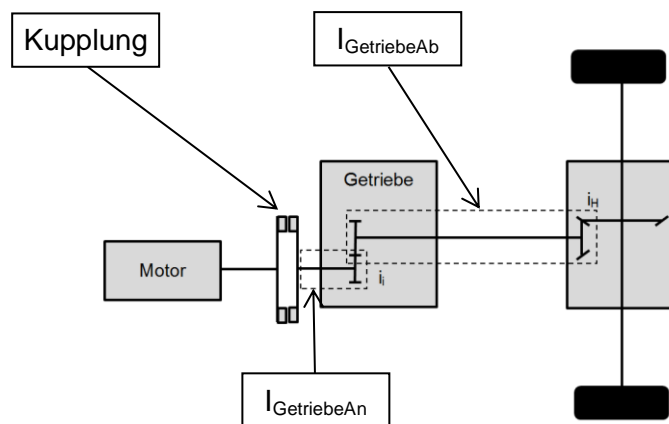


Abbildung 5-4: Skizze eines Antriebsstranges

$$I_{Ges} = (I_{Motor} + I_{Kupplung} + I_{GetriebeAn}) * i_i^2 * i_H^2 + I_{GetriebeAb} * i_H^2 + I_{Achse} + I_{Rad} \quad (5-5)$$

Die Leistung der zu beschleunigenden rotatorischen Massen ergibt sich durch:

$$P_{rot} = M_{rot} * \omega_r \quad (5-6)$$

Diese wird mit einer äquivalenten translatorisch zu beschleunigenden Masse gleichgesetzt:

$$P_{rot} = I_{Ges} * \dot{\omega}_r * \omega_r = m_{rot} * a_x * v_{FZ} \quad (5-7)$$

Mit

$$\dot{\omega}_r = \frac{a_x}{r_{dyn}}$$

und (5-8)

$$\omega_r = \frac{v_{FZ}}{r_{dyn}} \quad (5-9)$$

ergibt sich

$$m_{rot} = \frac{I_{Ges}}{r_{dyn}^2} \quad (5-10)$$

Diese translatorische Ersatzmasse kann nun der Fahrzeugmasse hinzugezählt werden. Somit sind die rotierenden Massen des Antriebsstrangs berücksichtigt.

$$F_A = (m + m_{rot}) * a_x + F_R + F_L + F_S \quad (5-11)$$

Roll- und Luftwiderstand können nun aus dem in Kapitel 2.2 beschriebenen Polynom, physikalischen Daten oder mittels Kennlinien geschwindigkeitsabhängig in CRUISE parametrisiert werden. Durch Multiplikation von Gleichung (5-11) mit der Fahrzeuggeschwindigkeit ergibt sich die Motorleistung. Aus der momentan anliegenden Motorleistung kann daraus nun die Fahrzeugbeschleunigung bestimmt werden.

Die Raddrehzahl kann nun aus der Winkelgeschwindigkeit des Rades ermittelt werden:

$$n_r = \frac{\omega_r}{2 * \pi} \quad (5-12)$$

$$n_r = \frac{v_{FZ}}{2 * \pi * r_{dyn}} \quad (5-13)$$

Dadurch kann mit der gewählten Achsübersetzung die Getriebeausgangsdrehzahl berechnet werden.

$$i_H = \frac{n_{Go}}{n_r} \quad (5-14)$$

$$n_{Go} = i_H * n_r \quad (5-15)$$

Damit erhält man für die Getriebeausgangsdrehzahl:

$$n_{Go} = i_H * \frac{v_{FZ}}{2 * \pi * r_{dyn}}. \quad (5-16)$$

Durch das Verhältnis von Eingangs- (Motor) zu Ausgangsdrehzahl von Getriebe lässt sich die Getriebeübersetzung ermitteln.

$$i_i = \frac{n_{mot}}{n_{Go}} \quad (5-17)$$

Zusammengefasst ergibt sich für die Übersetzungen der Getriebestufen  $i_i$ :

$$i_i = \frac{2 * \pi * r_{dyn} * n_{mot}}{i_H * v_{FZ}} \quad (5-18)$$

Als nächster Schritt ist nun die Auswertung der Messdaten notwendig. Diese stammen aus einem NEDC-Abgastest. Für alle Gänge ist die Auswertung von Fahrzeuggeschwindigkeit und Motordrehzahl notwendig. Es ist darauf zu achten, dass sich bei jedem Messpunkt das Fahrzeug im vollständig eingekuppelten Zustand befindet, damit kein Kupplungsrutschen und daraus folgend keine Differenz zwischen Motordrehzahl und Getriebeeingangsdrehzahl auftritt. Weiters ist darauf zu achten, dass alle Punkte, soweit möglich, bei stationären Bedingungen, also in Bereichen mit Konstantfahrt, ausgewertet werden. An diesen Stellen ist mit den kleinsten Messfehlern zu rechnen, da eine möglicherweise kleine Verschiebung der zeitlichen Zuordnung der beiden Signale einen kleineren Fehler zur Folge hat.



Abbildung 5-5 zeigt die Positionen der Messpunkte von Gang 1–6.

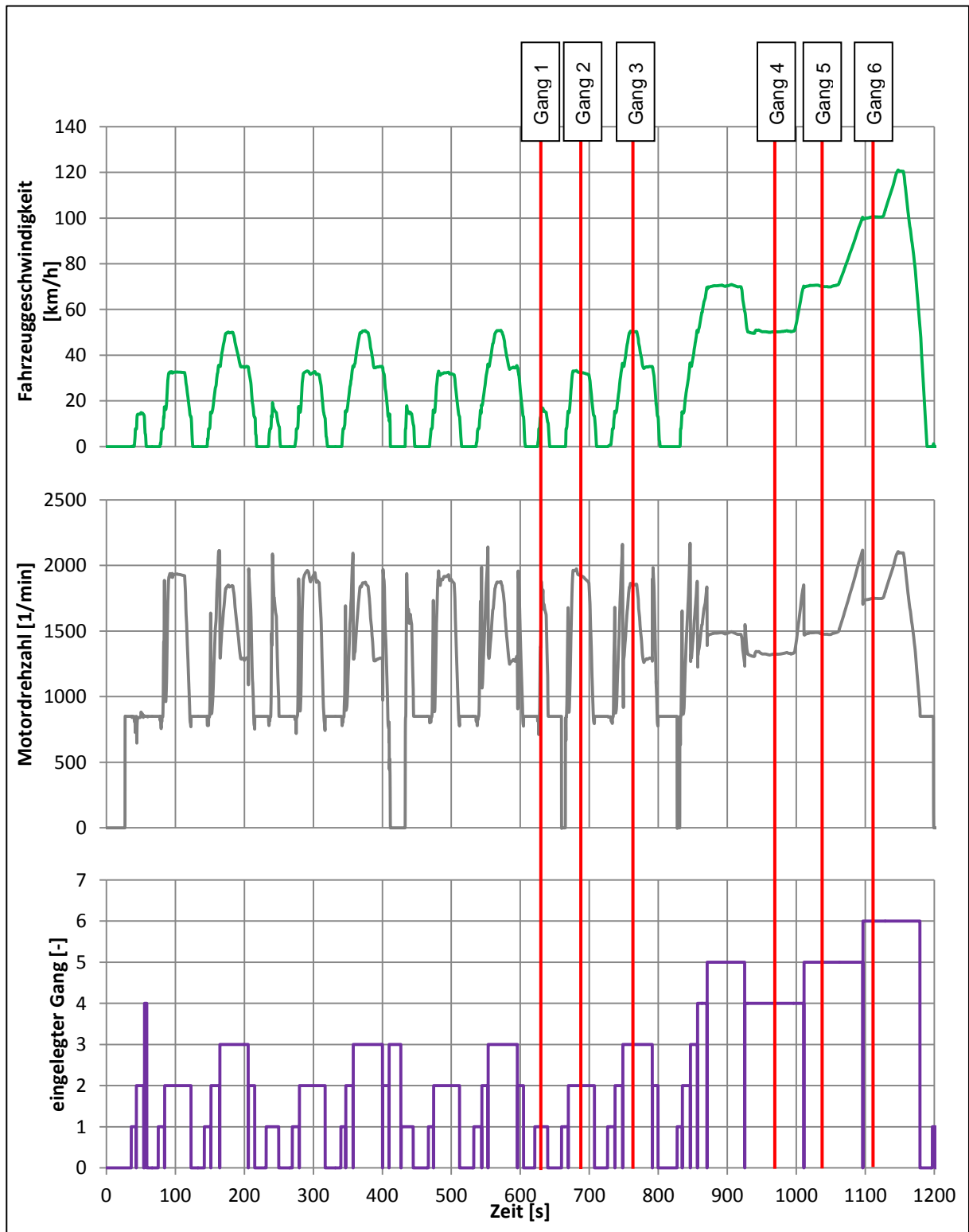


Abbildung 5-5: Zuordnung der Messpunkte zur Ermittlung der Getriebeübersetzung

Die errechneten Übersetzungsverhältnisse werden in CRUISE übertragen. Dort werden sie eventuell geringfügig angepasst, um ganzzahlige Werte für die Zähnezahlen der Zahnradpaare im Getriebe zu ermöglichen. Zusammen mit der gewählten Achsübersetzung ergeben die errechneten Werte die gleiche Gesamtübersetzung des realen Fahrzeuges. Die Übersetzungsverhältnisse der Gänge 1–6 lauten wie folgt:

Gear Ratio Table						
a = Gear	<->	b = Transmission Ratio	<->			
c = Inertia Moment In	<kg*m^2>	d = Inertia Moment Out	<kg*m^2>			
e = Number of Teeth Input	<->	f = Number of Teeth Output	<->			
	a	b	c	d	e	f
0	1,0	0,0015	0,005	10	10	
1	4,11111	0,0015	0,005	18	74	
2	2,25	0,0015	0,005	12	27	
3	1,4	0,0015	0,005	10	14	
4	1,0	0,0015	0,005	10	10	
5	0,8	0,0015	0,005	15	12	
6	0,66	0,0015	0,005	50	33	

Abbildung 5-6: Parametrierung des Übersetzungsverhältnisses in CRUISE

### Ermittlung der Volllastkurve

Da die Volllastkurve nicht aus dem NEDC-Abgastest ermittelt werden kann, weil das Fahrzeug dabei nur unter Teillast betrieben wird, stammt diese aus dem Herstellerprospekt der Vorgängerversion. Das vermessene Fahrzeug bietet eine maximale Leistung von 195 kW, das Vorgängermodell jedoch nur 180 kW. Die Korrektur der Volllastkurve erfolgt durch Multiplikation mit einem Faktor.

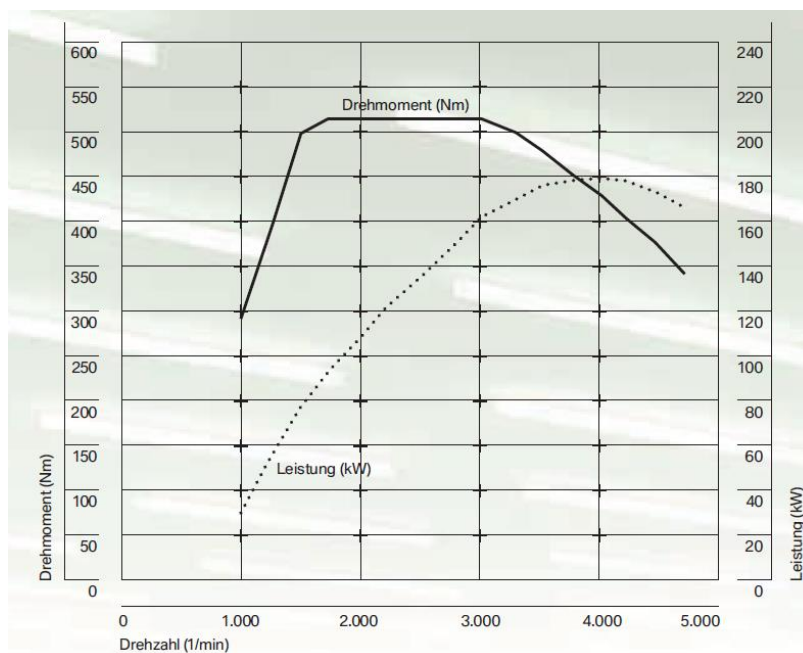


Abbildung 5-7: Volllastkurve des parametrierten Fahrzeuges

## Parametrierung der Bremse

Durch Definition der Dimension des Bremszylinders, des effektiven Reibradius und des Wirkungsgrades erfolgt die Berechnung des Bremsmomentes in Abhängigkeit vom Bremsdruck. Die von CRUISE vorgeschlagenen Dimensionen werden übernommen. Die Parametrierung der Bremse erfolgt über einen Faktor in MATLAB/Simulink.

## Schnittstelle zu MATLAB/Simulink

Nach Parametrierung der Vollastkurve, des dynamischen Reifendurchmessers sowie des Getriebes ist als abschließender Schritt die Integration der Schnittstelle zu MATLAB/Simulink notwendig. Das Fahrzeugmodell wird in CRUISE kompiliert und danach in MATLAB/Simulink importiert. In CRUISE kann mithilfe des Moduls „Cruise Interface“ (siehe Abbildung 5-1) vor dem Kompilervorgang definiert werden, welche Signale von Simulink beschrieben werden bzw. welche Signale von CRUISE an Simulink geschickt werden. Abbildung 5-8 zeigt den Signalfluss zwischen CRUISE, Simulink und PUMA Open. Signale zur Verifikation der korrekten Berechnung sind nicht dargestellt.

