

Universität Karlsruhe (TH)

Schriftenreihe des Instituts für Technische Mechanik

Bd. 2

Dipl.-Ing. Clemens Reitze

**Closed Loop
Entwicklungsplattform
für mechatronische
Fahrdynamikregelsysteme**



Dissertation, Universität Karlsruhe (TH), Fakultät für Maschinenbau, 2004

Impressum

Universitätsverlag Karlsruhe
c/o Universitätsbibliothek
Straße am Forum 2
D-76131 Karlsruhe

www.uvka.de

© Universitätsverlag Karlsruhe 2004
Print on Demand

ISSN 1614-3914
ISBN 3-937300-19-8

Closed Loop Entwicklungsplattform für mechatronische Fahrdynamikregelsysteme

Zur Erlangung des akademischen Grades eines

Doktors der Ingenieurwissenschaften

von der Fakultät für Maschinenbau
der Universität Karlsruhe (TH)

genehmigte
Dissertation

von

DIPL.-ING. CLEMENS REITZE
aus Heidenheim

Hauptreferent:

Prof. Dr.-Ing. Walter Wedig

Korreferent:

Priv. Doz. Dr.-Ing. Dieter Ammon

Tag der mündlichen Prüfung:

14. Juli 2004

Zusammenfassung

Reitze, Clemens

Closed Loop Entwicklungsplattform für mechatronische Fahrdynamikregelsysteme

Für die Dokumentation: Hardware-in-the-Loop – Echtzeit – Closed Loop – Simulationsumgebung – Steuergeräte Test – Fahrdynamik – Pkw, Motorrad, Anhänger – Modellbildung – räumliche Reifenkinematik – Schlupf-Definition

Schwerpunkt der vorliegenden Arbeit ist, durch ein aufgabenorientiertes Entwicklungswerkzeug und entsprechende Technologien dazu beizutragen, die Komplexität der Entwicklungsaufgaben besser zu beherrschen. Die vorliegende Arbeit behandelt die Konzeption und Realisierung einer solchen Entwicklungsumgebung. Speziell für den Entwicklungsschwerpunkt Fahrdynamikregelsysteme werden hierbei folgende Anforderungen berücksichtigt: Bedienbarkeit, Durchgängigkeit, Modell- und Datenmanagement, Modularität, Skalierbarkeit, Echtzeitfähigkeit und Automatisierbarkeit. Es wurden leistungsfähige, objektorientierte Modellbibliotheken für die Fahrzeuge Pkw, Motorrad und Pkw-Anhänger und deren verschiedene Baugruppen entwickelt. Die Modelle sind Herstellerübergreifend einsetzbar und genügen den aktuellen Erfordernissen der Fahrdynamik-Reglerentwicklung. Der Reifen-Fahrbahn-Kontakt hat maßgeblichen Einfluss auf das Fahrverhalten und die Fahrsicherheit von Fahrzeugen und stellt für die meisten Fahrdynamik-Regler die wesentliche Eingriffsmöglichkeit auf die Fahrdynamik dar. Die Kinematik des Reifen-Fahrbahn-Kontakts auf gewölbten Fahrbahnen wird als ein Problem der räumlichen Kinematik verstanden und durchgearbeitet. Die Größen Schlupf und Schräglauf werden zum ersten Mal auch für räumliche Fahrbahnen präzise definiert.

Alle verwendeten Produktnamen sind Warenzeichen der betreffenden Firmen.

Vorwort

Diese Dissertation entstand während meiner Berufstätigkeit als Diplomingenieur im Bereich Applikation und Entwicklung bei IPG Automotive GmbH Karlsruhe vorwiegend ausserhalb der regulären Arbeitszeit.

Herrn Professor Dr.-Ing. Walter Wedig danke ich sehr für die Betreuung dieser Arbeit, seine wohlwollende Unterstützung, für den Freiraum, den er mir bei der Bearbeitung der Aufgabe ließ und für die Übernahme des Erstgutachtens.

Herrn Privatdozent Dr.-Ing. Dieter Ammon danke ich für das der Arbeit entgegengebrachte Interesse und die Übernahme des Korreferats. Er gab mir manchen hilfreichen fachlichen Rat.

Dem Vorsitzenden des Prüfungsausschusses, Herrn Professor Dr.-Ing. Christoph Stiller, gilt ebenfalls mein Dank.

Mein Dank gilt ebenso den beiden Geschäftsführern der IPG Automotive GmbH Dr.-Ing. Andreas Riedel und Dr.-Ing. Alexander Schmidt. An dieser Stelle will ich es nicht veräumen, allen Kolleginnen und Kollegen für ihre Hilfsbereitschaft und freundliche Zusammenarbeit zu danken. Im Speziellen erwähnt seien hier Dr. Felix Pfister für seine vielen wertvollen Anregungen zu dieser Arbeit und die zahlreichen fruchtbaren Diskussionen und Ratschläge sowie Dipl.-Ing. Eberhard Schmidt für die Bereitstellung der benötigten Fahrwerk-Simulationsprogramme zur Parametrierung meiner Fahrzeugmodelle.

Meinem Bruder Felix bin ich besonders für seine Unterstützung bei der Literaturbeschaffung zu Dank verpflichtet. Ohne seine kompetente und zuverlässige Hilfe wäre manche Literaturstelle unerwähnt geblieben. Für das kritische Korrekturlesen des Manuskripts danke ich meinen lieben Eltern. Auch danke ich ihnen für die vielfältige Unterstützung und den persönlichen Rückhalt, ohne den mein Ausbildungsweg und diese Arbeit sicherlich nicht möglich gewesen wären.

Inhaltsverzeichnis

Zusammenfassung	I
Vorwort	II
Inhaltsverzeichnis	III
1 Nomenklatur	1
2 Einleitung	9
3 Simulationsumgebung und Softwarearchitektur	13
3.1 Mechatronische Systeme im Fahrzeug	13
3.2 Architektur der Simulations- und Prüfstandsumgebung	15
3.2.1 Bedienkonzept, Struktur und Bestandteile	15
3.2.2 Systematik eines Zeitschrittes	17
3.2.3 Hardware-in-the-Loop-Echtzeit-Simulation	19
3.3 Funktionsmodule der Umgebung: Anpassung und Erweiterung	22
3.3.1 Protokollierung, Fehlerhandling	22
3.3.2 Client-Server-Kommunikation zur Steuerung und Überwachung	26
3.3.3 Modell-Management – Varianten und Alternativen	29
3.3.4 Einbinden von Modellen als Simulink S-Function	30
3.4 Funktionsmodule der Umgebung: Durchführung von Tests	31
3.4.1 Signal-/Größen-Verzeichnis	31
3.4.2 Aufzeichnen von Signalen	33
3.4.3 Manipulation von Simulations- und elektrischen Signalen	34
3.4.4 Datenmanagement	36
3.4.5 Fahrbahn und Umwelt	37
3.4.6 Fahrmanöver, Fahrer, Fahrzeugbedienung	39
3.4.7 Stimulation	41
3.4.8 Verkehrsmodell	42
3.4.9 Testautomatisierung, Versuchsreihen	44
3.4.10 Steuergeräte Diagnose	47
3.5 Funktionsmodul Fahrzeugmodelle	48
3.5.1 Fahrzeugaufbau, Massen und Trägheiten	50
3.5.2 Fahrwerk: Kinematik und Kraftelemente	51
3.5.2.1 Realisierung eines Luftfeder-CDC-HIL-Prüfstandes	53
3.5.2.2 Wankstabilisierung	53

3.5.3	Lenksystem	54
3.5.4	Reifen	56
3.5.5	Bremsanlage	58
3.5.5.1	Realisierung eines ESP-HIL-Prüfstandes	59
3.5.6	Antriebsstrang	60
3.5.7	Aerodynamik	63
3.5.8	Anhänger	64
3.5.9	Kinematik-Sensoren	66
3.5.10	Kraft- und Momenten-Manager	68
4	Analytische Fahrzeugdynamik	71
4.1	Motivation und Aufgabenstellung	71
4.2	Grundlagen	75
4.2.1	Topologie offener Gelenkketten	75
4.2.1.1	Koordinatensystem \mathcal{F}_j (Frame)	75
4.2.1.2	Pfadmatrix D_{ij}	76
4.2.1.3	Generalisierter Körper \mathcal{G}_j (Generalized Body)	76
4.2.2	Notation und Formelapparat	76
4.2.2.1	Synthetische Kinematik	76
4.2.2.2	Analytische Kinematik	78
4.2.3	Massengeometrie	79
4.2.3.1	Generalisierter Körper \mathcal{G}_j (Generalized Body)	80
4.2.3.2	Erweiterter Körper \mathcal{B}_j^+ (Augmented Body)	80
4.2.3.3	Kinetik und Dynamik	81
4.2.4	Massengeometrische Rekursionen	82
4.2.5	Hamel-Transformation und Maggi-Matrix	82
4.2.6	Algebraisierung der Kinematik	83
4.2.7	Bewegungsgleichung nach dem Prinzip von D'Alembert/Lagrange	84
4.3	Topologie des Kraftfahrzeugs	84
4.3.1	Grundidee	84
4.3.2	Topologische Struktur des Fahrzeugs	85
4.3.3	Fahrdynamikmodell eines Pkw – "Car"	87
4.3.4	Fahrdynamikmodell eines Pkw-Anhängers – "Trailer"	88
4.3.5	Fahrdynamikmodell eines Motorrads – "Motorcycle"	88
4.4	Koordinaten des Fahrzeugs	89
4.4.1	Externaler Aspekt – ein kinematisch ungebundenes System	90
4.4.2	Internaler Aspekt – Modellierung der Radaufhängungen	92
4.4.2.1	Kinematik Handling	96
4.4.3	Drehbewegung der Räder – Sonderrolle des drehenden Rades	98
4.4.4	Koordinaten des Fahrzeugmodells – Übersicht	100
4.5	Massenmatrix des Fahrzeugs	101
4.5.1	Bemerkungen zur direkten Auswertung des Prinzips von D'A.	101
4.5.2	Struktur der Massenmatrix, Übersicht	101

4.5.3	Generalisiertes Fahrzeug-System \mathcal{F}_1 : Externale Bewegung	103
4.5.4	Radaufhängungen: Einfeldern, Lenken	103
4.5.5	Räder: Rotation	104
4.5.6	Elemente der Massenmatrix	104
4.5.7	Optimierung der Transformation eines Tensors 2-ter Stufe	106
4.6	Aufstellen der Bewegungsgleichungen des Fahrzeugs	107
4.6.1	Rekursive Verfahren der Kinetik und das Fischer-Frame	108
4.6.2	Kinematische Basisdaten ($\mathcal{F}_0 \rightarrow \mathcal{F}_1 \rightarrow \mathcal{F}_2$, “aufwärts”)	109
4.6.3	Massengeometrie ($\mathcal{F}_2 \rightarrow \mathcal{F}_1 \rightarrow \mathcal{F}_0$, “abwärts”)	111
4.6.4	Kinematik und Dynamik	111
4.7	Zusammenfassung	113
5	Reifen-Kinematik auf gewölbter Fahrbahn	115
5.1	Motivation und Aufgabenstellung	115
5.2	Kontaktkraft-Modell	116
5.2.1	Kinematik	116
5.2.2	Kräfte und Momente	117
5.2.3	Dynamik – Seitenwandtorsion und Querelastizität	118
5.3	Notation und Formelapparat	119
5.3.1	Differentialgeometrie gekrümmter Flächen	119
5.3.2	Levi-Civita Bewegung	120
5.4	Geometrie-Mapping: Von Radmitte C zum Kontaktpunkt P	121
5.4.1	Geometrische Begriffe	121
5.4.2	Bezugssysteme	122
5.4.3	Karlsruher Kinematische Kette – K3	128
5.5	Reifenkinematische Referenzgrößen	129
5.5.1	Längsschlupf s , Querschlupf s_\perp und Schräglaufwinkel β	129
5.5.2	Bemerkungen	130
5.6	Organisation des Auswertungsablaufs	133
6	Zusammenfassung und Ausblick	135
	Literaturverzeichnis	137

Kapitel 1

Nomenklatur

Abkürzungen und Begriffe, allgemein

ABS Antiblockiersystem

ACC Adaptive Cruise Control

ASR Antriebsschlupfregelung

CAN Controller Area Network

CPU Central Processing Unit

ECU Steuergerät, Electronic Control Unit

ESP Elektronisches Stabilitäts-Programm

TTP Time-Triggered Protocol

KWP2000

Keyword protocol 2000, Protokoll zur Steuergeräte-Diagnose, [ISO03]

Datensatz

Unter Datensatz . . .

Echtzeit

Ein System, das auf ein Ereignis innerhalb einer vorgegebenen Zeitspanne, der Reaktionszeit, garantiert reagiert, arbeitet “in Echtzeit”. Die Zeitskalen der Simulation auf einem Echtzeitsimulator und die der real ablaufenden, absoluten Zeit sind identisch.

I/O Input/Output, Erfassung und Ausgabe von Signalen

Versuch

Unter Versuch ...

Vektor-, Matrix- und Tensorrechnung

a, A, ω, Ω

Skalare Größen

$\bar{a}, \bar{A}, \bar{\omega}, \bar{\Omega}$

Vektoren im dreidimensionalen Raum

$\mathbf{a}, \mathbf{A}, \boldsymbol{\omega}, \boldsymbol{\Omega}$

Matrizen, Spaltenmatrizen

$\bar{\bar{a}}, \bar{\bar{A}}, \bar{\bar{\omega}}, \bar{\bar{\Omega}}$

Tensoren 1. Stufe (:= Vektor)

$\bar{\bar{\bar{a}}}, \bar{\bar{\bar{A}}}, \bar{\bar{\bar{\omega}}}, \bar{\bar{\bar{\Omega}}}$

Tensoren 2. Stufe

$[\bar{n}]_i$

Vektor \bar{n} , dargestellt in der Basis \mathcal{F}_i

$\mathbf{T}_{i \rightarrow j}$

Transformationsmatrix, die einen Vektor von der Basis i in die Basis j überführt:

$$[\bar{n}]_j = \mathbf{T}_{i \rightarrow j} [\bar{n}]_i$$

Funktionen und Rechenregeln

$\text{tr}(\)$ Spur-Funktion

$(\)^T$ transponiert-Funktion

$\bar{a} \cdot \bar{b}$ Skalarprodukt von \bar{a} und \bar{b}

$\bar{a} \times \bar{b}$ Vektor- oder Kreuzprodukt von \bar{a} und \bar{b} .

\hat{a} Das Symbol \hat{a} steht für schiefssymmetrischen Tensor 2. Stufe des Vektors \bar{a} .

Es gilt:

$$[\hat{a} \cdot \bar{b}] = \begin{bmatrix} a \\ \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \quad (1.1)$$

$$= \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \quad (1.2)$$

$$\bar{a} \times \bar{b} = (\bar{a} \times) \bar{b} = \hat{a} \bar{b} \quad (1.3)$$

$\bar{a} \otimes \bar{b}$ Tensor- oder Dyadisches Produkt der Vektoren \bar{a} und \bar{b}

Es gelten folgende Beziehungen:

$$(\bar{a} \otimes \bar{b}) \cdot \bar{c} = \bar{a} (\bar{b} \cdot \bar{c}) \quad (1.4)$$

$$[\bar{a} \otimes \bar{b}] = \begin{bmatrix} a_1 b_1 & a_1 b_2 & a_1 b_3 \\ a_2 b_1 & a_2 b_2 & a_2 b_3 \\ a_3 b_1 & a_3 b_2 & a_3 b_3 \end{bmatrix} \quad (1.5)$$

$$(\bar{a} \otimes \bar{b})^T = \bar{b} \otimes \bar{a} \quad (1.6)$$

$\bar{\bar{A}} \cdot \cdot \bar{\bar{B}}$

Doppelkontrahierendes Produkt

$$\bar{\bar{A}} \cdot \cdot \bar{\bar{B}}^T = \text{tr}(\bar{\bar{A}} \cdot \bar{\bar{B}}) = \sum_{i,j} a_{ij} b_{ji} \quad (1.7)$$

$$\frac{\partial}{\partial q_i} (\bar{a} \times \bar{b}) = \frac{\partial \bar{a}}{\partial q_i} \times \bar{b} + \bar{a} \times \frac{\partial \bar{b}}{\partial q_i} \quad (1.8)$$

Mechanik

i, j, k, l, m, \dots

Indizes (= 1, ..., n)

N Anzahl der Partikel eines Systems

$\mathbf{q} \equiv (q_i, \dots, q_n)$

Holonome oder globale oder Lagrangesche oder System-Koordinaten, generalisierte Koordinaten

\mathbf{q}, q_i Generalisierte Koordinaten, i -te generalisierte Koordinate

π_i i -te Quasi-Koordinate.

π_i ist das Integral der zugehörigen Quasi-Geschwindigkeit: $\pi_i = \int q_i dt$

$\omega_i \equiv \dot{\pi}_i$

i -te Quasi-Geschwindigkeit. Quasi-Geschwindigkeiten sind (i.Allg. nicht integrierbar) Linearkombinationen generalisierter (oder Lagrange'scher) Geschwindigkeiten \dot{q}_i .

Generalisierte Geschwindigkeiten sind zeitliche Ableitungen generalisierter Koordinaten $\dot{q}_i = dq_i/dt$, siehe auch Hamel-Transformation, Gleichung (1.26).

$$\omega_i = \sum H_{ij} \dot{q}_j$$

wobei H_{ij} Funktionen der q_j sind.

O_i, M Punkte im Raum

$$O = O_0$$

Ursprung (origin), Ursprung des Systems \mathcal{F}_0

C Schwerpunkt, (Massen-)Schwerpunkt (center of mass)

$\overline{O_i M}$ Vektor vom Punkt O_i zum Punkt M

$$\mathcal{F}_i \equiv \{O_i, \overline{X}_i, \overline{Y}_i, \overline{Z}_i\}$$

Rechtwinkliges, kartesisches, rechtssinniges Koordinatensystem i mit dem Ursprung im Punkt O_i und den drei Basisvektoren $\overline{X}_i, \overline{Y}_i, \overline{Z}_i$.

\mathcal{F}_0 Galileisches (Newtonsches) Koordinatensystem $\mathcal{F}_0 = \{O_0, \overline{X}_0, \overline{Y}_0, \overline{Z}_0\}$, absolutes Bezugssystem, Inertialsystem

\mathcal{F}_1 Mitbewegtes Bezugssystem, "Basis"-Koordinatensystem des Fahrzeugs.

Es dient der Beschreibung der globalen Fahrzeugbewegung im Raum. Alle anderen Körper des Fahrzeugs werden bez. dieses (mitbewegten) Koordinatensystems "lokal" parametrisiert.

\mathcal{F}_i^F , Fischer Frame

Koordinatensystem mit dem Ursprung im Gelenkpunkt O_i und stets zum Inertialsystem \mathcal{F}_0 parallel orientierten Koordinatenachsen, d.h. es geht durch reine Translation aus \mathcal{F}_0 hervor.

$$\overline{v}^i(A), \overline{a}^i(A)$$

Geschwindigkeit [Beschleunigung] des Punktes A bez. Koordinatensystem \mathcal{F}_i

$$\overline{v}_j^i(A), \overline{a}_j^i(A)$$

Geschwindigkeit [Beschleunigung] desjenigen \mathcal{F}_j festen Punktes, der (im betrachteten Zeitpunkt) mit Punkt A koinzidiert, bez. Koordinatensystem \mathcal{F}_i

$$\overline{\Omega}_j^i, \overline{\alpha}_j^i$$

Winkelgeschwindigkeit und Winkelbeschleunigung des Koordinatensystems \mathcal{F}_j bez. \mathcal{F}_i

$$\phi = \angle(\overline{u}, \overline{v})$$

Definition: Der Vektor \overline{u} wird durch die Drehung mit dem Winkel ϕ um $\overline{r} = \overline{u} \times \overline{v}$ in den Vektor \overline{v} übergeführt.

$$\overline{P} = \overline{P}(\mathbf{q})$$

Lage des Punktes P bez. des Referenzsystems \mathcal{F}_0 , abhängig von den Koordinaten \mathbf{q} , Ortsvektor $\overline{OP} = \overline{P} = \overline{P}(\mathbf{q})$

$$\overline{\eta} = \overline{\eta}(\mathbf{q})$$

Orientierung, abhängig von den Koordinaten \mathbf{q}

$$\bar{e}_k \equiv \partial \bar{P} / \partial q_k$$

System-Vektor oder ‘‘Heun’s Begleitvektor’’ [Heu02], partielle Ableitung des Punktes P nach der Koordinaten q_k

$$\bar{p}_k \equiv \bar{p}_k(M)$$

Translationsachse der Koordinaten q_k
prismatischer Anteil des Begleitvektors der Koordinaten q_k

$$\bar{r}_k$$

Rotationsachse der Koordinate q_k
rotatorischer Anteil des Begleitvektors im Punkt M der Koordinaten q_k

\mathcal{B}_i Körper (body) i , stets fest mit Koordinatensystem \mathcal{F}_i verbunden

\mathcal{B}_1 Körper 1, Fahrzeugaufbau; stets fest mit Koordinatensystem \mathcal{F}_1 verbunden

$\{\mathcal{B}_2\}$ Körper 2; alle Körper des Fahrzeugs, die sich bez. des Fahrzeugaufbausystems \mathcal{F}_1 bewegen, werden als \mathcal{B}_2 bezeichnet.
‘‘Körper \mathcal{B}_2 ’’ und ‘‘Körper vom Typ \mathcal{B}_2 ’’ werden synonym verwendet.

\mathcal{G}_i Generalisierter Körper (generalized body) i

\mathcal{B}_i^+ Erweiterter Körper (augmented body) i

dm Masse eines Partikels

Σ_i Materielles System

$$M \cdot \in \Sigma$$

Materieller Punkt M des Systems Σ

m^k Masse des Partikels k

M^i Masse des Systems Σ_i

$$\bar{\bar{\Pi}}^j$$

Binet’scher Trägheitstensor des Systems Σ_j bez. des Punktes O_j bzw. bez. des Punktes O_i

$$\bar{\bar{\Pi}}^j \equiv \int_{\Sigma_j} \overline{O_j M} \otimes \overline{O_j M} \, dm \quad (1.9)$$

$$\bar{\bar{\Pi}}^j(O_i) = \bar{\bar{\Pi}}^j + \overline{O_i O_j} \otimes \overline{O_i O_j} \cdot m_j \quad (1.10)$$

$$\bar{\bar{I}}^j$$

‘‘Segner’scher’’ oder klassischer Trägheitstensor des Systems Σ_j bez. des Punktes O_j bzw. bez. des Punktes O_i

$$\bar{\bar{I}}^j \equiv - \int_{\Sigma_j} \widehat{\overline{O_j M}} \cdot \widehat{\overline{O_j M}} \, dm \quad (1.11)$$

$$\bar{\bar{I}}^j(O_i) = \bar{\bar{I}}^j + \widehat{\overline{O_i O_j}} \cdot \widehat{\overline{O_i O_j}} \cdot m_j \quad (1.12)$$

A Massenmatrix des DGL-Systems, Matrix $[n \times n]$

a_{ij} Element (i, j) der Matrix **A** oder eines Tensors in Zeile i und Spalte j

b rechte Seite des DGL-Systems, Vektor $[n \times 1]$

Restbeschleunigung

Beschleunigungsanteil, der keine zweiten Ableitungen nach Koordinaten q_i, π_i enthält

$\bar{F}_{i \rightarrow j}$ Kraft (Force) von System i auf System j

$\bar{T}_{i \rightarrow j}(P)$

Moment (torque) bez. des Punktes P von System i auf System j

$\mathcal{W}_{i \rightarrow j}(P)$

Kraftwinder (Wrench) 2-Tupel von Kraft und Moment (bez. eines Punktes),

$$\mathcal{W}_{i \rightarrow j}(P) = \begin{bmatrix} \bar{F}_{i \rightarrow j} \\ \bar{T}_{i \rightarrow j}(P) \end{bmatrix} \quad (1.13)$$

Kraft $\bar{F}_{i \rightarrow j}$ von System Σ_i auf System Σ_j ($:= [3 \times 1]$) und Moment $\bar{T}_{i \rightarrow j}(P)$ von System Σ_i auf System Σ_j bez. des Angriffspunktes P ("Point of Attack") ($:= [3 \times 1]$).

$\mathcal{T}_j^i(P)$ "Kinematische Schraube" (twist) in dem mit Punkt P im System \mathcal{F}_j koinzidierenden Punkt bez. des Systems \mathcal{F}_i

Unter Verwendung obiger Notation lauten die relativkinematischen Grundformeln wie folgt:

$$\bar{\Omega}_k^i = \bar{\Omega}_j^i + \bar{\Omega}_k^j \quad (1.14)$$

$$\bar{v}_k^i(A) = \bar{v}_j^i(A) + \bar{v}_k^j(A) \quad (1.15)$$

und

$$\bar{\alpha}_k^i = \bar{\alpha}_j^i + \bar{\alpha}_k^j + \bar{\Omega}_j^i \times \bar{\Omega}_k^j \quad (1.16)$$

$$\bar{a}_k^i(A) = \bar{a}_j^i(A) + \bar{a}_k^j(A) + 2\bar{\Omega}_j^i \times \bar{v}_k^j(A) \quad (1.17)$$

Die analytische Mechanik formuliert die Kinematik des Punktes A mit Hilfe der Quasi-Geschwindigkeiten ω_i und partiellen Ableitungen \bar{A}_i entsprechend:

$$\bar{v}^0(A) = \sum_{i=1}^n \bar{A}_i \omega_i \quad (1.18)$$

$$\bar{a}^0(A) = \sum_{i=1}^n \bar{A}_i \dot{\omega}_i + \sum_{i,j=1}^n \bar{A}_{ij} \omega_i \omega_j \quad (1.19)$$

$$\bar{\Omega}_k^0 = \sum_{i=1}^n \bar{\Theta}_i \omega_i \quad (1.20)$$

$$\bar{\alpha}_k^0 = \sum_{i=1}^n \bar{\Theta}_i \dot{\omega}_i + \sum_{i,j=1}^n \bar{\Theta}_{ij} \omega_i \omega_j \quad (1.21)$$

wobei die Vektoren \bar{A} und $\bar{\Theta}$

$$\bar{A}_i \equiv \frac{\partial \bar{A}}{\partial \pi_i} = \frac{\partial \bar{v}^0(A)}{\partial \omega_i} \quad (1.22)$$

$$\bar{A}_{ij} \equiv \frac{\partial \bar{A}_i}{\partial \pi_j} = \frac{\partial^2 \bar{v}^0(A)}{\partial \pi_j \partial \omega_i} = \frac{\partial^2 \bar{\alpha}^0(A)}{\partial \omega_j \partial \omega_i} \quad (1.23)$$

$$\bar{\Theta}_i \equiv \frac{\partial \bar{\Omega}_k^0}{\partial \omega_i} \quad (1.24)$$

$$\bar{\Theta}_{ij} \equiv \frac{\partial \bar{\Theta}_k^0}{\partial \omega_i} = \frac{\partial^2 \bar{\alpha}_k^0}{\partial \pi_j \partial \omega_i} = \frac{\partial^2 \bar{\alpha}_k^0}{\partial \omega_j \partial \omega_i} \quad (1.25)$$

als Begleitvektoren (accompanying vectors) bezeichnet werden. $\omega_i \equiv \pi_i$ ist die i -te Quasi-Geschwindigkeit, deren Zeit-Integral die i -te Quasi-Koordinate π_i ergibt. n bezeichnet die Anzahl der Freiheitsgrade des betrachteten Systems.

Die Quasi-Geschwindigkeiten ω_i werden definiert durch die lineare kinematische Differentialgleichung (Hamel-Transformation)

$$\omega_i = \sum_j \bar{\bar{H}}_{ij} \dot{q}_j \quad (1.26)$$

\dot{q}_j sind wahre oder Lagrange'sche Geschwindigkeiten, d.h. Ableitungen von Lagegrößen q_i , und $\bar{\bar{H}}_{ij}$ ist die Hamel'sche Transformationsmatrix. Die Elemente $\bar{\bar{H}}_{ij}$ der Hamel-Matrix $\bar{\bar{H}}$ sind i.Allg. Funktionen der generalisierten Koordinaten q_i : $\bar{\bar{H}}_{ij} = \bar{\bar{H}}_{ij}(q_1, \dots, q_N)$.

Fahrdynamik und Reifenkinematik

β Schwimmwinkel, Winkel zwischen der Fahrzeug-Schwerpunktsgeschwindigkeit und der Mittelebene des Fahrzeugs

δ Schräglaufwinkel

γ Sturzwinkel des Rades gegenüber der Fahrbahn

ψ Gierwinkel

Radachse

Raddrehachse (C, \bar{Y}_C), d.h. Achse, um die sich die Felge dreht: $\bar{\Omega}_R^0 = \dot{\psi} \bar{Y}_C$

\bar{Y}_C Einheitsvektor der Radachse, um die sich Rad und Felge drehen

Radebene $\rho(C)$

Die Radebene $\rho(C)$ steht senkrecht auf \bar{Y}_C und bildet die Mittelebene des Rades.

C Radmittelpunkt C , Durchstoßpunkt der Radachse durch die Radebene

P Kontaktpunkt P Reifen/Fahrbahn.

Definition: P ist derjenige Punkt der Schnittgeraden der Radebene mit Fahrbahnoberfläche S , der den Abstand zum Radmittelpunkt C minimiert.

Spur (P, \bar{X}_H)

Schnittgerade der Radebene $\rho(C)$ mit der Fahrbahn-Tangentialebene $\tau(P)$ im Kontaktpunkt P . \bar{X}_H bezeichnet einen Einheitsvektor in Richtung der Spur.

Latsch, \mathcal{F}_B

“Gummistollen” des Reifen “belt” in der “Umgebung” des Kontaktpunktes P . Der Latsch befindet sich im Kontakt mit der Fahrbahn-Tangentialebene $\tau(P)$ (“ausgedehnte” Kontaktfläche bzw. Kontaktzone Reifen-Fahrbahn). Das Koordinatensystem \mathcal{F}_B ist Latsch-fest. Eine exakte Definition wird im Abschnitt 5.4.2 gegeben.

Kapitel 2

Einleitung

Die Wettbewerbssituation auf globalisierten Märkten hat sich auch für die Automobilindustrie verschärft. Nur wer mit ausgereiften und bezahlbaren Lösungen am schnellsten auf den Markt kommt – eine Reduktion der durchschnittlichen Time-to-Market-Zeiten um fast 20% von 2004 auf 2005 wird prognostiziert [Kif02], wer die gestiegene Komplexität und die Verwendung elektronischer Bauteile beherrscht und sie zu erweiterter Funktionalität und erhöhtem Bedienkomfort umzusetzen weiß, der hat Erfolg. Der Anteil der Elektronik an der Wertschöpfung des Autos steigt kontinuierlich und wird nach Expertenschätzungen bis 2010 die Marke von 40 Prozent erreichen [Hub04]. Eines der Hauptziele bei der Entwicklung moderner Fahrzeuge ist ein möglichst hohes Maß an wahrnehmbarem Kundennutzen in Form von Fahrkomfort, Funktionalität und Fahrsicherheit, bei verkürzten Entwicklungszeiten und reduzierten Entwicklungs- und Herstellungskosten. Stärkste Antriebskraft sind hierbei die Entwicklungen im Elektronikbereich. Ein wesentliches Leitprinzip stellt die Entwicklung von mechanischen und elektrischen Systemkomponenten als integrierte Einheiten unter Ausnutzung von Synergieeffekten zwischen den einzelnen Fachdisziplinen (Abbildung 2.0.1) dar.

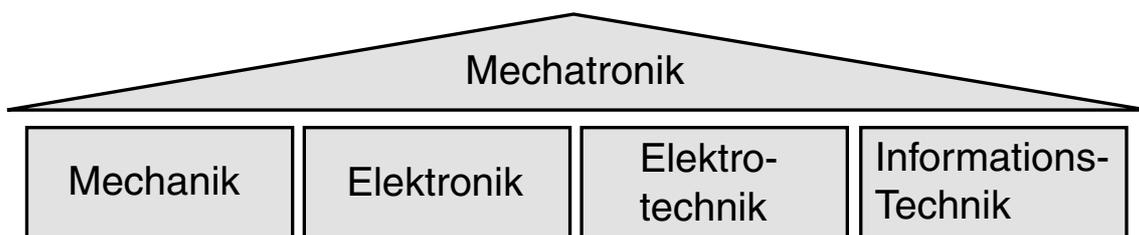


Abbildung 2.0.1: Mechatronik – Synergie-Effekt durch disziplinübergreifende Entwicklung

Mechatronische Systeme zeichnen sich oft durch komplizierte Eigendynamik aus. Ehemals rein mechanische Wirkprinzipien und Kopplungen zwischen Aggregaten werden

ersetzt durch elektronische Wirkprinzipien und elektrische Verbindungen. Neue Funktionalität und Flexibilität wird durch Softwareentwicklung ohne neue mechanische Hardware realisiert. Die funktionale und räumliche Integration von intelligenten Sensoren und Aktuatoren ermöglicht es den beschränkten Bauraum im Fahrzeug optimal zu nutzen. Diese Integrationsleistung wird von den Entwicklungsingenieuren aus verschiedenen Abteilungen beim Fahrzeughersteller und seinen Zulieferern in interdisziplinärer Zusammenarbeit erbracht. Systemkomponenten und ganze Module werden bei Zulieferern und Systemlieferanten entwickelt und gefertigt und beim Automobilhersteller zu einem Fahrzeug zusammengefügt. Der gesamte Entwicklungs- und Produktionsprozess muss an diesen Randbedingungen ausgerichtet werden.

Aufgrund der gestiegenen Anzahl komplexer, mechatronischer Komponenten mit komplizierter, variabler und regelbarer Eigendynamik und umfangreicher Vernetzung, die sich alle auf das Systemverhalten auswirken, ist eine Erprobung und Absicherung mit Prototypen in voller Breite ausgeschlossen. Vielmehr muss während des gesamten Entwicklungsprozesses konsequent das Werkzeug der Simulation eingesetzt werden: Bereits in frühen Konzept- oder Entwicklungsphasen wird das Verhalten von Systemkomponenten analysiert. Die einzelne Komponente muss zum einen autark den spezifizierten Anforderungen genügen, zum anderen muss sich auch im Gesamtsystem-Verbund das geforderte Verhalten einstellen.