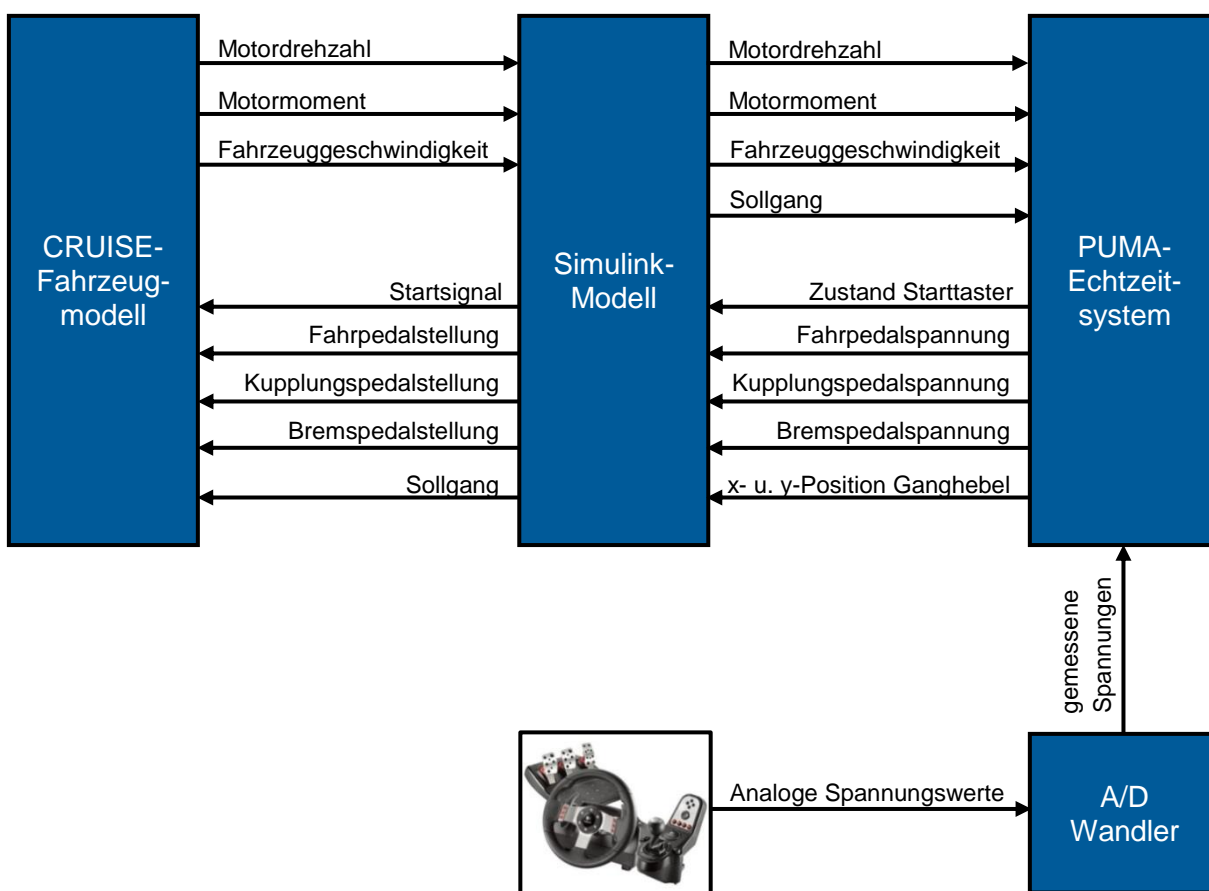


Abbildung 5-8: Signalfluss in der Fahrzeugmodellierung

## Kompilervorgang des CRUISE-Modells

Um das Modell in Echtzeit auf der Plattform PUMA Open zu berechnen, sind zwei Kompilierungsschritte notwendig. Zuerst wird in CRUISE das Modell so kompiliert, dass es in MATLAB/Simulink importiert werden kann. MATLAB/Simulink bietet eine Schnittstelle zu PUMA Open. Somit wird über diesen Zwischenschritt eine Schnittstelle zwischen CRUISE und PUMA Open geschaffen.

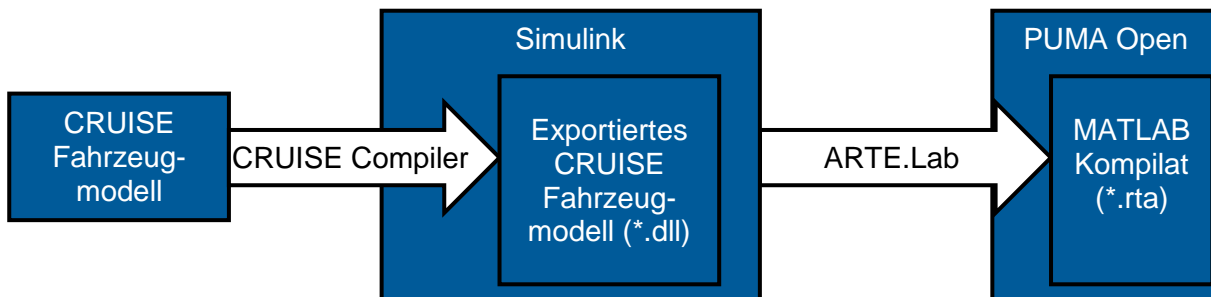


Abbildung 5-9: Kompilierreihenfolge des Fahrzeugmodells

In dem CRUISE-Menüpunkt „Calculation Center“ erfolgt jede in CRUISE verfügbare Simulationsmethode. Wird das CRUISE-Modell nicht für Echtzeitanwendungen verwendet, so ist im „Calculation Center“ der Ablauf der Simulation tabellarisch zu definieren. Damit ist für die Simulation der zeitliche Ablauf von Gangwechsel, Kupplungsverhalten, Fahrpedalwert etc. vorgegeben. Dies erbringt die Möglichkeit einer Validierung des Fahrzeugmodells vorab des Echtzeitbetriebs in CRUISE. Da wie in Kapitel 5.2.1 beschrieben in Simulink einige Funktionalitäten hinzugefügt sind, wird auch die Modellvalidierung in MATLAB/Simulink durchgeführt. Dazu wird das Fahrzeugmodell in CRUISE kompiliert und in MATLAB importiert.

Die dazu im Echtzeitbetrieb nötige fixe Zeitschrittweite erfordert die Simulation in CRUISE als „SAM“-Task. Dort wird die Zeitschrittweite der Berechnung und der zu verwendende Solver festgelegt. „Infinite“ bedeutet eine Berechnung des Modells ohne Stopzeitpunkt.

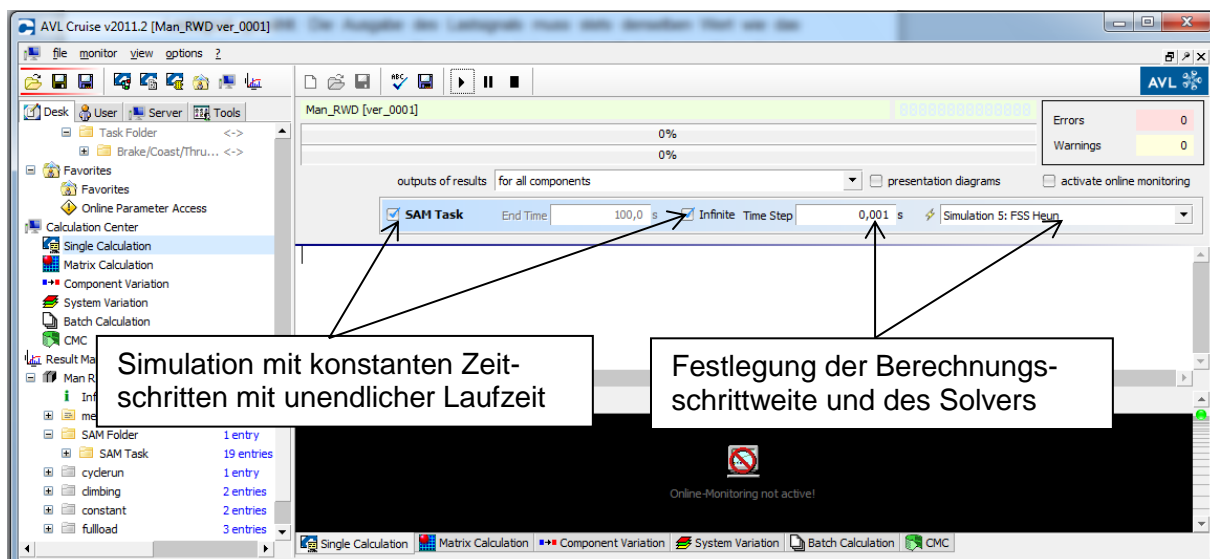


Abbildung 5-10: Kompilervorgang des Fahrzeugmodells in CRUISE

Über die „CMC-Calculation“ (CMC = Common Model Compiler) ist es nun möglich, das Modell zu kompilieren, um es in MATLAB/Simulink oder auch für andere Plattformen verfügbar zu machen. Als Plattform für das „PUMA Open“-Automatisierungssystem wird INTime 3.1 gewählt. Für den Echtzeitkernel (siehe Kapitel 3.2.2) steht somit ein echtzeitfähiger Programmcode zur Verfügung. Für die Simulation in MATLAB/Simulink erstellt CRUISE eine in Simulink ausführbare \*.dll-Datei (dynamik link library).

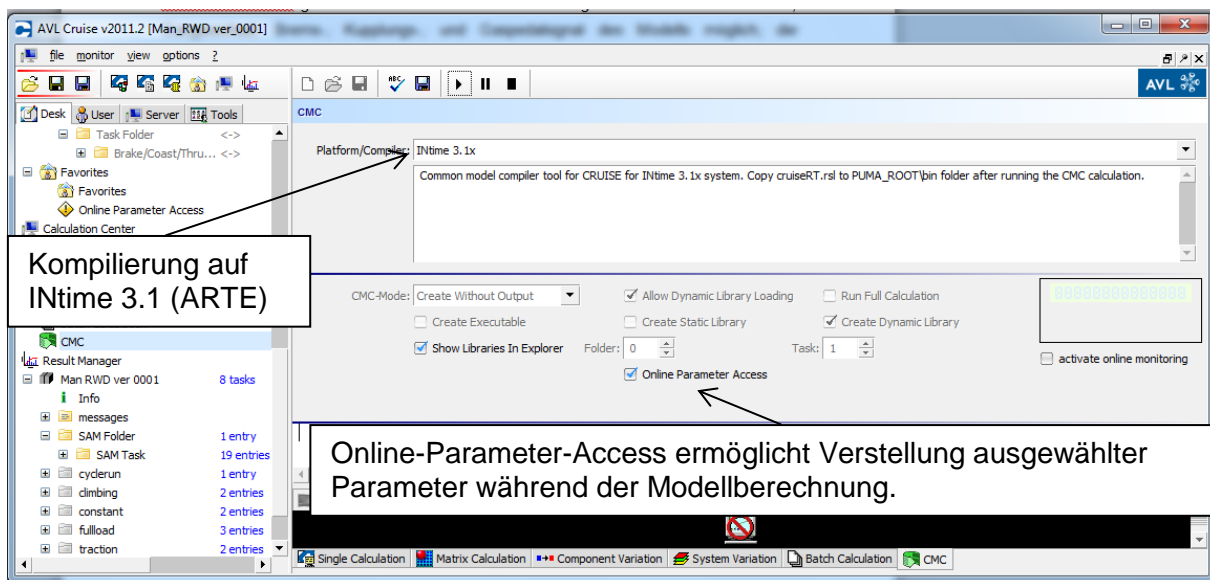


Abbildung 5-11: Festlegung des Zielsystems und des Online-Parameter-Zugriffs

Skalare Parameter werden in CRUISE in die Online-Parameter-Datenbank eingefügt. Bei Auswahl des „Online-Parameter-Access“ sind die Parameter im Simulink-Modell veränderbar. Die erhaltene MATLAB-Library steht somit zur Erweiterung und Simulation in MATLAB/Simulink bereit. Der Simulink-Block des CRUISE-Modells verfügt jeweils über zwei

Ein- und Ausgänge. Diese stellen den jeweiligen Datenbus für Inputs, Parameter, Outputs und Ausgabe der Parameter dar.

## 5.2 Echtzeitberechnung in MATLAB/Simulink

Die Kombination von MATLAB und AVL ARTE.Lab Studio ermöglicht es, sämtliche Simulink Modelle in Echtzeit lauffähig zu machen. In Simulink können Normnamen bzw. Systemnamen von der „PUMA Open“-Automatisierungsplattform gelesen und beschrieben werden. Auf diese Weise erfolgt die Weiterverarbeitung der gemessenen Spannungen von Pedalerie und Gangschaltung im Simulink-Modell.

In Simulink müssen folgende Aufgaben zusätzlich zum Cruise-Modell ergänzt werden:

- Umrechnung aller gemessenen Spannungen an den Pedalen in einen Pedalwert
- Umrechnung der gemessenen Drehwinkel des Schalthebels auf den jeweiligen Gang
- Umrechnung einer Spannung in ein Startsignal des Motors
- Motorabstellung bei Überschreitung des maximal möglichen Leerlaufmoments
- Parameterzugriff bei laufender Echtzeitberechnung
- Umrechnung der Ausgangssignale auf die vom Fahrerleitgerät geforderten Signale

Diese Anforderungen werden mit Simulink-Funktionsblöcken umgesetzt. Die Definition der zu verstellenden Parameter erfolgt so, dass ein erneuter Kompilervorgang nicht nötig ist. Dazu wird der Model-Parameter-Editor (MPE) von ARTE.Lab verwendet. Dieser listet alle verstellbaren Parameter des Simulink-Modells tabellarisch auf und ermöglicht dessen Änderung zur Laufzeit des Modells.

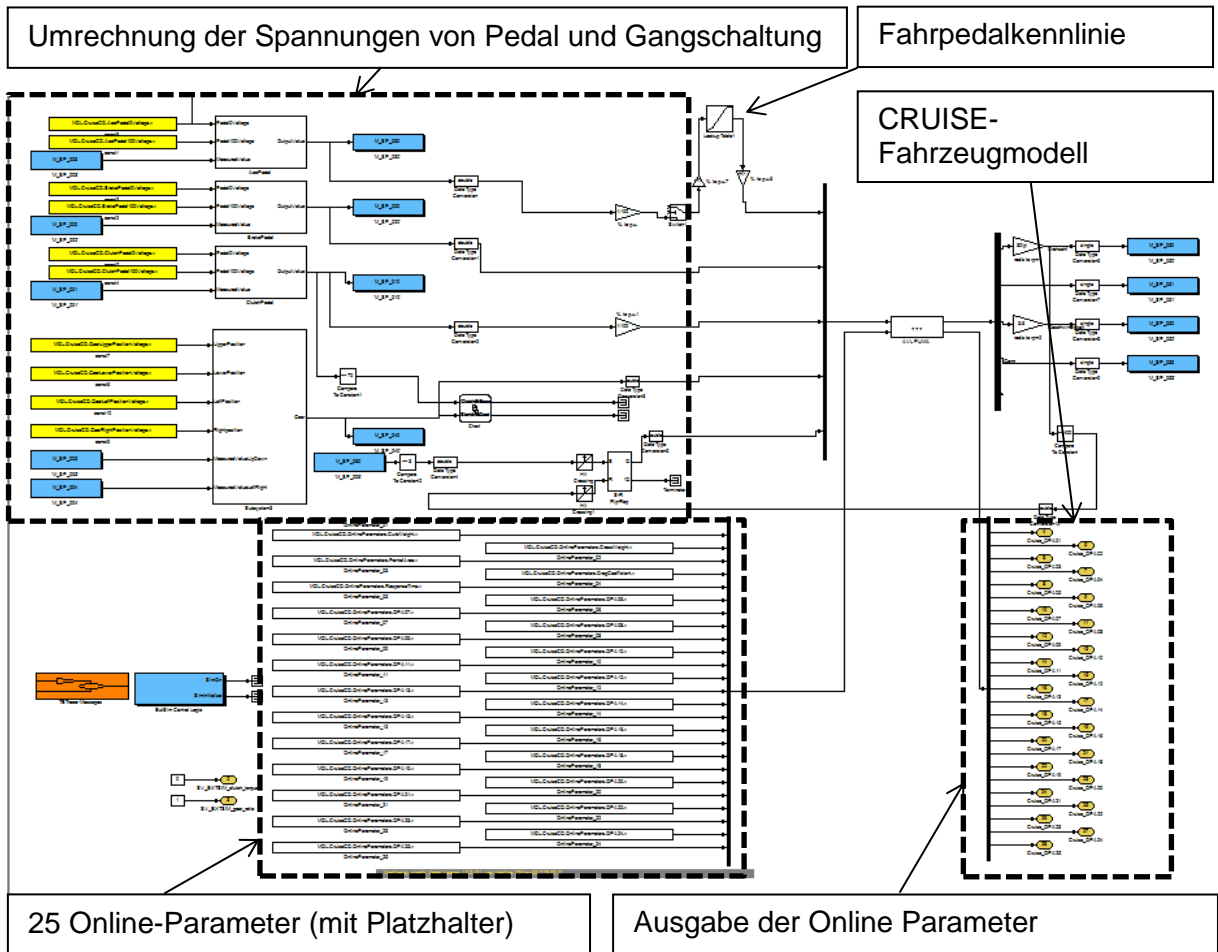


Abbildung 5-12: Aufbau des Simulink-Modells

Das Simulink-Modell des Gesamtsystems in Abbildung 5-12 zeigt die Subsysteme der Pedalwertberechnung, der Ein- und Ausgänge des „PUMA Open“-Systems, das CRUISE-Modell, die Ein- und Ausschaltautomatik und die in Kapitel 6.3 beschriebene Fahrpedalkennlinie. Da bei aktivierten Online Parametern von CRUISE immer die maximal mögliche Anzahl an Parametern vorhanden sein muss, sind für nicht verwendete Parameter Platzhalter einzufügen. Alle Werte werden aus dem CRUISE-Block wieder ausgegeben, um die korrekte Funktion der Online-Parametrierung zu überprüfen.

### 5.2.1 Signalkonditionierung

Der Analog/Digital-Wandler wertet die Spannungen von Pedalerie und Gangschaltung aus. Im Echtzeitsystem erfolgt dann die Berechnung auf die gewünschte Größe. Die Signale werden in Pedalwerte von 0–100 % und auf die Gänge 1–6 umgerechnet.

#### Signale der Pedalerie

Der Verlauf der Pedalspannungen ist bei der gewählten Pedalerie indirekt proportional zum Pedalweg. Bei Verwendung anderer Pedalerie oder von Eigenkonstruktionen kann dies aber auch direkt proportional sein, was bei der Integration neuer Pedale zu berücksichtigen ist. Aus diesem Grund wird in Simulink die Interpolation mit Funktionsblöcken durchgeführt.

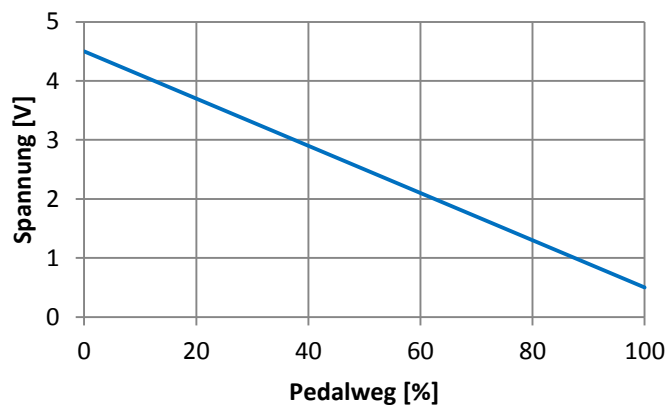


Abbildung 5-13: Spannungsverlauf der Pedalpotentiometer

Die einfachste Methode zur Umrechnung der Spannung in einen Pedalweg wäre die Verwendung des Simulink-Blocks „Lookup Table“. Hier ist die Bedatung einer Kennlinie mit mehreren Stützstellen mit Begrenzung auf bestimmte Limits möglich, um zugleich eine Begrenzung auf den 0 %- und 100 %-Pedalwert zu erhalten. Die Parametrierung des Lookup Tables kann auch nach dem Kompilervorgang vorgenommen werden, um ohne Anpassung des Simulink Modells eine neue Pedalerie oder Gangschaltung einzusetzen.

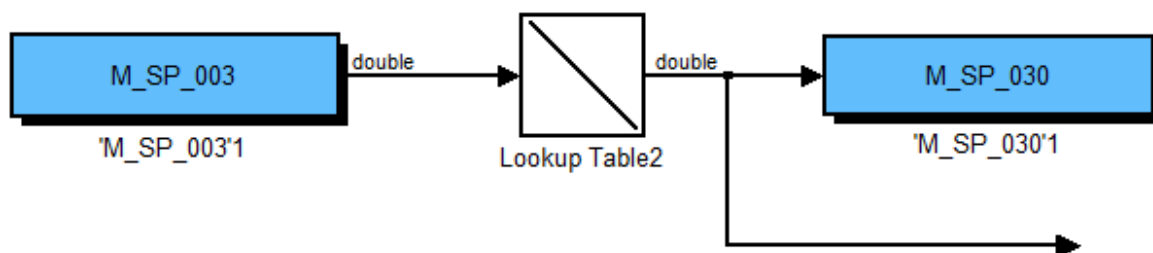


Abbildung 5-14: Beispiel eines „Lookup Tables“ in Simulink



Der Nachteil dieser Methode besteht lediglich in der unübersichtlichen Bedienung im Echtzeitsystem. Der Model-Parameter-Editor stellt den Lookup Table mit zwei Zeilenvektoren dar, deren Namensgebung ohne zusätzliche Anleitung nicht eindeutig ist. Aus diesem Grund wird die Berechnung auf Basis einer Geradengleichung in Simulink umgesetzt. Zur späteren Parametrierung sind folglich nur zwei Parameter mit eindeutigen Namen zu ändern. Dieselbe Routine kommt bei allen drei Pedalsignalen zum Einsatz. Dazu wird das zur Interpolation des Pedalwertes angefertigte Subsystem dreimal wiederverwendet und somit die Übersichtlichkeit des Gesamtmodells erhöht.

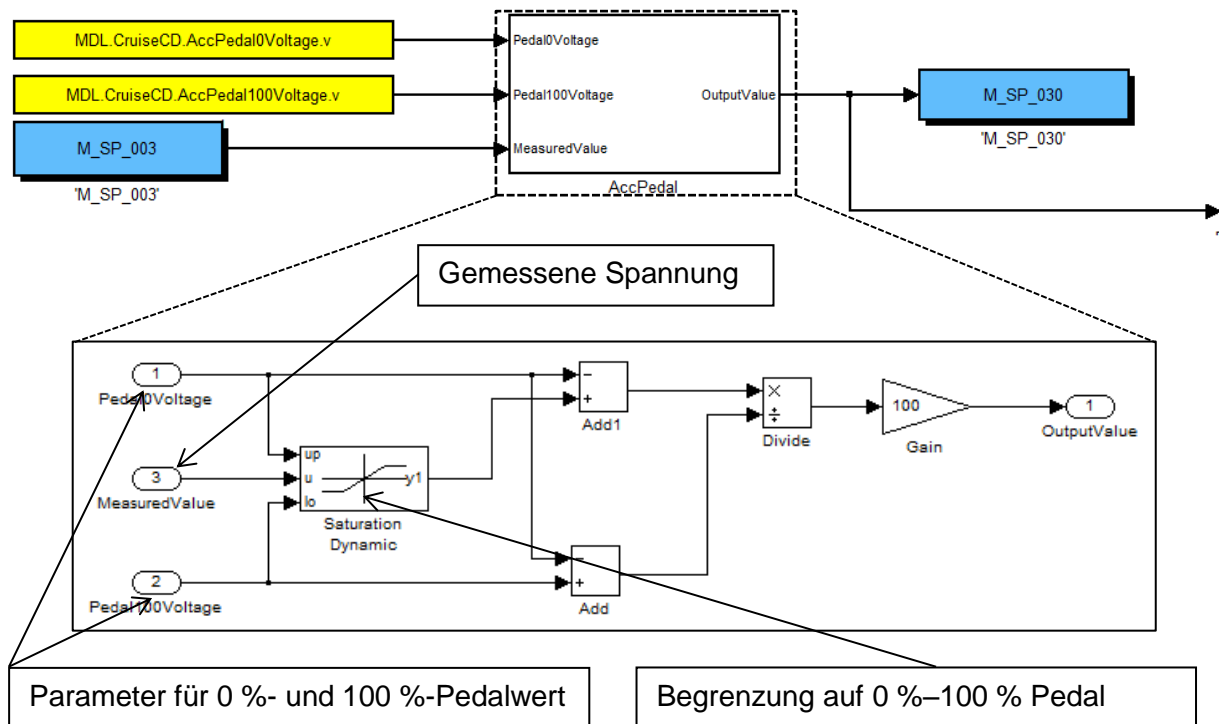


Abbildung 5-15: Berechnung der Pedalwerte in Simulink

## Signale der Gangschaltung

Die Gangschaltung ist mit zwei Potentiometern versehen, die den Drehwinkel um die Drehachsen des Schalthebels messen. Als erster Schritt müssen Bereiche für die Gänge 1–6 ermittelt werden.

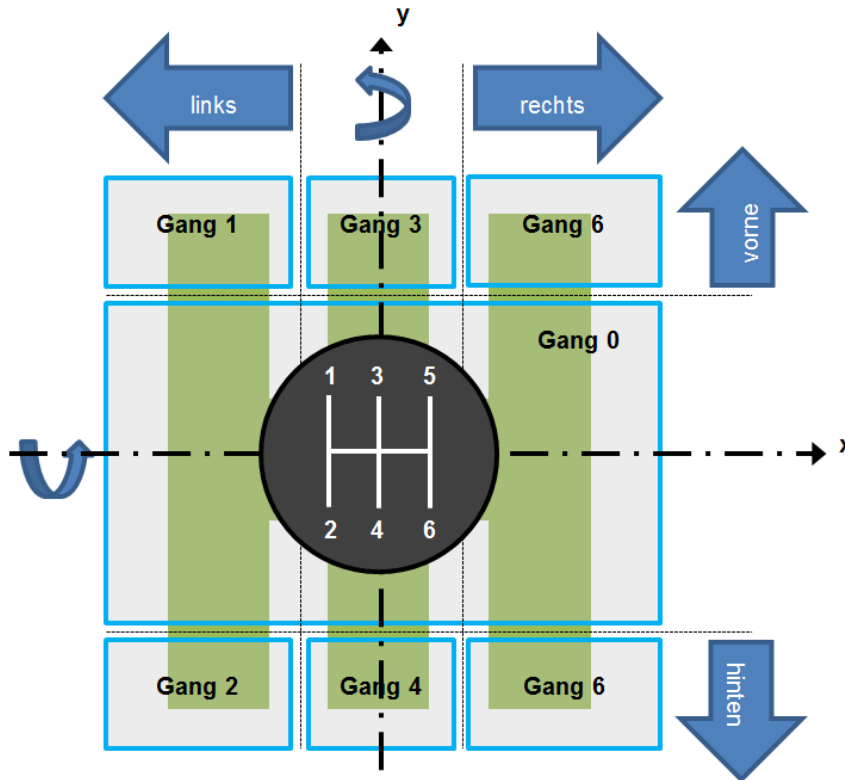


Abbildung 5-16: Koordinaten der Gänge 1–6

Die Pfeile beschreiben die Einteilung der Bereiche für die Gangschaltung. Für die Berechnung des gewählten Ganges sind nun insgesamt vier Parameter erforderlich. Befindet sich die Gangschaltung in vertikaler Richtung zwischen den Markierungen „vorne“ und „hinten“, so wird das als „Neutralgang“ interpretiert. Zusammen mit den Informationen der vertikalen Position wird über die horizontale Richtung die Position „links“, „rechts“ oder „Mitte“ erkannt und der eingelegte Gang eindeutig bestimmt.

Als nächster Schritt wird ein Modell zur Auswertung der Pedalsignale erstellt.

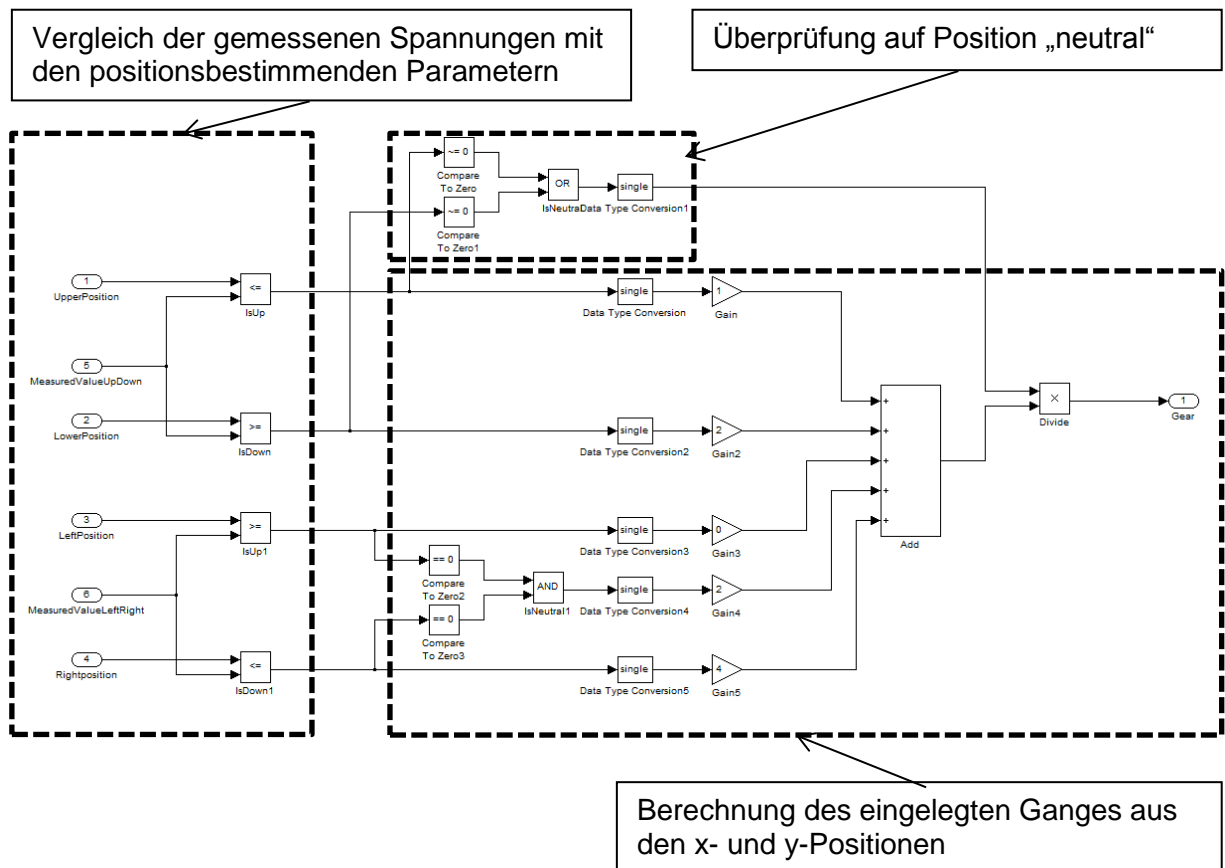


Abbildung 5-17: Berechnung des eingelegten Ganges

Der gemessene Wert für die horizontale Position wird nun mit den gewählten Parametern verglichen, dabei beschreiben folgende Werte den Zustand der horizontalen Position:

- Links = 0
- Mitte = 1
- Rechts = 2

Der gemessene Wert für die vertikale Position wird ebenfalls mit den Parametern verglichen, die die Schwellwerte der Positionen „vorne“ und „hinten“ beschreiben. Der Zustand für die Vertikalposition wird folgendermaßen interpretiert:

- Vorne = 1
- Hinten = 2

Abschließend erfolgt die Addition der Variablen für Horizontal- und Vertikalposition. Als Ergebnis erhält man den eingelegten Gang. Befindet sich der Ganghebel in vertikaler Richtung in mittlerer Position, so wird der zuvor errechnete Wert mit „0“ multipliziert, andernfalls mit „1“. Dies ergibt die korrekte Interpretation des „Neutralgangs“.