Den Ausgangspunkt der Entwicklung bilden meist sogenannte reine *Model-in-the-Loop* (MIL)-Experimente, bei denen die in reine Softwaremodelle umgesetzten Spezifikationen überprüft werden. Sobald aus Konzepten konkrete Entwicklungsvorgaben geworden sind, wird auf *Software-in-the-Loop* (SIL) übergegangen. Konkrete Funktionsmodelle einzelner Komponenten und Aktuatoren, wie z.B. von Fahrwerksdämpfern, Servo-Motoren zur Lenkunterstützung oder einem Zündschloss, werden entsprechend konkreter Vorgaben aufgebaut und parametrisiert. Der zugehörige Reglercode wird implementiert und als reines Software-Modul ohne elektrisches Signalinterface in die Simulationsumgebung integriert. Sind diese Tests erfolgreich durchlaufen und erste Prototypen-Steuergeräte als Hardware verfügbar, folgen Tests auf *Hardware-in-the-Loop* -Prüfständen (HIL). Hierbei werden die Software-Regler gegen reale Steuergeräte getauscht. Das Signal-Interface dieser Hardware-Steuergeräte wird über spezielle Schnittstellen mit leistungsstarken Echtzeit-Computern verbunden. Per Simulation wird den Steuergeräten eine vollständige Umgebung in Echtzeit vorgetäuscht: Fahrversuche können so reproduzierbar virtuell durchgeführt werden und dabei das Verhalten von Reglern, Signalinterface, Aktuatoren, Diagnose und Gesamtsystem überprüft werden. Elektrische Fehler, Diagnosefunktionalität und Kompatibilität im Steuergeräteverbund werden kontrolliert. Der Austausch von Prototypen-Steuergeräten durch Serien-Steuergeräte folgt.

Aufgabenstellung und Ziele

Schwerpunkt der vorliegenden Arbeit ist, durch ein aufgabenorientiertes Entwicklungswerkzeug und entsprechende Technologien dazu beizutragen, die Komplexität der Entwicklungsaufgaben besser zu beherrschen. Im ersten Teil der vorliegenden Arbeit, Kapitel 3, wird die Konzeption und Realisierung einer solchen Entwicklungsumgebung vorgestellt, die für Fahrzeug-Regelsysteme allgemein und speziell für den Entwicklungsschwerpunkt Fahrdynamikregelsysteme folgende Anforderungen berücksichtigt:

- **Bedienbarkeit:** Anwender des Entwicklungswerkzeugs sind nicht nur Simulationsspezialisten aus den Berechnungsabteilungen der Automobil-Hersteller, sondern vor allem auch Applikationsingenieure und Versuchsfahrer. Die intuitive, an aus der Praxis bekannten Begriffen, Funktionseinheiten und Strukturen, wie z.B. Fahrzeug, Reifen, Fahrwerk, Achsfeder, Versuch, Fahrbahnbreite, orientierte Bedienung hält den Schulungsaufwand gering und erlaubt einen flexiblen Einsatz der Mitarbeiter. Außerdem wird hierdurch schon im Vorfeld Fehlinterpretationen aufgrund von Fehlbedienung entgegen getreten.
- **Durchgängigkeit:** Der gesamte Entwicklungsprozeß soll durchgängig von der Konzeptphase bis hin zur Erprobung auf Hardware-in-the-Loop -Simulatoren unter harten Echtzeitbedingungen in einer Entwicklungsumgebung mit den gleichen Simulationsmodellen, Parametersätzen und Testszenarien erfolgen. Bei der Konzeption und Realisierung des Entwicklungswerkzeugs ist auf eine im Hinblick auf geringen und vor allem pro Zeitschritt konstanten Rechenzeitbedarf der Simulationsmodelle zu achten.
- **Aufgabenorientiertheit:** Die Entwicklungsumgebung soll die Untersuchung von Fahrdynamik-Regelsystemen für Pkws und Motorräder, als Einzelfahrzeug und im Gespann-Betrieb mit einem Anhänger, ermöglichen. Die Modellgüte muss den hohen Anforderungen von Fahrdynamik-Regelsystemen genügen.
- **Modell- und Datenmanagement:** Im Rahmen der interdisziplinären, firmenübergreifenden Zusammenarbeit ist der stete Austausch von Parameter-, Modell- und Hardware-Ständen zwischen allen Projektpartnern und den verschiedenen Simulationsplattformen (Software-in-the-Loop, Hardware-in-the-Loop) notwendig. Hierfür ist es wichtig, dass Parametersätze einzelnen Baugruppen oder Funktionseinheiten zugeordnet sind und so modular ausgetauscht werden können. Gleiches gilt für die Simulationsmodule oder Hardware-Komponenten. Der einfache Wechsel zwischen verschiedenen Versionsständen ist für die Entwicklungs- und besonders für die Validierungsarbeit notwendig.
- **Modularität und Skalierbarkeit:** Wachsender Bedarf an Input/Output -Signalen oder detailliertere Simulationsmodelle stellen höhere Anforderungen an die eingesetzte Hardware in Bezug auf Zeitbedarf und Hauptspeicher. Neuere, schnellere Rechner oder die Verteilung von Aufgaben auf mehrere CPUs bieten hier Auswege.

Industriestandards bei Hardware, Infrastruktur (Bus-System) und Systemsoftware garantieren herstellerunabhängige Systemerweiterungen entsprechend der am Markt gerade verfügbaren maximalen Rechnerleistung und Funktionalität.

- **Automatisierbarkeit, Reproduzierbarkeit:** Bei jeder Änderung an Steuergerätesoftware, elektrischer Verkabelung oder an den Simulationsmodellen und ihrer Parametrierung müssen automatisiert Testreihen mit Standard-Szenarien durchgeführt werden können. Die reproduzierbaren Versuchsbedingungen sichern die Rückwirkungsfreiheit von Softwareänderungen und damit den Entwicklungsprozess und das Entwicklungswerkzeug ab. Änderungen im Reglercode lassen sich unter verschiedenen Witterungsverhältnissen unabhängig von der aktuellen Jahreszeit überprüfen. Darüber hinaus steht durch die Automatisierung die Möglichkeit zur Absicherung von Toleranzen und Systemkomponenten-Haltbarkeit zur Verfügung.

Schwerpunkt des zweiten Teils, Kapitel 4, ist die Bereitstellung einer modular strukturierten, auf geringen Rechenzeitbedarf hin optimierten Simulationsplattform für die Fahrzeugtypen Pkw, Pkw mit Anhänger und Motorrad. Die Modelle müssen herstellerübergreifend einsetzbar sein und den aktuellen Erfordernissen der Fahrdynamik-Reglerentwicklung genügen.

Der dritte Teil, Kapitel 5, behandelt die spezielle Thematik “Kinematik des Reifen-Fahrbahn-Kontakts auf gewölbten Fahrbahnen”. Der Reifen-Fahrbahn-Kontakt hat maßgeblichen Einfluss auf das Fahrverhalten und die Fahrsicherheit von Fahrzeugen und stellt für die meisten Fahrdynamik-Regler die wesentliche Eingriffsmöglichkeit auf die Fahrdynamik dar. Zur Parametrierung der dort übertragenen Kräfte und Momente wird das Reifenverhalten auf Prüfständen oder mit Messfahrzeugen vermessen. Den damit parametrisierten Simulationsmodellen kommt anschließend die Aufgabe zu, die unter speziellen “analytischen” Randbedingungen vermessenen Reifeneigenschaften auf die aktuellen Gegebenheiten beim Befahren modellierter, realer Fahrbahnoberflächen zu übertragen. Hierzu müssen den aktuellen Kontaktkräften, Geschwindigkeitsbedingungen und Reibwertverhältnissen im Kontaktbereich die korrespondierenden Messsituationen zugeordnet werden.

Kapitel 3

Simulationsumgebung und Softwarearchitektur einer modernen Entwicklungsplattform: CarMaker von IPG

Das CarMaker-Simulationspaket von IPG Automotive GmbH ist ein umfangreiches Programmpaket für Entwicklung und Test von Steuergeräten in Kraftfahrzeugen. Darin integriert sind zahlreiche, von den Ingenieuren, Informatikern und Mathematikern von IPG entwickelte Soft- und Hardware-Module wie z.B. Modelle für Fahrbahn, Reifen und Fahrer, Programme zur Darstellung von Zeitverläufen und 3D-Animationen, die Integration in die Matlab/Simulink-Welt, Hardware zum Generieren elektrischer Fehler oder Bibliotheken zur Kommunikation mit Steuergeräten und deren Diagnose-Interface. Der Autor hat am Konzept und an der Realisierung des CarMaker-Simulationspakets maßgeblichen Anteil.

3.1 Mechatronische Systeme im Fahrzeug

In die Automobilindustrie hat eine technologische Veränderung Einzug gehalten. Aus ehemals rein mechanischen Komponenten sind mittlerweile hoch integrierte Mechatronik-Systeme geworden. Hierdurch werden Systemeigenschaften in weiten Bereichen variabel. Damit einer mechatronischen Lösung gegenüber der traditionellen, rein mechanischen der Vorzug gegeben wird, muss sie Vorteile in Bezug auf hohe Zuverlässigkeit, Funktionalität und Sicherheit für den Endanwender, reduziertes Bauvolumen, höheren Wirkungsgrad bieten und kostengünstiger bei Herstellung, Wartung und späterer Entsorgung sein. Mechatronische Systeme stellen gegenüber traditionellen Lösungen wesentlich erweiterte Anforderungen:

- Erfassung des Systemzustands durch Sensoren: Mit Hilfe von Sensoren verschafft sich der Regler alle relevanten Zustandsgrößen, sowohl über das zu regelnde System,

als auch über die von ihm kontrollierten Aktuatoren. Signale müssen eindeutig plausibilisiert werden. Den Ausfall einzelner Sensoren muss die Regelstrategie möglichst kompensieren können und ein sicheres und für den Fahrer beherrschbares Systemverhalten gewährleisten. Fällt z.B. das Dehzahlsignal eines Rades aus, wird auf Basis der verbleibenden Drehzahl-signale eine Ersatzdrehzahl ermittelt. Häufig stehen weitere Alternativ-Signale bei vernetzten Steuergeräten zur Verfügung: Z.B. ist die Größe Fahrgeschwindigkeit in der Bremsanlage über die Raddrehzahl-signale vorhanden und im Antriebsstrang über Getriebedrehzahl unter Annahme des Reifenradius; aus Lenkradwinkel, Gierrate und Querbeschleunigung kann ebenfalls auf die Fahrgeschwindigkeit geschlossen werden.

- **Energiemanagement:** Elektromechanische Aktuatoren benötigen in einem 12 Volt Bordnetz hohe Stromstärken und damit kurze Kabellängen und große Leitungsquerschnitte. Das Zusammenbrechen der Versorgungsspannung muss auch bei einem meist nur kurzzeitigen hohen Energiebedarf zuverlässig verhindert werden. Bei gleichzeitigem Bedarf mehrerer elektrischer Verbraucher verhindert eine gezielte, priorisierte Verteilung der zur Verfügung stehenden Ressourcen Engpässe. Hierzu ein Beispiel: Die Aktivierung der Hydraulikpumpe durch das Bremsen-Steuergerät ist ein sicherheitsrelevanter Eingriff. Eine Leistungs-Anforderung des Klimakompressors parallel hierzu wird abgewiesen.
- **Zuverlässigkeit, Ausfallsicherheit:** Durch den Einsatz in sicherheitsrelevanten Bereichen wie z.B. Airbag, elektrische Lenkung und Bremsanlage kommt der "Null-Fehler-Toleranz" besondere Bedeutung zu. Das mechatronische System muss unter allen Randbedingungen und über die gesamte Lebensdauer des Fahrzeugs zuverlässig funktionieren. Fehlfunktion wird nicht akzeptiert, Totalausfall ist durch Rückfallebenen zu verhindern.
- **Gefährdungsvermeidung:** Passive Systeme werden nicht von selbst aktiv, sie reagieren, sie bedürfen einer Einwirkung aus der Umwelt oder durch den Fahrer. Passive Systeme enthalten in diesem Sinne kein zusätzliches Gefahrenpotential. Aktive Systeme dagegen greifen selbständig in das Systemverhalten ein: Eine Aktiv-Lenkung ändert die Radstellung ohne dass der Fahrer am Lenkrad dreht, ein Fahrdynamikregler bremst einzelne Räder oder fordert zur Verhinderung blockierender Antriebsräder zusätzliches Motormoment an. Fragen der Systemsicherheit und der Produkthaftung kommt zentrale Bedeutung zu.
- **Vernetzung:** Im Verbund arbeitende, vernetzte Regelsysteme erschließen ein deutlich erweitertes Potential an realisierbarer Funktionalität. Durch Kommunikation der Regler untereinander wird die Qualität der Regelung deutlich gesteigert. Redundanz und Sicherheit wird durch die breite Verfügbarkeit der im Systemverbund ermittelten Zustandsgrößen erhöht. Mehrfachnutzung von Informationen eröffnet die Möglichkeit zur Einsparung von Sensoren und damit zur Reduktion der Bauteilanzahl und möglicher Fehlerquellen.

Funktionalität ist nicht mehr an ein spezielles Steuergerät gekoppelt: So lassen sich brachliegende Ressourcen vorhandener Steuergeräte “fremdnutzen” oder gezielt in einem Steuergerät zusammenfassen.

Dezentrale, intelligente Sensor-Aktuator-Module werden von übergeordneten Modulen im Netzwerk kontrolliert und übernehmen eigenständig Regelungsaufgaben.

Diesen Anforderungen kann durch den Einsatz angepasster Entwicklungsprozesse und geeigneter Entwicklungs- und Testwerkzeuge begegnet werden.

3.2 Architektur der Simulations- und Prüfstandsumgebung

3.2.1 Bedienkonzept, Struktur und Bestandteile

Für Entwurf und Realisierung von regelungstechnischen Systemen und für die Simulation von dynamischen Systemen stehen leistungsfähige Werkzeuge zur Verfügung. Ihr Schwerpunkt liegt meist entweder im Bereich regelungstechnischer Systeme oder im Bereich allgemeiner Simulation. Was der Entwickler eines mechatronischen Fahrdynamikregelsystems benötigt, ist eine Umgebung, die beide Bereiche miteinander verbindet (siehe Abbildung 3.2.1). Der Anwendungsbereich erstreckt sich von der Simulation mit konzeptionellen Software-Regelalgorithmen und einfach parametrisierten Fahrzeugmodellen als Regelstrecke bis hin zu Simulationen im fahrdynamischen Grenzbereich auf Hardware-in-the-Loop-Prüfständen mit als Hardware ausgeführten Steuergeräten und aufwändig im Fahrversuch validierten Fahrzeugmodellen.

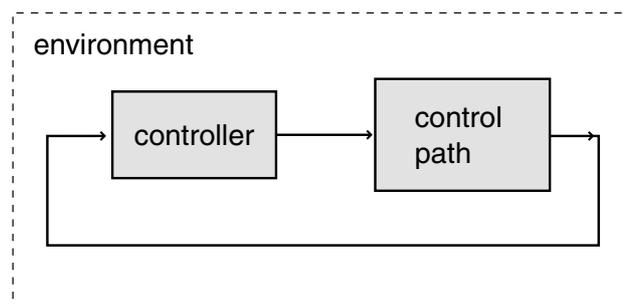


Abbildung 3.2.1: Regler und Regelstrecke – Bestandteile einer gemeinsamen Umgebung

Die hier vorgestellte Entwicklungsplattform für mechatronische Fahrdynamikregelsysteme ist auf dieses Arbeitsgebiet an der Schnittstelle unterschiedlicher Fachdisziplinen ausgerichtet. In einer Matlab/Simulink-Version steht sie dem Reglerentwickler zur Verfügung. Hier kann er sich in der ihm vertrauten Arbeitsumgebung auf seine Entwicklungsarbeit

konzentrieren. Als Streckenmodelle für die Regelsysteme stehen Fahrzeugmodelle von Pkws und Motorrädern zur Verfügung. Diese Modelle sind modular strukturiert und darauf ausgerichtet, einzelne Baugruppen, wie z.B. den Motor oder den Reifen, oder auch ganze Funktionseinheiten, wie z.B. das Lenksystem oder den kompletten Antriebsstrang, durch unterschiedliche Simulationsmodelle abdecken zu können. Die Fahrzeugmodelle bieten hierfür zahlreiche Schnittstellen und ein leistungsfähiges Modul zur Verwaltung und Auswahl verschiedener Modellvarianten einer Baugruppe. Die Ergebnisse der verschiedenen, speziellen Entwicklungsumgebungen sind in Form von exportierten Modellen und Parametersätzen direkt übernehmbar. In Matlab/Simulink oder in C formulierte Modelle sind direkt nutzbar.

Fertige, vom Spezialisten validierte Parametersätze können für die Simulation per Referenz einfach angewählt werden. Durch ihre bauteilorientierte, modulare Struktur ist die Parametrierung sehr anschaulich. Gewünschte Variationen lassen sich von vorhandenen Parametersätzen meist auch ohne Spezialkenntnisse direkt ableiten.

Eine Anforderung an das Bedienkonzept war die intuitive Bedien- und Anwendbarkeit. Zu der angesprochenen Zielgruppe zählen speziell Versuchsfahrer und Applikationsingenieure, deren Alltag ansonsten weniger von der numerischen Simulation und Berechnung geprägt ist. Spezialisten aus dem Bereich Simulation dürfen für den alltäglichen Einsatz nicht erforderlich sein. Die Bedienung ist deshalb am praktischen Fahrversuch ausgerichtet. Es wurde versucht, das gewohnte Arbeitsumfeld als virtuelle Umgebung bereitzustellen. Dies bedeutet:

- Mit einem Fahrzeug, bereift mit einem bestimmten Reifensatz, werden
- mit oder ohne angehängtem Hänger,
- auf einer Teststrecke oder einem Testgelände unter speziellen Umweltbedingungen, wie z.B mit Seitenwind oder bei nasser, glatter Fahrbahn,
- von einem Fahrer
- Fahrmanöver und Tests durchgeführt.

Die Zusammenstellung dieser Angaben werden als Versuch oder TestRun bezeichnet. Sie umfasst alle Angaben, die zur vollständigen Beschreibung einer Simulation notwendig sind, sei es, dass sie direkt im Versuchsdatensatz spezifiziert wird oder dass sie per Referenz erfolgt. Eine Simulation ist die Durchführung eines Versuchs.

Der Versuch, die Umgebung, das Arbeitsumfeld der Fahrdynamikregler, stehen als virtuelle Umgebung zur Verfügung. Mit dieser muss der Entwicklungsingenieur interagieren. Hierfür benötigt er geeignete Hilfsmittel:

- Konfiguration von Versuchen: Dialoge zur Parametereingabe, Editoren zur Definition der Fahrstrecke oder der Fahraufgabe etc.
- Durchführung und Überwachung: Start- und Stop-Befehle, Überwachung aussagekräftiger Größen in Form von numerischen Anzeigen, als Zeigerinstrumente oder

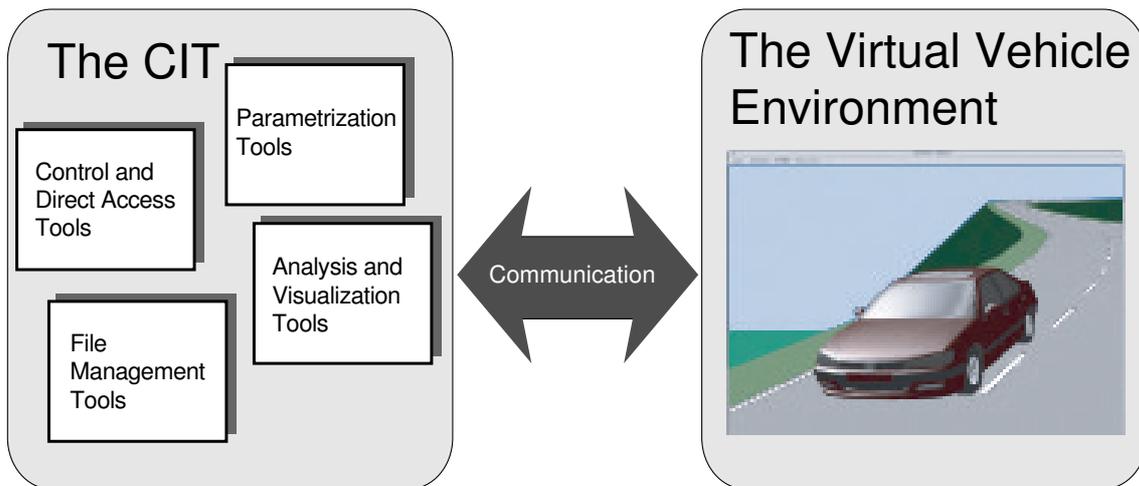


Abbildung 3.2.2: Virtuelle Umgebung und Interface-Toolbox

anderer spezieller Anzeigen, als Zeitverläufe und als 3D-Animation des in der virtuellen Umgebung ablaufenden Geschehens. Die räumliche Animation stellt ein sehr wichtiges Hilfsmittel zur Beobachtung und Beurteilung von Fahrmanövern dar.

- Analyse und Auswertung: Aufzeichnung von Signalen, deren Analyse, Ermittlung von Kennwerten
- Protokollierung: Dokumentation der durchgeführten Versuche und deren Ergebnisse.

Die virtuelle Fahrzeug-Umgebung (Virtual Vehicle Environment, VVE) mit leistungsfähigen Simulationsmodellen und angeschlossenen Hardware-Reglern auf der einen Seite und die Toolbox spezialisierter Werkzeuge (Car Interface Toolbox, CIT) zu deren Bedienung bilden die zwei großen Teilbereiche der Entwicklungsplattform (siehe Abbildung 3.2.2).

3.2.2 Systematik eines Zeitschrittes

In einer Closed-Loop-Simulations- und Prüfstandsumgebung für mechatronische Fahr-dynamikregelsysteme sind zur Berechnung eines Zeitschrittes folgende, funktional untergliederte Teilaufgaben durchzuführen:

1. Signal-Erfassung (I/O = Input/Output)
Einlesen und Auswerten der Ausgaben von Steuergeräten, Sensoren und anderer Messelektronik. Die Signale werden eingelesen, kalibriert und – falls notwendig – mit entsprechenden Sensormodellen aufgearbeitet, so dass in der virtuellen Umgebung “reale Signale” zur Verfügung stehen.
2. Aktuatoren
Input-Signale werden von als Hardware vorhandenen Aktuatoren entgegengenommen oder an Aktuator-Modelle weitergeleitet. Es handelt sich dabei z.B. um

Servomotoren, die den Fahrer beim Lenken unterstützen, oder Fahrwerks-Dämpfer, deren Dämpferhärte von einem Regler gezielt gesteuert wird.

3. Umwelt des Fahrzeugs

Ermitteln der äußeren Bedingungen für die aktuelle Kursposition, an der sich das Fahrzeug gerade befindet. Dazu zählen u.a. die Windverhältnisse, die Temperatur und der Luftdruck.

Bei der Beschaffenheit der Fahrbahnoberfläche, z.B. trockener oder nasser Asphalt oder Spiegeleis, handelt es sich um ortsgebundene, Eigenschaften des befahrenen Kurses. Das Reifenmodell ermittelt während seiner Berechnung die Beschaffenheit der Fahrbahnoberfläche für die Kontaktstellen Reifen-Fahrbahn direkt über eine Anfrage an das Fahrbahnmodul.

4. Fahrer- und Manöver-Aktivitäten

Der Führer eines Kraftfahrzeugs übernimmt zahlreiche allgemeine Bedienungsaufgaben. Dazu zählen z.B. das Aufschliessen des Fahrzeugs per Funkschlüssel, Öffnen und Schließen von Türen, Drehen des Schlüssels im Zündschloß, Drücken von Tastern zur Fahrwerks- oder Getriebe-Abstimmung oder das Einschalten des Fernlichts.

Für Fahraktivitäten im Speziellen regelt oder steuert der Fahrer über Lenkrad, Fahrpedale und Gangwahl die Quer- und Längsdynamik des Fahrzeugs.

5. Fahrzeug und weitere Modelle

Nachdem alle Bedieneingriffe erfolgt sind und auch die Einwirkung der Aktuatoren bereitstehen, werden das Fahrzeugmodell und seine Submodule berechnet. Hierzu zählen, angeordnet nach der Chronologie ihrer Berechnung

- das Lenksystem,
- die Fahrzeug-Kinematik und
- die Fahrwerkskräfte,
- die Reifen,
- der Anhänger (falls vorhanden),
- die Fahrzeugdynamik,
- das Bremssystem und
- der Antriebsstrang.

6. Signal-Ausgabe (I/O Output)

Zustandsgrößen aus den verschiedenen Simulationsmodulen des Fahrzeugs und der restlichen virtuellen Umgebung werden durch Sensormodelle geleitet oder direkt übernommen und damit Ausgangssignale für Steuergeräte, Aktuatoren oder andere Module generiert.

7. Simulationsüberwachung, -Steuerung und Basisdienste

Die Zustandsinformationen der verschiedenen Simulationsmodule werden ausgewertet. Verlassen einzelne Werte den zulässigen Bereich oder werden Fehlerzustände detektiert, werden entsprechende Reaktions-Maßnahmen eingeleitet. Dies bedeutet gegebenenfalls auch das Abschalten der Simulation oder des Simulators.

Relevante Ereignisse wie der Start eines Tests, sein fehlerfreies Ende oder sein Abbruch aufgrund von Fehlerzuständen oder ermittelte Kennwerte werden protokolliert.

Zur Kommunikation mit Interface Tools werden von dort eingetroffene Nachrichten ausgewertet, darin enthaltene Befehle ausgeführt und Anfragen beantwortet. Datenvektoren und zyklische Nachrichten werden versendet.

Die skizzierte Systematik trifft auf den gesamten Bereich von reinen Software-Simulationsumgebungen bis hin zu komplexen Hardware-in-the-Loop-Prüfständen mit realen Steuergeräten und in Hardware ausgeführten Fahrzeug-Baugruppen zu.

3.2.3 Hardware-in-the-Loop-Echtzeit-Simulation

Fahrdynamik-Regelsysteme sind mechatronische Systeme: Die Regelstrecke ist das mechanische System Fahrzeug. Die Situation wird über elektrische Sensoren erfasst und von dem im Steuergerät über Software implementierten Regelalgorithmus verarbeitet. Umgesetzt werden die Regeleinriffe über elektromechanische oder hydraulische Aktuatoren.

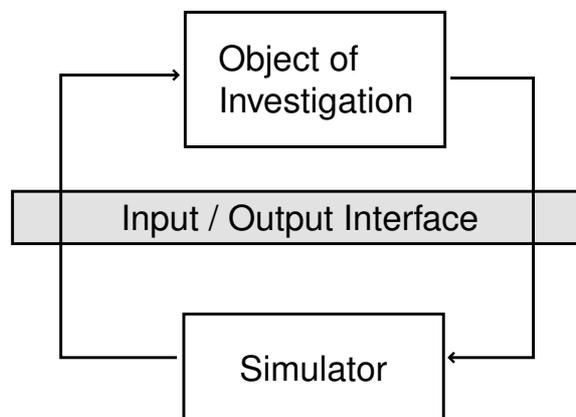


Abbildung 3.2.3: Regler und Mechatronik-Komponenten, der Gegenstand der Entwicklungsarbeit (Object of Investigation), über ein Input/Output-Interface gekoppelt mit dem Simulator

Modellbildung und Erprobung in der reinen Offline-Simulation sind die ersten Phasen in der Entwicklung eines Reglers. Sind diese abgeschlossen, werden das Signal- und

Aktuatorinterface der Steuergeräte-Hardware spezifiziert und nach diesen Vorgaben erste Hardware-Prototypen aufgebaut, die anschließend experimentell erprobt werden müssen. In dieser Entwicklungsphase kommt das Werkzeug der Hardware-in-the-Loop-Simulation zum Einsatz: Die Regelstrecke wird über numerische Simulation von Softwaremodellen nachgebildet und verhält sich gegenüber dem über ein Signalinterface angekoppelten Hardware-Prototypen-Regler wie die echte Regelstrecke. Die physikalischen Größen aller Sensorsignale werden von Simulationsmodellen berechnet und konditioniert auf die elektrischen Sensoreingänge des Reglers gelegt. Die Steuersignale des Reglers werden an seinen Aktuator-Ausgängen erfasst und von den Aktuator-Simulationsmodellen weiterverarbeitet. Bei Bedarf werden reale Aktuatoren an das Steuergerät angeschlossen, um Lasten, d.h. vor allem die Ströme, wie im realen Betrieb fließen zu lassen (siehe Abbildung 3.2.3).

Simulation in Echtzeit ist deshalb notwendig, da ein als Hardware realisierter Regler immer in Realzeit arbeitet, Signale einliest, verarbeitet und Aktuatoren bedient.

In der hier vorgestellten Entwicklungsumgebung erfolgt der Übergang von der reinen Offline-Simulation zur Hardware-in-the-Loop-Echtzeitsimulation fast unmerklich:

- Die Entwicklungsumgebung und die Simulationsmodelle sind die gleichen.
- Die Struktur des Berechnungsablaufs bleibt unverändert. Software-Simulationsmodelle für den Regler werden deaktiviert, die Signalleitungen werden abgetrennt und über einen Block zur Signalkonditionierung auf den Treiber eines I/O-Moduls zur Signalerfassung bzw. Signalausgabe gelegt.
- Die Simulation läuft nicht mehr direkt auf dem Bedienrechner, sondern in Echtzeit auf einem separaten Rechner. Dieser Simulationsrechner läuft unter dem Echtzeitbetriebssystem LynxOS. LynxOS ermöglicht einen harten Echtzeit-Betrieb und stellt alle notwendigen Dienste, wie z.B. die Netzwerkprotokolle TCP/IP und UDP und den lesenden und schreibenden Filesystemzugriff über Netzwerk per NFS, zur Verfügung. Hierdurch können die CarMaker-Interface-Tools zur Konfiguration und Steuerung der Simulation unverändert eingesetzt werden.

Die Zykluszeit beträgt typischerweise 1 Millisekunde. Verschiedene, hochdynamische Signale, wie z.B. die Ansteuerung der Hydraulikventile eines ESP-Systems werden mit 0.25 Millisekunden oder schneller erfaßt.

Zum Einsatz kommen PowerPC-Prozessoren auf VMEbus-Platinen. Zur Signalerfassung und -ausgabe werden Industriestandard-Module eingesetzt auf Basis von Mezzanine-I/O-Modulen. Es handelt sich hierbei um Trägerplatinen mit einem VMEbus-Interface, auf die die eigentlichen I/O-Module aufgesteckt werden. Die auf einem Simulator eingesetzten Module richten sich nach benötigter Signalanzahl, Signaltyp und den Anforderungen an die Signale. Gängige Signal- bzw. Modultypen sind

- Analog Input (analog digital Wandlung A/D)
- Digital Input

- Analog Output (digital analog Wandlung D/A)
- Digital Output
- Signal-/Frequenzgenerator Output
- CANbus [Law94][Ets00] Input/Output

Zur Signalerfassung ist teilweise zusätzliche Elektronik erforderlich: Ein hochfrequentes, pulsweitenmoduliertes Signal (PWM) wird für die anschließende, "niederfrequente" Analog-Digital-Wandlung mit einem Analog-Input-Modul vorbereitet; die hohen Ansteuerströme eines elektrisch verstellbaren Fahrwerksdämpfers werden über eine Strommessbrücke zugänglich gemacht; die über Stromniveaus kodierten Drehzahl-Informationen aktiver Raddrehzahlsensoren werden von dem analogen Spannungssignal eines hochohmigen Digital-Analog-Output-Moduls abgeleitet. Spezielle Ausgabesignale: Im Bereich Motor, Kurbelwelle und Getriebe sind Signale häufig winkelsynchron und das Steuergerät triggert auf die Signalfanke. Die Signale intelligenter Raddrehzahlsensoren enthalten zusätzlich zum eigentlichen Rechteck-Drehzahl-Signal weitere kodierte Informationen für Stillstand, Drehrichtung etc. Solche Signale müssen über I/O-Module mit programmierbarer Signalform realisiert werden. Entsprechend der Parametrie-Vorgabe werden die Signale vom Modul autonom erzeugt.

Der Simulationsrechner greift über den VMEbus direkt auf die I/O-Module zu. Mit zusätzlichen PowerPC-CPU's lässt sich der Simulator problemlos aufrüsten und an gesteigerten Bedarf für Rechenleistung anpassen. Über VMEbus werden hierzu weitere CPU's in den Simulator integriert und übernehmen Simulationsaufgaben oder auch die zeitintensive Signalerfassung oder Signalausgabe.

Simulatorkopplung

Die Anforderungen an die Detaillierung der Simulationsmodelle oder die Komplexität der zu regelnden Prüfstandshardware machen bisweilen den Einsatz mehrerer, untereinander gekoppelter Simulatoren notwendig:

- Simulation der Umfelderkennung und Objektverfolgung mittels eines Radarsensors (Abbildung 3.4.7),
- Simulation der Verbrennungsvorgänge und Sensorsignale zum Test und zur Applikation von Motorsteuergeräten,
- Hydraulik-Aktuatoren zur Belastung eines als Hardware ausgeführtes Lenksystems,
- Simulationsmodelle, für deren Echtzeit-Berechnung die Rechenleistung eines einzelnen Computers nicht ausreicht.

Die Realisierung der Kopplung richtet sich nach den Erfordernissen und den Gegebenheiten. Die verfügbare Hardware, die Anforderungen an die Direktheit der Kopplung,

die auszutauschende Datenrate, die Geschwindigkeit und die Zuverlässigkeit sind Auswahlkriterien für den Kopplungsmechanismus: Verwenden mehrere Simulatoren jeweils Rechner mit VMEbus-Interface, so können diese in einem gemeinsamen VMEbus-Rechner-Gehäuse installiert werden und über Shared-Memory-Bereiche auf dem VMEbus miteinander kommunizieren. Aktion und Reaktion werden innerhalb weniger Millisekunden ausgetauscht. Ebenfalls sehr enge Kopplungen lassen sich auf Basis einer CANbus-Verbindung oder über Lichtwellenleiter realisieren. Eine kompatible Rechner-Hardware ist hierfür nicht erforderlich. Für Kopplungen mit geringer Datenrate und relativ hoher Zykluszeit lassen sich z.B. auch Ethernet-Netzwerk-Verbindungen einsetzen.

3.3 Funktionsmodule der Entwicklungsumgebung: Anpassung und Erweiterung

Die in den Fahrdynamikreglern und anderen Steuergeräten implementierten Regelalgorithmen unterliegen der kontinuierlichen Weiterentwicklung und Verbesserung. Damit einher gehen auch Modifikationen in der angeschlossenen Sensorik und Aktuatorik: Sensoren für bisher nicht detektierte Signale kommen hinzu, werden durch verbesserte ersetzt oder entfallen, da sich das Steuergerät die notwendige Information auf anderem Wege beschafft. Die Ausgabesignale der Regler werden entsprechend der veränderten Anforderungen neu definiert. Modifizierte oder neue Aktuatoren bieten verbesserte Eingriffsmöglichkeiten. Speziell im Bereich der Vorentwicklung mechatronischer Systeme unterliegen deshalb auch die Entwicklungswerkzeuge einem kontinuierlichen Anpassungs- und Weiterentwicklungsprozess. Die hierfür in der vorgestellten Entwicklungsplattform vorgesehenen Schnittstellen und Module sind Gegenstand dieses Abschnitts.

3.3.1 Protokollierung, Fehlerhandling

Wesentliche Bestandteile jeder Art von Steuergeräteentwicklung sind Protokollierung und Dokumentation. Damit Entwicklungsschritte später nachvollziehbar und Ergebnisse korrekt bewertbar sind, wird die Konfiguration der Testumgebung festgehalten, Art, Umfang und Anzahl der durchgeführten Versuche, Versionsstände von Regler-Software und Steuergeräte-Hardware usw. Während der Tests selbst sind eingetretene Ereignisse zu dokumentieren, wie z.B. vom Steuergerät diagnostizierte Fehler, Signale außerhalb des zulässigen Toleranzbandes oder auch dass alles korrekt ablief und keinerlei Auffälligkeiten festgestellt wurden. All diese Informationen müssen ohne manuelle Aktivierung automatisch festgehalten werden. Jede manuelle Aktivierung wird im entscheidenden Fall potentiell vergessen.

Für die Simulation bietet sich hierfür ein eigenständiges Modul an. Es muss u.a. folgende Funktionalität und Eigenschaften besitzen:

- Ausgabe aus dem Programm heraus: Unterschiedliche Arten von Ausgaben sind zu unterstützen.

“Fehlermeldungen” werden ausgegeben, wenn ein harter Fehler auftritt, d.h. ein Ereignis eintritt, bei dem die Fortsetzung des Tests nicht sinnvoll oder möglich ist. Die Simulationssteuerung leitet hierauf das Simulationsende ein oder bricht den Test als Notaus-Maßnahme unmittelbar ab.

“Warnungen” sind Hinweise auf besondere Ereignisse, die nicht unmittelbar und beim ersten Auftreten zum Ende des Tests führen sollen. Sie können auch als “weiche Fehler” bezeichnet werden.

“Meldungen” sind textuelle Ausgaben, die z.B. dazu dienen, zu einem bestimmten Zeitpunkt die zurückgelegte Wegstrecke oder den notwendigen Bremsweg auszugeben.

- Protokolldatei: Alle Ausgaben werden zusätzlich zu den auf dem Bildschirm angezeigten Dialogen dauerhaft in einer Protokoll-Datei archiviert. Zur eindeutigen Zuordnung wird automatisch der Beginn jedes Tests mit einem Weltzeit-Zeitstempel und jede Ausgabe während des Tests mit einem Zeitstempel bezogen auf Testbeginn versehen.
- Einfach zu handhabende Anwenderschnittstelle: Nur wenn ein Modul einfach zu nutzen ist, wird es konsequent eingesetzt.

Der Entwickler kann eine Meldung an beliebiger Stelle im Simulationsprogramm genau dort erzeugen und ausgeben, wo es notwendig ist. Die Simulationsumgebung prüft an zentraler Stelle zu Beginn jedes Simulationszeitschrittes, ob in der Zwischenzeit Fehler oder Warnungen aufgetreten sind. Gegebenenfalls leitet sie das Simulationsende ein oder bricht die Simulation ab.

- Portabel: Verfügbarkeit in den verschiedenen Simulationsumgebungen und Fachabteilungen, von Software-in-the-Loop bis zum Hardware-in-the-Loop-Prüfstand. Die Anpassungsarbeiten erfordern aufgrund der einfachen, über nur wenige Funktionen realisierten Schnittstelle nur geringen Aufwand. Meist ist nur das Öffnen von Fehlerdialogen anzupassen.
- Fehlerklasse, individuell erweiterbar: Auftretende Fehler lassen sich nach Zeitpunkt und Stelle des Auftretens klassifizieren. Der Anwender muss die Liste bekannter Fehler nach seinen Bedürfnissen einfach erweitern können.
 - Fehler bei der Initialisierung: Notwendige Parameter sind nicht angegeben oder haben unzulässige Werte, Parametersätze sind nicht vorhanden, Prüfstands-Hardware befindet sich nicht im geforderten Zustand, z.B. der Druckspeicher des Pneumatik-Zylinders zur Betätigung des Bremspedals lässt sich nicht befüllen oder eine Versorgungsspannung fehlt.
 - Simulations-Fehler: Simulationsmodelle detektieren eine Situation, in der die Fortsetzung der Simulation nicht sinnvoll erscheint oder möglich ist: Das Fahrzeug ist von der Strasse abgekommen, der Motor wurde abgewürgt und das Fahrzeug steht.

- Steuergerätefehler: Ein Steuergerät erkennt ganz allgemein auf Fehler oder meldet einen bestimmten Fehlercode, z.B. “Drehzahlfühler vorne links ausgefallen”.
- Allgemeiner Fehler: Diese Fehlerklasse wird verwendet, wenn ein Fehler keiner speziellen Klasse zugeordnet wurde.
- Unterdrückung bzw. Verzögern von Reaktionen bei Auftreten von Fehlern: Zum Test der Diagnosefunktionalität eines Steuergerätes werden bewusst Fehler generiert, wie z.B. Sensorsignale verfälscht oder unterdrückt. Erkennt das Steuergerät einen extra herbeigeführten Fehler nicht, ist dieses Nicht-Erkennen als Fehler festzuhalten.

Test der Abschaltstrategie und Rückfallebene: Wird untersucht, wie sich das Gesamtsystem z.B. nach Ausfall eines Sensorsignals verhält, darf dieses Ereigniss nicht mehr zum sofortigen Simulationsende führen. Es muss aber dennoch protokolliert werden. Hierfür ist die Möglichkeit vorgesehen, Fehler auf den Status einer Warnung zurückzustufen.

Untersuchung als sporadisch detektierter Fehler: Nur selten, scheinbar nicht reproduzierbar auftretende, wieder verschwindende Fehler sind schwierig zu analysieren. Sie sind bei jedem Eintreten zu protokollieren. Der Test soll jedoch fortgesetzt werden. Erst bei Überschreiten einer individuell festzuliegenden Auftretens-Häufigkeit liefert ein Versuch keine zusätzliche Aussage mehr und ist deshalb abzubrechen. Über die Angabe von “Ereignisse pro Zeit” kann der Anwender eine individuelle Konfiguration vornehmen.

- Abspeichern im Fehlerfalle: Zur zeitversetzten Analyse automatisch ablaufender Versuche ist es notwendig, bei Auftreten von Fehlern die Zeitverläufe der involvierten Größen abzuspeichern. Für jede Fehlerklasse kann deshalb “Abspeichern ja/nein” parametrisiert werden. Generell während aller Simulationen abzuspeichern ist nicht förderlich: Die wenigen, relevanten Ergebnisse gehen in der hohen Zahl von Datensätzen unter, es ist mit hohem Speicherbedarf verbunden und läuft einer konsequenten Archivierung und Backup-Strategie zuwider.

C-Code Interface

Das Modul zur Protokollierung und zum Fehlerhandling verfügt im wesentlichen über folgende Schnittstelle: Jede Fehlerklasse (error class) wird eindeutig über ihren Ident gekennzeichnet.

```

/* Error Classes */
enum {
    EC_General =    0,      /* general fault          */
    EC_Init =      1,      /* initialisation fault   */
    EC_Sim =       2,      /* simulation fault       */
    EC_CycleTime = 3,      /* cycletime exceeding    */
    ....
};

```

Initialisiert wird das Modul einmalig unter Angabe des Namens der Protokolldatei über den Aufruf der Funktion

```
LogInit (const char *LogFileName) .
```

Mit der Funktion

```

LogDefineErrClass (
    unsigned      ECId,
    tLogEAction   Action,
    const char    *ErrTxt,
    const int     SaveMode)

```

können weitere Fehlerklassen erzeugt werden. Über weitere Argumente wird das Default-Verhalten dieser Fehlerklasse konfiguriert: *Action* legt fest, wie bei Auftreten dieser Klasse zu verfahren ist, ob es sich um eine Warnung handelt oder einen Fehler, bei dem der Versuch beendet, sofort oder verzögert abgebrochen oder ob der Fehler unterdrückt werden soll. *ErrTxt* ist der Text, der ausgegeben werden soll, falls bei Auftreten des Fehlers keine Erläuterung angegeben wurde. Der Parameter *SaveMode* gibt an, ob Abspeichern von Zeitverläufen aktiviert werden soll oder nicht.

Fehler, Warnungen und Meldungen werden aus dem Simulationsprogramm heraus über die Funktionen

```

LogErrF (unsigned ECId, const char *format, ...)
LogWarnF (unsigned ECId, const char *format, ...)
Log      (const char *format, ...)

```

ausgegeben. Fehler und Warnungen sind jeweils einer Fehlerklasse *ECId* zugeordnet und können über einen erläuternden Text, der über Formatstring und variable Argumentliste individuell ausgeführt werden kann, verfügen.

Durch den Aufruf der Funktion

```
LogClose ()
```

wird die Protokollierung beendet und die Protokolldatei geschlossen.

3.3.2 Client-Server-Kommunikation zur Steuerung und Überwachung

Die Simulationsumgebung ist realisiert als ein modulares “Baukastensystem” mit in Client-Server-Architektur verteilten Aufgaben. Einzelne Komponenten (Anwendungen, Programme) bieten Informationen, Funktionalität oder allgemein Dienste an, sogenannte Server (deutsch: Diensteanbieter) oder Back-End-Komponenten. Ein Beispiel hierfür ist ein Simulationsprogramm, das ein virtuelles Fahrzeug bereitstellt, mit dem Fahrmanöver durchgeführt werden können. Andere Komponenten, sogenannte Clients (deutsch: Kunden) oder Front-Ends, greifen auf Server zurück. Dies ist z.B. eine Tachometeranzeige, die vom Simulations-Server die Information über die aktuelle Fahrzeuggeschwindigkeit bezieht und sie auf dem Bildschirm anzeigt, oder ein Bedienknopf, über den der Anwender einen Start-/Stop-Befehl an den Simulations-Server schickt. Die Verteilung

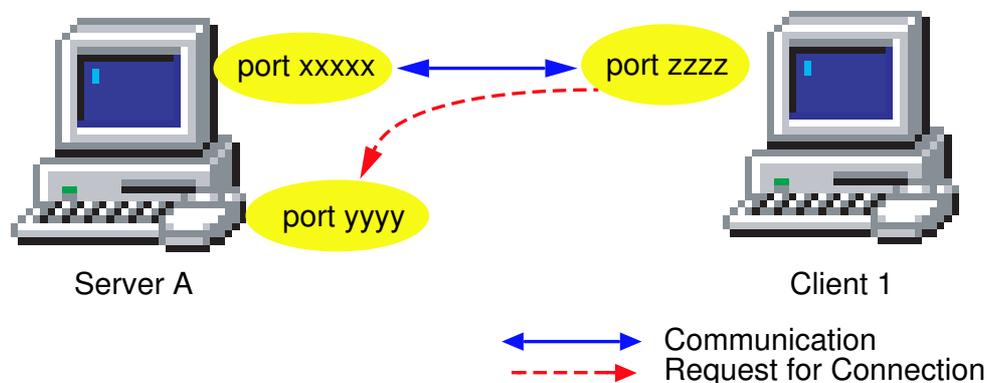


Abbildung 3.3.1: Client-Server-Kommunikation

beschränkt sich nicht auf verschiedene Anwendungsprogramme, sondern umfasst auch die Nutzung verteilter, z.B. über Netzwerk, gekoppelter Hardware. Bei einem Hardware-in-the-Loop-Prüfstand berechnet ein Echtzeitrechner die Simulationsmodelle und liest Input-Signale von und generiert Output-Signale für die angekoppelten Hardware-Komponenten. Der Testingenieur steuert und überwacht den Prüfstand von einem Bedienrechner mit Anzeigen, Eingabe-Oberflächen, Zeitverläufen und 3D-Animationen.

Eine Client-Server-Architektur vermeidet die Konzentration der gesamten Funktionalität an einer Stelle. Unterschiedliche Aufgaben werden von verteilten, untereinander gekoppelten Systemen wahrgenommen. Eine solche Systemarchitektur ist flexibel erweiterbar und durch neue Spezialkomponenten an besondere Anforderungen anpassbar. Existierende Komponenten müssen hierfür nicht modifiziert werden. Diese Vorteile gegenüber einer monolithischen Struktur sind mit zusätzlichen Anforderungen an anderer Stelle verbunden: Die Kopplung verschiedener Teilsysteme wird lockerer. Ein Kommunikationsprotokoll wird notwendig. Ein Client muss einen Server suchen, finden und mit ihm Kontakt aufnehmen können. Der Client muss Befehle an den Server senden und Statusinformationen vom Server empfangen. Bei der Kommunikation entstehen zwischen

Befehl und Antwort Verzögerungen.

Die Umgebung verfügt über ein eigenes, von IPG Automotive GmbH entwickeltes und speziell auf ihre Anforderungen zugeschnittenes Kommunikationsprotokoll. Entsprechende Bibliotheken stehen sowohl als C-Schnittstelle für Anwendungsprogrammierung als auch für die bei der Testautomatisierung eingesetzte Scriptsprache zur Verfügung.

Jeder Server wird über eine eindeutige Kennung, über Bezeichnung und Klassifizierung der Anwendung (z.B. "Steuergerät 0815/HIL", "Wankregler/SL", "Fahrzeug abc/ESP xyz"), den Rechnernamen und die Benutzer-Kennung, mit der der Server läuft und weitere Merkmale identifiziert. Ein Client fragt nach einer Liste der vorhandenen Server an. Aus dieser Liste wählt er den passenden Server aus und nimmt Verbindung zu ihm auf. Die Kommunikation zwischen Client und Server findet in zweierlei Modi statt:

1. **Gesicherte Verbindung:** Der Sender erhält den Empfang seiner versandten Botschaft bestätigt bzw. er erfährt, dass der Versuch ihrer Zustellung erfolglos war. Dieser Kommunikationsmodus wird für Steuerbefehle und für Anfrage-Antwort-Dialoge eingesetzt.
2. **Ungesicherte Verbindung:** Es ist für den Sender nicht notwendig, dass alle versandten Botschaften beim Gegenüber auch ankommen und empfangen werden. Für den Empfänger gilt im umgekehrten Sinne das Gleiche. Kurzzeitige Unterbrechungen oder Verluste, z.B. weil der Empfänger gerade mit anderen Aufgaben beschäftigt ist, sind zulässig. Dieser Modus wird verwendet bei der kontinuierlichen Übermittlung von Zustandsvektoren an Anzeigen und Plott- oder Animationsprogramme.

Über Zugriffsbeschränkungen und Zugriffsrechte wird sichergestellt, dass auf einen Server nur privilegierte Clients zugreifen können, um ihn zu überwachen und zu steuern.

Beispiel: Server-Anwendung (C-Code Interface)

Serverseitig initialisiert `AposInit()` die Kommunikation.

Signale werden per `AposDeclare<DataType>()` bekanntgegeben. Ist der Server bereit, schließt `AposReady()` die Vorbereitungsphase ab und gibt den Server für Clients frei. Während der Simulation können Botschaften per `AposSendMsg()` versendet werden. Eintreffende Botschaften werden mit `AposRecvMsg()` beschafft und ausgewertet. Die Kommunikation mit den Clients und der ungesicherte Datenaustausch wird von `AposPoll()` abgewickelt.

```

/** preparation */
AposInit (ServerClass, 0, ChannelMask);
AposDeclareDouble ("Car.ax", "m/s^2", &Car.ax);
AposReady();

/** simulation loop */
while (Simulating) {
    /* saved connection */
    AposSendMsg(...);
    while (AposRecvMsg(channel, ...) == 0) {
        ...
    }
    /* unsaved connection */
    AposPoll();
}
/** clean up, finish */
AposExit();

```

Beispiel: Client-Anwendung (Tcl-Code Interface)

Verfügbare Server werden ermittelt, der passende ausgewählt und die Verbindung mit ihm aufgebaut. Der Client abonniert vom Server per ungesicherter Verbindung die in der Subscriptions-Liste aufgeführten Signale. Die Funktion `HandleDataVec` nimmt die hierfür eintreffenden Datenvektoren entgegen, wertet sie aus und weist sie an die Client-Datenstruktur zu. Für den Datenkanal 2 wird die Auswertefunktion `RecvMsg_Channel2` eingerichtet. Eine Nachricht wird mit dem gesicherten Befehl `sendmsg` an den Server verschickt.

```

# connect
apoc query 1000
SelectServer [apoc servers]
dataprocess dictionary $Server
dataprocess subscribe $Server 0 100 HandleDataVec $Subscriptions
dataprocess tracemsg $Server 2 RecvMsg_Channel2

# working
dataprocess sendmsg $Server $Channel "K115 on"

# clean up, finish
dataprocess delete $Server

```

3.3.3 Modell-Management – Varianten und Alternativen

Im Zentrum der virtuellen Modelle der Simulationsumgebung steht das Fahrzeugmodell. Es übernimmt die Rolle einer integrativen Plattform, die entsprechend der Projektanforderungen angepasst und erweitert werden kann.

Während des Entwicklungsprozesses wird das Wissen über die einzelnen beteiligten Baugruppen kontinuierlich immer weiter präzisiert. Ein Parameter, der anfangs einer Konzeptstudie entstammt, wird später aus Messungen an realen Prototypen oder aus detaillierten FEM-Modellen gewonnen. Auch die Modellierung selbst wird entsprechend dem aktuellen Wissensstand verfeinert werden. Vergleiche mit (einfacheren) Vorgängermodellen dienen der Kontrolle und der Beurteilung möglicher Veränderungen im Systemverhalten.

Die Variantenvielfalt moderner Fahrzeuge ist sehr hoch. Verschiedene Motorisierungen, Benzin- oder Dieselmotor, Handschalt- oder Automatikgetriebe, Sportfahrwerk mit geänderter Bremsanlage, Klimaanlage, Navigationssystem, um nur einige zu nennen, kann der Käufer wählen. Diese verschiedenen Konfigurationen mit der Simulation abdecken zu können, erfordert ein ausgereiftes geeignetes Konzept. Für jede Konfiguration ein eigenes Simulationsprogramm aufzubauen ist keine befriedigende und praxistaugliche Lösung. Gefragt ist die Möglichkeit, automatisiert, aus Versuchsreihen heraus, zwischen verschiedenen Modellkonfigurationen wechseln zu können. Diese Möglichkeit darf sich nicht nur auf die Standard-Modelle beschränken. Sie muss auch für die vom Anwender selbst beigestellten Modelle verfügbar sein.

Gerade im Bereich Hardware-in-the-Loop-Simulation steht immer nur eine beschränkte Rechenleistung zur Verfügung. Mit dieser müssen alle Modelle und anfallenden Aufgaben realisiert werden. Zwischen detaillierteren Modellen im Fokus der aktuellen Fragestellung und vereinfachten an deren Peripherie wechseln zu können, stellt einen Ausweg aus dieser Situation dar. Beispiel: Das Hydraulik-Modell einer Bremsanlage mit Ventilen, Bremszylindern, Drücken und Volumenströmen zum Betrieb eines ABS- oder ESP-Steuergerätes; ein einfaches Modell zur Bremsmomentenverteilung abhängig von der Bremspedalkraft im Zusammenhang mit einem Kurvenlicht-Regler.

Die Simulationsumgebung bietet für diese Anforderung eigens ein Modul zum Modell-Management an, das folgende Standard-Submodell-Klassen des Fahrzeug-Kernmodells verwaltet: Reifen, Fahrwerk, Lenksystem, Bremsanlage, Antriebsstrang, Motormodell, Kupplung, Getriebe, Triebstrang, incl. Differentialsperre und Hang-On-Kupplung, Aerodynamik. Für Details zu einzelnen Modellen sei auf Abschnitt 3.5 verwiesen.

Unabhängig von seiner Modell-Klasse kann oder muss jedes Modell über folgende Funktionalität verfügen:

- **Initialisierung:** Beim Start einer Simulation wird eine Instanz des Modells erzeugt. Sie wird entsprechend des gewählten Datensatzes parametrisiert und für die eigentliche Simulation vorbereitet.
- **Berechnung:** Während der Zeitsimulation ruft das Modell seine Eingangsgrößen ab, führt die Berechnung durch und liefert seine Ausgangsgrößen zurück.

- **Freigabe:** Nach der Simulation wird die eigens für diese erzeugte Instanz und alle dynamisch angeforderten Ressourcen, wie z.B. Hauptspeicher, wieder freigegeben.
- **Signale zur Beobachtung:** Ein Modell kann eigene Signale in das Signal-Verzeichnis eintragen und so dem Anwender Einblick in interne, modellspezifische Größen bieten (siehe Abschnitt 3.4.1).
- **Registrierung:** Damit die Simulationsumgebung ein Modell “kennt”, muss es “bekannt” gemacht worden sein. Zusammen mit einer eindeutigen Kennung, über die das Modell vom Anwender später angesprochen wird, werden obige Schnittstellenfunktionen an das Modell-Management-Modul übergeben.

Durch das Modell-Management-Modul können im Simulationsprogramm beliebig viele Modelle einer Modell-Klasse parallel vorgehalten werden. Der Anwender spezifiziert das Modell, das er bei einer Simulation verwenden möchte, einfach über die Angabe dessen Kennung im Parameterfile. Das Modell-Management wählt über diese Kennung und die Modell-Klasse das zugehörige Modell aus und aktiviert dessen Initialisierung, Signal-Deklaration, Berechnung und abschließend wieder dessen Freigabe.

Das Modell-Management-Modul bietet große Flexibilität. Dadurch, dass die an zentraler Stelle gepflegte und weiterentwickelte Standard-Bibliothek für den Kern der Fahrzeug-Plattform in allen Anwendungen zum Einsatz kommt, steigt deren Qualität und Zuverlässigkeit. Weiterentwicklungen stehen automatisch allen Anwendern zur Verfügung.

Das Modell-Management ist Bestandteil des Fahrzeugmodells. Sein Einsatz zum Austausch des Fahrzeugmodells selbst ist nicht vorgesehen. Die Berührungspunkte mit der restlichen Simulationsumgebung sind zu umfangreich, als dass ein Umschalten auf ein anderes Fahrzeugmodell mit vertretbarem Aufwand realisierbar wäre.

3.3.4 Einbinden von Modellen als Simulink S-Function

Matlab/Simulink ist vor allem im Umfeld der Reglerentwicklung eine weit verbreitete Entwicklungsumgebung. Simulationsmodelle können direkt im Simulink-Blockschaltbild aufgebaut werden. Im Rahmen des “Concurrent Engineering” werden häufig Entwicklungsstände abgeglichen und notwendige Simulationsmodelle unter den beteiligten Entwicklungspartnern ausgetauscht. Hierfür werden Modelle als ganzes oder einzelne Funktionsblöcke aus der eigenen Simulationsumgebung herausgelöst und – versehen mit geeigneten Schnittstellen – für eine andere Umgebung exportiert. Dies geschieht durch den direkten Austausch des Simulink-Modellblocks, dem Quellcode des Modells. Seine interne Struktur, seine Implementierung und Parametrierung ist in diesem Fall offengelegt. Ist der damit verbundene Know-How-Transfer nicht erwünscht, erfolgt der Export als Black-Box-Funktionsmodul in Form einer “Simulink-Funktion” oder “S-Function”. Eine S-Function kann in zwei Arten vorliegen:

- **binäre, ausführbare Modellbibliothek (DLL):** Das Modell lässt sich direkt als S-Function-Block in ein Simulink-Modell einbinden.

- über den RealTimeWorkshop als C-Quellcode exportiert: Diese Form des Modellaustauschs wird gewählt, wenn es sich bei der Ziel-Umgebung um eine von der eigenen abweichende Umgebung handelt. Ein anderes Betriebssystem oder eine Prozessor-Architektur mit unterschiedlichem Binärformat machen eine spezielle Entwicklungsumgebung notwendig. Meist steht sie und entsprechendes Know-How nur auf Seiten des Modell-Imports zur Verfügung. Der Weg über C-Quellcode ermöglicht es dem Importierenden Parameter, wie z.B. die Compiler-Einstellungen, selbst zu wählen. Ein typisches Beispiel hierfür ist der Modellexport von einem Windows-PC für einen mit dem Betriebssystem LynxOS laufenden Echtzeit-Rechner eines Hardware-in-the-Loop-Simulators.

Über bei IPG Automotive GmbH entwickelte speziell auf die Entwicklungsumgebung zugeschnittene Export-“Targets” gestaltet sich der Modell-Import in diese Umgebung besonders einfach: Die Initialisierungs-, Berechnungs- und Freigabe-Funktionen werden automatisch mit den passenden Schnittstellen und Verhalten erzeugt. Die Simulink Block-Library wurde um Funktions-Blöcke erweitert. Über sie wird im Simulink-Modell der Zugriff auf Funktionalität der Entwicklungsumgebung ermöglicht. Z.B. “messen” Kinematik-Sensor-Blöcke im Fahrzeug für beliebige Punkte Position, Geschwindigkeit und Beschleunigung. Mit sog. Dictionary-Blöcken werden Simulink-Signale in das Signal-Verzeichnis eingetragen (siehe Abschnitt 3.4.1) und damit für das Abspeicher-Modul (siehe Abschnitt 3.4.2) und das Manipulationsmodul (siehe Abschnitt 3.4.3) zugänglich gemacht. Über spezielle Read/Write-Blöcke ist der lesende und schreibende Zugriff direkt auf Variablen der Umgebung realisiert.

3.4 Funktionsmodule der Entwicklungsumgebung: Durchführung von Tests

3.4.1 Signal-/Größen-Verzeichnis

Für den interaktiven oder den automatisierten Simulationsbetrieb ist es wichtig, das Simulationsgeschehen “online” überwachen zu können. “Online” meint in diesem Zusammenhang “schon während der laufenden Simulation” die Zeitverläufe repräsentativer Größen zu verfolgen, anstatt sich nach Simulationseende abgespeicherte Ergebnisdaten anzusehen.

Die vorliegende Simulationsumgebung ist komplex und verfügt aus ihren verschiedenen Teilmodellen und Steuergeräten über sehr viele – häufig sind es mehrere hundert – Signale. Von Anwendung zu Anwendung oder abhängig von der aktuellen Konfiguration der Simulationsumgebung stehen unterschiedliche Signale zur Verfügung.

Damit auf die Signale zugegriffen werden kann, werden sie in einem zentralen Verzeichnis zusammengefasst und verwaltet. In Abbildung 3.4.1 sind einige Einträge in einem solchen Verzeichnis beispielhaft wiedergegeben. Jedes Modell trägt dort bei der Initialisierung die von ihm bereitgestellten Signale mit folgenden Angaben ein:

Ambient.AirPressure	bar	8	double	0	0	0x81aa108
Ambient.Temperature	K	8	double	0	0	0x81aa0f8
Brake.FPedal	N	8	double	0	0	0x821f444
Brake.pMC	bar	8	double	0	0	0x821f4e8
Brake.pWB_FL	bar	8	double	0	0	0x821f4f0
Car.SimPhase		4	long	12	0	0x8223e44
Car.Yaw	rad	4	float	0	0	0x8223f20
Car.YawRate	rad/s	4	float	0	0	0x8223f24
Car.YawAcc	rad/s ²	4	float	0	0	0x8223f28
Car.alHori	m/s ²	4	float	0	0	0x8223f0c
Car.atHori	m/s ²	4	float	0	0	0x8223f08
Car.v	m/s	4	float	0	0	0x8223f00
DM.Gas		4	float	0	0	0x8228fcc
DM.GearNo		4	long	0	0	0x8228fe8
PT.Engine.rotv	rad/s	8	double	0	0	0x82245e8
SC.State		4	long	14	0	0x82296c0
Steer.WhlAng	rad	8	double	0	0	0x8224204
Steer.WhlTrq	Nm	8	double	0	0	0x822421c
Vhcl.sRoad	m	8	double	0	0	0x8229920

Abbildung 3.4.1: Ausschnitt aus dem Signal-/Größen-Verzeichnis eines Simulationsprogramms. Dargestellt sind Name, Einheit, Größe in Byte und Typ, Anzahl möglicher Zustände, Speicheradresse etc.

- Name: Eine sprechende, lesbare Bezeichnung, über die der Anwender auf das Signal zugreift. Dieser Realisierung wurde der Vorzug gegenüber fehlerträchtigen und in der Handhabung unkomfortablen Indices innerhalb einer Tabelle gegeben.
- Einheit: Der Zahlenwert bezieht sich auf diese Einheit. Fast alle Signale liegen programmintern in SI-Einheiten vor.
- Datentyp: Die einem Signal zugehörige Variable hat einen bestimmten, vom Modellentwickler gewählten, Datentyp. Z.B. wird für die Gangnummer ein Integer verwendet und für die Gangübersetzung ein Float oder Double. Damit die Signale korrekt interpretiert und übertragen werden können, muss ihr Datentyp bekannt sein.
- Adresse: Adresse im Hauptspeicher des Computers, an der die Variable liegt und von der ihr Wert gelesen werden kann.
- Zusätzliche Angaben: Anzahl möglicher Zustände bei zählbaren Größen, Monotonität, u.a.

Eine Client-Anwendung, z.B. ein Programm zur Darstellung von Zeitverläufen, kann sich jetzt über Netzwerk mit dem Simulations-Server verbinden (siehe Abschnitt 3.3.2) und eine Liste aller aktuell verfügbaren Signale anfordern. Der Simulations-Ingenieur

wählt sich dann aus dieser Übersicht die Signale aus, die für ihn bei der aktuellen Simulation von Interesse sind. Steht in der aktuellen Versuchskonfiguration ein Signal gerade nicht zur Verfügung, kann meist auf eine ähnliche Größe ausgewichen werden. Bei der Testautomatisierung werden alternative Signale angegeben. Nur wenn auch diese nicht verfügbar sind, wird eine Fehlermeldung ausgelöst.

Vom Simulations-Server zum verbundenen Client werden nur die ausgewählten Größen übertragen. Hierdurch wird die verfügbare Rechenleistung und die Übertragungskapazität der Netzwerkverbindung effektiv genutzt.

Das Signal-Verzeichnis schafft die Voraussetzung zu weiterer Funktionalität, wie dem Abspeichern (siehe Abschnitt 3.4.2) oder dem Manipulieren von Signalen (siehe Abschnitt 3.4.3).

3.4.2 Aufzeichnen von Signalen

Beim Versuchs- und Simulationsbetrieb besitzt das Aufzeichnen und Abspeichern von Signalen und der Vergleich zweier oder mehrerer Tests einen großen Stellenwert. Zeitverläufe von Signalen werden deshalb zur späteren detaillierten Analyse, Weiterverarbeitung und Archivierung festgehalten.

Im Gegensatz zur Messung im Fahrversuch stehen bei Simulationsmodellen sehr viele Signale meist automatisch zur Verfügung, ohne dass aufwändige Messtechnik installiert und kalibriert werden muss. Das Erfassen und Abspeichern von Signalen verbraucht während der Simulation Rechenzeit und anschließend Speicherkapazität auf der Festplatte. Die Aufgabe lautet hier deshalb, aus der Vielzahl zur Verfügung stehender Signale diejenigen auszuwählen, die für den aktuellen Arbeitsschwerpunkt relevant und notwendig sind. Hierfür wurde ein leistungsfähiges Konzept in Form eines Aufzeichnungs-Moduls realisiert.

Vor Beginn eines Versuchs konfiguriert der Anwender das Aufzeichnungs-Modul. Die aufzuzeichnenden Signale werden über ihren Namen spezifiziert. Zusammenstellungen von Namen lassen sich in Datenbasen ablegen und referenzieren. So können z.B. für jeden Arbeitsschwerpunkt einheitlich die gleichen Signale erfasst werden. Die Datenrate wird festgelegt, d.h. mit welcher Zeitschrittweite bzw. jeden wievielten Simulationszyklus ein Datenvektor aufzuzeichnen ist. Über das Signal-Verzeichnis (siehe Abschnitt 3.4.1) gelangt das Aufzeichnungsmodul an die einzelnen Variablen und ihre Werte.

Das Abspeichern über Netzwerk auf eine Festplatte ist nicht echtzeitfähig. Für Hardware-in-the-Loop-Anwendungen muss jedoch Echtzeitfähigkeit garantiert werden. Deshalb werden in der Echtzeitanwendung die Datenvektoren in einen mehrere Megabyte großen Puffer im Hauptspeicher des Simulationsrechners geschrieben. Ein zweiter Prozess oder Thread übernimmt die Daten aus dem Puffer und wickelt unter Nicht-Echtzeit-Bedingungen, mit niedrigerer Priorität als der eigentliche Simulationsprozess, das Übertragen und Abspeichern der Daten ab.

Von Beginn bis zum Simulationsende alles abzuspeichern ist auch ohne einen Datenpuffer möglich. Mit Hilfe des Datenpuffers sind zusätzliche, sehr nützliche Abspeicher-Modi realisiert: Rückwirkendes Abspeichern zuvor gepufferter Datenvektoren. Während der

Simulation werden die Datenvektoren nur gepuffert. Der Anwender sendet von einer Bedienoberfläche aus per Client-Server-Kommunikation (siehe Abschnitt 3.3.2) den Befehl, mit dem Abspeichern zu beginnen. Es kann entweder die verfügbare Historie komplett oder anteilig mit ausgegeben werden. Über Ereignisse (siehe Abschnitt 3.3.1) getriggertes Abspeichern incl. eines Zeitabschnitts vor Auftreten des Ereignisses ist möglich.

Die Ergebnisdatensätze tragen automatisiert den Namen des Versuchs, bei dem sie aufgezeichnet werden. Durch Datum und Uhrzeit des Versuchsbeginns und einer fortlaufenden Nummer für jedes Abspeichern innerhalb des Versuchs sind sie eindeutig bezeichnet. Die Aufzeichnung selbst erfolgt in einem kompakten Binärformat. Importfilter für den Matlab-Workspace, für Excel und IPG-CONTROL sind verfügbar. Importiert werden die Namen der Signale, ihre Einheit und die Anzahl ihrerer Zustände. Andere Postprocessing-Werkzeuge werden über einen Exportfilter für ein ASCII-Tabellenformat unterstützt.