### Starttaster

Der Zustand des Starttasters kann über einen digitalen Eingang gemessen werden. Da der Simulator für Pedalerie und Gangschaltung nur analoge Eingänge benötigt, wird der Starttaster ebenfalls mit einem analogen Eingang ausgewertet. Als Stromquelle dient ein USB-Port des PUMA-PCs. Der Taster schließt bei Betätigung diesen Stromkreis und das Simulink-Modell erkennt über einen definierten Schwellwert der Eingangsspannung den gewünschten Motorstart.

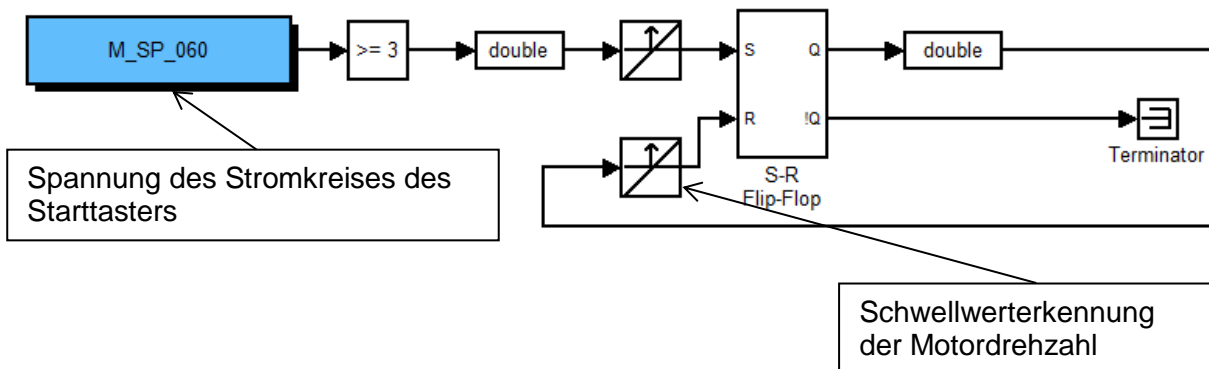


Abbildung 5-18: Simulink-Funktion für Motorstart und -stopp

Bei Erreichen des Schwellwertes von 3 V wird dieser über eine Flankenerkennung an ein Flipflop weitergegeben, das das Bit für den Motorstart setzt. Der Reset-Eingang des Flipflops kommt aus einer Schwellwertüberprüfung der Motordrehzahl. Das CRUISE-Modell verfügt über einen Leerlaufregler, der die Drehzahl im Leerlauf je nach Größe des anliegenden Moments verringert. Über eine Schwellwernerkenkung wird bei zu geringer Drehzahl der „Motor gestartet“-Befehl des Modells abgeschaltet, was einem „Abwürgen des Motors“ entspricht.

### 5.2.2 Kompiliervorgang in MATLAB/Simulink

Das Simulink-Modell muss für das Echtzeitsystem kompiliert werden. Dazu kommt das ARTE.Lab Studio zum Einsatz, welches nach erfolgter Installation direkt in MATLAB aufrufbar ist. Zunächst erfolgt die Vorbereitung des Simulink-Modells für den Kompiliervorgang. ARTE.Lab bietet dazu einen Befehl, der einen neuen Ordner mit Beispielmmodell und allen nötigen Dateien erstellt. Das Simulink-Modell wird in diese Vorlage eingefügt. Als nächster Schritt wird der Solver auf dieselbe Berechnungsfrequenz eingestellt, die auch für CRUISE gewählt ist. Danach müssen alle zur Laufzeit änderbaren Parameter in einem Textfile eingetragen werden. Eine vorgeschriebene Namenskonvention versieht dort alle Parameter mit ihrem Startwert, dem zulässigen Wertebereich, der Anzahl der Dezimalstellen und der Schrittweite der Verstellung im Parameter-Editor.

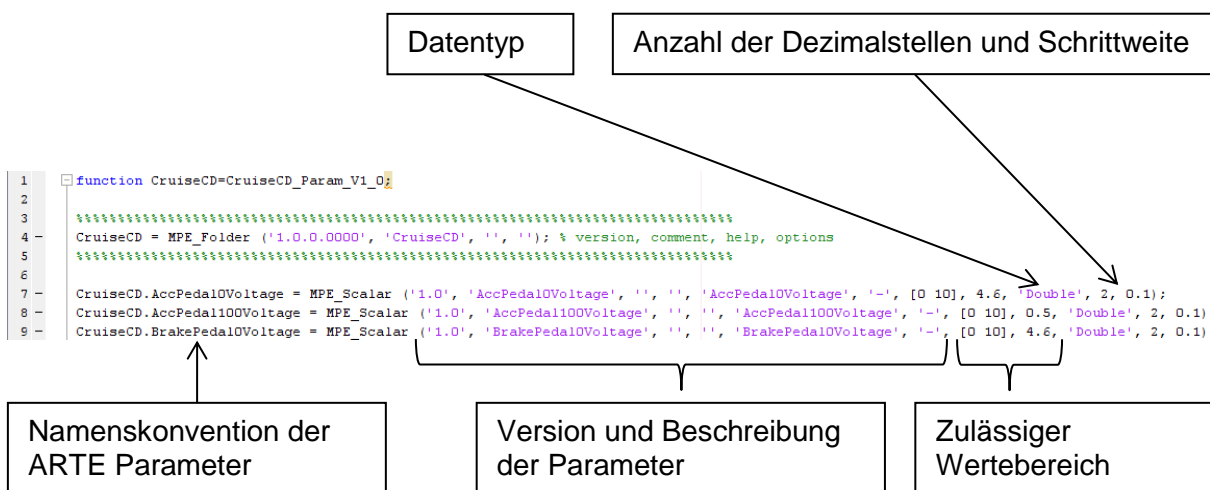


Abbildung 5-19: Parameterdefinition in Simulink

Nach der Fertigstellung wird das Modell mit dem ARTE.Lab-Befehl „startup“ geöffnet und über den Befehl „build“ der Kompiliervorgang gestartet. Das Ergebnis dieses Vorganges ist eine „\*.rta“-Datei. Dieses File enthält den Programmcode des Fahrzeugmodells. Der Ordner „Deployment“ enthält zusätzlich zu einer Kopie der „\*.rta“-Datei noch die Datei „setup.exe“. Bei Aufruf dieser Datei am PUMA werden automatisiert alle nötigen Änderungen an der Startkonfiguration von PUMA Open durchgeführt. Beim nächsten Start von PUMA Open ist das Modell bereits lauffähig.

## 6 Experimentelle Untersuchungen

Anhand der Messdaten des bereits zur Parametrierung des CRUISE-Modells verwendeten NEDC-Abgastests wird das Fahrzeugmodell mit dem realen Fahrzeug abgeglichen. Hierbei ist eine genaue Übereinstimmung der Geschwindigkeiten von Modell und den Messdaten des realen Fahrzeuges nicht notwendig, sofern sich das Fahrzeugmodell hinsichtlich Drehmomentenaufbau und Reaktion auf die Eingabe von Kupplung, Bremse und Fahrpedal qualitativ gleich verhält. Dies gewährleistet eine effektive Nutzung der aus dem Training am Simulator gewonnenen Erfahrungen bei Tests am Rollenprüfstand.

### 6.1 Vorgehensweisen zum Modellabgleich mit vorhandenen Messdaten

Der Vergleich der Modelldaten mit den Messdaten kann über mehrere Vorgehensweisen durchgeführt werden. Da die Messdaten den Fahrpedalwert des Fahrzeuges beinhalten, wird dieser als Eingang des exportierten CRUISE-Fahrzeugmodelles in MATLAB/Simulink verwendet. Das Ergebnis ist ein Geschwindigkeitsverlauf, der mit der gemessenen Fahrzeuggeschwindigkeit verglichen wird. Alternativ kann ein Regler mit dem Fahrpedalwert als Ausgangsgröße und der Sollgeschwindigkeit als Eingangsgröße, also in diesem Fall die gemessene Geschwindigkeit aus den Messdaten, verwendet werden. Nach Abschluss der Simulation kann der aus dem Regler stammende Fahrpedalwert mit dem gemessenen verglichen werden. Diese Methode ist für Konstantfahrten geeignet, da das Erreichen der gleichen Geschwindigkeit von Modell und realem Fahrzeug bei gleichem Fahrpedalwert überprüft wird. Bei Beschleunigungsteilen tritt aber das Problem auf, das aus unterschiedlicher Parametrierung des Reglers unterschiedliche Verläufe des Pedalwerts resultieren. Zur Verdeutlichung dieses Problems kann angenommen werden, dass ein konstanter Pedalwert die gleiche Beschleunigung erwirken kann wie ein pulsierender Pedalwert, der denselben Mittelwert hat. Die Vergleichbarkeit mit dem gemessenen Pedalwert ist daraus folgend nicht möglich. Aus diesem Grund kommt die Methodik der Pedalwertvorgabe mit anschließendem Vergleich von simulierter und gemessener Geschwindigkeit zum Einsatz.

Der in Kapitel 5 beschriebene Vorgang zur Modellerstellung ermöglicht eine Validierung des Modells auf unterschiedliche Weisen. Es ist möglich, die Berechnung des Modells direkt in CRUISE zu starten und dabei alle Eingangsgrößen in CRUISE vorzugeben. Eine andere Möglichkeit besteht darin, das in MATLAB/Simulink exportierte Modell zu verwenden und dort zu simulieren. Die Eingänge können dort filebasierend in MATLAB vorgegeben werden. Die dritte Möglichkeit ist, bereits das in Echtzeit gerechnete Modell zu verwenden und den Sollwert über ein in der AVL entwickeltes Pedalbediengerät (AVL DriCon) vorzugeben. Das

hat jedoch den Nachteil, dass die Echtzeitsimulation einen höheren Zeitaufwand mit sich bringt. Die Sollwertvorgabe in MATLAB ist mit nur geringen Änderungen am Simulink-Modell möglich. Auch die im Simulink-Modell ergänzten Funktionen, wie Abschaltung bei zu hoher Last im Leerlauf, werden getestet. Somit stellt die Simulation in MATLAB die beste Möglichkeit dar und kommt bei der Abstimmung des Fahrzeugmodells zum Einsatz.

## 6.2 Erstellung des Simulationsaufbaus

Zur Validierung des Fahrzeugmodells wird der Fahrpedalwertverlauf des realen Fahrzeuges bei einem NEDC-Abgastest verwendet. Zuerst erfolgt die Konvertierung der Messdaten in das textbasierende Dateiformat „\*.csv“ zur Weiterverarbeitung in Microsoft Excel. Um das Fahrzeugmodell auf gleiche Weise wie unter Echtzeit am Simulator zu steuern, sind zusätzlich zu der Fahrpedalstellung noch Kupplungs- und Bremspedalstellung, einzulegender Gang und das Startsignal vorzugeben. Zur Vereinfachung des Vergleichsprozesses werden dem Vorgabefile die Messdaten der Fahrzeuggeschwindigkeit und eine Zeitspur hinzugefügt. MATLAB bietet die Möglichkeit, die Daten dieses Files in MATLAB zu importieren und als \*.m-File abzuspeichern. Mittels des Funktionsblocks („From File“) werden nun in Simulink die Sollwerte an das Modell übergeben. Dieses Vorgabefile besitzt eine zeitliche Auflösung von 10 Hz, die Simulation rechnet das Modell jedoch mit Rechenschrittweiten von 1 ms. Aus diesem Grund bietet der Funktionsblock „From File“ eine Interpolation, sodass die Sollwerte in einer Frequenz von ebenfalls 1 kHz verfügbar sind. Eine Umschaltung der Datenquelle bestimmt, ob die Sollwerte von dem Vorgabefile oder den Eingängen des „PUMA Open“-Systems (bei Echtzeitbetrieb) stammen. Parallel zur Simulation in MATLAB erfolgt die Darstellung der Ergebnisse in einem Diagrammfenster. Gleichzeitig wird es als Array in den Arbeitsspeicher abgelegt. Ein MATLAB Skript vereinfacht diese Daten nach Abschluss der Berechnung auf eine Frequenz von 10 Hz und speichert sie als Textfile ab.

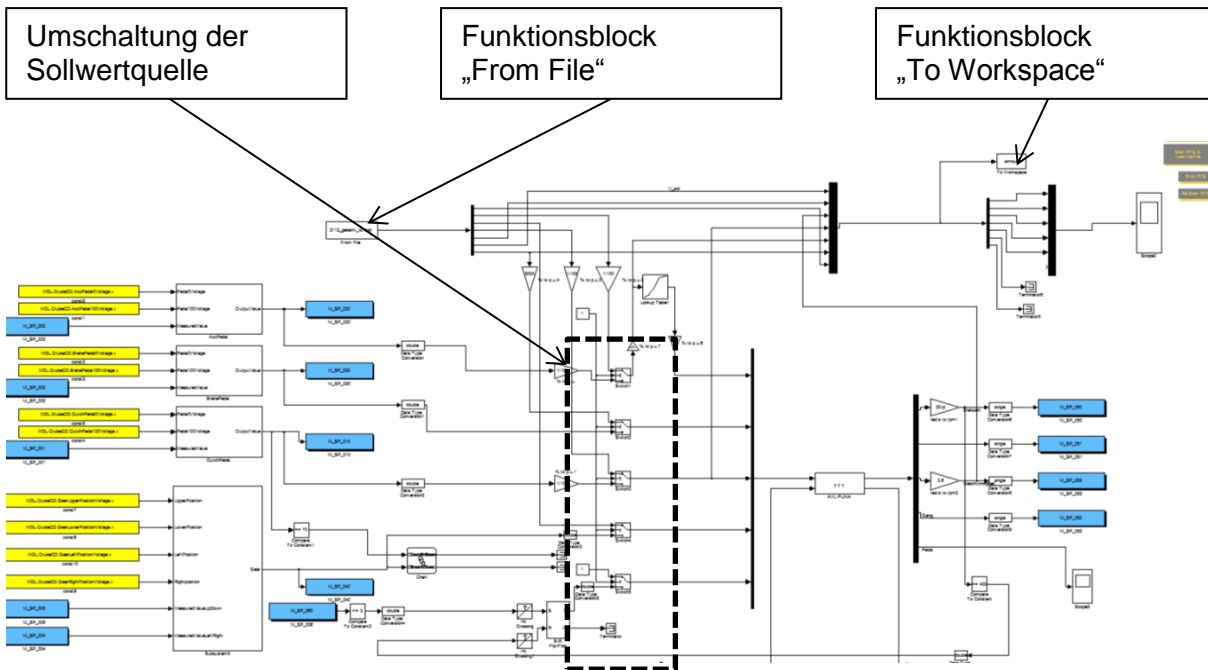


Abbildung 6-1: Anpassung des Fahrzeugmodells zur Simulation in Simulink

Die Sollwerte des Gangwechsels und die Sollgeschwindigkeit stammen aus der Norm des NEDC-Tests für manuelle Getriebe. Für die Bestimmung des Einkuppelvorganges bei Beschleunigung aus dem Stillstand werden die aus den Rollenmessdaten stammenden Größen Motordrehzahl und Fahrzeuggeschwindigkeit miteinander dividiert. Dies geschieht an Stellen, bei denen sichergestellt ist, dass im Testlauf bereits voll eingekuppelt ist. Der aus der Division der Motordrehzahl mit der Geschwindigkeit errechnete Faktor repräsentiert die Umrechnung von Motordrehzahl zur Fahrzeuggeschwindigkeit laut Kapitel 5.1.2, die auch zur Parametrierung des Getriebes eingesetzt werden. Dieser Faktor wird dann an allen Stellen der Messdaten gebildet. Weicht dieser bei gleichem eingelegten Gang von dem ermittelten Faktor ab, so repräsentiert das einen nur teilweise eingekuppelten Zustand. Aus den Punkten „Voll ausgekuppelt“ bei Stillstand und „voll eingekuppelt“ wird dann eine lineare Kupplungsbetätigung angenommen. Beim Gangwechseln während der Fahrt wird ein dreieckiger Kupplungsverlauf verwendet.