3.4.3 Manipulation von Simulations-Größen und elektrischen Signalen

Die virtuellen Modelle werden durch Eingriffe eines Fahrers oder andere Eingriffe bedient. Hierfür werden üblicherweise vorab Versuche definiert, die anschließend von der virtuellen Umgebung geladen und durchgeführt werden. Dem gegenüber steht eine interaktive Bedienung oder eine Bedienung aus der Testautomatisierung heraus während eines laufenden Versuchs. Das Simulationsgeschehen wird beobachtet und zum gewünschten Zeitpunkt oder Zustand eines Modells wird in das Simulationsgeschehen eingegriffen. Hierbei handelt es sich z.B. darum, das Gaspedal voll durchzutreten, das Lenkrad loszulassen oder die Fahrwerksabstimmung von Sport auf Komfort umzustellen. Ein zweites, für die Reglerentwicklung besonders wichtiges Aufgabenfeld ist die Manipulation von Sensor-Signalen. Signale können durch Wackelkontakt oder Leitungsbruch gestört werden. Sie können wegdriften oder ein Sensor kann seine Empfindlichkeit ändern. Zur softwaretechnischen Manipulation von Signalen wurde ein DirectVariableAccess Modul (DVA) realisiert. Es bietet folgende Möglichkeiten:

- Zugriff auf Signale über ihren Namen, realisiert über das Signal-Verzeichnis (siehe Abschnitt 3.4.1)
- Manipulation der Signale an verschiedenen vordefinierten Positionen innerhalb eines Simulations-Schrittes: Nach Einlesen von Input-Signalen, nach der Manöversteuerung und dem Fahrermodul, nach Berechnung des Fahrzeugmodells, vor der Ausgabe der Signale an die Steuergeräte, u.a.
- Zeitdauer der Manipulation: für eine bestimmte Zeitdauer oder unbegrenzt
- Arten der Manipulation: absoluter Wert oder relativer Offset, Skalierungsfaktor, konstante oder anwachsende Manipulation
- Freigabe aller aktiven Manipulationen

Client-Anwendung können per Online-Kommunikation (siehe Abschnitt 3.3.2) diese Abfragen und Manipulationen vornehmen. Dies sind z.B. eine Bedienoberfläche mit Tastern oder die Testautomatisierung. In der Manöversteuerung stehen sie als Erweiterung der Funktionalität der Mini-Manöver (siehe Abschnitt 3.4.6) zur Verfügung. Darüber hinaus stellt dieses Modul ein gutes Werkzeug bei der Modellentwicklung und Inbetriebnahme von Prüfständen dar. Mit ihm lassen sich einfach gewünschte Konstellationen von Signalen setzen, wie es z.B. zur Kalibrierung von Sensorsignalen für ein Steuergerät hilfreich ist.



Abbildung 3.4.2: FailSafeTester zur elektrischen Manipulation von Signalen. Ausführung mit vertieftem Einbau oder normal, verschiedene Einsteckkarten (Quelle: IPG Automotive GmbH)

Zur elektrischen Manipulation von Signalen auf Hardware-in-the-Loop-Prüfständen wurde der von IPG Automotive GmbH entwickelte FailSafeTester in die Simulationsumgebung integriert [WH03]. Er kann mit unterschiedlichen Karten bestückt werden (siehe Abbildung 3.4.2):

- Standard- oder Hochstrom-Relais-Karten zur Erzeugung von Leitungsbrüchen durch Trennen oder elektrischen Kurzschlüssen durch untereinander Verschalten von Signalen
- Widerstandskarten zur Simulation von Leitungs-Alterung und Korrosion von Kontakten durch Einbauen von programmierbaren Widerständen in einzelne Leitungsstränge
- Wackelkontakt-Karten zur hochfrequenten Unterbrechung von Signalen

Der FailSafeTester ist über CAN-Bus an den Hardware-in-the-Loop-Simulator angeschlossen. Bedient werden kann er entweder interaktiv über eine eigene Bedienoberfläche, über Befehle aus den Manövern eines Versuches oder über die Testautomatisierung per Kommunikations-Botschaften.

3.4.4 Datenmanagement

Für alle Bereiche der Simulation – Modelle, Simulationsumgebung, Visualisierung – sind Daten notwendig. Sie besitzen eine hierarchische Struktur, bei der ein Datensatz auf Subdatensätze zurückgreift. Die einzelnen Datensätze haben unterschiedliche Formate und Umfänge. Beispiele: Ein Fahrzeug-Modell erfordert Angaben zu Geometrie, Massen und Trägheiten, Steifigkeitskennlinien, Reibwerte, Zeitverhalten, Hydraulik, Gangübersetzung; eine Simulationsumgebung muss konfiguriert und Signale konditioniert werden; für die Visualisierung sind 3D Grafikmodelle notwendig. Für den Anwender ist es wichtig, die für ihn relevanten Daten und Informationen einfach aufzufinden und zu verwalten. Der Datenaustausch zwischen verschiedenen Rechner- und Betriebssystemplattformen muss vor allem vor dem Hintergrund der Hardware-in-the-Loop-Simulation und der abteilungs- und firmenübergreifenden Entwicklungsprozesse möglich sein. Ein einheitliches, plattformübergreifendes Datenformat ist erforderlich.

Für das Datenmanagement ist eine über Verzeichnisse und Dateien realisierte Datenbank implementiert. Die Daten selbst werden in einem lesbaren ASCII-Dateiformat abgelegt. Unterstützt werden die Datentypen Double, Long, String und Text. Kennlinien und Kennfelder werden mit den Typen String oder Text abgespeichert. Das Dateiformat ist plattformunabhängig. Eine Kontrolle der Datenbestände mit einem normalen Texteditor ist möglich. Die Datensätze können flexibel aufgebaut sein, da schlüsselwortorientiert zugegriffen wird. Die hierarchische Strukturierung der Daten kann über ebenenbildende Punkte im Schlüsselnamen erfolgen – analog zu Strukturen in der Programmiersprache C. Durch die Verwendung eines einheitlichen Formates für alle Datensätze, kombiniert mit dem schlüsselwortorientierten Zugriff und der Definition von Standard-Schlüsseln zur Kennzeichnung und Beschreibung eines Datensatzes, können dem Anwender einheitliche

Dialoge zur Suche und Auswahl der Datensätze angeboten werden. Schnittstellen zum lesenden und schreibenden Zugriff auf die Datenbank stehen für Bedienoberflächen, für Anwendungsprogramme und für die Testautomatisierung zur Verfügung. In der Bedienoberfläche wurden hiermit Importfunktionen realisiert, z.B. für die Minimanöver, die Fahrercharakterisierung, die Fahrbahn etc.

Modellparameter Zur Organisation der Daten wurde ein kombinierter Ansatz gewählt. Die einzelnen Modelle werden modulbasiert, d.h. an Bauteilen oder Baugruppen und funktionalen Einheiten orientiert parametrisiert. Dies ermöglicht es, sehr viele Daten aus bereits vorhandenen Bauteil-Datenbanken direkt zu übernehmen. Aus den einzelnen Modulen werden anwendungsorientierte Einheiten gebildet. So werden z.B. die Daten zur Fahrzeug-Geometrie und Massenverteilung, zum Lenksystem, zum Antriebsstrang etc. in einem Fahrzeugdatensatz abgelegt. Nachteilig ist hierbei eine Redundanz in den Daten. Sie ermöglicht es jedoch, mit einer geringen Anzahl von Datensätzen insgesamt auszukommen, die mit wenigen Bedienschritten ausgewählt und damit simuliert werden kann. Es besteht die Möglichkeit, einzelne Subdatensätze, wie z.B. den kompletten Antriebsstrang oder die Bremsanlage aus dem Fahrzeugdatensatz herauszulösen und ihn separat abzuspeichern. Im Fahrzeugdatensatz wird dann eine Referenz auf diesen Datensatz eingetragen. Folgende Klassen von Datensätzen sind bekannt: Versuch, Fahrzeug, Reifen, Fahrwerkskinematik und Elastokinematik, Anhänger, digitalisierte Fahrbahn, Simulationsumgebung, Sonstiges u.a.

Da die Daten in einer Datenbank mit standardisierter Schnittstelle abgelegt sind, können elegant Modell-Varianten konfiguriert und Parameterstudien durchgeführt werden. Gleiches gilt auch für Versuche, Manöver und Testszenarien.

Versuche und Testszenarien Der Versuchsdatensatz ist dasjenige, welches an das Simulationsprogramm zur Durchführung übergeben wird. Ein Versuchsdatensatz enthält zum einen Referenzen auf andere, aus seiner Sicht abgeschlossene, konstante Einheiten. Dies sind das Fahrzeug, dessen Bereifung, der Anhänger – falls gewünscht –, vermessene Fahrstrecken und vermessene Zeitverläufe zur Stimulation (siehe Abschnitt 3.4.7). Direkt im Versuchsdatensatz abgelegt sind alle anderen Angaben, die speziell diesem Versuch zugehörig sind. Dies sind die Bedingungen auf dem Testgelände, auf dem der Versuch stattfindet, der zu befahrende Kurs, die Beladung des Fahrzeugs, das durchzuführende Fahrmanöver incl. Fahrercharakteristik, weitere Bedienaktionen wie z.B. das Stören von Signalen, das Verhalten bei Auftreten von (Fehler-)Zuständen oder die konkrete Zuordnung von Stimulationsgrößen.

3.4.5 Fahrbahn und Umwelt

Virtuelle Modelle benötigen genauso wie reale eine Umgebung, in der sie sich bewegen und agieren können. Für das Fahrzeug ist dies in erster Linie die Fahrbahn. Auf ihrer Oberfläche stützt es sich mit den Reifen ab und der Fahrer versucht ihrem Verlauf zu folgen. Hinzu kommen Umweltbedingungen wie Temperatur, Luftdruck, Wind und

weitere strecken- und/oder zeitabhängige Ereignisse. Bei IPG Automotive GmbH wurde hierfür das Modul IPG-ROAD entwickelt.

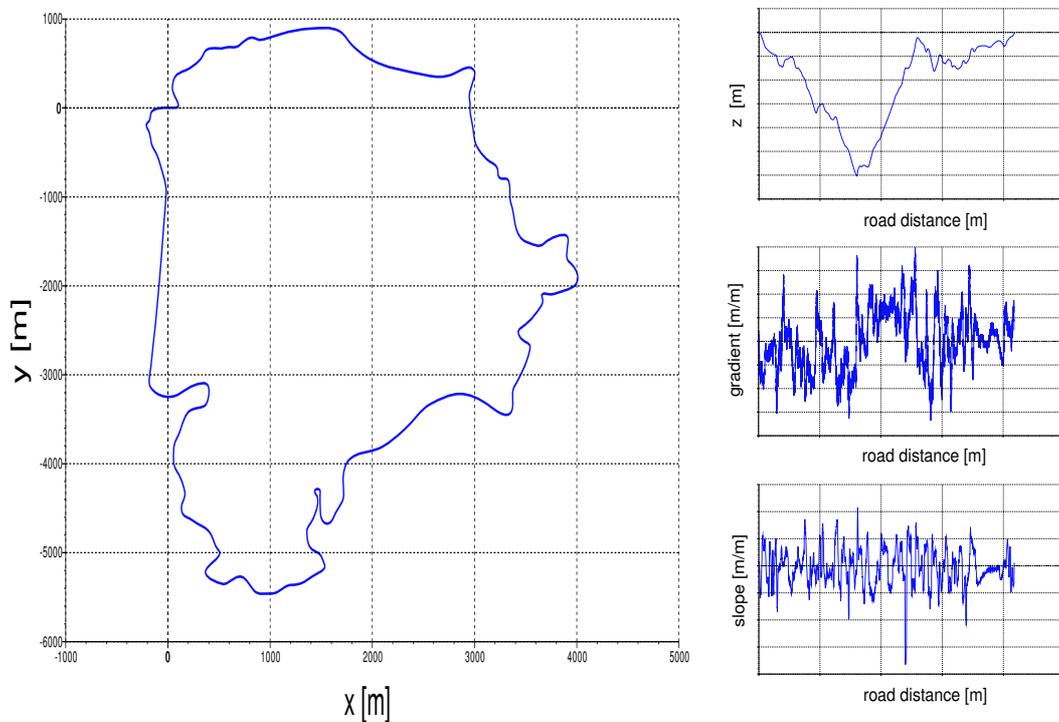


Abbildung 3.4.3: Umwelt- und Fahrbahnmodell: Digitalisierter Kurs der Nürburgring Nordschleife

Für die Entwicklung von Fahrdynamikreglern ist es wichtig, in der Simulation die gleichen Strecken befahren zu können wie im Fahrversuch. Es werden hierfür reale Strecken, wie z.B. die Nürburgring Nordschleife und das Stilfser Joch, von Messfahrzeugen, die mit GPS und Kreiselpattform bestückt sind, vermessen. Aufgezeichnet wird hierbei die befahrene Fahrspur über den x, y -Streckenverlauf, die Steigung, die Querneigung und die Spurbreite. Mit diesen Werten lässt sich das Fahrbahnmodell im Modus “Digitalisierte Strasse” parametrieren. Im Modus “Synthetischer Kurs” wird der Fahrbahnverlauf mit aneinander gereihten Segmenten der Grundtypen Gerade, Kurve, Klothoide aufgebaut. Segmente haben Länge und Querausdehnung, Steigung, Querneigung. Über ihr Querprofil lassen sich Spurrillen und gewölbte Fahrbahnen modellieren. Differenzen in der Steigung und dem Querprofil zweier Segmente werden in einem Übergangsbereich aneinander angeglichen. Schlaglöcher, Querfugen, Schwellen etc. werden als diskrete Einzelhinder-nisse der zuvor definierten Fahrspur überlagert. Gleiches gilt für Bereiche mit speziellem Reibwert.

Zugriff über absolute Koordinaten Reifenmodelle greifen über absolute Koordinaten wahlfrei auf das Fahrbahnmodell zu. Sie erfragen für eine bestimmte Position den Zustand der Fahrbahnoberfläche, d.h. die z -Koordinate, den Normalenvektor bzw. die Steigungen in x - und y -Richtung und die Oberflächenbeschaffenheit in Form eines Reibwertes.

Zugriff über die Bahnkoordinate Entlang der Fahrbahn lassen sich Trigger und Aktionen definieren. Dies sind z.B. Geschwindigkeitsbeschränkungen, zu durchfahrende Pylonengassen oder Slaloms, Abschnitte mit Windmaschinen oder Zeitmesspunkte. Über Fahrmanöver-Trigger wird z.B. der Bremsbeginn ausgelöst (siehe Abschnitt 3.4.6), oder es werden über einen Signalmanipulator-Trigger Störungen aufgeschaltet, wie z.B. das Durchtrennen der Anschlüsse eines Drehzahlfühlers mitten in einer Kurve (siehe Abschnitt 3.4.3). Die vorhandenen Aktionen können vom Anwender durch neue, selbst definierte ergänzt werden. Über die Bahnkoordinate stehen außerdem Temperatur, Luftdruck, Luftfeuchtigkeit, Windverhältnisse und der Fahrbahnzustand zur Verfügung.

Die Schnittstelle zum Animationspaket IPG-MOVIE generiert automatisiert aus der Fahrbahn- und Umweltdefinition des aktuellen Versuchs eine dreidimensionale Grafik der Umgebung mit dem Fahrbahnverlauf, Schildern, Windmaschinen etc. Für eine gute Beurteilung und Kontrolle der ablaufenden Simulation ist diese realistische Visualisierung sehr hilfreich. Für Präsentationen können diese Animationen als Filme (".avi"-Dateien) exportiert werden.

3.4.6 Fahrmanöver, Fahrer, Fahrzeugbedienung

Grundlage eines Versuchs bildet die Versuchsbeschreibung. In ihr ist festgelegt, mit welcher Fahrzeugkonfiguration (Fahrzeug, Bereifung, Beladung etc.) auf welcher Teststrecke welche Fahrmanöver und Bedienabläufe durchzuführen sind. Im realen Fahrversuch ist es Aufgabe des Testfahrers, diese Vorgaben umzusetzen. In der virtuellen Simulationsumgebung tritt an dessen Stelle das Modul der Manöversteuerung.

Strukturierung: Jeder Versuch wird über die Zeitdauer und/oder die zurückgelegte Fahrstrecke beschränkt. Komplexe Vorgaben werden in überschaubare, klar definierte Einzelschritte, sogenannte Minimanöver, untergliedert. Das einzelne Minimanöver ist über Zeitdauer und/oder Weg beschränkt. Es setzt sich zusammen aus den Komponenten

- Längsdynamik: Geschwindigkeit, Beschleunigen und Verzögern
 - Manuelle Bedienung von Gas-, Brems-, Kupplungspedal, Gangnummer bzw. Gangwahlhebel und Handbremse: Startzeitpunkt der Betätigung, Endwert absolut oder relativ, Dauer zum Erreichen des Endwertes
 - Geschwindigkeitsregelung: Wunschgeschwindigkeit, Toleranz, maximale Beschleunigungen
 - Beschleunigungsregelung: Wunschbeschleunigung
 - Fahrermodell IPG-DRIVER: Befahren eines Kurses entsprechend der Vorgaben [RW03]

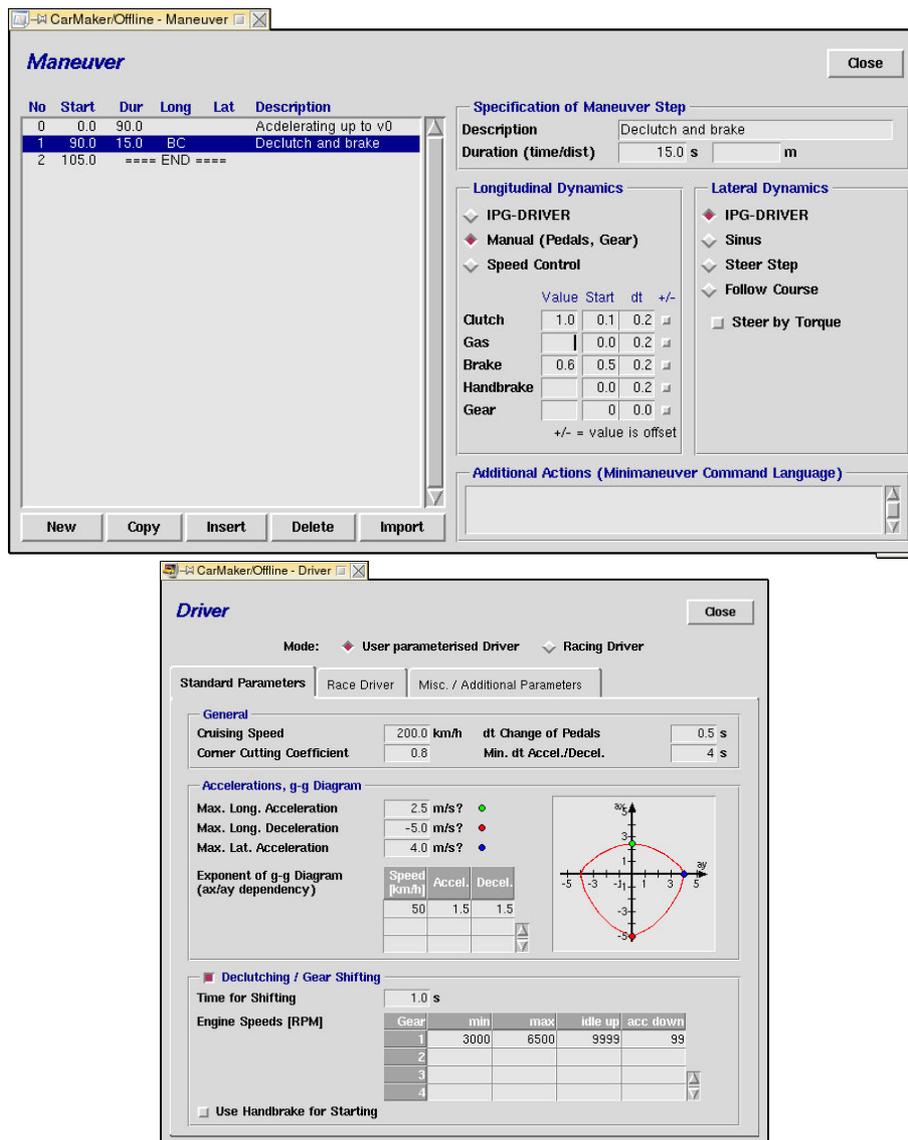


Abbildung 3.4.4: Minimanöver- und Fahrermodell-Dialog zur Parametrierung der Fahraufgabe

- Querdynamik: Lenken

Gelenkt werden kann über Vorgabe des Lenkwinkels oder des Lenkmomentes. Die Leistungsfähigkeit des im Fahrzeug parametrierten Lenksystems ist hierbei zu beachten.

- Lenkwinkelrampe: Startzeitpunkt, Amplitude absolut oder relativ, Dauer zum Erreichen des Endwertes
- Sinuslenken: Startzeitpunkt, Amplitude, Frequenz bzw. Periodendauer, Anzahl

der Perioden

- Sinus-Sweep: Startzeitpunkt, Anfangs- und End-Amplitude und -Frequenz, Dauer der Frequenz-Änderung
 - Querschleunigung: Wunschbeschleunigung
 - Kurs-Folge-Regler: Einfacher Regler, um zu verhindern, dass das Fahrzeug von der Fahrspur abtrifft
 - Fahrermodell IPG-DRIVER: Befahren eines Kurses entsprechend der Vorgaben
- Bedienkommandos: Die Funktionalität der gesamten Simulationsumgebung steht zur Verfügung: Signal-Manipulation zur Bedienung von Schaltern, zur Störung von Signalen und zum Umschalten elektrischer Fehler (siehe Abschnitt 3.4.3), Protokollierung und Fehlerhandling (siehe Abschnitt 3.3.1), Abspeichern (siehe Abschnitt 3.4.2), an bestimmte Bedingungen gekoppelter Wechsel in ein anderes Minimanöver u.v.m. Hiermit lassen sich selbst komplexe Bedienvorgänge übersichtlich automatisieren, wie z.B. ein Versuch im Zusammenhang mit einem aktiven Luftfeder-Fahrwerk: Kofferraum auf, Beladen, Kofferraum zu, Türe auf, Einsteigen, Türe zu, Motor anlassen, Anfahren, Kurs befahren, Anhalten, Aussteigen und wieder Entladen.

Es sind sowohl Open-Loop-Fahraufgaben möglich, bei denen eine Aktion gesteuert durchgeführt wird, wie z.B. ein Lenkwinkelsprung mit 30 Grad Amplitude, als auch Closed-Loop-Manöver, bei denen eine Regelung stattfindet.

Mit dem Fahrermodell IPG-DRIVER wird das Befahren von Kursen unter Berücksichtigung des fahrdynamischen Grenzbereiches möglich. Längs- und Querdynamik werden entsprechend der Fahrercharakteristik gewählt und aufeinander abgestimmt. Fahrertyp, Aufmerksamkeit in Bezug auf Einhalten von Kurs- und Geschwindigkeitsvorgabe, Schaltdrehzahlbereiche, Kurvenschneiden etc. können parametrisiert werden. Bevor das Fahrermodell das Fahrzeug im optimalen Leistungsbereich bewegen kann, wird eine kurze Adaptions-Fahrt durchgeführt. Hierbei wird die Fahrzeugcharakteristik identifiziert.

Über die Funktionalität des Stimulations-Moduls (siehe Abschnitt 3.4.7) können reale Versuchsfahrten oder aufgezeichnete Simulationen nachgestellt und Steuergeräte wiederholt mit identischen Eingangssignalen stimuliert werden. Solche Stimulations-Tests sind vor allem zur Absicherung neuer Regler-Software-Stände von Bedeutung. Es handelt sich dabei vor allem um die Überprüfung von Schwellwerten bei der Detektion von Fehler- und Fahrzuständen.

3.4.7 Stimulation

Ein wichtiger Test zur Kontrolle und Absicherung neuer Regler-Software-Stände ist die Stimulation. Hierfür werden speziell Fahrversuche oder Simulationen durchgeführt, mit denen die reglerinterne Detektion von Fahr- und Fehlerzuständen angesprochen wird und ihr im Lastenheft definiertes Verhalten überprüft werden kann. Alle Eingangssignale des

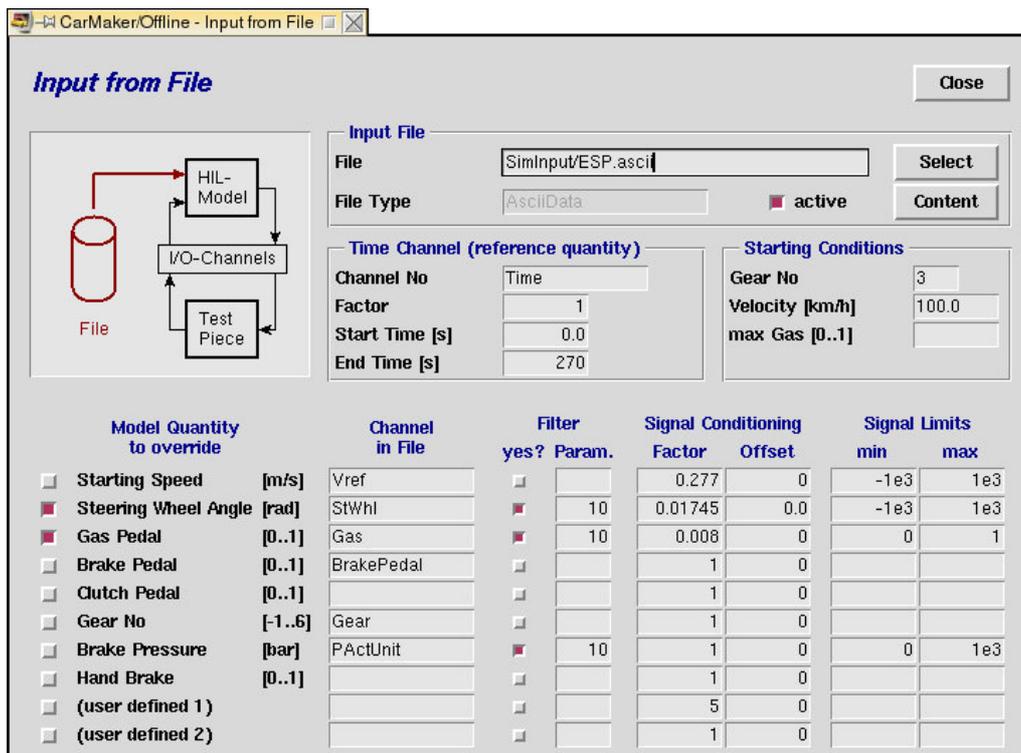


Abbildung 3.4.5: Dialog zur Parametrierung der Stimulation über Zeitverläufe

Reglers werden dabei aufgezeichnet. Mit diesen aufgezeichneten, beliebig oft reproduzierbaren Zeitverläufen wird anschließend jeder neue Regler stimuliert. Kontrolliert wird, dass Fehler und Zustände wie spezifiziert erkannt und Toleranzen und Reaktions-Zeiten eingehalten werden. Sobald der Regler aktiv wird und in das Systemverhalten eingreift, endet der Versuch. Eine sinnvolle System-Reaktion ist durch den reinen Stimulations-Modus ab diesem Zeitpunkt nicht gegeben.

Aufgezeichnete Zeitverläufe für die wesentlichen Fahrereingriffe lassen sich in einem eigenen Dialog auswählen, konditionieren und in die Simulation einspielen (siehe Abbildung 3.4.5).

3.4.8 Verkehrsmodell

Das Fahrerassistenzsystem Adaptive Cruise Control (ACC) regelt die Fahrzeuggeschwindigkeit und berücksichtigt dabei – im Gegensatz zum normalen Tempomat – das Fahrzeugumfeld. Ein ACC-System sensiert bevorzugt mit einem Radar-Sensor die Umgebung vor dem Fahrzeug in einem Abstand von ca. 100 bis 200 Metern und einem horizontalen Winkelbereich von ca. 5 Grad. Unter Zuhilfenahme von Gierrate und Raddrehzahlsignalen werden die für die Regelung relevanten Objekte auf der eigenen Fahrspur extrahiert. Kommt man einem langsameren Fahrzeug auf der eigenen Fahrspur näher, wird von der Geschwindigkeitsregelung bei Freifahrt auf Abstandsregelung bei Fahrgang übergegangen.

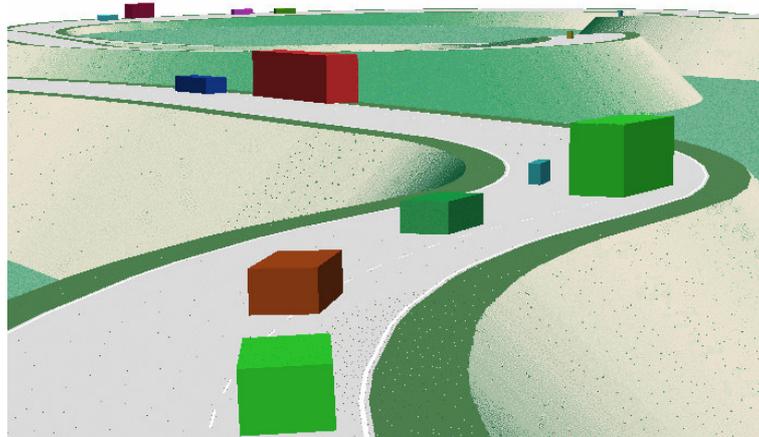


Abbildung 3.4.6: Darstellung von Einzelhindernissen als 3D-Animation

gen. Ohne Zutun des Fahrers wird Bremsdruck aufgebaut und das Fahrzeug verzögert. Verlässt das Hindernis die Fahrspur oder steuert der Fahrer das Fahrzeug auf die Überholspur, wird das Fahrzeug wieder auf die eingestellte Reisegeschwindigkeit beschleunigt. Die Beschleunigungen bewegen sich hierbei in einem Bereich von $-2.5m/s^2$ bis $+1.5m/s^2$.

Zur Entwicklung und Test des Zusammenspiels der ACC-Regelung mit den Steuergeräten für Bremsanlage und Motorsteuerung und den Eingaben durch den Fahrer wurde die Simulationsumgebung um ein zusätzliches Einzelhindernis- oder Verkehrs-Modell erweitert: Einzelhindernisse werden durch ihre Objekt-Klasse (Pkw, Motorrad, Lkw, etc.), ihre Abmessungen und zusätzliche Eigenschaften wie z.B. ihren "Radar"-Eigenschaften repräsentiert. Ihre Bewegung erfolgt längs und quer zum Verlauf der Fahrspur, abschnittsweise parametrisiert über Geschwindigkeits- oder Beschleunigungsprofile und Querposition über der Bahnkoordinate oder der Zeit. Schwabungen oder Zufallskomponenten können dieser Grund-Bewegung überlagert werden. Ein am Fahrzeug angebrachter virtueller Sensor ermittelt für jedes Hindernis Abstand, Richtung, Differenzgeschwindigkeit und Differenzbeschleunigung im Koordinatensystem des Sensors. Diese Informationen werden an ein Simulationsmodell des Radarsensors und der Objektsensierung übergeben, die wiederum das eine, für die ACC-Regelung relevante Objekt ermittelt und an den ACC-Regler übergibt.

Anwendung: Realisierung eines ACC/ADR-Prüfstandes

Zur interaktiven Bedienung wurde eine Bedienoberfläche erstellt, über die die vom Fahrer möglichen Einstellungen für Geschwindigkeit, Abstand, Aktivierung, Abschaltung und Wiederaufnahme der Regelung vorgenommen werden können.

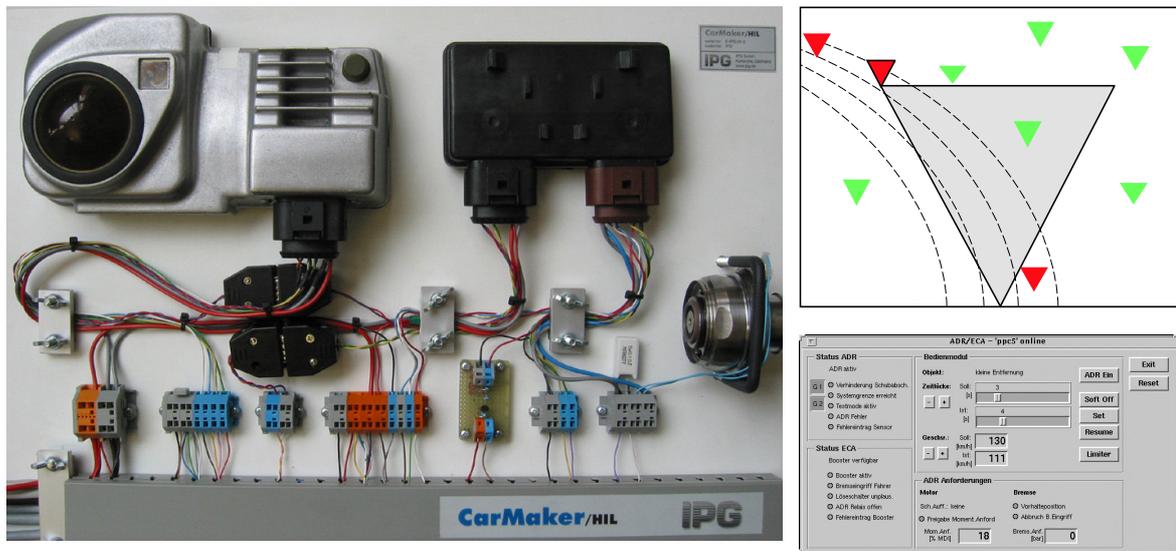


Abbildung 3.4.7: ACC/ADR Prüfstand: Hardware-Aufbau mit Radarsensor, Steuergerät und Verkabelung; Bedienoberfläche und Anzeige (Quelle: IPG Automotive GmbH)

3.4.9 Testautomatisierung, Versuchsreihen

Speziell im Hinblick auf die steigende Komplexität vernetzter elektrischer und mechatronischer Systeme im Fahrzeug kommt dem automatisierten Testen und Validieren von Software und Steuergeräten eine stark wachsende Bedeutung zu. Vor allem bei sicherheitsrelevanten Steuergeräten, wie z.B. dem Fahrdynamikregler ESP oder der elektrischen Lenkung Steer-by-Wire, wird eine Null-Fehler Toleranz angestrebt. Jede Änderung der Parametrierung eines Steuergerätes hat Einfluss auf das Verhalten des Gesamtsystems. Zur Freigabe eines neuen Softwarestandes muss deshalb das im Lastenheft spezifizierte Verhalten erneut überprüft werden. Zur Umsetzung bedarf es eines leistungsfähigen Automatisierungswerkzeugs (siehe Abbildung 3.4.8). Das entwickelte Automatisierungswerkzeug verfügt im Wesentlichen über nachfolgende Funktionalität und Schnittstellen.

Scriptbasierte Tests Es existiert eine Vielzahl an Alternativen zur Prüfstandsautomatisierung, wie z.B. Tcl, Java-Script, VisualBasic, Python, Perl. Das bei IPG Automotive GmbH entwickelte Automatisierungswerkzeug basiert auf der Open Source Scriptsprache "Tool Command Language" Tcl (siehe [tcl04]). Tcl ist logisch klar strukturiert, einfach zu erlernen, sehr leistungsfähig und hat durch einen integrierten Laufzeit-Compiler sehr gute Ausführungszeiten. Besonders hervorzuheben ist die Verfügbarkeit von Tcl auf allen gängigen Rechnerplattformen. Es enthält alle wesentlichen Eigenschaften einer sogenannten Kommandosprache, bietet darüber hinaus aber Möglichkeiten, Folgen von Kommandos zu Prozeduren zusammenzufassen, Variablen einzuführen, Kontrollstrukturen wie Schleifen und Fallunterscheidungen zu programmieren und vieles andere mehr. Eine Vielzahl an Paketen zur Erweiterung des Leistungsumfangs sind verfügbar. Das wohl bekannteste ist Tk zur Gestaltung graphischer Bedienoberflächen. Die Scripte sind einfach

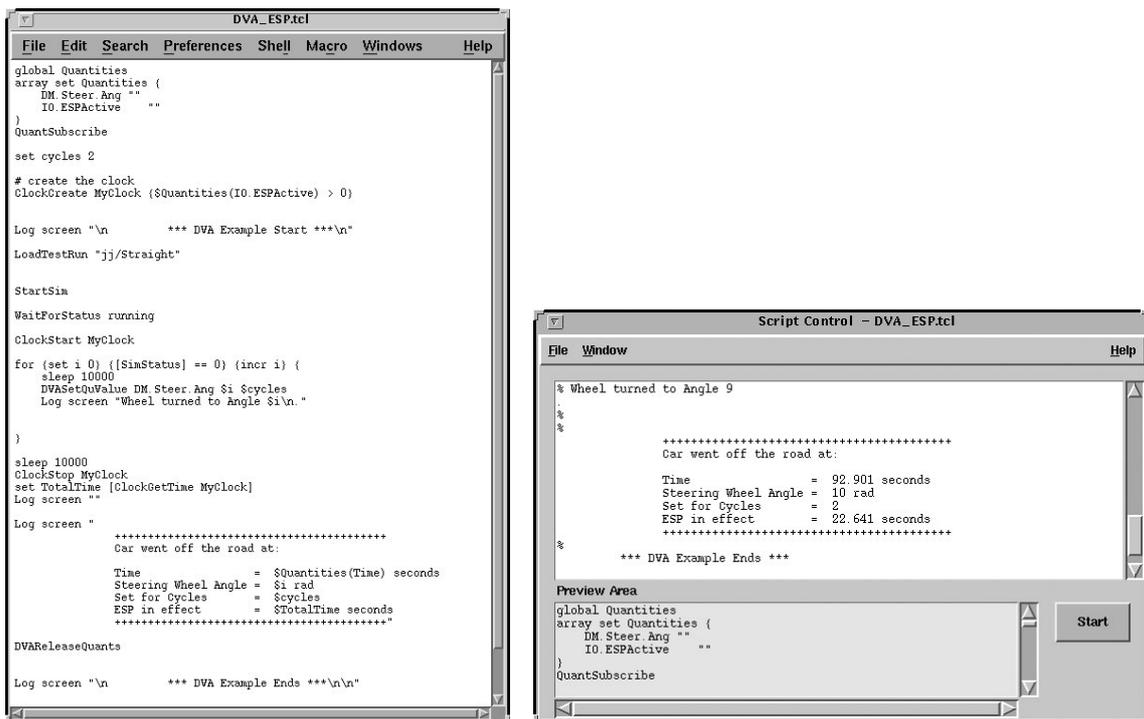


Abbildung 3.4.8: Script zur Testautomatisierung

zu formulieren und durch den Sprachumfang einer mächtigen Programmiersprache nahezu beliebig erweiterbar.

Die Testscripte werden in Form lesbarer ASCII-Text-Dateien vom Test-Ingenieur erstellt. Ein einfacher Austausch mit Kollegen und Zulieferern ist hierdurch gegeben.

Lesender und schreibender Zugriff auf die Versuchs- und Modellparameter-Datenbank Zuvor definierte Standard-Tests lassen sich laden, modifizieren oder neu generieren und reproduzierbar beliebig oft wiederholen. Die Parametrierung von Teil-Modellen, wie z.B. die Charakteristik der Achsfederung oder die Förderleistung einer Hydraulikpumpe, kann abgefragt und modifiziert werden. Ganze Teilmodelle wie z.B. der Motor, die Kupplung oder die Fahrzeugbeladung können gegen andere ausgetauscht werden. Alle am Markt angebotenen Fahrzeug-Varianten sind in Form eines “Baukasten-Systems” per Script zusammenstellbar. Fahrzeuge, Anhänger, Fahrmanöver werden zu Versuchen zusammengefügt. Aus einzelnen Tests und Datensätzen werden umfangreiche, mehrere Stunden, Tage oder Wochen ununterbrochen andauernde Testreihen abgeleitet. Jeder Fahrer verhält sich unterschiedlich; verschiedene Beladungen oder ein alter, defekter Fahrwerksdämpfer beeinflussen das Systemverhalten. Mit automatisierter Variation von Parametern innerhalb spezifizierter Wertebereiche lässt sich die Prüftiefe und Prüfbreite deutlich erhöhen. Gleiches gilt für die Absicherung von Bauteiltoleranzen.

Steuern und Überwachen des Versuchsablaufs Aus den Testscripten heraus lässt sich der Simulator bedienen. Versuche werden in die Bedienoberfläche geladen, gestartet

```

TestRun::Read  StraightForwardDriving      # modify an existing testrun
TestRun::Set   Driver.CruisingSpeed 100.0
TestRun::Write TestRun-temp"

GUI::TestRunLoad TestRun-temp              # prepare and start simulation
APO::SubscribeQuants Time Car.v Car.RoadDist
GUI::TestRunStart

# accelerate and drive to braking area
while {$Car.RoadDist < 400} ; # stay here

set T0 $Time                               # remember time
DVA::Write Driver.Gas          0.0
DVA::Write Driver.Clutch      1.0 0.75
DVA::Write Driver.BrakePedal 1.0 0.1

# after 3 seconds of braking, wheel speed sensor defect
set T1 [expr $T0 + 3.0]
while {$Time < $T1} ; # stay here
FST::Cut IO.WheelSpeedFL.In

set T1 [expr $Time + 1.2]                   # wait 1.2 seconds
while {$Time < $T1} ;
DVA::Read IO.ESPFaultLED

while {$Car.v > 0.1} ;                       # documentation after car stands still

if { $IO.ESPFaultLED == 0n } {
    LogOk    "WheelSpeedSensor fault detection: ok"
} else {
    LogError "WheelSpeedSensor fault detection: failed"
    DataStorage::SaveAll "WheelSpeedSensor_fault_detection"
}

```

Abbildung 3.4.9: Script zum Test der Fehlererkennungslogik eines Steuergerätes bei Ausfall eines Drehzahlsensor-Signals während eines Bremsmanövers

und wieder beendet. Durch den Zugriff auf die aktuellen Werte aller Größen der Simulationsumgebung (Abschnitt 3.4.1, Abschnitt 3.3.2) ist der Testautomatisierung der Systemzustand ständig bekannt. Trigger-Bedingungen können darauf basierend formuliert werden. Ein typisches Beispiel hierfür ist, eine bestimmte Zeitdauer oder Wegstrecke warten, nachdem ein Ereignis eingetreten ist. Automatisiert lässt sich z.B. bei einem Lenkwinkelsprung die Lenkrad-Drehgeschwindigkeit steigern, so lange bis die ESP-Fahrdynamikregelung zum ersten Mal aktiv eingreift.

Anmerkung: Es ist zu unterscheiden zwischen Bedingungen, die exakt für jeden Simulationsschritt oder unter Echtzeitbedingungen zu überprüfen sind, und solchen, bei denen eine so exakte Kontrolle nicht notwendig ist. Die Ersteren müssen direkt im Simulator abgewickelt werden. Für Zweitere ist es ausreichend, sie innerhalb des Automatisierungsscripts zu behandeln, das über Netzwerk an den Simulator angekoppelt ist.

Eingriffe in den Versuchsablauf All die Eingriffe, die ein Prüfstandsbediener manuell ausführen kann, können über die Testscripte automatisiert werden: Fernsteuern der Bedienoberfläche, Manipulation von Simulations-Signalen per DirectVariableAccess (siehe Abschnitt 3.4.3), Aufschalten elektrischer Fehler auf HIL-Simulatoren über die FailSafe-Tester Breakout-Box (siehe Abschnitt 3.4.3), Umschalten auf ein anderes Minimanöver, Abspeichern von Zeitverläufen, um nur einige zu nennen.

Durchführung komplexer Versuche Steuergeräte aktiver Fahrwerke, z.B. eines Luftfedersystems, sind nicht nur während der Fahrt aktiv. Sie gehen bereits bei öffnen der Fahrertüren oder des Kofferraums in den Bereitschaftszustand über, befüllen den Druckspeicher oder gleichen den durch Zuladung veränderten Höhenstand des Fahrwerks wieder aus. Nach der Fahrt verbleiben sie im Bereitschaftszustand und wachen in definierten Abständen aus dem Schlafmodus wieder auf und regeln das Fahrwerk nach. Es sind komplexe Versuchsläufe notwendig, um alle definierten Verhaltensmuster und Zustandsübergänge der Steuergeräte abzuprüfen.

Der Testingenieur baut aus mehreren, klar abgegrenzten Einzelschritten Aktionen auf, die er zu Prozeduren zusammenfasst. Z.B. besteht die Prozedur “Kofferraum beladen mit 50 kg” aus den Einzelschritten Kofferraum auf, Masse in den Kofferraum legen, Kofferraum schließen. Aus diesen Basis-Aktionen erstellt er anschließend umfangreichere Abläufe. Zur Erhöhung der Testtiefe kann die Generierung von Versuchsszenarien aus Basis-Aktionen automatisiert werden.

Auswertung und Dokumentation Für eine effiziente Testautomatisierung ist es wichtig, jedem Versuch eine klare Fragestellung zugrunde zu legen, die mit “In Ordnung” oder “Nicht in Ordnung” beantwortet werden kann. Zulässige Wertebereiche und Toleranzen müssen festgelegt werden. Nur so lassen sich die Ergebnisse umfangreicher Testreihen kompakt und eindeutig darstellen und vergleichen. Bewährt hat sich hierfür eine tabellarische Auflistung der durchgeführten Tests und ihres Ergebnisses, u.U. ergänzt um einen kurzen Kommentar. Im Falle eines negativ verlaufenen Tests wird automatisiert ein ausführlicheres Versuchsprotokoll erstellt: Versuchsbedingungen, Fehlerbeschreibung, Zeitverläufe relevanter Größen, Diagramme, Abweichungen vom Sollwert etc. werden archiviert und in der Gesamtübersicht per Hyperlink verknüpft.

3.4.10 Steuergeräte Diagnose

Die Steuergeräte moderner Fahrzeuge verfügen über Speicherbereiche (Fehlerspeicher), in denen aufgetretene Fehlersituationen über längere Zeit hinweg festgehalten werden. Abgelegt werden die Kennnummer des detektierten Fehlers, die Fehlerart, z.B. ob es sich um ein dauerhaftes oder ein sporadisches Ereignis handelt, der Fehlerort und weitere für die Fehlersituation relevante Informationen. In der Werkstatt wird dann später ein Diagnosegerät an das Fahrzeug angeschlossen und über die Diagnoseschnittstelle (K-Line oder CAN/KWP2000[ISO03]) der Fehlerspeicher ausgelesen. Mit diesen Informationen

kann die Problem- oder Fehlerursache eingegrenzt, erkannt und anschließend gezielt beseitigt werden.

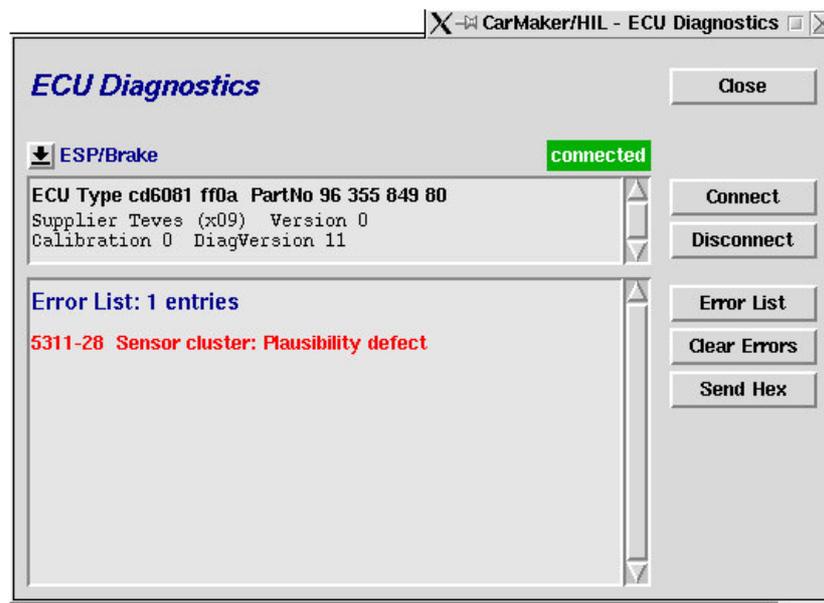


Abbildung 3.4.10: Grafische Bedienoberfläche zur Steuergerätediagnose

Für den automatisierten Test auf einem Hardware-in-the-Loop-Simulator ist es unerlässlich, Zugriff auf diese Diagnosemöglichkeit zu haben. Am Simulator steht zur interaktiven Diagnose eine Bedienoberfläche zur Verfügung (siehe Abbildung 3.4.10). Zur automatisierten Diagnose aus Testscripts heraus sind entsprechende Script-Befehle implementiert. Darüber kann

- der Fehlerspeicher ausgelesen werden. Über die hinterlegten Fehler kann erkannt werden, ob das Steuergerät die aufgeschalteten Fehler korrekt erkannt und abgespeichert hat,
- der Fehlerspeicher zurückgesetzt bzw. gelöscht werden, um ihn vor dem nächsten Versuch in einen definierten Grundzustand zu versetzen,
- lesend und schreibend auf weitere Bereiche des Steuergerätes zugegriffen werden, um z.B. Messwertblöcke auszulesen, Kodierungen vorzunehmen, Funktionen freizuschalten oder Applikationsparameter zu verändern.

3.5 Funktionsmodul Fahrzeugmodelle

Im Zentrum einer Entwicklungsplattform für mechatronische Fahrdynamikregelsysteme steht das zu regelnde System, das Fahrzeug. In der Fachliteratur, an Universitäten und

als kommerzielle Lösungen sind unterschiedliche Fahrzeugmodelle verfügbar: lineare Modelle, Einspur- oder Zweispurmodelle, einfache oder sehr detaillierte MKS-Modelle mit eigener Simulationsumgebung und Oberfläche (ADAMS-CAR, DADS, Simpack), Finite-Elemente-Modelle, bei denen Karosserieverformung, -schwingungen oder Crash-Verhalten im Vordergrund stehen (ABACUS). Trotz dieser Fülle dafür ein eigenes Fahrzeugmodell-Konzept zu realisieren, sprachen vor allem entscheidende Argumente: notwendige Echtzeitfähigkeit, Konfigurierbarkeit und Erweiterbarkeit. Dieser Abschnitt beschäftigt sich mit der Strukturierung des Systems Fahrzeug, seinen Subsystemen und Schnittstellen. Die Formulierung des MKS-Modells für das Fahrzeug ist Gegenstand von Abschnitt 4.

Der Entwicklungsprozess eines Fahrdynamikreglers erstreckt sich von der Konzeptphase über "reine" Simulationen bis hin zur Hardware-in-the-Loop-Simulation mit Prototypen- und Seriensteuergeräten. Die Testszenarien erstrecken sich von Versuchen mit stehendem Fahrzeug, über Manöver auf Fahrdynamikflächen, bis hin zur Fahrt auf vermessenen Strassen, Rennstrecken und Alpenpässen. Die Modelle müssen alle diese Einsatzbereiche abdecken und dabei mehrere Tage Simulationszeit und mehrere hundert Kilometer am Stück erlauben. Eine Lauffähigkeit des Modells ohne spezielle Oberfläche oder Simulationsumgebung ermöglicht die Integration des Modells in die verschiedenen Arbeitsumgebungen. Um die dabei eingesetzten Betriebssysteme und Rechnerplattformen unterstützen zu können, muss das Modell als C-Quellcode vorliegen. Alle beteiligten Modelle müssen sich im Hinblick auf ihre Anforderungen an die Rechenzeit und den Hauptspeicher effizient verhalten. Diese Anforderung gilt speziell für den Bereich Hardware-in-the-Loop, bei dem zusätzlich die Echtzeitfähigkeit sichergestellt sein muss. Mit zunehmendem Wissen und Detail-Kenntnis über das zu regelnde Fahrzeug und seine Subsysteme wachsen die Anforderungen an die Güte der Simulationsergebnisse. Neben einer validierten Parametrierung macht dies meist den Übergang auf verbesserte Subsystem-Modelle notwendig. Andererseits ist es für die frühe Projektphase wichtig, mit überschaubar parametrierbaren Modellen arbeiten zu können. Durch die Fertigung der Fahrzeuge entsprechend der vom Kunden gewählten Konfiguration ist hohe Variabilität und wechselweiser Einsatz verschiedener Submodelle und Modellierungsvarianten gefordert. Der Konfigurationswechsel muss einfach und automatisierbar sein.

Aktuell sind Simulationsmodelle für folgende Fahrzeug-Konfigurationen vorhanden:

- Pkw,
- Pkw-Anhänger bzw. Pkw mit angehängtem Anhänger und
- Motorrad.

Diese Modelle sind mit einem einheitlichen Plattform-Konzept realisiert, ähnlich der Plattform-Strategie von Automobilherstellern: Ein MKS-Modell für den Fahrzeug-Aufbau und das Fahrwerk bildet das Kernmodell und die integrative Plattform für alle anderen Substrukturen des Fahrzeugs. Es ist in Form einer allgemeinen, herstellerunabhängigen

Bibliothek realisiert und verfügt über die notwendigen Schnittstellen, sich entsprechend der Projekt-Anforderungen anpassen und erweitern zu lassen. Verschiedene Hilfsmodule stehen hierfür zusätzlich bereit.

Durch die Wiederverwendung bzw. gemeinsame Nutzung von Funktionsmodulen und Programm-Codes tritt eine Fehlerreduktion ein und die Modellqualität steigt. Voraussetzung hierfür sind objektorientierte Ansätze und gekapselte Subsysteme mit klar definierten Schnittstellen.

Modellstruktur

Die realisierte allgemeine Fahrzeugmodell-Plattform besitzt folgende, (weitestgehend) einheitliche Struktur bzw. Untermodule:

- Fahrzeugaufbau, Massen und Trägheiten: Fahrzeug-Aufbau, aufbaufeste Baugruppen (Antriebsstrang, Bremsanlage, ...), feste und bewegliche Ladungen. Beim Motorrad ist als Fahrer ein zusätzlicher beweglicher Körper vorhanden.
- Fahrwerk: Kinematik, Massen und Trägheiten
- Fahrwerk: Kraftelemente (Feder, Dämpfer, Wankstabilisator, ...)
- Lenksystem
- Antriebsstrang: Motor, Kupplung, Getriebe, Triebstrang, Räder
- Bremsanlage
- Reifen
- Aerodynamik

In den nachfolgenden Kapiteln werden diese im einzelnen vorgestellt und ihre Schnittstellen zu anderen Submodellen und zum Gesamtsystem Fahrzeug erläutert. Hierbei sind jeweils die für dieses Untermodul notwendigen, allgemeinen Schnittstellengrößen aufgeführt. Zusätzliche, häufig im Zusammenhang mit Aktuatoren und Steuergeräten auftretende Signale sind individuell auf eine Realisierung dieses Moduls ausgerichtet und deshalb im folgenden nicht mit aufgeführt.

3.5.1 Fahrzeugaufbau, Massen und Trägheiten

Ein Fahrzeug ist aus mehreren Einzelkörpern aufgebaut. Im Simulationsmodell sind diese direkt oder durch generalisierte Körper repräsentiert. Parametriert werden sie für die Konstruktions- oder Zeichenstellung (Design-Konfiguration) und nicht für die Gleichgewichtslage (Equilibrium-Konfiguration). Das Fahrzeugmodell – außer dem Fahrwerk – ist mit folgenden Einzelkörpern aufgebaut:

Der *Fahrzeugaufbau* fasst die Karosserie und weitere karosseriefeste Bauteile zusammen. Der *Motor* ist ein separater Körper.

n stets fest mit dem Fahrzeugaufbau verbundene *Trimm-Beladungen* passen einen vorhandenen Fahrzeugdatensatz an eine spezielle Konfiguration an, z.B. Prototypenträger, Fahrzeugvariante, Zusatzausstattung, Bestückung mit Messtechnik etc. Diese n parametrisierten Beladungen werden im Modell durch einen generalisierten Körper repräsentiert.

3 *versuchsabhängige Beladungen* passen den Beladungszustand den Versuchsbedingungen an. Dies sind Tankfüllung, Passagiere, Gepäck auf dem Dachträger und im Kofferraum und verrutschende Ladungen. Die versuchsabhängigen Beladungen sind während der Simulation gegenüber dem Fahrzeugaufbau verschieblich. Die Bewegung erfolgt rückwirkungsfrei. Ihre Positionierung erfolgt in einem messtechnisch einfach erfaßbaren Koordinatensystem, dessen Ursprung in der Fahrzeug-Mittenebene hinter dem Fahrzeug auf Fahrbahnniveau liegt.

Beim Motorradmodell ist der Fahrer zusätzlich als beweglicher Einzelkörper vorhanden. Er wird vom Fahrermodell fahrsituationsabhängig kontrolliert.

Details zum der MKS-Modellierung zugrunde liegenden Formalismus finden sich in Abschnitt 4.3.2.

3.5.2 Fahrwerk: Kinematik und Kraftelemente

Der Einsatzbereich des Fahrzeugmodells ist die Fahrdynamiksimulation; es soll nicht der Entwicklung und Optimierung von neuen Achskonzepten dienen. Unterschiedliche Achskonstruktionen und Parametrierungen müssen unterstützt werden. Das Modul Fahrwerk modelliert die Kinematik und Elastokinematik und die Kraftelemente der Radaufhängungen. Hierzu zählen:

- Bewegung des Radträgers und weiterer Einzelkörper mit Position und Orientierung bez. eines fahrzeugaufbaufesten Referenzpunktes je Radaufhängung,
- Bewegung des Rades (außer der Drehung um die Raddrehachse) über einen Punkt auf der Raddrehachse; die Raddrehachse ist starr mit dem Radträger verbunden,
- Abstände und Relativgeschwindigkeit der Anlenkpunkte der Kraftelemente Achsfeder, Achsdämpfer, Wankstabilisator, Puffer und Zugansschläge,
- Kräfte der konventionell oder aktiv ausgeführten Kraftelemente der Radaufhängungen.

Eine Vorder- und eine Hinterachse bilden das Fahrwerk. Sie sind jeweils als eigenständiges Modul realisiert. Sie sind innerhalb des MKS-Formalismus dem Fahrzeugaufbau untergeordnet und fester Bestandteil des MKS-Kernmodells. Die Achse wird mit ihrem in Achsmitte liegenden Referenzpunkt an ihn im Einfügapunkt (axle mount point) für Vorder- bzw. Hinterachse gekoppelt. Die Kinematik der Aufhängungen wird über maximal zwei unabhängige, generalisierte Freiheitsgrade pro Radaufhängung, beschrieben. Für die Vorderachse sind dies Einfedern und Lenken, für die Hinterachse meist Einfedern links und/oder Einfedern rechts. Die Berechnung der Dynamik der Radaufhängung erfolgt nicht

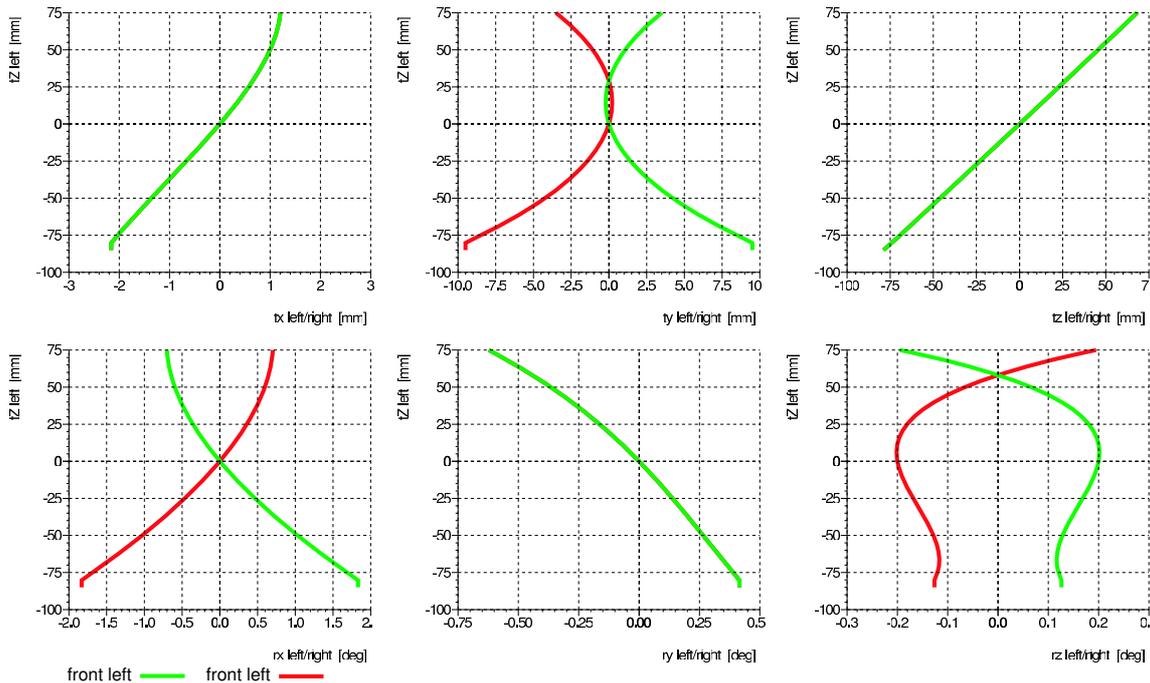


Abbildung 3.5.1: Kinematik einer McPherson-Vorderradaufhängung beim parallelen Einfedern

für jedes ihrer Bauteile einzeln sondern als generalisiertes System. Hiermit wurde die Forderung nach Unabhängigkeit von der tatsächlichen Konstruktion erfüllt.

Folgende Achsvarianten werden unterstützt:

- Vorderachse: Einzelradaufhängung gelenkt
- Hinterachse: Einzelradaufhängung, Verbund-Achse rechts/links gekoppelt, Starrachse

Für $n \geq 1$ Einzelkörper einer Radaufhängung sind Position und Orientierung anzugeben. Der Radträger muss immer vorhanden parametrisiert sein. Die bauteilorientiert modellierten Kraftelemente benötigen die Bewegung ihrer Anlenkpunkte. Die Kinematik kann über einfache, lineare Ansatzfunktionen oder über umfangreiche Kennfelder vorgenommen werden. Hierzu werden reale Achsen auf einem Kinematik-Prüfstand vermessen, Kinematik-Rechnungen mit Spezialprogrammen zur Achsauslegung durchgeführt oder Design-Größen aus Lastenheften übernommen. Elastokinematik: Die Positions- und Orientierungsänderungen aufgrund der wirkenden Kräfte und Momente werden der reinen Kinematik quasistatisch überlagert.

Kommen Luftfedern, regelbare Dämpfer oder aktive Stabilisatoren bei einer Fahrwerkskonstruktion zum Einsatz, müssen hierfür spezielle Simulationsmodelle ergänzt werden. Sie erhalten vom MKS-Fahrzeugmodell die Kinematik-Größen – speziell den Abstand und die Abstandsänderungsgeschwindigkeit Bauteil-Angriffspunkte – übergeben und liefern die in Bauteilrichtung wirkenden Kräfte zurück.

Motorrad Beim Motorrad besteht eine "Achse" aus einer einzelnen Radaufhängung. Die Konstruktion der Vorder- und Hinterradaufhängung besteht bei Motorrädern meist aus nur wenigen Bauteilen. Elastokinematische Effekte haben keinen oder nur untergeordneten Einfluss auf das Fahrverhalten. Es wurde die Möglichkeit geschaffen, das Fahrzeugmodell um eigene Aufhängungs-Module zu ergänzen. Hierüber wurden für die Konstruktionsvarianten Telelever und Paralever über Geometriepunkte parametrisierte, auf kinematischen Betrachtungen basierende, analytische Module implementiert, die Kinematik und Kräfte ermitteln. Eine einfache als Hülsenführung ausgeführte Aufhängung ist ebenfalls verfügbar.

Pkw-Anhänger Ein Pkw-Anhänger ist stets un gelenkt. Sein Fahrwerk ist ausgeführt als Einzelachse oder als Tandem-Achse mit zwei einzelnen, nahe beieinander angeordneten Achsen. Als über Geometriepunkte direkt parametrisierbare Achskonstruktionen sind die Pendelachse und die gefeilte Pendelachse implementiert. Da hiermit fast alle Pkw-Anhänger modellierbar sind, wurde – im Gegensatz zum Motorrad – auf die Möglichkeit zur Erweiterung des Modells um eigene Achskonstruktionen verzichtet. Eine einfache als Hülsenführung ausgeführte Aufhängung ist ebenfalls verfügbar.

3.5.2.1 Realisierung eines Luftfeder-CDC-HIL-Prüfstandes

Der VW-Phaeton verfügt über ein aktives Fahrwerk. Die Radaufhängungen sind über vier Luftfederbeine gefedert. Das Luftfedersystem ist vom Fahrer konfigurierbar und wird über ein eigenes, an den Fahrzeug-Datenbus angeschlossenes Steuergerät kontrolliert. Es bedient einen Kompressor und mehrere Pneumatik-Ventile. Der Höhenstand des Fahrzeugs wird u.a. abhängig von der Fahrgeschwindigkeit und den Strassenverhältnissen gewählt. Die Dämpfer-Abstimmung wird vom Fahrer vorgewählt und von einem Dämpfer-Steuergerät über elektrische Bestromung von Ventilen innerhalb der Dämpfer dem fahrdynamischen Zustand kontinuierlich angepasst. Die Bewegung des Fahrzeugaufbaus wird mittels einer Sky-Hook-Regelung reduziert. Zur Entwicklung der Fahrdynamikregler wurde ein umfangreiches Laborauto realisiert (siehe auch [Gor02]).

3.5.2.2 Wankstabilisierung

Fahrzeuge der Oberklasse und Sports Utility Cars verfügen häufig zur Steigerung des Fahrkomforts und der Fahrsicherheit über eine aktive Wankstabilisierung. Hierüber wird die Wankbewegung des Fahrzeugaufbaus bei Kurvenfahrt reduziert und das Fahrverhalten durch Verändern des Eigenlenkverhaltens verbessert. Erfolgt die Wankregelung über aktive Stabilisatoren an Vorder- und Hinterachse, so bestehen diese meist aus zwei gegeneinander – meist hydraulisch – verdrehbaren Teilen. Durch Verteilung des Stabilisierungsmomentes zwischen beiden Achsen wird die Radlastverteilung beeinflusst und das Eigenlenkverhalten des Fahrzeugs gezielt verändert. Eine wesentliche Eingangsgröße der Regelung ist das von einem am Fahrzeugaufbau montierten Sensor erfasste Querschleunigungssignal. Zur Kontrolle der Regelung wird die Stellung der Hydraulikventile überwacht und der Druck an den hydraulischen Aktuatoren der Stabilisatoren erfasst.

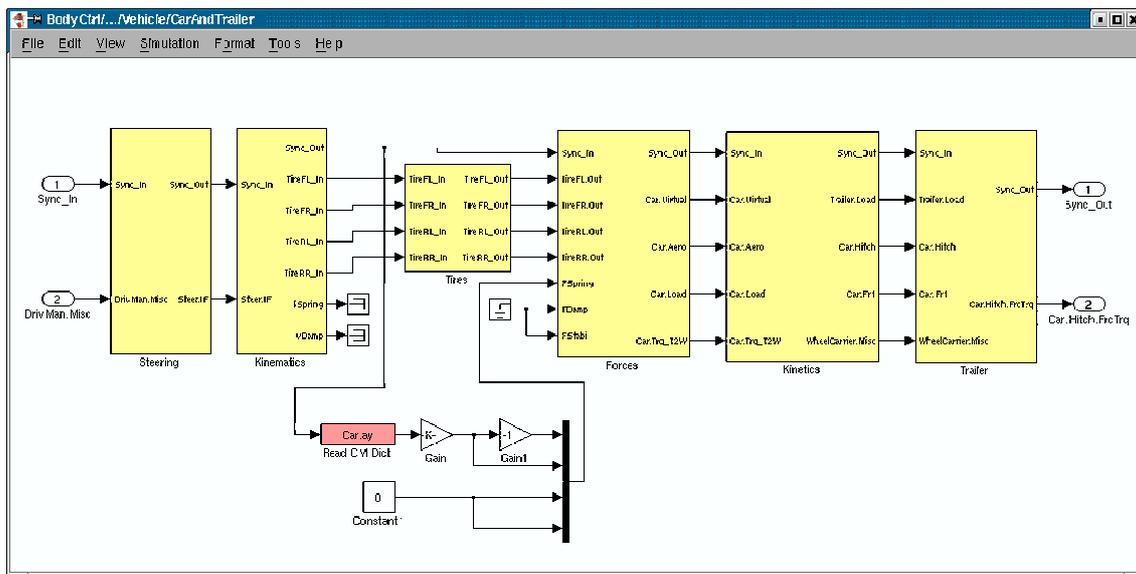


Abbildung 3.5.2: Prinzip-Modell zur Wankstabilisierung, implementiert als Simulink-Modell

Zur Plausibilisierung werden weitere Größen vom Fahrzeug-CAN herangezogen. Im Falle eines Systemfehlers sind die beiden verdrehbaren Stabilisatorteile mechanisch starr gekoppelt und erhalten so die normale, rein mechanische Stabilisatorwirkung. Ein Prinzip-Modell einer aktiven Wankstabilisierung findet sich in Abbildung 3.5.2.