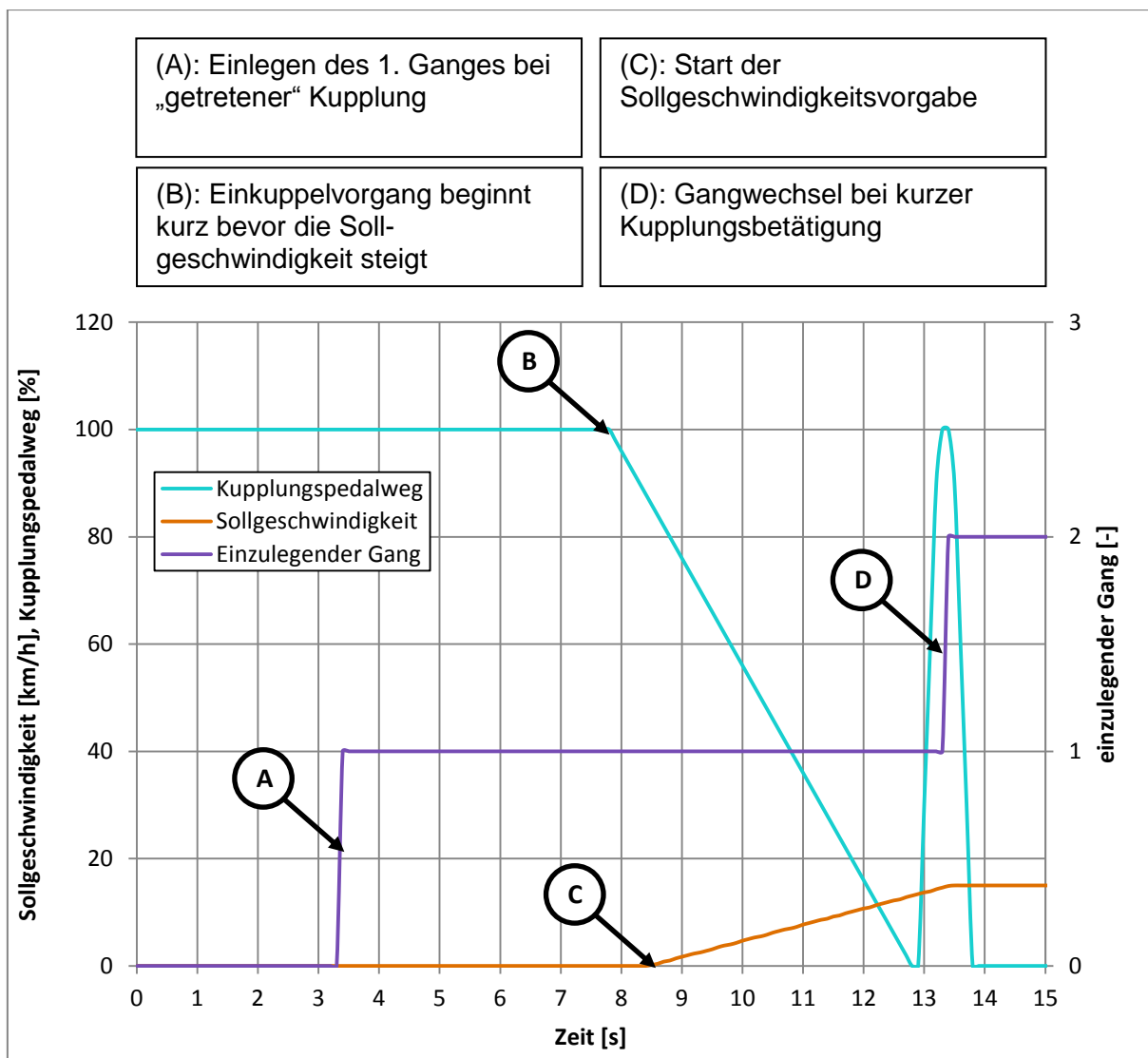


Abbildung 6-2: Festlegung des Kupplungsverlaufs

Das Hauptaugenmerk bei der Entwicklung des Modells ist das Verhalten bei Beschleunigungs- und Konstantfahrt. Bei Verwendung von Messdaten eines NEDC-Abgastests als Referenz muss sichergestellt sein, dass das modellierte Fahrzeug ausreichend abgebremst wird, um vor jedem Hügel aus dem Stillstand zu beschleunigen. Erfolgt die Beschleunigung mit falschen Anfangsbedingungen, so haben die Daten des untersuchten Hügels keine Aussagekraft. Diese Problematik ist in Abbildung 6-3 dargestellt.

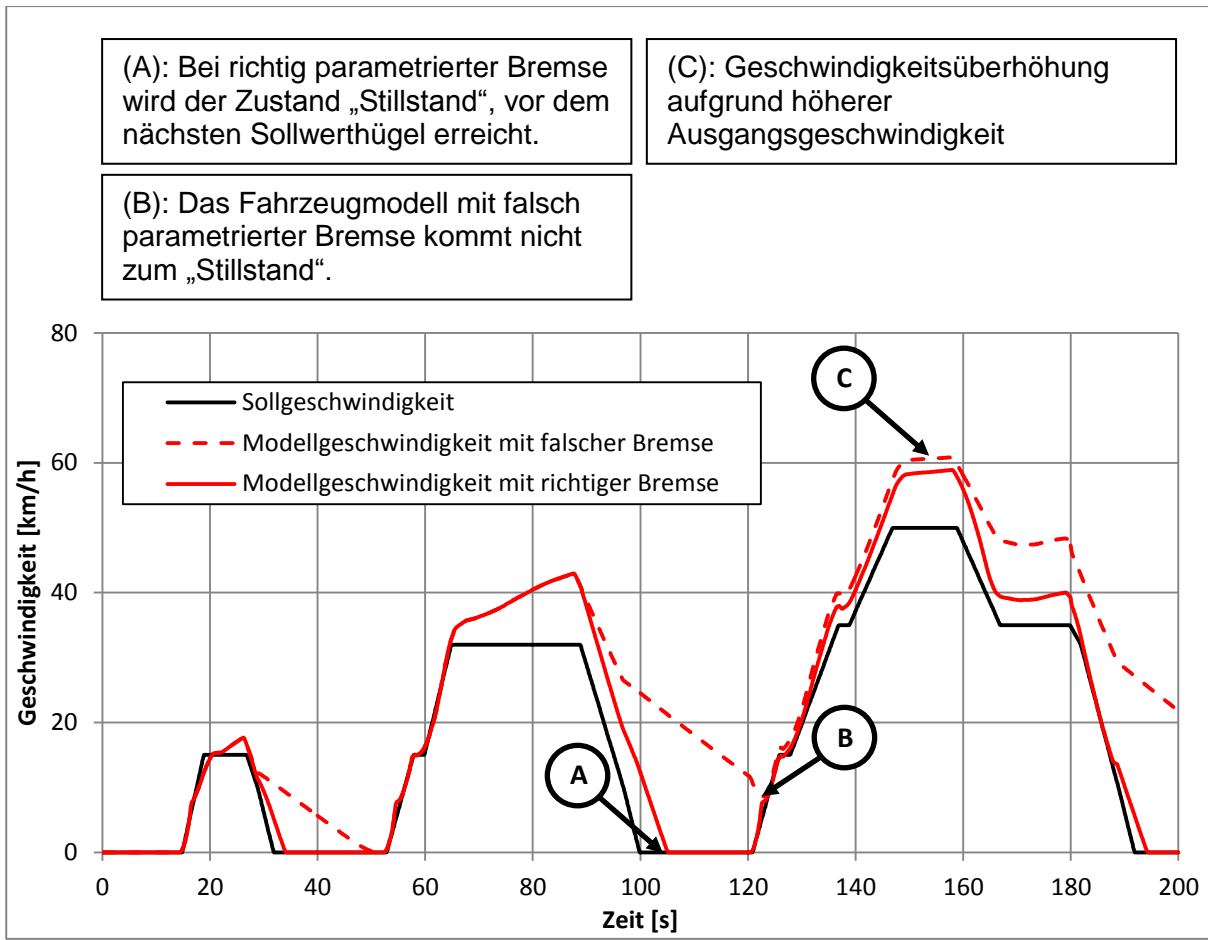


Abbildung 6-3: Fehlerfortpflanzung bei falsch parametrierter Bremskraft (Beispiel eines Simulationsergebnisses bei grob abgestimmten Fahrpedalwerten)

Bei der Dimensionierung der Bremse müssen Passagen beachtet werden, die nach der Verzögerung eine Konstantfahrt aufweisen und dort somit nicht bis zum Stillstand verzögert wird. Das hat, ähnlich zum Problem mit zu schwach parametrierter Bremse, zur Folge, dass sich der damit resultierende Fehler bis zum Erreichen des nächsten Stillstandes fortpflanzt.

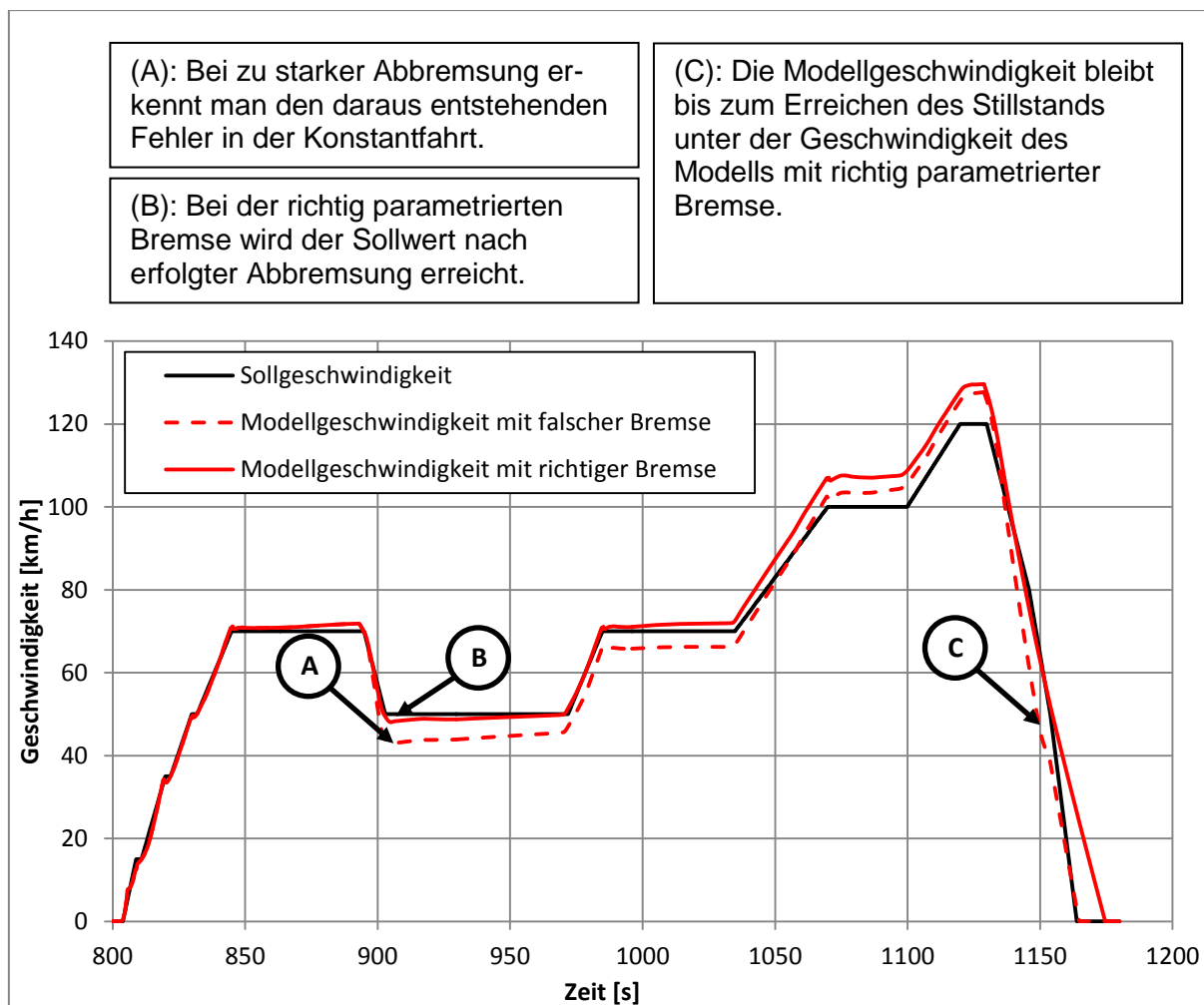


Abbildung 6-4: Dimensionierung der Bremse

Die Vorgabe des Bremsensollwertes entspricht dem Bremsdruck in Pa. Über die Parameter von Reibradius, Bremskolbenfläche und Reibwert werden die Bremsmomente der Räder errechnet. Um bei der Parametrierung der Bremse nicht bei jedem Simulationsvorgang das Sollwertfile ändern zu müssen, wird der Sollwert in Simulink mit einem Faktor multipliziert.

Da die Parametrierung der Bremse auch vom Wert der Geschwindigkeit vor Bremsbeginn abhängt, ist die Anpassung der Bremse bereits Teil des Modellabgleichs, das in folgendem Kapitel beschrieben wird.

### 6.3 Vergleich von Fahrzeugmodell und Messdaten

Die Grundeinstellung des Motormodells von CRUISE enthält kein Pedalkennfeld, d.h. der Weg des Fahrpedals von 0–100 % wird direkt als Momentenanforderung von 0–100 % Motorlast interpretiert. Als Ansprechzeit des Motors, die die Zeit bis zum Erreichen des maximalen Drehmoments beschreibt, wird ein Wert von 0,1 s gewählt, der für einen Dieselmotor mit Turboaufladung möglicherweise zu gering ist. Reale Fahrzeuge verändern aus Gründen verbesserter Fahrbarkeit den Fahrpedalwert mittels Kennlinie oder auch mittels Kennfeld, welche von Größen wie Motordrehzahl, Fahrzeuggeschwindigkeit oder Fahrmodus (Komfort, Sport) abhängen. Ebenso kann aus Gründen der Schadstoffentstehung und des Verbrennungsvorgangs nicht schlagartig die für das Fahrerwunschmoment geforderte Einspritzmenge umgesetzt werden. Der eingelesene Pedalwert wird also von mehreren Kennfeldern und Steuergerätefunktionen in eine Momentenanforderung umgerechnet, die dann die nötige Einspritzmenge bestimmt. Ohne Beachtung dieser Einflüsse resultiert eine große Differenz zwischen realem Fahrzeug und Fahrzeugmodell.

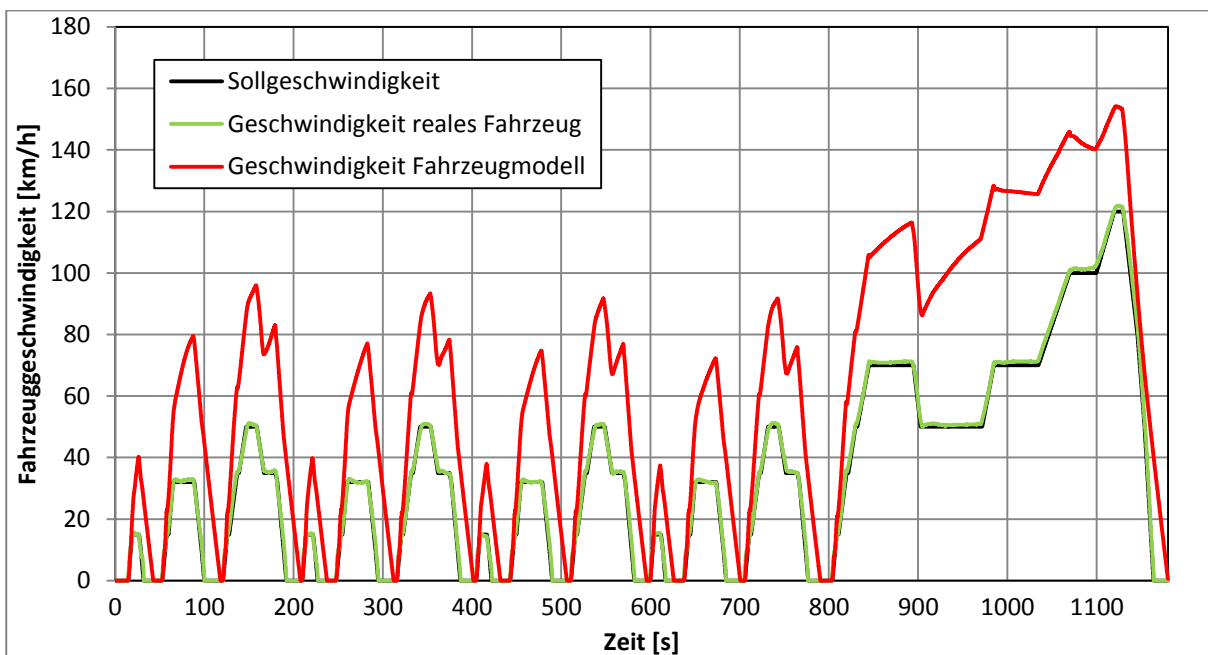


Abbildung 6-5: Geschwindigkeitsverlauf ohne Pedalkennlinie

Das Diagramm zeigt eine starke Abweichung von Modell zu realem Fahrzeug. Man erkennt sowohl, dass der Geschwindigkeitsgradient zu hoch als auch die Geschwindigkeit bei Konstantfahrt nicht konstant ist. Zur Kompensierung der beschriebenen Verzögerungen wird als erster Schritt eine mit einem „S-Schlag“ versehene Pedalkennlinie integriert. Da sich der erste Teil des Zyklus aus vier Wiederholungen zusammensetzt, wird vorerst nur einer der vier identen Abschnitte untersucht. Es ist zu beachten, dass sich die Messdaten auf ein Fahrzeug mit Kaltstart beziehen, da der konditionierte Zustand Vorschrift für den Test ist.

Aus diesem Grund ist es nicht sinnvoll, den ersten Teil zu verwenden, da hier mit einem höheren Pedalwert aufgrund höherer Reibung im Motor und Antriebsstrang zu rechnen ist und der Kaltstart nicht modelliert wird.

Zur Analyse der Pedalwertanpassung wird nun der Bereich von 600–1180 s genauer betrachtet. Der Verlauf des Sollprofils in diesem Bereich erfordert mit zunehmender Zeit eine höhere Geschwindigkeit, d.h. wenn die Anpassung des Pedalkennfeldes von kleinen nach großen Pedalwerten schrittweise erfolgen soll, so kann man die Anpassung im Sollwertprofil von links nach rechts durchführen. Die Umrechnung von Fahrpedalwert in Istmoment kann nun über eine Kennlinie oder eine Funktion erfolgen. Aufgrund größerer Parametriermöglichkeiten fiel die Wahl auf die Verwendung einer Kennlinie.

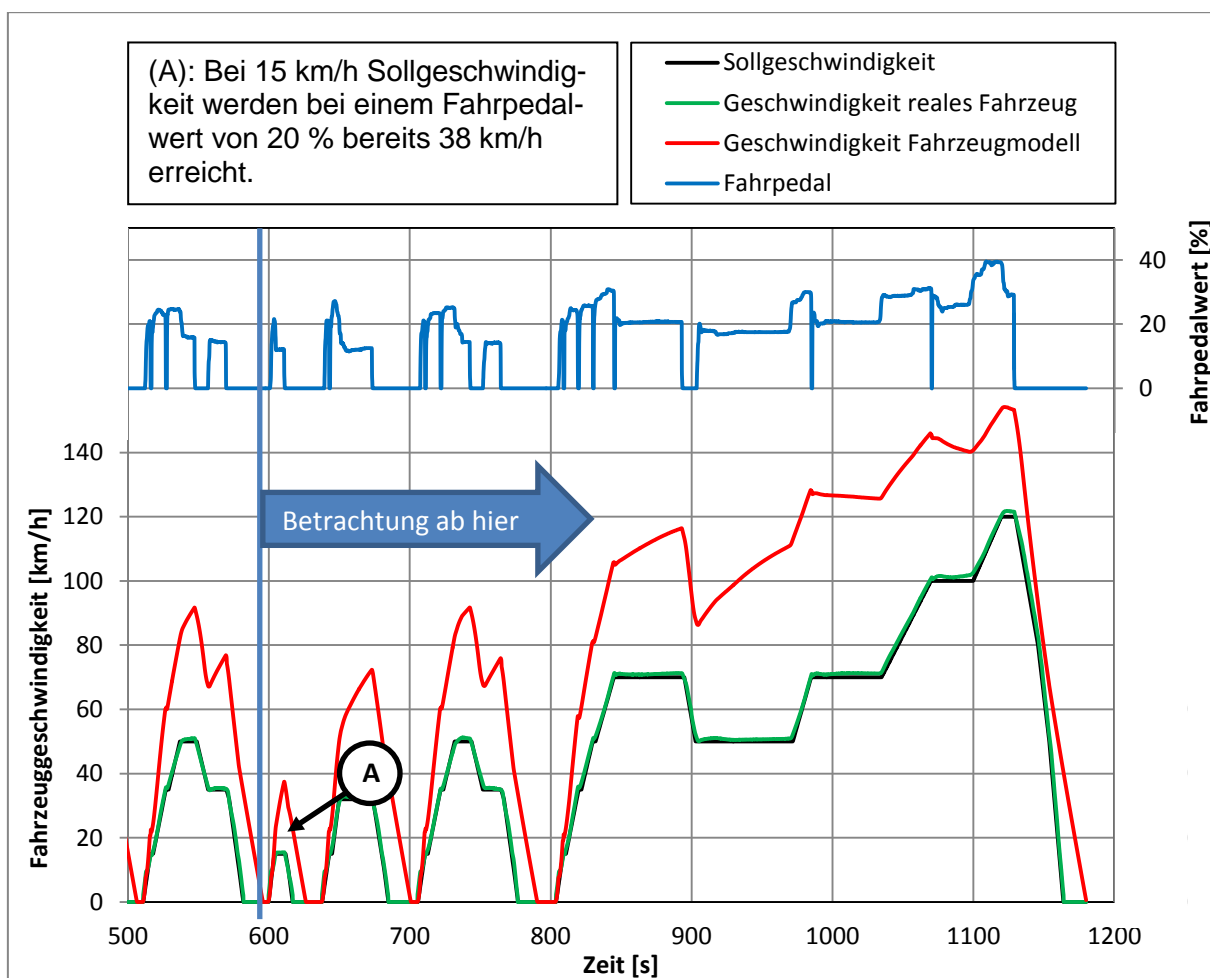


Abbildung 6-6: Für die Integration des Pedalkennfeldes verwendeter Abschnitt

Wie schon in Abbildung 6-5 zu erkennen ist, ergibt sich bei der Simulation eine mehr als doppelt so hohe Geschwindigkeit, wenn der aus den Messdaten stammende Fahrpedalwert verwendet wird. Aus diesem Grund erfolgt die Parametrierung der Fahrpedalkennlinie so, dass bei 20 % Fahrpedalstellung ein Wert von 12 % in das Modell eingegeben wird. Die Nachbarwerte sind dann so angepasst, dass der beschriebene S-Schlag entsteht.

Mit dieser Parametrierung der Pedalkennlinie erfolgt eine erneute Simulation und Auswertung. Das Ergebnis bringt bereits eine deutliche Annäherung an die Sollspur.

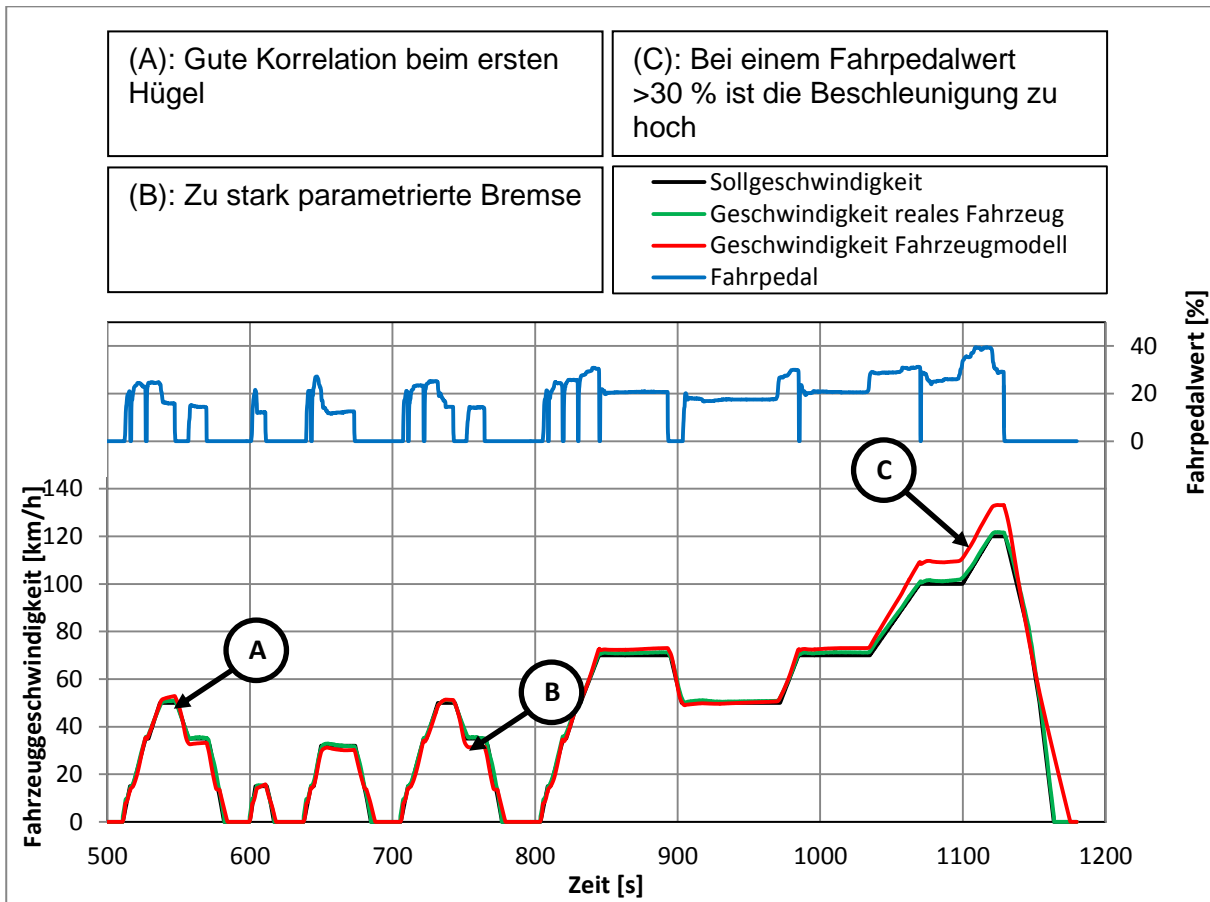


Abbildung 6-7: Simulierte Geschwindigkeit nach Integration einer Fahrpedalkennlinie

Als nächster Schritt wird der Fahrpedalwert bei 30 % und 40 % Input verringert, um die Geschwindigkeitsüberhöhung beim letzten Hügel auszugleichen. Die neu erstellte Pedalkennlinie enthält bereits stärkere Unstetigkeiten, als die der letzten Simulation.

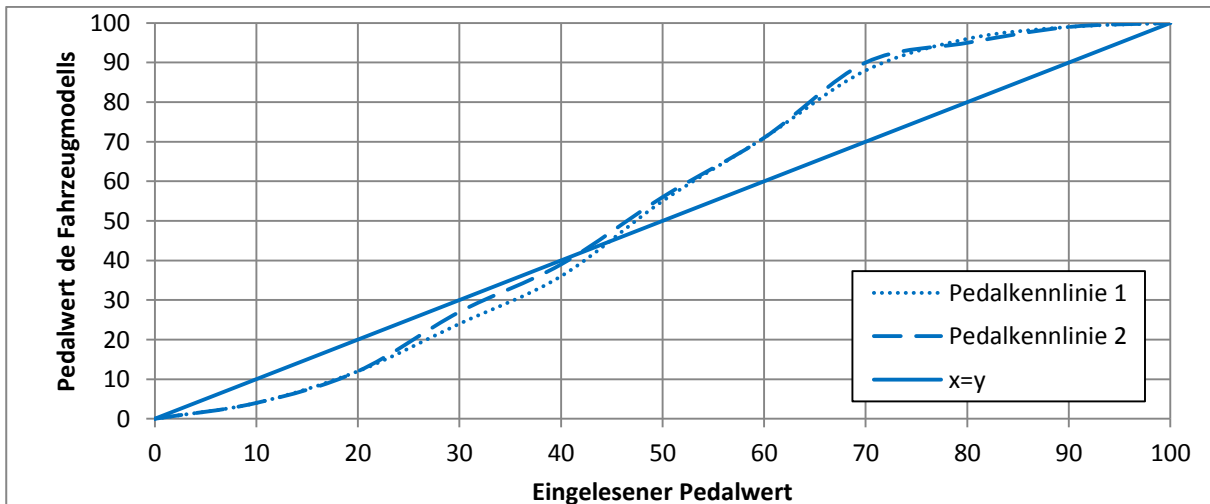


Abbildung 6-8: Fahrpedalkennlinien

Die neu gestaltete Kennlinie übergibt bei eingelesenen Fahrpedalwerten zwischen 30 % und 40 % weniger Fahrpedalwert an das Fahrzeugmodell.