3.5.3 Lenksystem

Das Lenken ist die wichtigste Eingriffsmöglichkeit des Fahrers auf das fahrdynamische Verhalten. Bei heutigen Lenksystemen sind Lenkrad und Rad mechanisch gekoppelt oder verfügen zu mindest über eine mechanische Rückfallebene. Das vom Fahrer am Lenkrad aufgebraachte Moment wird meist hydraulisch oder elektrisch unterstützt. Schon seit mehreren Jahren laufen Entwicklungsarbeiten, das Lenksystem “aktiv” auszuführen. Ziel ist hierbei, die starre, mechanische Kopplung zwischen Lenkrad und Rad aufzuheben und über hydraulische oder elektrische Aktuatoren zu ersetzen. Die Lenkübersetzung wird veränderlich – meist über ein Überlagerungsgetriebe – und die Lenkmomentenunterstützung wird situativ anpassbar oder ein Lenkmoment kann zur fahrdynamischen Stabilisierung aktiv aufgebaut werden.

Ein Lenksystem wird jeweils für ein bestimmtes Fahrzeug entwickelt und erfordert aufgrund seiner speziellen Konstruktion auch speziell angepasste Simulationsmodelle. Das Lenksystem ist deshalb als eigenständiges, über eine definierte Lenkmodell-Schnittstelle gekoppeltes Untermodul des Fahrzeugmodells ausgeführt. Das Lenksystem beginnt am Lenkrad und erstreckt sich über Lenksäule, Lenkgetriebe und Zahnstange. Es endet an der Schnittstelle zum Fahrwerk an den Gelenkpunkten rechts/links zwischen Zahnstange und Spurstange. Die verfügbaren Lenkmodelle können über das Modell-Management-Modul

(siehe Abschnitt 3.3.3) mit weiteren speziellen Modellen ergänzt und versuchsabhängig ausgewählt werden.

Lenksystem-Schnittstelle Der Fahrer bedient das Lenksystem in zweierlei Modi:

1. Lenken per Lenkradwinkel: Der Fahrer gibt die Stellung des Lenkrades über der Zeit vor. Die für diese Bewegung notwendigen Kräfte sind ohne Bedeutung. Dieser Modus stellt modellierungstechnisch die geringeren Anforderungen.
2. Lenken über Moment am Lenkrad: Der Fahrer wirkt über ein (externes) Moment auf das Lenkrad ein. Die Bewegung und Stellung des Lenkrades wird von einem dynamischen Lenksystem ermittelt. Die Trägheiten von Lenksystem und Fahrwerk und alle angreifenden Kräfte (Lenkmoment, Reifenkräfte, Servounterstützung, ...) werden dabei berücksichtigt.

Eingangssignale in das Lenksystem – von Fahrer und Fahrzeug – sind:

- Modus der Bedienung: Lenken über Lenkradwinkel oder über Lenkradmoment,
- Lenkradwinkel, Lenkradgeschwindigkeit,
- Lenkradmoment,
- Trägheiten und Kräfte aus dem Fahrwerk, bezogen auf die Bewegung am Gelenkpunkt Spurstange–Zahnstange.

Ausgangssignale des Lenksystems zum Fahrzeug sind:

- Position, Geschwindigkeit und Beschleunigung des Gelenkpunktes Spurstange–Zahnstange, an der Schnittstelle Lenksystem–Radaufhängung links/rechts.

Als zusätzliche Beobachter-Größen sind vorgesehen die aktuelle Lenkübersetzung von Lenkrad zu xyz und das statische Lenkradmoment, das erforderlich ist, das Lenksystem in der aktuellen Stellung zu halten.

Lenken über Kinematik

Der Fahrer gibt über der Zeit die Bewegung des Lenkrades vor. Kinematisch daran gebunden ergibt sich die Bewegung am Gelenkpunkt zur Spurstange. Über aktive Elemente können Regler die Übersetzung verändern. Durch die Modellierung über rheonome Bindungen von Lenksystem und Radaufhängung ohne Kraftwirkung verfügt ein Fahrzeugmodell mit einem solchen Lenksystem über keine Lenkrückstellung bei losgelassenem Lenkrad. Eine Rückwirkung z.B. von ABS-Regeleingriffen auf das Lenkrad ist nicht vorhanden.

Ein Motorrad-Modell stabilisiert sich beim Geradeauslauf über die Lenkinematik selbst und verhindert so ein Umfallen. Diese Lenksystem-Modellierung ist deshalb hierfür nicht anwendbar.

Lenken über Moment

Die Bewegung einer Radaufhängung ist über die zwei unabhängigen generalisierten Freiheitsgrade Einfedern und Lenken modelliert (siehe Abschnitt 3.5.2). Der Freiheitsgrad Einfedern wird vom Fahrzeugmodell abgedeckt. Der Freiheitsgrad Lenken wird vom Lenksystem berechnet. Zur Kopplung von Lenksystem und Radaufhängung auf Beschleunigungsebene werden vom Fahrwerks-Modul die Trägheiten der Radaufhängungen und die auf sie wirkenden Kräfte und Momente auf den Lenk-Freiheitsgrad projiziert und an das Lenksystem übergeben. Dort werden diese generalisierten Trägheiten und Kräfte im Modell zusammen mit der Trägheit des Lenkrades und der anderen am System beteiligten Komponenten berücksichtigt. Bei der Entwicklung der Regellogik für Lenk-Aktuatoren oder beim Betrieb eines Hardware-in-the-Loop-Lenksystem-Simulators sind detaillierte Modelle für Reibung, Spiel und Hysterese und die Dynamik der Aktuatoren notwendig.

3.5.4 Reifen

Eine Vielzahl an Reifenmodellen mit unterschiedlicher Funktionalität existiert. Phänomenologische Modelle, wie z.B. IPG-TIRE [SW88], Magic-Formula in unterschiedlichen Varianten und SWIFT [Pac02], “easy to use” [Ril94]), u.a. werden vor allem für Fahrdynamikuntersuchungen eingesetzt. Auf Finite-Elemente-Ansätzen basierende Strukturmodelle liefern Antworten auf Fragestellungen des Fahrkomforts, der Akustik und der Struktur des Reifens selbst. Sie sind für eine effiziente Fahrdynamiksimulation meist nicht geeignet. Andere Reifenmodelle, wie z.B. CDTire (ehemals RMod-K), schalten intern abhängig vom Anwendungsfall und den Betriebsbedingungen zwischen mehreren Modellierungsansätzen um. Für eine Bewertung verschiedener Reifenmodelle aus fahrdynamischer Sicht sei auf [BAK02] verwiesen.

Der Aufbau eines Reifendatensatzes ist ein Zeit- und Kostenfaktor. Er erfolgt über Vermessung von Reifen auf Prüfständen oder in Messfahrzeugen oder mit Hilfe detaillierter Strukturmodelle, sofern sie zur Verfügung stehen. Liegen bereits validierte Datensätze eines Reifens vor, wird man im konkreten Projekt meist darauf zurückgreifen wollen. Dies macht es notwendig, das dazu gehörende Reifenmodell einzusetzen.

Ein Fahrzeugmodell muss deshalb über eine standardisierte, allgemeine, auf das Wesentliche reduzierte Schnittstelle zwischen Fahrzeug und Reifen verfügen. Von der Automobilindustrie wurden die Größen, die über eine solche Schnittstelle ausgetauscht werden sollen, im Rahmen des TYDEX-Workshops[UZ97] erarbeitet.

Das Fahrzeug liefert folgende Eingangsgrößen an das Reifenmodell:

- Position des Radmittelpunktes C bez. Absolut-Systems \mathcal{F}_0
- Orientierung und Drehgeschwindigkeit des Radträgers bez. \mathcal{F}_0
- Drehwinkel und Drehgeschwindigkeit der Felge. Die Felge ist über ein reines Drehgelenk an den Radträger gekoppelt. Bezugssystem ist das Radträgersystem.

Das Reifenmodell berechnet:

- Reifen-Fahrbahn-Kontakt: unter Berücksichtigung der Fahrbahn-, Reifeneigenschaften und der Reifengeometrie
- Fahrbahn-(Normal)-Kraft: abhängig vom Abstand des Fahrzeugs von der Fahrbahnoberfläche
- Reifen-Reaktionskräfte in der Reifen-Fahrbahn-Kontaktzone (“Fahrbahn-Tangentialebene”)
- Rollwiderstand, Bohrmoment, Sturzmoment, ...

Das Reifenmodell gibt an das Fahrzeugmodell zurück:

- Kraft und Moment bez. Radmittelpunkt
- Speziell: Moment auf das Rad um die Raddrehachse

Weitere Größen wie z.B. Schlupf, Schräglaufwinkel, Sturzwinkel bez. Fahrbahn, dynamischer Reifenradius etc. können von einem Reifenmodell bereit gestellt werden. Es handelt sich hierbei um Ausgabegrößen, die einzig der Beobachtung dienen. Auf das Fahrzeugmodell haben sie keinerlei Einfluss.

In der Praxis sind häufig reine “Horizontal”-Reifenmodelle anzutreffen: Sie beschränken sich auf die Berechnung der Kraft in der Fahrbahnebene und das Moment senkrecht dazu. Um solche Modelle einfach in ein Fahrzeugmodell integrieren zu können, das über oben skizzierte allgemeine Schnittstelle verfügt, werden folgende Funktions-Blöcke bereitgestellt:

Kinematik-Transformator-Block Er berücksichtigt die Geometrie von Reifen und Fahrbahn und die Reifeneigenschaften, wie z.B. Vertikalsteifigkeit, entspannter Reifenradius. Berechnet werden der Reifen-Fahrbahn-Kontaktpunkt P , der Geschwindigkeitszustand im Kontaktpunkt, der Sturz bez. der Fahrbahn und die Fahrbahnnormalkraft \overline{F}_{ZH} . Sie sind die eigentlichen Eingänge des integrierten Horizontal-Modells. Die Berechnung dieser Größen stellt ein erhebliches Problem dar. Siehe hierzu Abschnitt 5.

Kraft-Momenten-Transformator-Block Dieser Block transformiert die von “Horizontal”-Modellen berechneten Reifenkräfte und -momente vom Reifen-Fahrbahn-Kontaktpunkt in den Radmittelpunkt zurück.

Die Auswahl und Parametrierung der zur Simulation zu verwendenden Reifenmodelle wird über die Modell-Manager-Schnittstelle, siehe Abschnitt 3.3.3, abgewickelt. Durch die Auswahl des Reifendatensatzes wird das zugehörige Reifenmodell aktiviert. Unterschiedliche Reifenmodelle in einem Fahrzeug oder einem Fahrzeug-Anhänger-Gespann sind genauso Simulationsalltag, wie Fahrten auf Schnee mit einem speziellen Schnee-Reifenmodell im Wechsel mit Fahrten auf Asphalt mit einem Standard-Modell, automatisiert aus Scripten der Testautomatisierung heraus.

3.5.5 Bremsanlage

Die Bremsanlage gehört zu den wichtigsten Komponenten des Fahrzeugs. Ihre Funktion umfasst die Reduktion der Fahrzeuggeschwindigkeit, die Verhinderung einer Fahrzeugbeschleunigung auf Gefällestrrecken und die Verhinderung einer Bewegung bei festgestelltem Fahrzeug. Diese Grundfunktionalität wird heutzutage durch intelligente Fahrdynamik-Regelsysteme ergänzt, die das Blockieren einzelner Räder beim Bremsen (AntiBlockierSystem ABS, Elektronische BremskraftVerteilung EBV) und das Durchdrehen beim Beschleunigen verhindern (AntriebsSchlupfRegelung ASR) oder über Bremsengriffe an einzelnen Rädern das Fahrzeug stabilisieren (Elektronisches StabilitätsProgramm ESP); Sicherheitssysteme wie der Bremsassistent helfen vor allem dem ungeübten Fahrer das verfügbare Kraftschlusspotential besser auszuschöpfen. Bremssysteme sind bisher vor allem aufgrund der notwendigen Systemdynamik und der zu erzeugenden Kräfte bzw. Momente als hydraulisches System ausgeführt. Elektrische oder elektro-hydraulische Systeme befinden sich in der Entwicklung.

Im Simulationsmodell ist die Bremsanlage als eigenständiges Untermodul realisiert, dessen Aufgabe es ist, die an den Rädern verfügbaren Bremsmomente zu ermitteln. Die Modelle für das Bremssystem können über das Modell-Management-Modul (siehe Abschnitt 3.3.3) mit weiteren speziellen Modellen ergänzt werden und werden versuchsabhängig ausgewählt.

Bremsanlagen-Schnittstelle Die Schnittstelle umfasst explizit Signale, die notwendig sind oder die heutzutage bei hydraulischen Bremssystemen üblicherweise anzutreffen sind. Für weitere, darüber hinausgehende Signale sind zusätzlich individuell konfigurierbare Ein- und Ausgänge vorgesehen. Im folgenden wird die Schnittstelle am Beispiel des Pkws vorgestellt.

Eingangsgrößen des Bremssystems sind

- Bremspedalkraft durch Betätigung des Fahrers,
- Ansteuerung der Hydraulikpumpe,
- Ansteuerung eines unterstützenden Boosters, der für ausreichend Druck auf der Saugseite der Hydraulikpumpe sorgt,
- Ansteuerung von Einlass-, Auslass- und anderer Hydraulik-Ventile zur Beeinflussung des Drucks in den verschiedenen Systemteilen,
- optional: Druck im Hauptbremszylinder, anstatt der Bremspedalkraft. Dieser Eingang ist vor allem während der Entwicklung eines Hydraulikmodells oder zur Validierung eines Parametersatzes hilfreich,
- Umgebungstemperatur.

Ausgangsgrößen des Bremssystems sind

- die an den Rädern aktuell maximal verfügbaren Bremsmomente, die im Fahrzeugmodell bzw. zur Berechnung der Raddrehung im Antriebsstrang benötigt werden,
- der Druck im Hauptbremszylinder und in den Radbremszylindern zur Kontrolle und zum Vergleich mit Messungen,
- die Bewegung des Kolbens des Hauptbremszylinders, des Bremspedals und des Bremsenbooster-Kolbens (optional),
- die Spannung an der Hydraulikpumpe (die beim Nachlaufen der Pumpe erzeugte Generatorspannung wird vielfach vom Steuergerät überwacht),
- der Booster Löseschalter. Benötigt das Bremssystem keine Unterstützung durch den Booster, wird dieses Signal gesetzt.

Zur Modellierung der Feststellbremse ist im Fahrzeugmodell ein einfaches Modell implementiert, das die Betätigung des Handbremshebels in Radbremsmomente umsetzt. Sollte das eigentliche Bremssystem-Modell keine Feststellbremse enthalten, kann auf diesen Modellteil zurückgegriffen werden.

In Simulationsumgebungen ohne Bremsanlagen-Steuergerät wird häufig als Bremsanlage eine Bremsmomentenverteilung ohne aufwändige Hydraulik eingesetzt. Ein einfaches Bremsanlagenmodell zur Validierung mit deren Simulationsergebnissen ist vorhanden.

Motorrad Der Fahrer eines Motorrades betätigt das Bremssystem radindividuell oder kombiniert über einen Handbremshebel und ein Fussbremspedal [Rie02].

Pkw-Anhänger Im Gegensatz zu Pkw und Motorrad wird die Bremse eines Pkw-Anhängers nicht vom Fahrer betätigt, sondern ist – falls vorhanden – als Auflaufbremse ausgeführt: Verzögert das Zugfahrzeug, läuft der Anhänger auf das Fahrzeug auf und verschiebt ein in die Deichsel in Längsrichtung integriertes Schubgelenk. Diese Verschiebung betätigt die Radbremsen des Hängers. Vereinzelt verfügen Pkw-Anhänger über eigene Steuergeräte, die den Anhänger zur Stabilisierung aktiv bremsen.

3.5.5.1 Realisierung eines ESP-HIL-Prüfstandes

Beim Elektronischen Stabilitäts Programm (ESP) erfolgen Regeleingriffe zur Erhöhung der Fahrstabilität durch gezielte Bremseneingriffe und Eingriffe in die Motorsteuerung. Durch Verhindern des Blockierens (AntiBlockierSystem ABS, Elektronische Bremskraft-Verteilung EBV) oder Durchdrehen (AntriebsSchlupfRegelung ASR) der Räder wird die Seitenführung erhalten und die maximale Kraftübertragung zwischen Reifen und Fahrbahn erzielt. Zum Ausgleich von Differenzen zwischen gewünschter und realisierter Fahrzeugbewegung wird durch radindividuelle Bremseneingriffe ein Giermoment erzeugt. Aktive Eingriffe erfolgen unabhängig von einer Fahrerbetätigung durch den Aufbau von Bremsmomenten über eine Hydraulikpumpe und Hydraulikventile sowie Eingriffe in das Motormanagement zur Erhöhung bzw. Reduktion des Antriebsmomentes. Im Falle

eines Systemausfalls bleibt die Bremsfähigkeit durch die direkte hydraulische Verbindung zwischen Bremspedal und Radbremse erhalten. Drehzahlsensoren an allen Rädern,

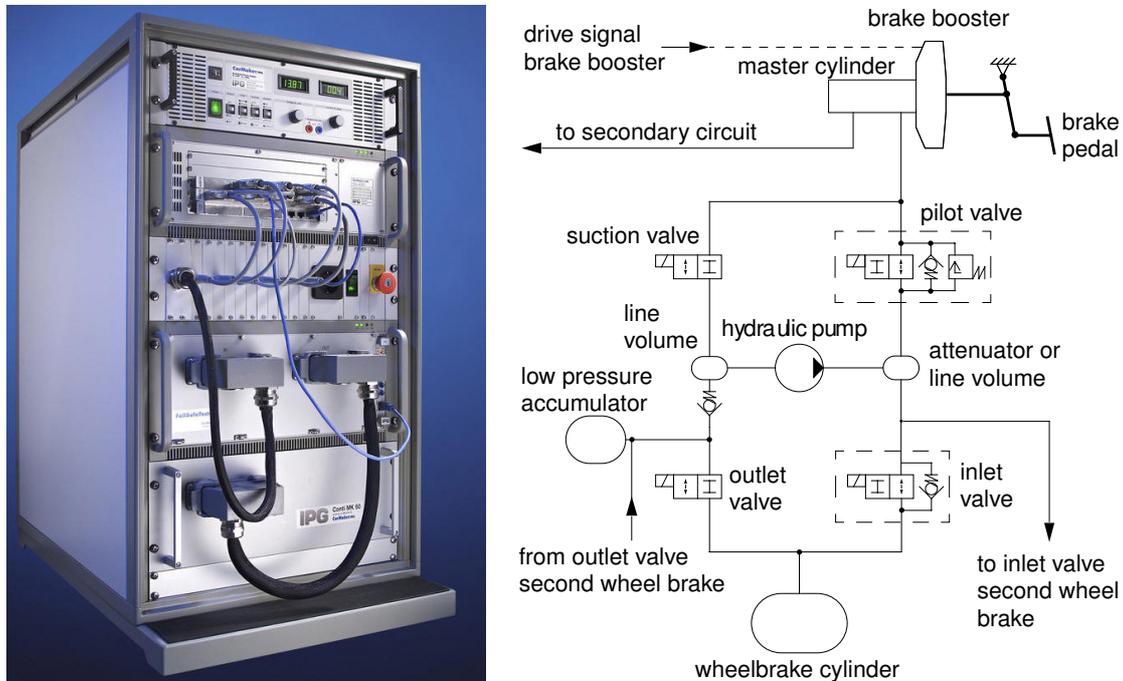


Abbildung 3.5.3: Hardware-in-the-Loop-Prüfstand für ein ESP-Steuergerät und Skizze des Hydraulik-Bremssystems (Quelle: IPG Automotive GmbH)

Querschleunigungs-, Gierraten-, Lenkradwinkel- und Bremsdruck-Sensor erfassen den Fahrerwunsch. Das ESP-Steuergerät ist an den Fahrzeug-CAN-Bus angeschlossen und kommuniziert hierüber mit der Motorelektronik. Auf dem Hardware-in-the-Loop-Simulator werden die Eingriffe des Steuergerätes an den Magnetspulen zur Ventilbetätigung, den Anschlüssen der Hydraulikpumpe und auf dem CAN-Bus erfasst und in die Simulation eingespeist. Die vom Hydraulikmodell berechnete Pumpen-Generatorspannung wird an den Anschlüssen der Hydraulikpumpe aufgeprägt.

3.5.6 Antriebsstrang

Der Antriebsstrang ist die zentrale Baugruppe im Fahrzeug, die auf Wunsch mit dem Motor Bewegungsenergie bereitstellt und diese zu den Rädern leitet, die sich mit den Reifen an der Fahrbahn abstützen. Für den Entwicklungsingenieur und Steuergeräteentwickler öffnet sich im Bereich Antriebsstrang ein weites Betätigungsfeld, mit dem er sich aus den unterschiedlichsten Blickwinkeln auseinandersetzt: Motorsteuergeräte werden auf minimalen Schadstoffausstoß und maximale Motorleistung hin optimiert. Im Verbund mit der Antriebs-Schlupf-Regelung ASR gilt es sowohl durchdrehende als auch

– auf glatter Strasse – beim Schalten oder Kuppeln blockierende Räder zu verhindern. Kupplungs- und Getriebesteuergeräte sorgen für die richtigen Motordrehzahlbereiche und für die Kraftübertragung vom Motor in den Triebstrang. In hochmotorisierten Fahrzeugen verhindern Steuergeräte eine Überlastung von Bauteilen des Antriebsstranges, wie z.B. der Kupplung, und verlängern damit deren Wartungsintervalle und Lebensdauer. Über Vierrad-Antrieb, Front- oder Heckantriebe mit geregelt angehängter zweiter Achse und unterschiedlichste Differentialsperren-Konstruktionen im Triebstrang wird für optimale Beschleunigung des Fahrzeugs gesorgt. Bei Verzögerungsvorgängen und speziell bei aktivem Eingriff der Fahrdynamikregelung über die Bremsanlage konfigurieren Steuergeräte den Antriebsstrang so um, dass die Antiblockier-Regelung unterstützt wird. Ziel ist die Entkopplung der einzelnen Räder, um jedes individuell an seiner Haftgrenze betreiben zu können. Durch gezielte Eingriffe in die Achsdifferentiale lässt sich das Eigenlenkverhalten anpassen und die Fahrstabilität steigern.

Innerhalb einer Fahrzeug-Baureihe existieren häufig unterschiedliche Antriebs-Konfigurationen: einzeln angetriebene Vorder- oder Hinterachse und Vierradantrieb mit über eine aktiv geregelte Kupplung angehängter zweiter Achse, manuelles Schaltgetriebe, automatisiertes Schaltgetriebe, stufenloses oder Automatikgetriebe. Je nach Aufgabenstellung sind unterschiedliche Modellierungsschwerpunkte zu setzen. Für automatisierte Testreihen ist die einfache Umkonfiguration der verschiedenen Modellvarianten wichtig.

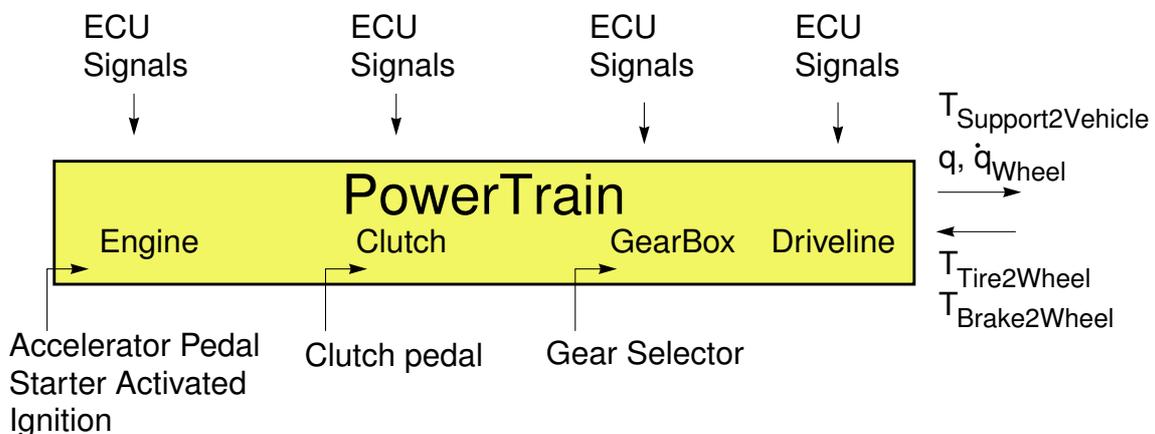


Abbildung 3.5.4: Schnittstelle des Antriebsstrang-Modules (Quelle: IPG Automotive GmbH [Rei04a])

Der Antriebsstrang ist als eigenständiges Modul in das Fahrzeugmodell integriert. Eingangsgrößen in das Antriebsstrang-Modul sind

- der Fahrerwunsch, geäußert über Gas- und Kupplungspedal, Gang- oder Fahrstufen-Wahlhebel, Zündung und Anlasserwunsch,

- Radbremsmomente,
- Reifenreaktionsmomente um die Raddrehachse.

Ausgangsgrößen des Antriebsstrang-Moduls sind

- Abstützmoment des Antriebsstranges am Fahrezugaufbau,
- Raddrehgeschwindigkeit, Motordrehzahl, verwendet vom Fahrer zur Geschwindigkeitsregelung und Wahl der Gangstufe,
- Antriebsmomente an den Rädern (optionale Ausgabe).

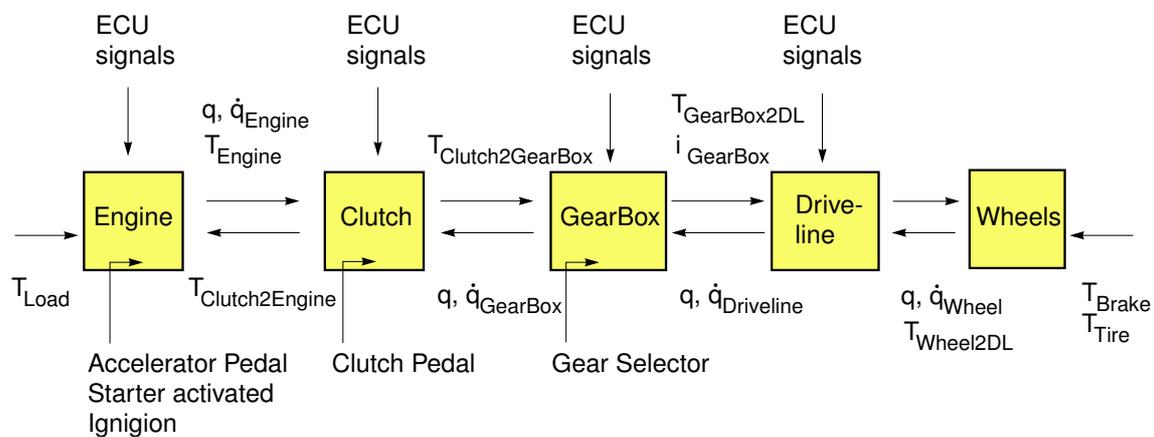


Abbildung 3.5.5: Struktur und Untermodule des Antriebsstranges (Quelle: IPG Automotive GmbH [Rei04a])

Intern ist der Antriebsstrang in die folgenden Module untergliedert (siehe hierzu auch Abbildung 3.5.5):

- **Motormoment:** Moment auf den Kupplungseingang als Funktion von Motordrehzahl, Fahrerwunschmoment, Zündung an/aus, Anlassermoment und Sekundärlasten (Klimakompressor, Servopumpe, ...),
- **Kupplung:** Vom Kupplungseingang auf ihren Ausgang bzw. den Getriebeeingang übertragenes Moment als Funktion der Kupplungsbetätigung,
- **Getriebe:** Übersetzungsverhältnis von Getriebeeingang zum Getriebeausgang als Funktion des gewählten Gangs bzw. der Fahrstufe,
- **Triebstrang und Räder:** Antriebsmomente an den Rädern als Funktion des Momentes am Triebstrang-Eingang,

- Antriebsstrang: Dynamische Berechnung der Rotoren des Antriebsstranges, vom Motor über Kupplung, Getriebe und Triebstrang bis hin zu den Rädern, unter Berücksichtigung aller Antriebsstrang-Baugruppen (siehe die Positionen direkt oberhalb) und der Momente, die aus der Bremsanlage, dem Reifen-Fahrbahn-Kontakt und aus der Aufhängungskinematik auf den Antriebsstrang wirken. Berechnung des resultierenden Abstützmomentes am Fahrzeug.

Über die Modell-Management-Schnittstelle (siehe Abschnitt 3.3.3) können die Simulationsmodelle für den Antriebsstrang selbst und die von ihm verwendeten Untermodule ausgewählt, parametrisiert und berechnet werden.

Vereinzelt treten in der Praxis Anwendungsfälle auf, bei denen die vorgeschlagene modular untergliederte Struktur des Antriebsstranges den Anforderungen nicht entspricht. Dies ist z.B. der Fall, wenn ein Steuergerät für seinen Betrieb einen speziellen Simulator erfordert, der an die Simulation anzukoppeln ist, oder wenn aus anderen Projekten bereits validierte Antriebsstrangmodelle als fertige Programmbibliotheken vorliegen und auf diese zurückgegriffen werden soll. Hierfür wird der "externe" Antriebsstrang als ein monolithisches Antriebsstrang-Modul über die Modell-Manager-Schnittstelle registriert. Das Modul enthält die volle Funktionalität und greift auf keine der ansonsten üblichen Untermodule Motor, Kupplung, Getriebe und Triebstrang über das Model-Management zurück. Modifikationen am Fahrzeugmodell sind hierfür nicht notwendig.

3.5.7 Aerodynamik

Die Aerodynamik eines Fahrzeugs beeinflusst seinen Kraftstoffverbrauch und seine Fahrstabilität. Aktuelle Entwicklungen gehen dahin, den Einfluss von Seitenwind, z.B. beim Verlassen des Windschattens neben einem Lkw, durch Lenk- und Bremseneingriffe auszugleichen und das Fahrzeug zu stabilisieren. Entsprechende Aerodynamik-Modelle betrachten die Geschwindigkeit und Anströmwinkel der das Fahrzeug umströmenden Luft und generieren hieraus Kräfte und Momente auf das Fahrzeug. Die fluiddynamischen Umströmungsvorgänge selbst sind bei dieser Fahrzeugstabilisierung nicht Gegenstand der Untersuchungen. Unabhängig von der gewählten Modellvorstellung lässt sich die Wirkung der Aerodynamik auf einen 6-dimensionalen Kraftwinder zurückführen. Parametrisiert werden solche Modelle üblicherweise durch Messungen im Windkanal, bei denen drei Kraft-Beiwerte (c_{Drag} , c_{Side} , c_{Lift}) und drei Momenten-Beiwerte (c_{Roll} , c_{Pitch} , c_{Yaw}) bez. eines Referenzpunktes, der meist in Fahrzeugmitte auf Fahrbahnniveau liegt, ermittelt werden. Eine andere verbreitete Variante ist die Parametrisierung über den Luftwiderstandsbeiwert c_w und zwei Abtriebsbeiwerte für die Vorder- und die Hinterachse.

Entlang der virtuellen Teststrecke können Windmaschinen aufgestellt werden. Ihre Windgeschwindigkeit, Windrichtung und Ausdehnung des (Seiten-)Windbereichs sind vorzugeben. Während der virtuellen Fahrt entlang der Teststrecke werden ständig für die aktuelle Fahrzeug-Position die Windverhältnisse ermittelt. Am Fahrzeug sind hierfür Trigger-Markierungen angebracht, über die das Passieren einer Windmaschine erkannt wird, bzw. über die die Windmaschinen aktiviert und deaktiviert werden.

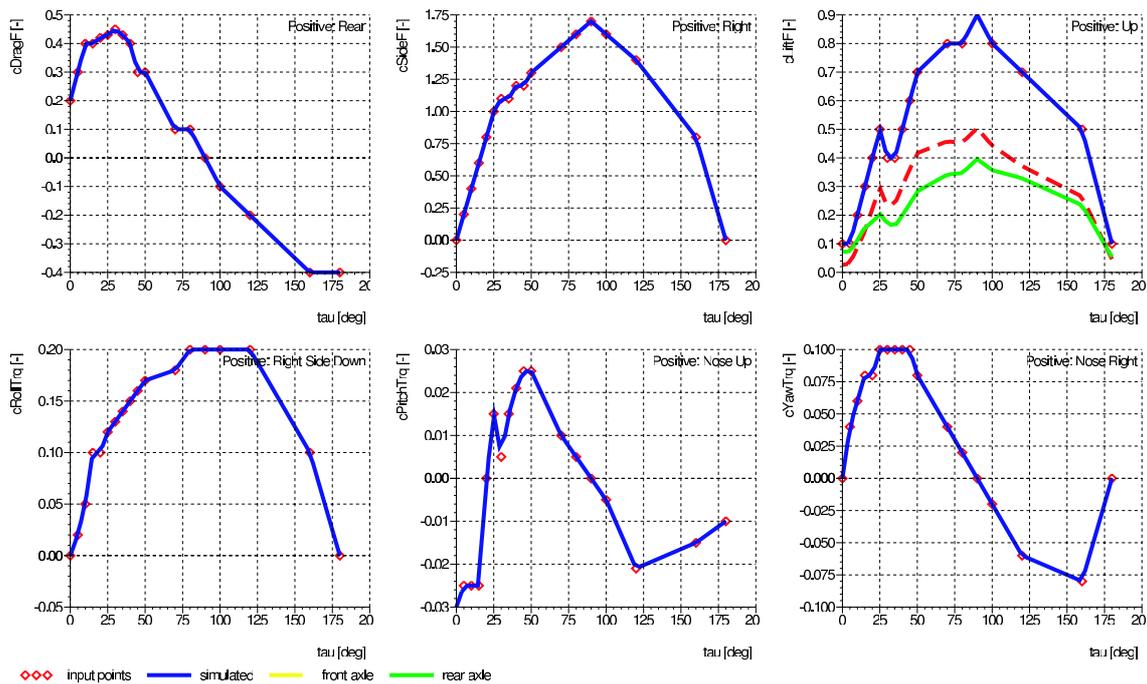


Abbildung 3.5.6: Aerodynamik-Beiwerte in Abhängigkeit vom Anströmwinkel τ

Das Fahrzeugmodell ermittelt auf Grundlage der aktuellen Fahrzeuggeschwindigkeit und der herrschenden Windverhältnisse den Vektor der Anströmgeschwindigkeit und den Anströmwinkel um die Hochachse des Fahrzeugs \bar{Z}_1 bez. \bar{X}_1 . Als Schnittstelle für die Aerodynamikwirkung verfügt das Fahrzeugmodell über einen 6-dimensionalen Kraftwinder. Über ihn sind auch beliebige andere Aerodynamik-Modelle integrierbar. So können z.B. gemessene oder synthetisch generierte Zeitverläufe eines Kraftwinders vorgegeben und zu einem beliebigen Zeitpunkt oder an einer beliebigen Fahrbahnposition in die Simulation eingespielt werden. Ein solcher “aus der Konserve” aufgeprägter Zeitverlauf ist ein ausgezeichnetes Werkzeug, Modifikationen am Reglercode eines Steuergerätes zu analysieren.

3.5.8 Anhänger

Die Sports Utility Vehicles, kurz SUV genannt, sind eine Fahrzeugklasse, die sich zur Zeit steigender Beliebtheit erfreut. Diese Fahrzeuge verfügen über hohe Fahrleistung und werden häufig als Gespann mit hohen Anhängerlasten betrieben, wie sie mit großen Boots, Wohn- oder Pferdeanhängern auftreten. Bei der Abstimmung der Fahrdynamikregler berücksichtigen die Fahrzeughersteller deshalb nicht nur das Fahrzeug alleine sondern verstärkt auch den Gespann-Betrieb.

Ein Pkw-Anhänger besitzt ein un gelenktes Fahrwerk. Es besteht aus einer Einzel-Achse oder einer Tandem-Achse, zwei nahe beieinander liegenden Achsen. Meist koppelt ein als Stabilisator ausgelegtes Achsrohr die rechte und linke Radaufhängung einer

Achse. Weit verbreitet ist die Kurbel- oder Schräglenker-Achs-Konstruktion. Im Simulationsmodell kann sie über wenige, einfach zu ermittelnde Geometrie-Größen direkt parametrisiert werden. Alternativ steht die Vorgabe der Radträgerbewegung über einfache lineare Gleichungen abhängig von rechter und linker Einfederung oder die Vorgabe über dreidimensionale (vermessen oder berechnet) Kennfelder – analog zum Fahrzeug – zur Verfügung. Der Aufbau des Anhängers ist als starrer Körper modelliert. Zusätzliche Beladungen sind frei positionierbar und während der Simulation verschieblich. Für die Aerodynamik stehen verschiedene Varianten zur Verfügung.

Pkw-Anhänger werden mit einer Deichsel an die Kugelgelenk-Kupplung am Pkw-Heck eingehängt. Die häufigste Kupplungsausführung ist die reine Kugelgelenk-Kupplung. Aufwändigere Kupplungskonstruktionen arbeiten zur Stabilisierung des Gespanns mit durch die Konstruktion virtuell verlagertem Kupplungspunkt, wie z.B. bei der Trapez-Kupplung. Kleine, leichte Anhänger sind ungebremst. Größere verfügen über eine Auflaufbremse, die über die Bewegung eines in die Deichsel integrierten Schubgelenks angesteuert wird. Zur Kopplung von Fahrzeug und Anhänger sind zwei Alternativen denkbar: Sie können

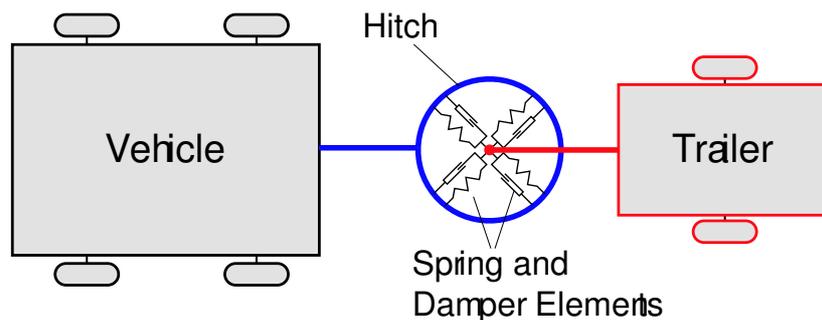


Abbildung 3.5.7: Kopplung von Fahrzeug und Anhänger

kinematisch über ein Gelenk verbunden werden. Dies bewirkt eine Kopplung beider Systeme auf Beschleunigungsebene. Die Simulation mit und ohne Anhänger erfordert für das Fahrzeug unterschiedliche Differentialgleichungen. Alternativ können Fahrzeug und Anhänger über äußere Kräfte gekoppelt werden. Beide Modelle werden unabhängig voneinander aufgebaut. Für die Anhängerkupplungskräfte werden in beiden Modellen entsprechende äußere Kräfte vorgesehen. Ihre Werte werden bei Betrieb mit Anhänger durch ein Feder-Dämpfer-Element ermittelt. Eine exakte, starre Kopplung lässt sich mit diesem Ansatz nicht realisieren. Da die Deichsel durch die Auflaufbremseinrichtung in Längsrichtung nachgiebig ist und die Kupplung in Realität Spiel besitzt, stellt diese Modellannahme keine unzulässige Einschränkung dar.

Gewählt wurde die Kopplungsvariante über äußere Kräfte. Ausschlaggebend war die Unabhängigkeit der Differentialgleichungen und dadurch der unkomplizierte Wechsel zwischen Simulation mit und ohne Anhänger. Außerdem lässt sich der Anhänger beim Wechsel auf ein anderes Fahrzeugmodell problemlos weiterverwenden.

Anmerkung: Der Ansatz einer Kopplung auf Beschleunigungsebene wurde für das Lenksystem verwendet, siehe Abschnitt 3.5.3.

3.5.9 Kinematik-Sensoren

In Fahrzeugen werden mit einer Vielzahl von Sensoren kinematische Zustandsgrößen erfasst: Die Beschleunigung des Fahrzeugaufbaus in Richtung seiner \bar{X}_1 -Längs- und seiner \bar{Y}_1 -Quer-Achse; die Rotationsgeschwindigkeit um seine \bar{Z}_1 -Hochachse; in \bar{Z}_1 -Richtung die Vertikalbeschleunigung des Fahrzeugaufbaus und der anderer Körper innerhalb der Radaufhängung, meist des Radträgers; die Position des Fahrzeugs auf dem befahrenen Kurs. Die Art und Anzahl der Sensoren und deren Montage ist hierbei fahrzeugvariantenspezifisch. Diese im realen Fahrzeug gemessenen Sensorsignale müssen die Simulations-

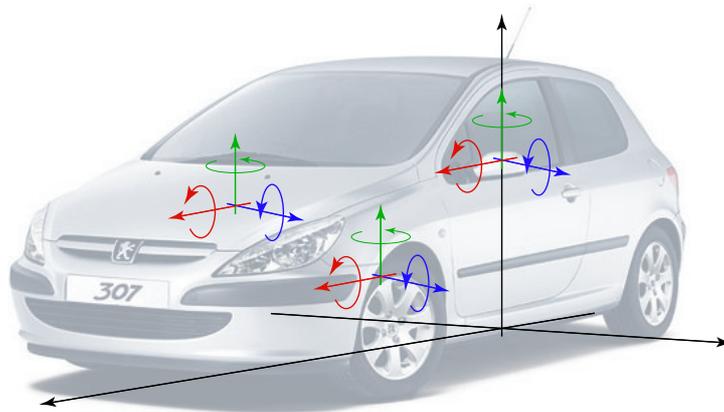


Abbildung 3.5.8: Kinematik-Sensoren

modelle bereitstellen. Für Beobachtersysteme und für zusätzlich in Versuchsfahrzeugen verbaute Messtechnik werden weitere Sensoren benötigt. Die Simulation selbst benötigt kinematische Größen, um z.B. Kraftgesetze zu stimulieren oder um Ereignisse auszulösen, wie z.B. den Bruch des Wankstabilisators beim Passieren eines Fahrbahnabschnitts oder beim Überschreiten eines Querbeschleunigungs-Schwellwertes.

Für den Entwicklungsingenieur ist es notwendig, individuell auf das konkrete Projekt zugeschnitten, sein Simulationsmodell mit allen benötigten Sensoren bestücken zu können. Unterstützt durch das Kinematik-Sensormodul reduziert sich die Aufgabe auf die Positionierung der Sensoren im Fahrzeugmodell, die Konfiguration der zu messenden Signale und die Formulierung des Sensor-Modells:

- **Sensorsignale:** Für folgende kinematische Größen stehen Sensoren zur Auswahl: Position, Translationsgeschwindigkeit, Translationsbeschleunigung, Orientierung, Rotationsgeschwindigkeit, Rotationsbeschleunigung. Beschleunigungssignale können mit oder ohne Berücksichtigung der Erdbeschleunigung ermittelt werden.

- Position und Orientierung des Sensors: Der Sensor wird auf einem Körper des Fahrzeugmodells montiert, meist dem Fahrzeugaufbau oder Radträger. Die Position und Ausrichtung wird relativ zu diesem Körper spezifiziert.
- Beobachtersystem: Als Koordinatensystem, von dem aus die Sensorgrößen beobachtet werden, stehen Fahrzeugaufbau \mathcal{F}_1 und das absolute System \mathcal{F}_0 zur Auswahl.
- Basis-System: Die Sensorsignale können in den Systemen \mathcal{F}_0 und \mathcal{F}_1 bereitgestellt werden.

Das Kinematik-Sensormodul liefert die an der Sensorposition vorhandenen physikalischen Größen, d.h. die mit dem realen Sensor zu erfassenden “Sollwerte”. Ein unter Kenntnis des Messprinzips und der Elektronik formuliertes Sensor-Modell kann nachgeschaltet werden und die eigentlichen Ausgabesignale des realen Sensors liefern. Bei gebräuchlichen Beschleunigungssensoren reduziert sich diese Aufgabe meist auf die Limitierung des Wertebereichs und die Transformation der Beschleunigung in eine elektrische Spannung. Die meist ungeliebte und vor allem auch fehlerträchtige Berechnung skalarer, geometrischer Parameter zwischen zwei Koordinatensystemen im dreidimensionalen Raum und deren ersten und zweiten zeitlichen Ableitungen ist vollständig in das Kinematik-Sensormodul ausgelagert. Sie steht dem Simulationsingenieur in Form von Funktionsblöcken zur Verfügung, die die benötigten Sensorsignale direkt als fertige Ergebnisse liefern. Relative Messgrößen wie Abstände zweier Punkte, Drehwinkel oder Differenzgeschwindigkeiten um spezielle Achsen, werden mit Hilfe zusätzlicher Funktionsblöcke nachgeschaltet ermittelt.

C-Code Interface

Jeder Sensor stellt seine Signale über folgende Ausgabe-Datenstruktur `struct tKinSens` zur Verfügung:

```

struct tKinSens {
    struct {
        double Pos[3];      /* position                */
        double Vel[3];      /* translational velocity  */
        double Acc[3];      /* translational acceleration */
        double Phi[3];      /* orientation ZYX          */
        double Omega[3];    /* rotational velocity      */
        double Alpha[3];    /* rotational acceleration  */
    } FrA,                  /* decomposed in FrameAbsolut Fr0/FrA */
    Fr1;                   /* decomposed in FrameCarBody Fr1      */

    double Tr2FrA[3][3];   /* transformation matrix to FrA        */
    double Tr2Fr1[3][3];   /* transformation matrix to Fr1        */
}

```

Zu Beginn der Simulation wird ein Sensor erzeugt und parametrisiert durch den einmaligen Aufruf der Funktion `KinSens_Start()`.

```
struct tKinSens *
KinSens_Start (
    const char    *MntBdyName, /* sensor is mounted on this body */
    double        Pos[3],      /* mount position                */
    double        Rot_ZYX[3], /* mount orientation             */
    unsigned char SigMask[10]); /* selection of sensor signal    */
```

`KinSens_Start()` liefert ein Handle vom Datentyp `struct tKinSens` zurück, das auf die Datenstruktur, in der der Sensor seine Signale bereitstellt, verweist. Außerdem referenziert es Sensor-Modul-interne Parameter, die zur Berechnung für diesen Sensor benötigt werden.

Die Sensorsignale werden mit der Funktion

```
int KinSens_Calc (struct tKinSens *KinSens, double dt);
```

berechnet. Sie greift direkt auf kinematische Basisgrößen des Fahrzeugmodells zurück. Durch Verwendung von Hilfsgrößen aus dem zuvor gelösten MKS-Formalismus, wie dem Beschleunigungsdistriktor \overline{A}_j^i , Gleichung (4.2.11), reduziert sich der numerische Aufwand bei der Signalberechnung deutlich gegenüber einer Berechnung ausschließlich aus den kinematischen Basisgrößen.

Nach Ende der Simulation wird der Sensor wieder freigegeben durch den Aufruf der Funktion

```
void KinSensor_End (struct tKinSens *KinSens);
```

3.5.10 Kraft- und Momenten-Manager

Grundgerüst des Fahrzeugmodells bildet ein speziell entwickeltes MKS-Modell. Seine Struktur ist fix, d.h. die Entwicklung des Kern-Modells ist abgeschlossen. Eine MKS-Entwicklungsumgebung mit Eingriffsmöglichkeit in die Modellstruktur, wie sie z.B. die Programme MESA VERDE [WWS90], ADAMS [Orl87] oder SimPack bieten, ist für den an der Entwicklung von Fahrdynamik-Regelsystemen interessierten Anwender nicht notwendig. Dennoch kann es bisweilen notwendig sein, für Aktuatoren neuer Regelkonzepte oder spezielle Simulationsszenarien zusätzliche, benutzerdefinierte, eingeprägte, vektorielle oder generalisierte Kräfte und Momente einzuführen. Folgende Beispiele aus der Simulationspraxis seien genannt:

- Kräfte zwischen einzelnen Körpern: Häufig im Umfeld aktiver Fahrwerke, z.B. um einen aktiven Wankstabilisator zu realisieren, der sich zwischen linker und rechter Radaufhängung abstützt.
- Kräfte auf einzelne Körper: Ankopplung weiterer externer Teilmodelle, meist am Fahrzeugaufbau, z.B. eine Hebebühne, die das Fahrzeug über vier unabhängige

Stempel aufbockt, ein Ständer eines Motorrades oder ein erweitertes Aerodynamikmodell mit mehreren Angriffspunkten.

- Aufbau- \mathcal{F}_1 -feste Kraft: Das Fahrzeug wird von einer zusätzlichen Antriebskraft in einen Geschwindigkeitsbereich hinein beschleunigt, den es aus eigenem Antrieb nicht erreichen könnte. Die Kraft wirkt stets in Fahrzeug-Längsrichtung.
- Inertial- \mathcal{F}_0 -feste Kraft/Moment: Seitenwind wirkt kurzzeitig als Störgröße auf den Fahrzeugaufbau ein. Die Störung behält ihre Richtung unabhängig von der Fahrzeugbewegung.

Allgemein ausgedrückt handelt es sich um eingeprägte Kräfte und Momente, die auf verschiedene Teile des Fahrzeugs einwirken. Die Definition des Kraftgesetzes erfolgt relativ zu einem Koordinatensystem, seine Wirkung relativ zu einem weiteren. Diese Erweiterungen des MKS-Systems um eingeprägte Kräfte und Momente muss aus dem Anwendercode heraus aktivierbar sein und darf keine Neugenerierung des Gleichungssystems des Fahrzeugmodell-Kerns erfordern.

Der hier eingesetzte und speziell auf die Topologie des Fahrzeugmodells ausgelegte MKS-Formalismus bietet für diese Funktionalität die entsprechenden Schnittstellen: Jeder Körper besitzt $N \geq 0$ eingeprägte Kraftwinder, die er an das ihm übergeordnete, generalisierte System weiterleitet. Der Anwender fügt einfach zu den N Fahrzeugmodell-eigenen Kraftwindern, die auf einen bestimmten Körper wirken, M eigene hinzu. Für jeden Kraftwinder sind folgende Angaben erforderlich:

- Körper, an dem der Kraftwind angreift,
- Punkt auf dem Körper, auf den sich die Momentenwirkung bezieht,
- System, in dem der Kraftwinder vorliegt. Zur Auswahl stehen das Galileische System \mathcal{F}_0 , der Fahrzeugaufbau \mathcal{F}_1 und das jeweils körpereigene System $\mathcal{F}_{\text{this}}$.

Während der Simulation weist er den Kraftwindern die gewünschten Werte zu. Zu ihrer Berechnung kann er auf z.B. mit Hilfe des Kinematik-Sensor-Moduls (siehe Abschnitt 3.5.9) ermittelte Messgrößen (Position, Orientierung, Geschwindigkeit, Abstand, Differenzwinkel) zurückgreifen. Der Softrunner-Algorithmus übernimmt anschließend die Weiterverarbeitung der Kraftwinder im MKS-Modell des Fahrzeugs.

Kapitel 4

Analytische Fahrzeugdynamik

4.1 Motivation und Aufgabenstellung

Die in Kapitel 3 vorgestellte Simulationsumgebung dient der Entwicklung von Steuergeräten für Pkw, auch im Gespann mit einem Anhänger, und für Motorräder. Ein Schwerpunkt liegt hierbei auf Fahrdynamik-Regelsystemen, wie z.B. dem Antiblockiersystem, dem Stabilitätsprogramm, der Dämpferregelung, der Wankstabilisierung, der Servo- oder Aktivlenkung, der Motorsteuerung, der Antriebsschlupfregelung u.v.m. Für die Entwicklung dieser Regelsysteme werden Simulationsmodelle benötigt, die den Reglern ihre normale Arbeitsumgebung, bestehend aus Sensoren, Aktuatoren und Streckenmodellen, vortäuschen. Im Zentrum der Simulationsmodelle steht das Fahrzeugmodell. Während der Konzeptphase wird es sich hierbei um vereinfachte Modelle für z.B. eine Matlab/Simulink-Umgebung handeln. Die Modelle und ihre Parametrierung orientieren sich mit fortschreitender Entwicklung zusehends am konkreten Fahrzeug und den eingesetzten Sensoren und Aktuatoren. Spätestens für den Betrieb von Hardware-Prototypen und Seriensteuergeräten unter Echtzeitbedingungen auf Hardware-in-the-Loop-Simulatoren werden validierte Simulationsmodelle benötigt, die das reale Systemverhalten umfassend abbilden.

Der Entwickler von Fahrdynamik-Regelsystemen – Regelungstechniker, Elektronik-Ingenieur oder Versuchsfahrer – benötigt während des gesamten Entwicklungsprozesses in seiner Entwicklungsumgebung Simulationsmodelle, die seinen Erfordernissen entsprechen: Die Modelle müssen einfach in der Handhabung sein. Sie müssen anschaulich und übersichtlich parametrierbar sein und dürfen nicht den Spezialisten aus der Berechnungsabteilung erfordern. Für einzelne Tests sind Modifikationen wie z.B. die Änderung der Dämpferhärte, der Beladung oder des Radstandes oder ein schiefziehendes Fahrwerk vorzunehmen. Auf dem Hardware-in-the-Loop-Simulator muss sich in der virtuellen Umgebung alles genau so abspielen, wie es auch im realen Fahrzeug geschieht. Überwacht ein Regler das Öffnen des Kofferraums oder der Türen, so müssen diese auch in der Simulation betätigt werden und jeder Test beginnt mit stillstehendem Fahrzeug auf dem Parkplatz und nicht z.B. mit 100 Stundenkilometern direkt vor Bremsbeginn in der Kurve. Für die Entwicklung relevante Simulationsszenarien sind u.a.:

- Funktionstests mit stillstehendem Fahrzeug. Hier wird z.B. die Zündung an- und wieder ausgeschaltet und das Aufleuchten der Lampen im Display während 100000 Betätigungen kontrolliert.
- Open-Loop-Manöver auf der Fahrdynamikfläche, wie z.B. Lenkwinkelsprung oder Bremsung.
- Closed-Loop-Manöver, wie z.B. stationäre Kreisfahrt oder Spurwechsel.
- Das Befahren von realen, vermessenen Alpenpass- oder Rennstrecken, auf denen die Regler unter extremen Bedingungen betrieben werden.

Bei diesen Tests müssen die Simulationsmodelle häufig mehrere hundert Kilometer zurücklegen und viele Stunden am Stück ohne Unterbrechung in Betrieb sein. Sind Tests und Parametersätze einmal in einer Simulationsumgebung aufgebaut, so sollen sie auch für die anderen, z.B. den Hardware-in-the-Loop-Simulator, überspielt und den dort erzielten Ergebnissen direkt verglichen werden können. Für diesen Daten- und Ergebnisaustausch müssen während des Entwicklungsprozesses in den verschiedenen Umgebungen die gleichen Simulationsmodelle verfügbar sein. D.h. die Modelle müssen in verschiedenen Umgebungen (standalone, Matlab/Simulink, Hardware-in-the-Loop), unter verschiedenen Betriebssystemen (Linux, Windows, LynxOS) und auf unterschiedlichen Rechner-Hardware-Plattformen (Intel, Motorola) laufen. Dies lässt sich nur realisieren, wenn die Modelle mit allen verwendeten Modulen im Quellcode vorliegen. Außerdem müssen sie ohne eigene Oberfläche und standalone, d.h. ohne eine Modell-Entwicklungsumgebung/-Oberfläche, lauffähig sein. Bei der Erprobung von Regelstrategien ist es wichtig, den Regler mit sehr vielen unterschiedlichen Szenarien zu konfrontieren und darüber seine Robustheit für den Praxiseinsatz abzusichern. Hierzu sind schnell berechenbare Modelle, die über große Distanzen, mehrere 100 Kilometer entfernt vom Ursprung des Koordinatensystems, und während lange andauernder Simulationsläufe genauso gut funktionieren, wie auf den ersten hundert Metern, unerlässlich.

Aus diesen Randbedingungen leiten sich folgende Anforderungen für die benötigten Fahrzeugmodelle ab:

- Aufgabe des Fahrzeugmodells ist die Simulation des fahrdynamischen Verhaltens des Fahrzeugs im Zusammenspiel mit aktiven Regelsystemen. Sein Einsatz zur Entwicklung oder Optimierung von z.B. Achskonstruktionen und zur Auslegung von Motor-Gummilagern ist nicht zu berücksichtigen.
- Das Fahrzeugmodell muss alle Berechnungen dreidimensionaler, räumlicher Kinematik abdecken. Wird ein Beschleunigungssignal an einer bestimmten Position im Fahrzeug benötigt, so soll es ausreichen, an dieser Stelle im Modell einen Sensor zu positionieren. Das gewünschte Signal wird dann vom Simulationsmodell bereitgestellt.

- Die Modelle sollen herstellerübergreifend einsetzbar sein und die verschiedenen herstellereigenen konstruktiven Besonderheiten abdecken. So sind u.a. unterschiedliche Antriebskonzepte, wie z.B. Front-, Heck-, Vierradantrieb oder Hang-On-Antriebe, in verschiedensten Konstruktionen und Konfigurationen genauso zu unterstützen, wie manuelle oder automatische Schaltungen oder konventionelle und aktive Fahrwerke und Lenksysteme.
- Zu Parametrierung geometrischer Größen sind am realen Fahrzeug einfache ermittelbare Koordinatensysteme zu definieren. Sie müssen mit Kreide und Meterstab bestimmt werden können.
- Die Parametrierung soll auch auf Basis von Messungen erfolgen: Dies bedeutet für die Fahrwerkskinematik, dass sie auf Grundlage von Kinematik-Prüfstandsmessungen über die Bewegung des Radträgers über Vorspur, Sturz etc. unabhängig von der Konstruktion der Achse vorgegeben werden soll. Einzelne Baugruppen wie z.B. der Motor, der Triebstrang oder der Tank sollen als einzelne generalisierte Körper mit Position, Masse und Trägheit parametrierbar sein. Über fahrzeugaufbau fest positionierbare Trimm-Beladungen sollen Radlasten und Trägheitstensor des Gesamtfahrzeugs abgestimmt werden können.
- Die Fahrzeugmodelle sollen über eine einheitliche, modulare Struktur verfügen. Dies bezieht sich sowohl auf ihre Parametrierung als auch auf die Namensgebung für ihre Beobachtersignale.

Eine bauteil- und baugruppenorientierte Modellierung ist anzustreben. Sie vereinfacht die Datenbeschaffung und den Austausch von Parametersätzen zwischen den beteiligten Entwicklungspartnern, z.B. des Datensatzes für eine Fahrwerksachse, für den Antriebsstrang oder für das Lenksystem. Eine bauteilorientierte Parametrierung führt meist auch zu einem aus der Praxis bekannten und anschaulich nachvollziehbaren Systemverhalten: Wird der Abstand der Federangriffspunkte verkürzt, erhöht sich die Federkraft mit der Folge, dass bei ansonsten gleicher Systemkonfiguration ausgefedert wird; zusätzliche Beladungen führen dazu, dass Reifen und Fahrwerk stärker belastet und dadurch zusammengedrückt werden.

- Regelsysteme greifen fast immer über Aktuatoren in das Systemverhalten ein. Die Systemdynamik dieser Aktuatoren ist sehr speziell und wird im Regelalgorithmus entsprechend berücksichtigt. Aktive Baugruppen, wie z.B. eine Luftfederung, eine elektrische Lenkung oder der Booster zur Bremsbetätigung durch einen ACC-Regler, erfordern deshalb für die konkrete Realisierung speziell entwickelte Teilmodelle. Teilweise sind auch für die Sensoren des Regelsystems eigene Sensor-Simulationsmodelle notwendig. Diese Spezialmodelle müssen in das Fahrzeugmodell als zusätzliche Modelle oder als Ersatz für vorhandene Modellteile integrierbar sein. Bei der Anwendung soll zwischen Standard-Teilmodellen und projektspezifischen Erweiterungen nicht unterschieden werden; sie sollen einheitlich parametrierbar und angewählt werden. Einige der notwendigen Schnittstellen und Funktionalitäten

wurden in Kapitel 3 bereits vorgestellt: Lenksystem in Abschnitt 3.5.3, Fahrwerkskinematik in Abschnitt 3.5.2, Krafterelemente (Feder, Dämpfer, Stabilisator) in Abschnitt 3.5.2, Aerodynamik in Abschnitt 3.5.7, Antriebsstrang in Abschnitt 3.5.6, virtueller Kraftwinder in Abschnitt 3.5.10, versuchabhängige Beladung in Abschnitt 3.5.1, Kinematiksensoren in Abschnitt 3.5.9.

- Für die Hardware-in-the-Loop-Simulation ist eine Modell-Zykluszeit von typischerweise ≤ 1 ms erforderlich. Das Rechenzeitverhalten der Modelle muss deterministisch sein. Die Modelle sind auf minimalen Rechenzeit- und Speicherbedarf und hohe Modellgüte hin auszurichten.

In den nachfolgenden Abschnitten wird eine einheitliche “Modellplattform” für Pkw, Pkw-Anhänger und Motorrad vorgestellt, die entsprechend obiger Anforderungen und Randbedingungen entwickelt wurde. Sie eignet sich deshalb optimal für den Einsatz zur Entwicklung von Fahrdynamik-Regelsystemen.

Die Modellstruktur und viele Detailspekte der vorgestellten Modellplattform wurden vom Autor bereits im Programmpaket CarMaker umgesetzt. Der MKS-Code der CarMaker-Fahrzeugmodelle wurde mit dem Gleichungsgenerierer MESA VERDE [WWS90] modelliert.

In Abschnitt 4.2 werden notwendige Grundlagen aus dem Bereich der Kinematik von Gelenkketten, der Relativkinematik, der Massengeometrie und dem Aufstellen von Bewegungsgleichungen nach dem Prinzip von D’Alembert zusammengetragen. In Abschnitt 4.3 wird die Topologie für die Kraftfahrzeuge Pkw, Anhänger und Motorrad vorgestellt. Die Wahl der Koordinaten zur Beschreibung der Bewegung des Fahrzeugs im Raum und innerhalb des Fahrzeugs ist Gegenstand von Abschnitt 4.4. Nach der Diskussion der verschiedenen Terme der Massenmatrix in Abschnitt 4.5 werden in Abschnitt 4.6 die Bewegungsgleichungen der Fahrzeugmodelle am Beispiel des Pkws formuliert. Eine Zusammenfassung in Abschnitt 4.7 beschließt das Kapitel.

4.2 Grundlagen

4.2.1 Topologie offener Gelenkketten

Abbildung 4.2.1 zeigt eine offene Gelenkkette oder Baumstruktur. Einzelne starre Körper sind miteinander über Gelenke gekoppelt. Die Gelenke besitzen 0 bis 6 Freiheitsgrade. B_0 sei die Wurzel, d.h. der (ruhende) Bezugskörper (“Inertialsystem”). Die Körper werden regulär, d.h. beginnend mit B_0 , “zu den Blättern hin” aufsteigend durchnummeriert.

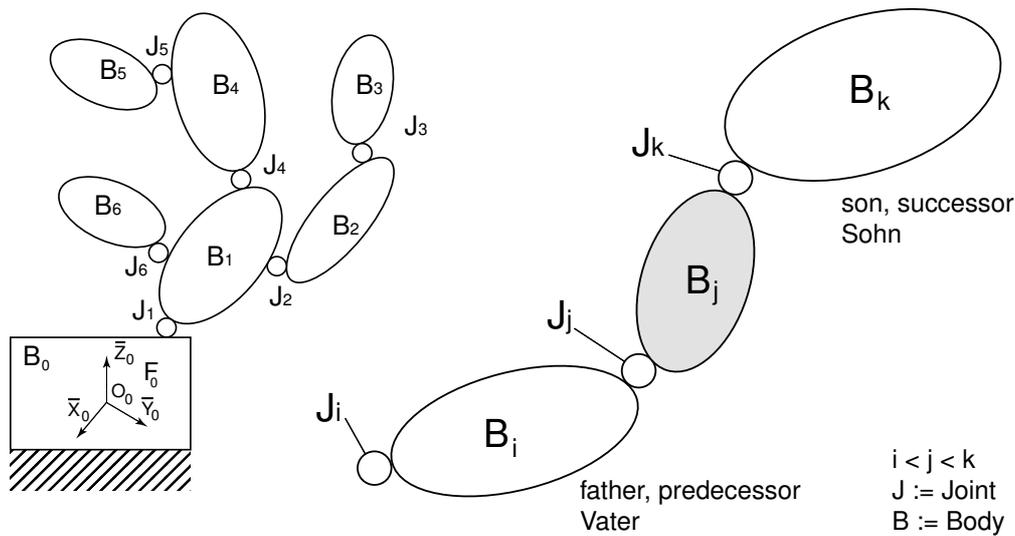
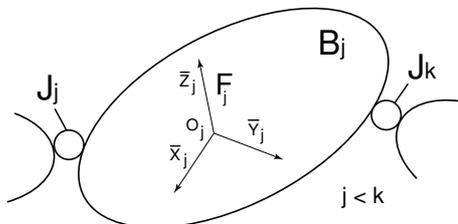


Abbildung 4.2.1: Offene Gelenkkette: Körper (Body) $B_i, B_j, B_k, i < j < k$

4.2.1.1 Koordinatensystem \mathcal{F}_j (Frame)



Jeder Körper B_j führt ein mit ihm stets fest verbundenes, zunächst beliebig festgelegtes, kartesisches, rechtssinniges Koordinatensystem (Frame) \mathcal{F}_j mit dem Ursprung O_j und den drei Basisvektoren $\bar{X}_j, \bar{Y}_j, \bar{Z}_j$ mit sich:

$$\mathcal{F}_j \equiv \{O_j, \bar{X}_j, \bar{Y}_j, \bar{Z}_j\} \quad . \quad (4.2.1)$$

Die Lage und Orientierung des Körpers B_j in Bezug auf seinen Vorgänger $B_i, i < j$, wird durch die generalisierten, direkten Koordinaten q_j im Gelenk J_j festgelegt.

4.2.1.2 Pfadmatrix D_{ij}

Mit Bezug auf Abbildung 4.2.1 sei folgende Sprechweise definiert: Der Körper \mathcal{B}_i ist “Vater” oder “direkter Vorgänger” von \mathcal{B}_j . Der Körper \mathcal{B}_j ist “Sohn” oder “direkter Nachfolger” von \mathcal{B}_i .

Diese Beziehung wird in einer “Ist direkter Vorgänger von” Matrix $\mathbf{D}_{N \times N}$ abgelegt, deren Elemente definiert sind über

$$D_{ij} = \begin{cases} 1 & \text{wenn } \mathcal{B}_i \text{ direkter Vorgänger von } \mathcal{B}_j \text{ ist} \\ 0 & \text{sonst} \end{cases}. \quad (4.2.2)$$

4.2.1.3 Generalisierter Körper \mathcal{G}_j (Generalized Body)

Der generalisierte Körper \mathcal{G}_j wird rekursiv wie folgt definiert [Ren80]:

$$\mathcal{G}_j \equiv \mathcal{B}_j \cup_{\{k | D_{jk}=1\}} \mathcal{G}_k \quad (4.2.3)$$

Der generalisierte Körper \mathcal{G}_j umfasst somit das gesamte Subsystem “jenseits” des Gelenkpunktes O_j .

4.2.2 Notation und Formelapparat

Umfassende und übersichtliche Darstellungen des Formelapparates sind unter anderem bei Papastavridis [Pap02] und Brossard [Bro94], [Bro95] zu finden.