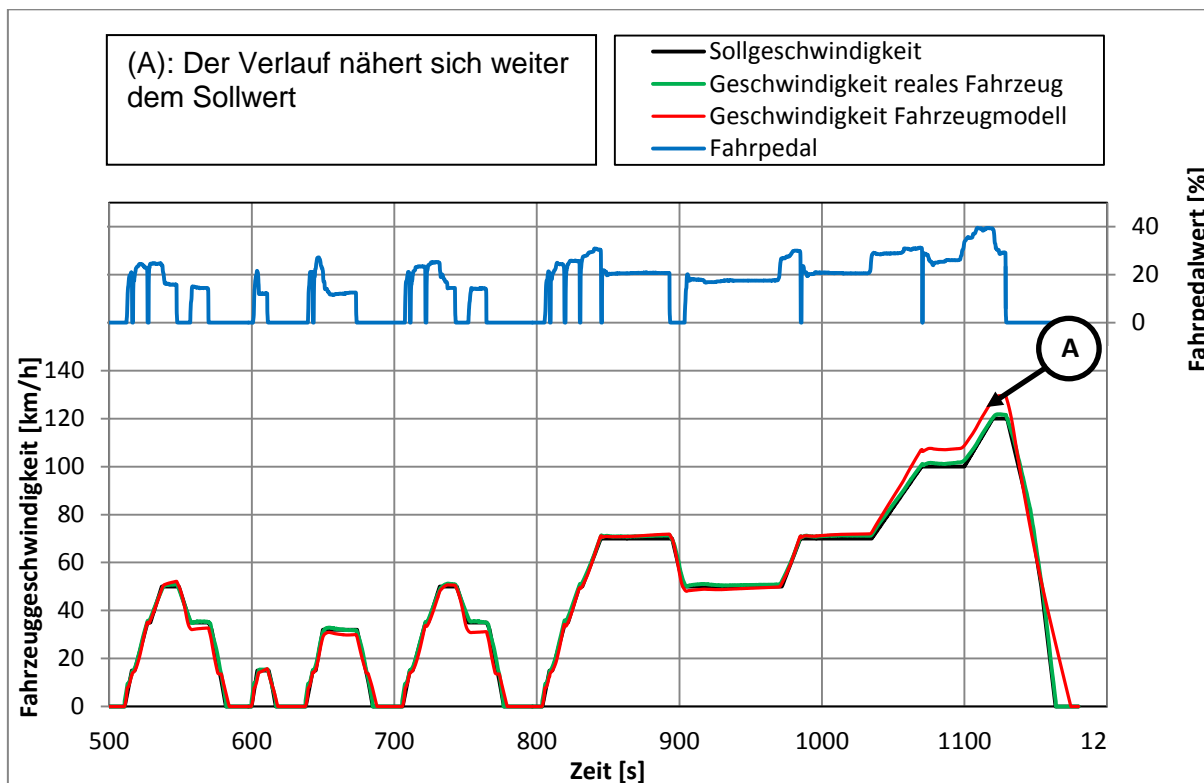


Abbildung 6-9: Simulierte Geschwindigkeit nach Anpassung der Fahrpedalkennlinie

Man erkennt die Verringerung der Geschwindigkeitsüberhöhung am letzten Hügel von 135 km/h auf 130 km/h. Würde man diese Vorgehensweise weiter verfolgen, so ist zu erwarten, dass sich die Qualität beim letzten Hügel weiter verbessern lässt. Die Anpassung der Pedalkennlinie weist dann aber eine große Unstetigkeit auf, die nicht realitätsnah ist. Dieses Verhalten lässt sich damit erklären, dass im realen Fahrzeug ein Kennfeld zum Einsatz kommt, das drehzahl- oder geschwindigkeitsabhängig ist. Mit dem Einsatz solcher Kennfelder wäre es möglich, den Geschwindigkeitsverlauf des Modells weiter anzunähern. Das Beschleunigungsverhalten und die Pedalwerte bei Konstantfahrt sind sehr gut nachgebildet und das Verhalten des Modells ist qualitativ richtig. Aus diesem Grund ist die Verwendung einer Kennlinie ausreichend.

## 7 Ergebnisse

In Kapitel 6 wird erläutert, wie die Kalibrierung des Modells vorstättenging. Dieses Kapitel beschäftigt sich mit der Analyse und Verbesserung der Modellqualität. Anschließend wird ein selbstgefahrener Test dargestellt sowie Eindrücke und Verbesserungsvorschläge von Prüfstandstechnikern gesammelt.

### 7.1 Ergebnis des Modellabgleichs mit realem Fahrzeug

#### Analyse der berechneten Fahrzeuggeschwindigkeit

Um den Optimierungsvorgang vollständig darzustellen, werden die Geschwindigkeitsverläufe der Optimierungsstufen zusammengefasst dargestellt. Bisher wurde nur der letzte Teil des NEDC-Zyklus betrachtet. Da der Zyklus vier Wiederholungen derselben Sollgeschwindigkeit aufweist, kann das Modellverhalten bei gleicher Sollspur mit unterschiedlichen Pedalverläufen getestet werden. Die unterschiedlichen Pedalwerte resultieren daraus, dass der Fahrer die sich wiederholenden Abschnitte nicht mit dem exakt gleichen Pedalwertverlauf durchfährt.

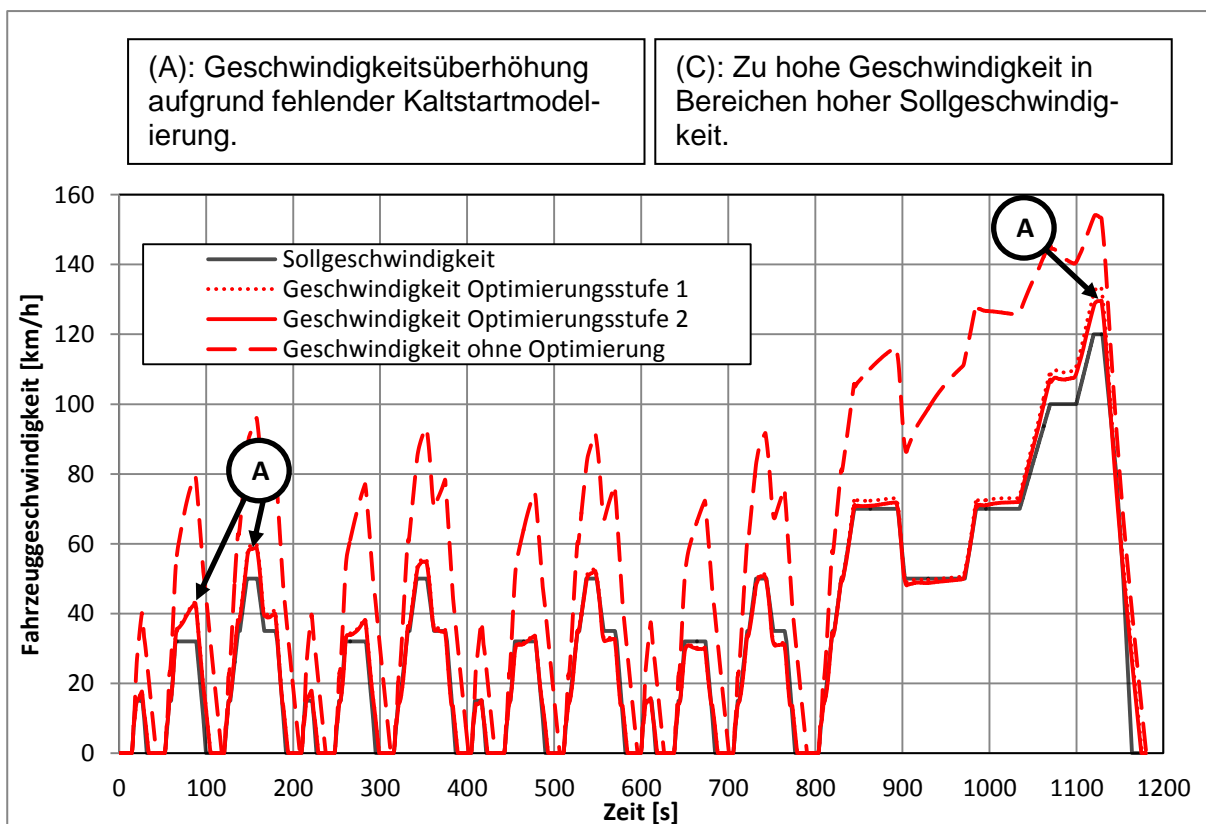


Abbildung 7-1: Vergleich der Geschwindigkeitsverläufe bei verschiedenen Kennlinien



Wie zuvor beschrieben, ist der Geschwindigkeitsverlauf ohne Optimierung des Pedalkennfeldes nicht zufriedenstellend. Durch die Einbindung der Pedalkennlinie liefert das Modell sowohl bei Konstantfahrten als auch bei Beschleunigungsfahrten gute Ergebnisse. Die Modelloptimierung in Verbindung mit der Verwendung des letzten Teils des NEDC-Zyklus zeigt, dass diese Optimierung Gültigkeit im gesamten Testzyklus hat. Beim ersten Hügel lässt sich erkennen, dass der Geschwindigkeitswert überhöht ist. Das resultiert aus der fehlenden Nachbildung des Kaltstarts. Die im Modell fehlende höhere Reibung beim Kaltstart verursacht, dass der Pedalwert des realen Fahrzeuges eine zu hohe Geschwindigkeit ergibt.

### Analyse der Ansprechzeit

Zur Analyse der Ansprechzeit erfolgt die Untersuchung des Geschwindigkeitsaufbaus am ersten Hügel des Freilandteils des NEDC-Zyklus, also ab Sekunde 800 im Testzyklus.

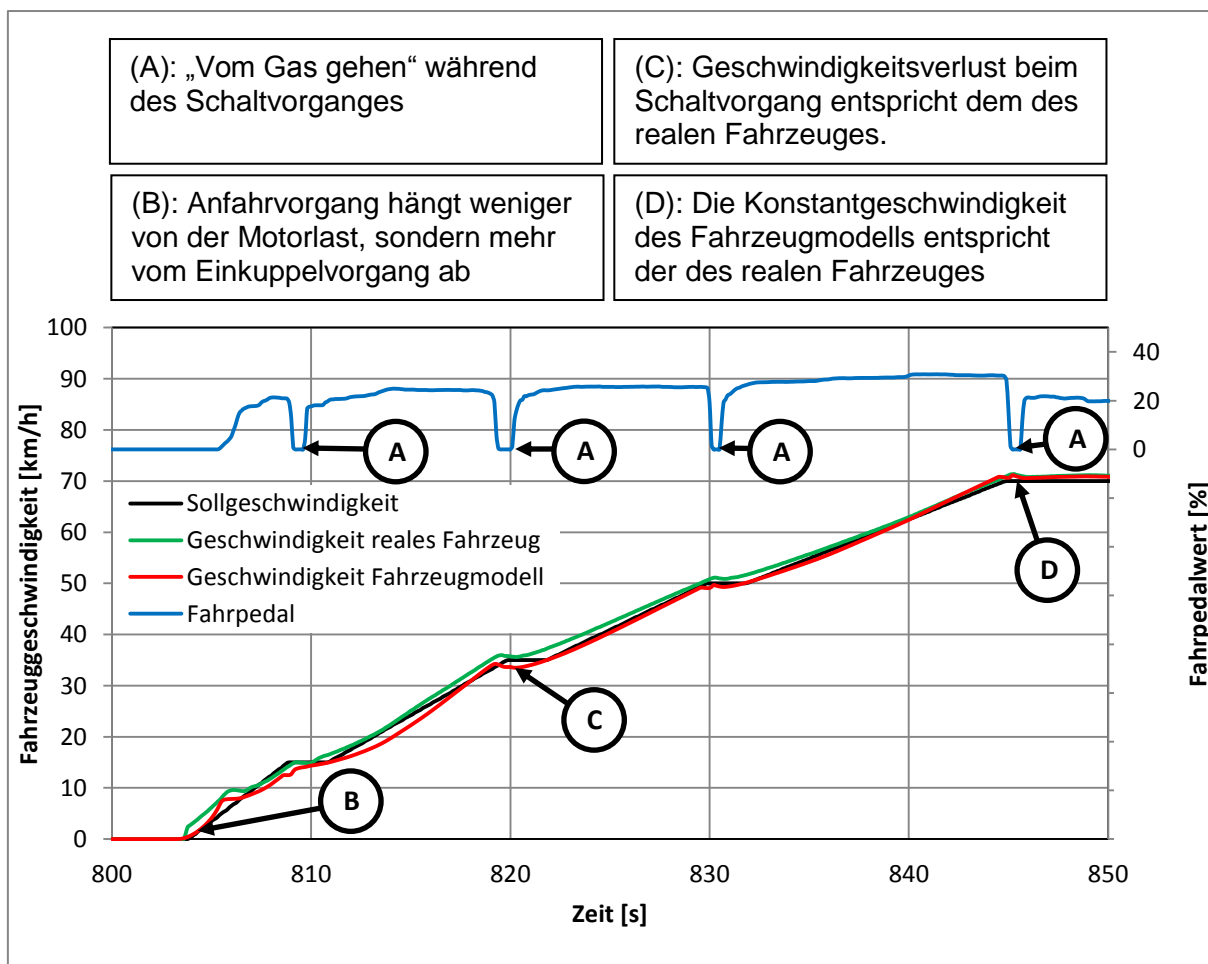


Abbildung 7-2: Analyse der Ansprechzeit

Aufgrund der kleinen parametrisierten Ansprechzeit ergibt sich die Erwartung, dass das Fahrzeugmodell zu schnell auf die Pedalwertänderungen reagiert. Zusätzlich ist bei Beschleunigung aus dem Stillstand bei einem Dieselmotor mit Abgasturbolader von einer höheren Ansprechzeit auszugehen als bei einem Saugmotor. Die trotz dieser Kriterien gute Übereinstimmung lässt sich damit erklären, dass das reale Fahrzeug bei der geringen Motorlast beim Anfahrvorgang noch keinen Ladedruck aufgebaut hat und sich im Saugbetrieb befindet. Die Beschleunigung aus dem Stillstand ist demnach hauptsächlich vom Einkuppelvorgang abhängig, der sowohl im Falle eines realen Tests als auch in Trainings am Simulator vom Prüfstandstechniker abhängig ist. Der NEDC-Abgastest fordert generell geringe Motorlasten, da er für alle PKW-Fahrzeugklassen gültig ist und auch für Fahrzeuge mit geringer Motorleistung fahrbar sein muss. Die Motorleistung von 195 kW des Versuchsfahrzeuges ist sehr groß, womit der Motor nur in unterer Teillast betrieben wird. Für Testzyklen, die eine höhere Motorlast erfordern, kann die Modellierung des Ladedruckaufbaus eine größere Rolle spielen.

Beim Gangwechsel, also im kurzzeitig ausgekuppelten Zustand, ist das Verhalten des Fahrzeugmodells ebenfalls sehr ähnlich zum realen Fahrzeug, man kann eine ähnlich große Geschwindigkeitsabnahme erkennen. Obwohl die ausgekuppelte Phase nur sehr kurz ist, ist dies ein Indiz für eine richtige Parametrierung der Ausrollkurve.

### **Analyse eines Trainings am NEDC**

Zur Überprüfung des Modellverhaltens bei Echtzeitbetrieb wird ein NEDC-Abgaszyklus am Simulator durchgeführt. Mittels Darstellung des Geschwindigkeitsverlaufs und des Fahrpedalwerts kann das Verhalten qualitativ verglichen werden. Da der Test am Simulator nicht von demselben Fahrer wie der Test, aus dem die Messdaten zur Modellkalibrierung stammen, durchgeführt wird, ist aber nicht von gleichen Fahrpedalverläufen auszugehen. Abbildung 7-3 zeigt den Fahrpedalwert und die sich daraus ergebende Fahrzeuggeschwindigkeit des am Simulator gefahrenenen Testlaufs.

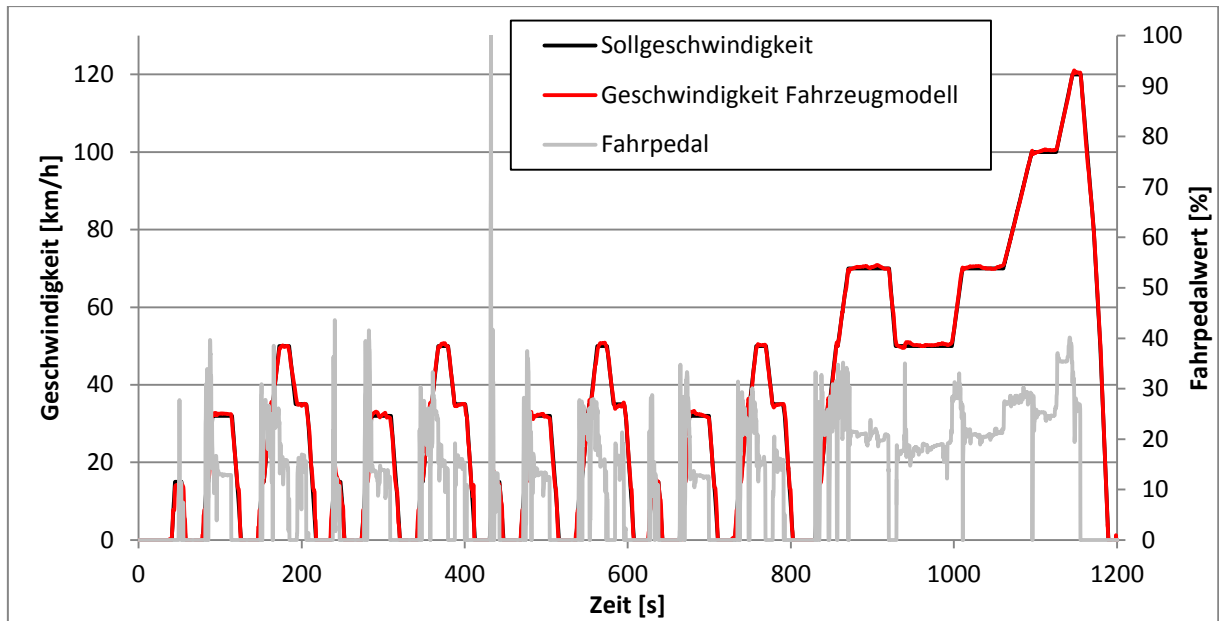


Abbildung 7-3: Fahrpedalwert und Fahrzeuggeschwindigkeit beim Training am Simulator

Nachfolgend wird die simulierte Geschwindigkeit mit dem realen Test am Prüfstand verglichen. Laut Abbildung 7-4 zeigen die Geschwindigkeitsverläufe eine gute Korrelation, daher können die Fahrpedalwerte vom Simulator und vom realen Fahrzeug miteinander verglichen werden.

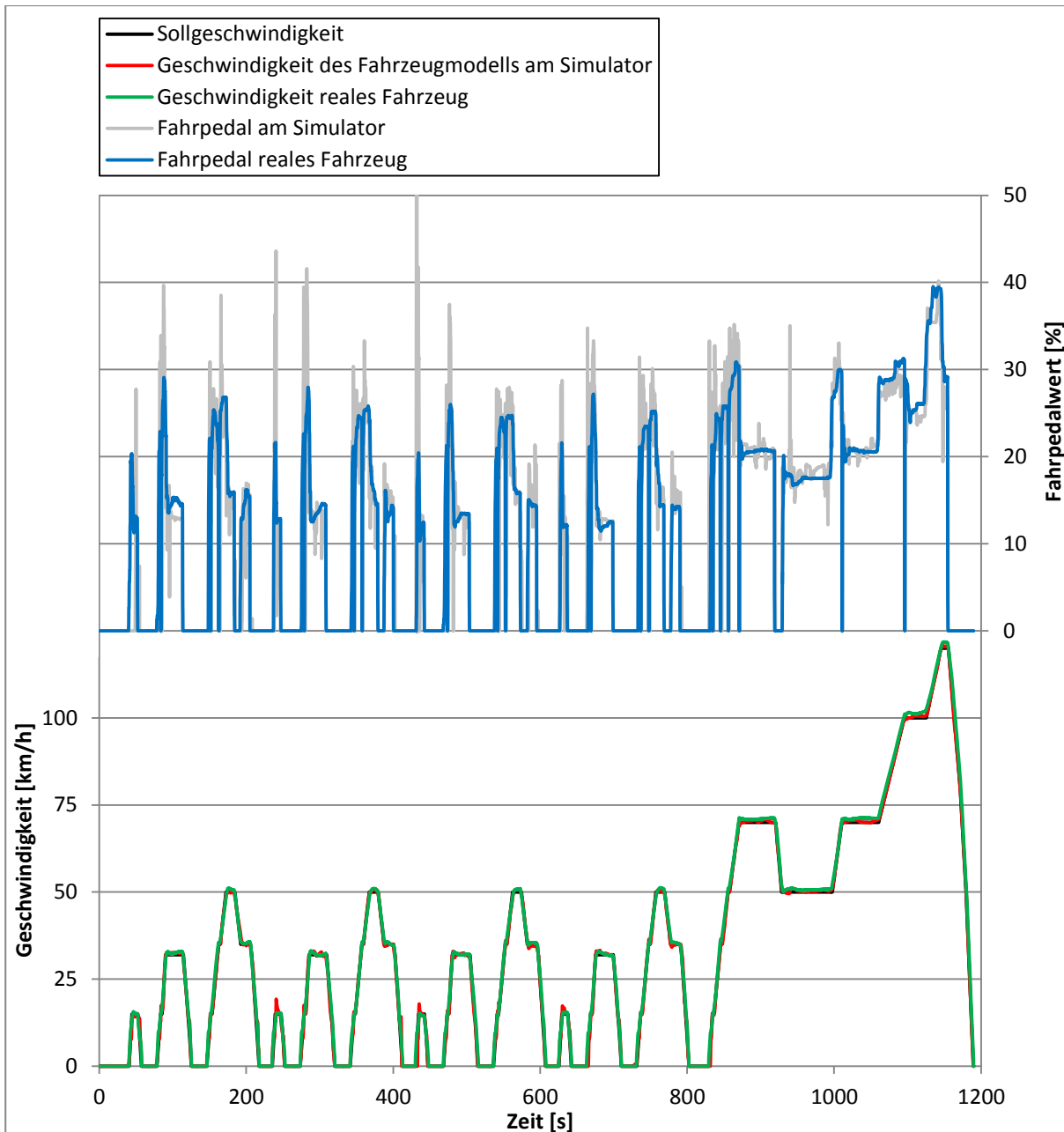


Abbildung 7-4: Vergleich der Messdaten am realen Fahrzeug mit einem Simulatortraining

Die Fahrpedalwerte liegen dabei stets im gleichen Bereich wie beim realen Test. Das Fahrpedalsignal vom Simulator weist jedoch höherfrequente Änderungen auf. Diese Abweichungen können mehrere Ursachen haben. Einerseits sind die Fahrweisen von Fahrer zu Fahrer stark unterschiedlich. Da die Messdaten vom Prüfstand nicht vom gleichen Fahrer stammen, kann dies eine Ursache für die Abweichung sein. Auch die aus einer Spielsteuerung stammenden Pedale haben ein anderes Kraft-Weg-Verhältnis, was höhere Schwankungen verursachen kann. Weiters besteht die Möglichkeit, dass das Pedalsignal des realen Fahrzeuges in dessen ECU gefiltert wird. Trotz dieses hochfrequenten Signals lässt sich erkennen, dass die in Kapitel 6 beschriebene Vorgehensweise bei der Modellkalibrierung im Echtzeitbetrieb brauchbare Ergebnisse liefert.

## 7.2 Eindrücke von Prüfstandstechnikern

Eine Befragung von erfahrenen Prüfstandstechnikern, die sich bereit erklärten, erste Funktionstests des Simulators durchzuführen, liefert folgende Eindrücke:

- Das Verhalten des Fahrzeugmodells bei Beschleunigungsfahrten und bei Gangwechsel ist gut.
- Die Pedalerie erzeugt an den Pedalen genug Widerstand, um ein realitätsnahes Training zu ermöglichen.
- Der Anzeige der Fahrzeuggeschwindigkeit am Fahrerleitgerät muss die Motordrehzahl hinzugefügt werden.
- Es ist schwierig, den Zustand des Fahrzeuges ohne Motorgeräusche einzuschätzen.
- Bei Konstantfahrt fehlen Störglieder im Modell, sodass sich die Konstantfahrt zu einfach gestaltet.

Da das Fahrerleitgerät es nicht ermöglicht die Drehzahl anzuzeigen, muss dies über einen Fachexperten hinzugefügt werden. Es ist jedoch möglich, das Signal des Fahrerleitgeräts über eine CAN-Schnittstelle an ein weiteres Anzeigegerät zu schicken. Die Anzeige der Motordrehzahl steht in Zusammenhang mit der Abwesenheit des Motorgeräusches. Die Generierung eines Motorgeräusches reicht aus, um ein „Gefühl“ für die Drehzahl zu entwickeln. Dazu ist es aber nötig, ein Programm zu entwickeln, das auf geeignete Weise auf die Variablen Motordrehzahl und Drehmoment von PUMA Open Zugriff hat, um abhängig von Drehzahl und Lastzustand ein Motorgeräusch zu generieren und über den Audioausgang des PUMA-PCs auszugeben.

Beim realen Betrieb am Rollenprüfstand entsteht durch die Beschleunigung des Fahrzeuges bzw. durch jede Laständerung eine Bewegung des Fahrzeuges, da der gesamte Antriebsstrang, die Reifen und die Aufspannung des Fahrzeuges elastische Elemente sind. Dies führt dazu, dass der Fahrer das Fahrpedal nicht konstant halten kann, womit Abweichungen in der Konstantfahrt entstehen, die wieder auszugleichen sind. Wegen fehlender Fahrzeugbewegung tritt dieses Problem am Simulator nicht auf. Durch Integration zufällig auftretender Fahrwiderstandsänderungen im Fahrzeugmodell wird das dynamische Verhalten zwar nicht nachgebildet, es bietet aber die Möglichkeit, den Schwierigkeitsgrad bei Konstantfahrt zu erhöhen. Zur Simulation der Fahrzeugbewegung auf der Rolle muss ein Teil des Prüfstands geringfügige Bewegungen erzeugen können. Beispielsweise kann der Sitz mit einem Aktuator beweglich gestaltet werden, um die Fahrzeugbewegung nachzubilden. Dazu kann man Beschleunigungsgradienten aus dem Fahrzeugmodell auswerten und als Längsbewegung an den Aktuator weitergeben. Die Integration beweglicher Teile erfordert jedoch eine stabilere Konstruktion des Aufbaus.

## 8 Schlussfolgerungen und Ausblick

Die Entwicklung einer Emissionsrollen-Simulationsumgebung mit Human Driver Interface bietet im entwickelten Stadium die Möglichkeit, sich mit dem Automatisierungssystem des Rollenprüfstands vertraut zu machen und auf die gleiche Weise wie im realen Betrieb die Bedienung des Rollenprüfstands zu trainieren. Nach Integration eines Echtzeitsystems mit Fahrzeugmodell lässt sich der Testzyklus, d.h. die Fahrt mit dem Fahrzeug am Prüfstand, ebenfalls nachstellen. Es hat sich gezeigt, dass sich für die Simulation eines Fahrzeuges mit manuellem Getriebe durch Wahl der richtigen Parameter das Verhalten eines üblichen Fahrzeugkonzeptes durch die Auswertung der Messdaten Fahrzeuggeschwindigkeit, Motordrehzahl, eingelegter Gang und Fahrpedalwert ein NEDC-Abgaszyklus hinreichend genau nachbilden lässt.

Am Simulator soll zukünftig der Betrieb von Fahrzeugen mit Automatikgetrieben möglich sein. Gleichzeitig wird die Durchführung von Tests mit modernen Fahrzeugkonzepten wie Hybridfahrzeugen immer wichtiger. Durch die Verwendung von CRUISE besteht durch den modularen Aufbau stets die Möglichkeit, neue Konzepte zu modellieren. Stehen Modelle für ein Fahrzeugkonzept bereit, kann die Kalibrierung auf ein neues Fahrzeug unter Eingabe vorhandener Messdaten automatisiert werden. Zur Zeiteinsparung kann dies von Prüfstandstechnikern selbst durchgeführt werden. Wird durch einen Fahrfehler ein Test ungültig, so kann man die Messdaten in den Simulator übertragen, um den Test am Simulator nachzubilden.

Zur Verbesserung der Haptik und Nachbildung der Eingabegeräte können Pedale mit realitätsgetreuen Abmessungen und eine Gangschaltung, die bei nicht betätigter Kupplung keine Schaltbewegung zulässt, integriert werden. Die Pedale können dann über Feder- und Dämpferelemente mit realen, einstellbaren Bedingungen versehen werden.

Für die Fahrerbewertung ist bereits die Aufzeichnung des ganzen Prüflaufes integriert. Die Bewertung kann sowohl zur Laufzeit des Trainings, z.B. durch Zählen von Toleranzbandverletzungen oder falsch durchgeführten Gangwechselforgängen, als auch nach Abschluss gestartet werden. Je nach Modelltiefe wird nach unterschiedlichsten Kriterien, wie Standardabweichung bei Konstantfahrt, Fahrpedalgradienten oder Kraftstoffverbrauch und ausgestoßene Emissionen, bewertet.

Der Simulator kann auch für Zwecke abseits des Fahrertrainings verwendet werden. Durch die integrierte CAN-Schnittstelle könnte er auch für den Test von Methodikelementen wie Pedalansteuerungsgeräten, Bremsaktuatoren und Schaltrobotern zum Einsatz kommen.