4.2.2.1 Synthetische Kinematik

$\bar{v}^i(P)$ Geschwindigkeit des Punktes P beobachtet vom System \mathcal{F}_i

$\bar{v}_j^i(P)$ Geschwindigkeit desjenigen \mathcal{F}_j -festen Punktes, der im betrachteten Zeitpunkt mit P koinzidiert, beobachtet vom System \mathcal{F}_i

$\bar{a}^i(P)$ Beschleunigung des Punktes P beobachtet vom System \mathcal{F}_i

$$\bar{a}^i(P) \equiv \frac{{}^i d}{dt} \bar{v}^i(P) \quad (4.2.4)$$

$\bar{a}_j^i(P)$ Beschleunigung desjenigen \mathcal{F}_j -festen Punktes, der im betrachteten Zeitpunkt mit P koinzidiert, beobachtet vom System \mathcal{F}_i

$\bar{\Omega}_j^i$ Rotationsgeschwindigkeit des Systems \mathcal{F}_j beobachtet vom System \mathcal{F}_i

$\bar{\alpha}_j^i$ Winkelbeschleunigung des Systems \mathcal{F}_j beobachtet vom System \mathcal{F}_i

$$\bar{\alpha}_j^i \equiv \frac{{}^i d}{dt} \bar{\omega}_j^i \quad (4.2.5)$$

$\hat{\Omega}_j^i$ Rotationsgeschwindigkeitstensor, $\hat{\Omega}_j^i \bar{n} = \hat{\Omega}_j^i \times \bar{n}$, vgl. Gleichung (1.3)

$\hat{\alpha}_j^i$ Winkelbeschleunigungstensor, vgl. Gleichung (1.3)

Beobachtersystemwechsel (Rotationsgeschwindigkeit, Rotationsbeschleunigung, Translationsgeschwindigkeit, Translationsbeschleunigung):

$$\bar{\Omega}_k^i = \bar{\Omega}_j^i + \bar{\Omega}_k^j \quad (4.2.6)$$

$$\bar{\alpha}_k^i = \bar{\alpha}_j^i + \bar{\alpha}_k^j + \bar{\Omega}_j^i \times \bar{\Omega}_k^j \quad (4.2.7)$$

$$\bar{v}_k^i(P) = \bar{v}_j^i(P) + \bar{v}_k^j(P) \quad (4.2.8)$$

$$\bar{a}_k^i(P) = \bar{a}_j^i(P) + \bar{a}_k^j(P) + 2 \bar{\Omega}_j^i \times \bar{v}_k^j(P) \quad (4.2.9)$$

Geschwindigkeitsfeld eines starren Körpers:

$$\bar{v}_j^i(P) = \bar{v}_j^i(M) + \bar{\Omega}_j^i \times \overline{MP} = \bar{v}_j^i(M) + \hat{\Omega}_j^i \overline{MP} \quad (4.2.10)$$

$\overline{\overline{A}}_j^i$ Beschleunigungsdistributor von \mathcal{F}_j bez. \mathcal{F}_i

$$\overline{\overline{A}}_j^i \equiv \hat{\alpha}_j^i + \hat{\Omega}_j^i \hat{\Omega}_j^i \quad (4.2.11)$$

Hierbei ist $\hat{\Omega}_j^i$ der in Gleichung (1.3) eingeführte Operator.

Beschleunigungsfeld eines starren Körpers:

$$\begin{aligned} \bar{a}_j^i(P) &= \bar{a}_j^i(M) + \bar{\alpha}_j^i \times \overline{MP} + \bar{\Omega}_j^i \times (\bar{\Omega}_j^i \times \overline{MP}) \\ &= \bar{a}_j^i(M) + \overline{\overline{A}}_j^i \cdot \overline{MP} \end{aligned} \quad (4.2.12)$$

Transformationsmatrix $\mathbf{T}_{i \rightarrow j}$

Zur Transformation von Größen von einem Bezugssystem in ein anderes wird eine Transformationsmatrix $\mathbf{T}_{j \rightarrow i}$ eingeführt. Sie überträgt die Komponenten des Vektors \bar{n} , dargestellt im System \mathcal{F}_j , in die Komponenten dargestellt im System \mathcal{F}_i . Sie ist definiert durch

$$[\bar{n}]_i = \mathbf{T}_{j \rightarrow i} [\bar{n}]_j \quad (4.2.13)$$

$$[\bar{n}]_j = \mathbf{T}_{j \rightarrow i}^T [\bar{n}]_i \quad (4.2.14)$$

Die Transformationsmatrix $\mathbf{T}_{j \rightarrow i}$ ist eine Funktion generalisierter Koordinaten: $\mathbf{T}_{j \rightarrow i} \equiv \mathbf{T}_{j \rightarrow i}(\bar{q})$.

4.2.2.2 Analytische Kinematik

\mathcal{G}_i Materielles System i

\overline{OM} Vektor vom Punkt O zum Punkt $M \cdot \in \mathcal{G}$

$\mathbf{q} \equiv (q_1, \dots, q_n)$

Holonome oder globale oder Lagrangesche oder System-Koordinaten

$\overline{OM} \equiv \overline{O_i M}(q_1, \dots, q_n)$

Lagrangesche Darstellung des Ortsvektors \overline{OM}

π_i, ω_i i -te Quasi-Koordinate bzw. i -te Quasi-Geschwindigkeit, siehe Gleichung (4.2.50)

$\frac{i \partial \bar{n}}{\partial q_j}$ Partielle Ableitung des Vektors \bar{n} nach der Koordinate q_j bez. des Systems i

$\bar{e}_i(M)$ "Begleitvektor" [Heu02] oder "Gradientenvektoren" [Pap02]

$$\bar{e}_i(M) \equiv \frac{{}^0 \partial \overline{OM}}{\partial q_i} \quad (4.2.15)$$

$\bar{e}_{i,j}(M)$

$$\bar{e}_{i,j}(M) \equiv \frac{{}^0 \partial \bar{e}_i}{\partial q_j} = \frac{{}^0 \partial}{\partial q_j} \left(\frac{{}^0 \partial \overline{OM}}{\partial q_i} \right) = \frac{{}^0 \partial^2 \overline{OM}}{\partial q_j \partial q_i} \quad (4.2.16)$$

$\bar{\epsilon}_i(M)$

$$\bar{\epsilon}_i(M) \equiv \frac{{}^0 \partial \overline{OM}}{\partial \pi_i} \quad (4.2.17)$$

$\bar{\epsilon}_{i,j}(M)$

$$\bar{\epsilon}_{i,j}(M) \equiv \frac{{}^0 \partial \bar{\epsilon}_i}{\partial \pi_j} = \frac{{}^0 \partial}{\partial \pi_j} \left(\frac{{}^0 \partial \overline{OM}}{\partial \pi_i} \right) = \frac{{}^0 \partial^2 \overline{OM}}{\partial \pi_j \partial \pi_i} \quad (4.2.18)$$

$\bar{v}^i(M), \bar{a}^i(M)$

Für die Geschwindigkeiten und Beschleunigungen von $M \cdot \in \Sigma$ bez. System \mathcal{F}_i erhält man [Pap02]

$$\bar{v}^i(M) = \sum_{j=1}^N \frac{{}^i \partial \overline{OM}}{\partial \pi_j} \omega_j, \quad (4.2.19)$$

$$\bar{a}^i(M) = \sum_{j=1}^N \frac{{}^i \partial^2 \overline{OM}}{(\partial \pi_j)^2} \dot{\omega}_j + \sum_{j,k=1}^N \frac{{}^i \partial^2 \overline{OM}}{\partial \pi_j \partial \pi_k} \omega_j \omega_k. \quad (4.2.20)$$

Im Falle von $\mathcal{F}_i = \mathcal{F}_0$ wird mit $\frac{\partial \overline{OM}}{\partial \pi_j} \equiv \frac{\partial \overline{OM}}{\partial \pi_j}$ abkürzend geschrieben:

$$\overline{v}^0(M) = \sum_{i=1}^N \frac{\partial \overline{OM}}{\partial \pi_i} \omega_i \quad , \quad (4.2.21)$$

$$\overline{a}^0(M) = \sum_{i=1}^N \frac{\partial^2 \overline{OM}}{(\partial \pi_i)^2} \dot{\omega}_i + \sum_{i,j=1}^N \frac{\partial^2 \overline{OM}}{\partial \pi_i \partial \pi_j} \omega_i \omega_j \quad . \quad (4.2.22)$$

Mit Gleichung (4.2.17) und Gleichung (4.2.18) ergibt sich die kompakte Schreibweise:

$$\overline{v}^0(M) = \sum_{i=1}^N \overline{\epsilon}_i \omega_i \quad , \quad (4.2.23)$$

$$\overline{a}^0(M) = \sum_{i=1}^N \overline{\epsilon}_i \dot{\omega}_i + \sum_{i,j=1}^N \overline{\epsilon}_{i,j} \omega_i \omega_j \quad . \quad (4.2.24)$$

$\mathcal{T}_j^i(M)$

Geschwindigkeitsfeld (twist) des starren Körpers

Das Geschwindigkeitsfeld des starren Körpers \mathcal{B}_j reduziert auf den Punkt M ist:

$$\mathcal{T}_j^i(M) = \left[\begin{array}{c} \overline{\Omega}_j^i \\ \overline{v}_j^i(M) \end{array} \right] \quad . \quad (4.2.25)$$

Hierbei ist $\overline{\Omega}_j^i$ die Rotationsgeschwindigkeit und $\overline{v}_j^i(M)$ die Geschwindigkeit des Punktes M . Über die Eulersche Formel

$$\overline{v}_j^i(N) = \overline{v}_j^i(M) + \overline{\Omega}_j^i \times \overline{MN} \quad (4.2.26)$$

wird die Geschwindigkeit für beliebige Punkte N des Raumes und somit das Geschwindigkeitsfeld festgelegt. Das Feld besitzt i.Allg. die ausgezeichnete Achse $\overline{A} + \mu \overline{\Omega}_j^i$ auf der $\overline{\Omega}_j^i$ und $\overline{v}_j^i(\overline{A})$ parallel sind. Der Bezugspunkt \overline{A} auf der Achse ist bestimmt durch

$$\overline{A} = \overline{M} - \frac{1}{(\overline{\Omega}_j^i)^2} \overline{v}_j^i(M) \times \overline{\Omega}_j^i \quad . \quad (4.2.27)$$

4.2.3 Massengeometrie

Folgende Schreibweisen wurden bereits vereinbart:

m^i Masse des Körpers \mathcal{B}_i

(O_i, M^i)

Punkt(-körper) O_i der Masse M^i

4.2.3.1 Generalisierter Körper \mathcal{G}_j (Generalized Body)

Die Trägheitstensoren 0., 1. und 2. Ordnung des in Abschnitt 4.2.1.3 eingeführten generalisierten Körpers \mathcal{G}_j bezogen auf den Punkt O_j lauten:

$$M^j \equiv \int_{\mathcal{G}_j} dm \quad (4.2.28)$$

$$\bar{U}^j \equiv \int_{\mathcal{G}_j} \overline{O_j M} dm \quad (4.2.29)$$

$$\bar{\bar{\Pi}}^j \equiv \int_{\mathcal{G}_j} \overline{O_j M} \otimes \overline{O_j M} dm \quad (4.2.30)$$

$$\bar{I}^j \equiv - \int_{\mathcal{G}_j} \widehat{\overline{O_j M}} \cdot \widehat{\overline{O_j M}} dm \quad (4.2.31)$$

Der aus dem dyadischen Produkt berechnete Trägheitstensor $\bar{\bar{\Pi}}^j$ wird als Binet'scher Trägheitstensor bezeichnet. \bar{I}^j ist der (klassische) Segner'sche Trägheitstensor. Von Fayet und Pfister [Fay90], [FP94], [Pfi95] werden zusätzlich die sich auf zwei Punkte beziehende Trägheitstensoren eingeführt:

$$\bar{U}^{ij} \equiv \int_{\mathcal{G}_j} \overline{O_i M} dm \quad (4.2.32)$$

$$\bar{\bar{\Pi}}^{ij} \equiv \int_{\mathcal{G}_j} \overline{O_i M} \otimes \overline{O_j M} dm \quad (4.2.33)$$

Falls $B_j \in \mathcal{G}_i$, $i \neq j$, d.h. der Bezugspunkt O_i "vor" \mathcal{G}_j liegt, gilt

$$\bar{U}^{ij} = \overline{O_i O_j} M^j + \bar{U}^{jj} \quad (4.2.34)$$

Wenn $i < j$ und $D_{ij} = 1$ gilt für den Trägheitstensor 2. Ordnung:

$$\bar{\bar{\Pi}}^{ij} = \overline{O_i O_j} \otimes \bar{U}^{jj} + \bar{\bar{\Pi}}^{jj} \quad (4.2.35)$$

$$\bar{\bar{\Pi}}^{ij} = \bar{\bar{\Pi}}^{ji T} \quad (4.2.36)$$

4.2.3.2 Erweiterter Körper \mathcal{B}_j^+ (Augmented Body)

Der erweiterte Körper \mathcal{B}_j^+ [RW66] [Wit77] [Ren80] besteht aus dem Körper \mathcal{B}_j und den Punkt-Körpern (O_k, M^k) , $(O_l, M^l), \dots$, wobei $\mathcal{B}_k, \mathcal{B}_l, \dots$ Söhne von \mathcal{B}_j sind:

$$\mathcal{B}_j^+ \equiv \mathcal{B}_j \cup_{\{k|D_{jk}=1\}} (O_k, M^k) \quad (4.2.37)$$

\mathcal{B}_j^+ ist somit der um die Punktkörper (O_k, M^k) erweiterte Körper \mathcal{B}_j . O_k, \dots sind hierbei die Gelenkpunkte der äußeren generalisierten Nachbarkörper \mathcal{G}_k, \dots und M^k, \dots deren Massen.

Die Trägheitstensoren 1. und 2. Ordnung des erweiterten Körpers \mathcal{B}_j^+ werden wie folgt angeschrieben und berechnet:

$$\bar{u}_+^j \equiv \int_{\mathcal{B}_j^+} \overline{O_j M} \, dm \quad (4.2.38)$$

$$= \int_{\mathcal{B}_j} \overline{O_j M} \, dm + \sum_k D_{jk} \overline{O_j O_k} M^k$$

$$\bar{\pi}_+^j \equiv \int_{\mathcal{B}_j^+} \overline{O_j M} \otimes \overline{O_j M} \, dm \quad (4.2.39)$$

$$= \int_{\mathcal{B}_j} \overline{O_j M} \otimes \overline{O_j M} \, dm + \sum_k D_{jk} \overline{O_j O_k} \otimes \overline{O_j O_k} M^k$$

4.2.3.3 Kinetik und Dynamik

$\bar{\Sigma}_{\mathcal{G}_j}^i$ Die *dynamische Summe* eines generalisierten Körpers \mathcal{G}_j bez. eines Beobachtersystems i ist definiert als

$$\bar{\Sigma}_{\mathcal{G}_j}^i \equiv \int_{\mathcal{G}_j} \bar{a}^i(M) \, dm \quad . \quad (4.2.40)$$

$\bar{\delta}_{\mathcal{G}_k}^i(O_j)$

Das auf den Punkt O_j reduzierte *dynamische Moment* eines generalisierten Körpers \mathcal{G}_k oder Systems Σ_k bez. des Beobachtersystems i ist definiert als

$$\bar{\delta}_{\mathcal{G}_k}^i(O_j) \equiv \int_{\mathcal{G}_k} \overline{O_j M} \times \bar{a}^i(M) \, dm \quad . \quad (4.2.41)$$

$\mathcal{W}_{i \rightarrow j}(P)$

Kraftwinder

Die von einem Körper \mathcal{B}_i auf \mathcal{B}_j wirkenden Kräfte und Momente lassen sich zu einer als Kraftwinder oder Wrench $\mathcal{W}_{i \rightarrow j}(P)$ bezeichneten Größe zusammenfassen. $\mathcal{W}_{i \rightarrow j}(P)$ ist im Allgemeinen definiert als Tupel zweier Vektoren, der Kraft (force) $\bar{F}_{i \rightarrow j}$ und des auf den Punkt P reduzierten Momentes (torque) $\bar{T}_{i \rightarrow j}(P)$

$$\mathcal{W}_{i \rightarrow j}(P) = \begin{bmatrix} \bar{F}_{i \rightarrow j} \\ \bar{T}_{i \rightarrow j}(P) \end{bmatrix} \quad . \quad (4.2.42)$$

Die Größen Wrench $\mathcal{W}_{i \rightarrow j}(P)$ und Twist $\mathcal{T}_j^i(P)$ gestatten es, die bei einer Bewegung verrichtete Leistung (power) in nachfolgender, einfacher Weise als Co-Moment durch Über-Kreuz-Multiplizieren zu formulieren:

$$\begin{aligned} \text{power} &= \mathfrak{C}(\mathcal{W}_{i \rightarrow j}(P), \mathcal{T}_j^i(P)) \\ &= \bar{F}_{i \rightarrow j} \cdot \bar{v}_j^i(P) + \bar{T}_{i \rightarrow j}(P) \cdot \bar{\Omega}_j^i \end{aligned} \quad (4.2.43)$$

4.2.4 Massengeometrische Rekursionen

Eine wesentliche Motivation für die zuvor eingeführten, gruppierenden Strukturen *generalisierter* und *erweiterter Körper* ist die Möglichkeit zur effizienten, rekursiven Berechnung massengeometrischer Größen. Der numerische Aufwand bei der Berechnung von Mehrkörpersystemen lässt sich mit ihrer Hilfe deutlich reduzieren.

Der generalisierte Körper \mathcal{G}_j und der erweiterte Körper \mathcal{B}_j^+ haben identische Masse. Sie setzt sich aus der Summe der Masse des Körpers \mathcal{B}_j selbst und der aller direkt nachfolgenden generalisierten oder erweiterten Körper zusammen.

$$M^j \equiv \int_{\mathcal{G}_j} dm \quad (4.2.44)$$

$$M^j = m^j \cup_{\{k|D_{jk}=1\}} M^k \quad (4.2.45)$$

$$M^j = M^{\mathcal{G}_j} = M^{\mathcal{B}_j^+} \quad (4.2.46)$$

Die Trägheitstensoren 1. und 2. Ordnung gliedern sich für $i = j$ in

$$\begin{aligned} \overline{U}^{jj} &= \int_{\mathcal{G}_j} \overline{O}_j \overline{M} dm = \int_{\mathcal{B}_j} \overline{O}_j \overline{M} dm + \sum_k D_{jk} \int_{\mathcal{G}_k} \overline{O}_j \overline{M} dm \\ &= \int_{\mathcal{B}_j} \overline{O}_j \overline{M} dm + \sum_k D_{jk} \overline{O}_j \overline{O}_k M^k + \sum_k D_{jk} \overline{U}^{kk} \\ &= \overline{u}_+^{jj} + \sum_k D_{jk} \overline{U}^{kk} \end{aligned} \quad (4.2.47)$$

$$\overline{\overline{\Pi}}^{jj} = \overline{\overline{\pi}}_+^{jj} + \sum_k \left\{ D_{jk} \left(\overline{O}_j \overline{O}_k \otimes \overline{U}^{kk} + \overline{U}^{kk} \otimes \overline{O}_j \overline{O}_k + \overline{\overline{\Pi}}^{kk} \right) \right\} . \quad (4.2.48)$$

4.2.5 Hamel-Transformation und Maggi-Matrix

Für die Parametrierung eines frei beweglichen Koordinatensystems \mathcal{F}_1 ist es bisweilen zweckmäßig, an Stelle der allgemeinen Geschwindigkeitskomponenten $\dot{\mathbf{q}} = d\mathbf{q}/dt$ lineare, nicht-integrable Kombinationen derselben, sogenannte Quasigeschwindigkeiten, einzuführen [LN27]. Als klassisches Beispiel hierfür sei die Orientierungsänderung zweier Koordinatensysteme zueinander genannt: Die Orientierung zweier Systeme zueinander, die über ein Gelenk miteinander verbunden sind, kann durch drei Euler-Winkel beschrieben werden. Auf Geschwindigkeitsebene, d.h. für die relative Rotationsgeschwindigkeit, bietet es sich häufig an, von den (generalisierten) Euler-Winkel-Koordinaten auf den $\overline{\Omega}$ -Vektor der Quasi-Geschwindigkeiten $\omega_i, i = 1, 2, 3$ überzugehen:

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix}_1 = \begin{bmatrix} \sin \vartheta \sin \phi & \cos \phi & 0 \\ \sin \vartheta \cos \phi & -\sin \phi & 0 \\ \cos \vartheta & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{\psi} \\ \dot{\vartheta} \\ \dot{\phi} \end{bmatrix} \quad (4.2.49)$$

Hierbei sind ω_1, ω_2 und ω_3 die Komponenten des Rotationsgeschwindigkeitsvektors $\overline{\Omega}_1^0$ bei einer Zerlegung im Koordinatensystem \mathcal{F}_1 . $\dot{\psi}, \dot{\vartheta}$ und $\dot{\phi}$ sind die Änderungen der drei Eulerwinkel einer sukzessiven Drehung mit der Drehreihenfolge $Z - X - Z$.

Dieser Zusammenhang führt, allgemein formuliert, auf lineare, kinematische Differentialgleichungen für die Quasi-Geschwindigkeiten

$$\boldsymbol{\omega} = \mathbf{H} \dot{\boldsymbol{q}} \quad . \quad (4.2.50)$$

Die im folgenden als *Hamel-Matrix* bezeichnete Matrix \mathbf{H} [Ham49] [Ham04] ist eine Funktion der generalisierten Koordinaten \boldsymbol{q} .

$$\mathbf{H} = \mathbf{H}(\boldsymbol{q}) \quad (4.2.51)$$

Die Transformation mit einer Hamel-Matrix \mathbf{H} kommt z.B. bei der Formulierung der globalen Bewegung des Fahrzeugs in Gleichung (4.4.5) zum Einsatz.

Bestehen zwischen den Geschwindigkeiten $\dot{\boldsymbol{q}}$ Z Zwangsbedingungen der Form $\mathbf{C}(\boldsymbol{q}) = 0 \rightarrow \mathbf{C}(\boldsymbol{q}) \dot{\boldsymbol{q}} = 0$, so gilt

$$\dot{\boldsymbol{q}} = \mathbf{M} \dot{\boldsymbol{q}}_{\text{indep.}} \quad , \quad (4.2.52)$$

wobei $\dot{\boldsymbol{q}}_{\text{indep.}}$ die Spaltenmatrix der $N - Z$ unabhängigen Variablen darstellt. Pfister [Pfi95] prägte für diese Matrix $\mathbf{M} \equiv \mathbf{M}(\boldsymbol{q})$ den Begriff *Maggi-Matrix* [Mag96] [Mag03].

Eingesetzt in die Hamel-Gleichung (4.2.50) ergeben sich aus den unabhängigen generalisierten Koordinaten direkt die Quasi-Geschwindigkeiten $\boldsymbol{\omega}$ [Pap02]:

$$\boldsymbol{\omega} = \mathbf{H} \mathbf{M} \dot{\boldsymbol{q}}_{\text{indep.}} \quad (4.2.53)$$

4.2.6 Algebraisierung der Kinematik

Bei infinitesimaler Variation der Quasi-Koordinate π_i geht die Bewegung des generalisierten Körpers \mathcal{G}_i in eine Starrkörperbewegung über:

$$\bar{\boldsymbol{e}}_i(M) = \bar{\boldsymbol{p}}_i + \bar{\boldsymbol{r}}_i \times \overline{\boldsymbol{O}_i M} \quad , \quad (4.2.54)$$

$$\bar{\boldsymbol{e}}_{i,j}(M) = \bar{\boldsymbol{r}}_i \times \bar{\boldsymbol{p}}_j + \bar{\boldsymbol{r}}_i \times (\bar{\boldsymbol{r}}_j \times \overline{\boldsymbol{O}_i M}) \quad . \quad (4.2.55)$$

$\bar{\boldsymbol{p}}_i \equiv \bar{\boldsymbol{p}}_i(M)$ ist der translatorische (prismatic) Anteil und $\bar{\boldsymbol{r}}_i$ der rotatorische (revolute) Anteil der geometrischen Schraube der Quasi-Koordinaten π_i (siehe Abschnitt 4.2.2.2). Für ein reines Drehgelenk $(O_i, \bar{\boldsymbol{r}}_i)$ ergibt sich der Ausdruck $\bar{\boldsymbol{e}}_i(M) = \bar{\boldsymbol{r}}_i \times \overline{\boldsymbol{O}_i M}$. Für ein reines Schubgelenk ergibt sich $\bar{\boldsymbol{e}}_i(M) = \bar{\boldsymbol{p}}_i$.

Für die Geschwindigkeit $\bar{\boldsymbol{v}}^0(M)$ mit Gleichung (4.2.54) algebraisiert ergibt sich in expliziter Form

$$\bar{\boldsymbol{v}}^0(M) = \sum_{i=1}^N (\bar{\boldsymbol{p}}_i + \bar{\boldsymbol{r}}_i \times \overline{\boldsymbol{O}_i M}) \dot{q}_i \quad . \quad (4.2.56)$$

Für den in dieser Arbeit verwendeten Formalismus ist speziell der Fall $M \cdot \in \mathcal{G}_i$ von Interesse. Falls $M \cdot \notin \mathcal{G}_i$ gilt $\bar{\boldsymbol{p}}_i = \bar{\mathbf{0}}, \bar{\boldsymbol{r}}_i = \bar{\mathbf{0}} \rightarrow \bar{\boldsymbol{e}}_i(M) = \bar{\mathbf{0}}$.

4.2.7 Bewegungsgleichung nach dem Prinzip von D'Alembert/Lagrange

Für beliebige Systeme von endlichen Freiheitsgraden ergeben sich aus dem Prinzip von D'Alembert/Lagrange die folgenden N Bewegungsgleichungen:

$$\int_{\Sigma} (\bar{a}^0(M) dm(M) - \bar{F}(M)) \cdot \bar{\epsilon}_i(M) = 0 \quad . \quad (4.2.57)$$

Hierbei sind

$dm \equiv dm(M)$	die Masse eines Partikels im Punkt M
$\bar{a}^0(M)$	die Beschleunigung des Masseteilchens dm bez. des Galileischen Systems \mathcal{F}_0 ,
$\bar{F}(M)$	die äußeren, eingepprägten Kräfte und Momente, die an dem Masseteilchen dm im Punkt M angreifen und
$\bar{\epsilon}_i(M) = \frac{\partial \bar{O}M}{\partial q_i}$	der Gradientenvektor des Masseteilchens dm .

Diese Formulierung ist gültig für starre Systeme, Systeme starrer Körper und beliebige Kontinua von endlichem Freiheitsgrad. Die Berechnung von $\bar{F}(M) \cdot \bar{\epsilon}_i(M)$ erfolgt als Summation über diskrete Kräfte und Momente.

Gegenstand der nachfolgenden Abschnitte ist die direkte Auswertung der Integrale I_k

$$I_k \equiv \int_{\Sigma} \bar{a}^0(M) dm(M) \cdot \bar{\epsilon}_k(M) \quad , \quad (4.2.58)$$

mit $\bar{a}^0(M) = \sum_{i,j} \bar{\epsilon}_i \ddot{w}_i + \bar{\epsilon}_{i,j} \dot{w}_i \cdot \dot{w}_j$.

4.3 Topologie des Kraftfahrzeugs

4.3.1 Grundidee

Die im Rahmen dieser Arbeit betrachteten Fahrzeuge – Pkw, Pkw-Anhänger und Motorrad – besitzen trotz all ihrer augenfälligen Unterschiede sehr viele Gemeinsamkeiten: Sie werden von einem Fahrer gelenkt, bewegen sich allesamt auf der Oberfläche von Strassen fort, besitzen eine Karosserie, ein Fahrwerk, eine Lenkung, Räder und Reifen, werden von einem Motor angetrieben und durch eine Bremsanlage verzögert, verfügen über die unterschiedlichsten Steuergeräte u.v.m. Die Grundidee dieser Arbeit ist es, diese Gemeinsamkeiten bei der Modellierung zu nutzen und eine einheitliche, von der konkreten Ausprägung des Fahrzeugs unabhängige Strategie zu verfolgen. Durch die Wiederverwendung der gleichen Teilmodelle und Funktionsblöcke wird der Entwicklungsaufwand reduziert ihre Zuverlässigkeit erhöht, da Fehler schneller aufgedeckt und beseitigt werden. Der Einsatz der Modelle in unterschiedlichen Betriebsbereichen – z.B. kleine Wankwinkel beim Pkw und sehr große beim Motorrad – erhöht ebenfalls die Modellqualität,

Zuverlässigkeit und Vertrauenswürdigkeit der Modelle. Für den Anwender bedeutet ein einheitliches Konzept bei Struktur, Parametrierung, Namensgebung und Programmierschnittstelle eine deutliche Reduktion des Einarbeitungs- und Schulungsaufwandes und der Fehlerrate bei der Parametrierung und der Interpretation von Simulationsergebnissen.

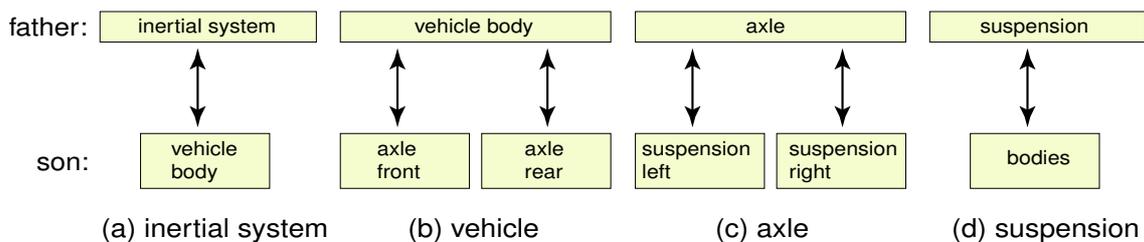


Abbildung 4.3.1: Vater-Sohn Strukturen auf verschiedenen Detaillierungsebenen: (a) Galileisches System \mathcal{F}_0 – Fahrzeugaufbau, (b) Fahrzeugaufbau – Vorderachse, Fahrzeugaufbau – Hinterachse, (c) Achse – Radaufhängung links, Achse – Radaufhängung rechts, (d) Radaufhängung – Bauteil der Aufhängung

Das komplexe Gesamtsystem *Fahrzeug* wird in möglichst einfache, funktional getrennte Subsysteme mit wenigen, klaren Beziehungen zerlegt. Von der globalen Bewegung des Fahrzeugs als generalisiertes Ganzes wird Detaillierungsebene für Detaillierungsebene in sein Inneres eingedrungen und ein generalisiertes System nach dem anderen in seine Bestandteile aufgelöst. Der angewendete Formalismus ist davon unabhängig, ob es sich bei den einzelnen Systemen um starre oder um elastisch verformbare Teilsysteme handelt.

4.3.2 Topologische Struktur des Fahrzeugs

Den im Rahmen dieser Arbeit behandelten Fahrdynamikmodellen wird die folgende einheitliche topologische Struktur zugrunde gelegt. :

Das Fahrzeug besitzt Baumstruktur. Die Wurzel bildet das Galileische System \mathcal{F}_0 . Den Stamm bildet der Fahrzeugaufbau. Er wird im Folgenden \mathcal{F}_1 (frame) bzw. \mathcal{B}_1 (body) genannt. \mathcal{F}_1 kommt die Funktion einer mitbewegten Referenz oder Träger-Plattform zu: Seine Bewegung bez. des absoluten Systems \mathcal{F}_0 beschreibt die Bewegung des Gesamtfahrzeugs als generalisiertes System. Mit \mathcal{F}_1 sind alle anderen Substrukturen des Fahrzeugs verbunden und über ihn miteinander gekoppelt. Eine üblicherweise gelenkte Vorder- und eine Hinterachse bilden das Fahrwerk. Eine Achse kann aus einer einzelnen oder aus zwei Radaufhängungen, einer linken und einer rechten, bestehen. Jede Radaufhängung wird an ihrem \mathcal{F}_1 -festen Bezugspunkt (mount point) fix in die Karosserie eingehängt. Die Radaufhängung selbst setzt sich aus dem Körper Radträger, der Achsfederung und der Achsdämpfung und einer frei wählbaren Anzahl weiterer Körper zusammen. Die Konstruktionsweise der Radaufhängung ist nicht näher spezifiziert, z.B. ob als Einzelradaufhängung, als gelenktes Vorderrad oder als links-rechts gekoppelte

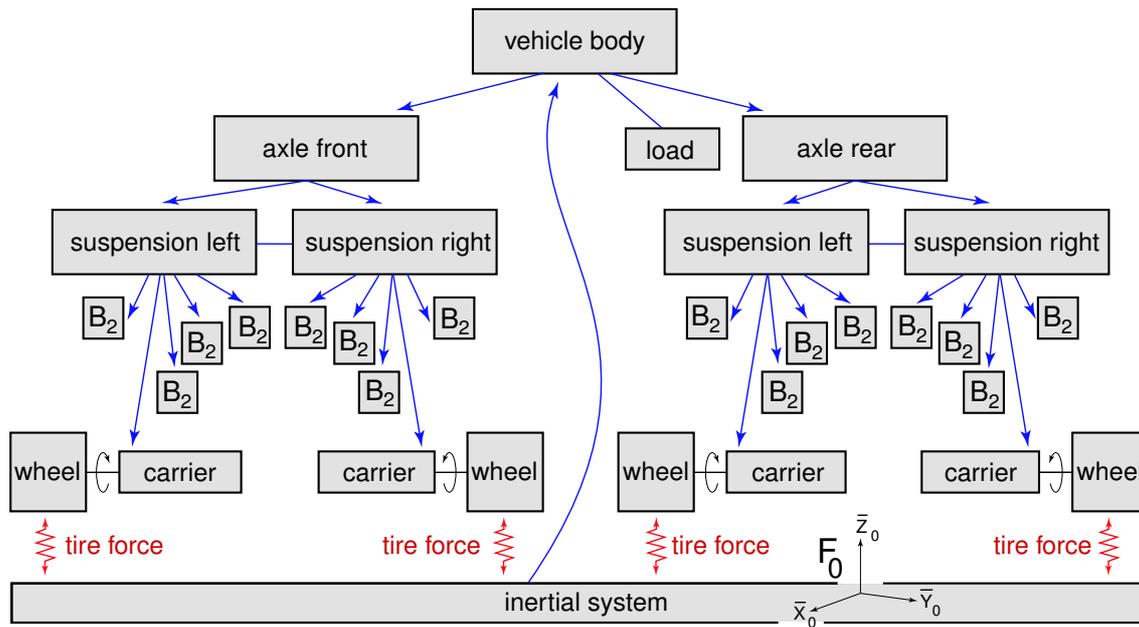


Abbildung 4.3.2: Kinematische Grundstruktur des Fahrdynamikmodells eines Pkw

Verbundlenker-Konstruktion. Unabhängig von ihrer konstruktiven Realisierung ist die Hauptaufgabe der Radaufhängung die Führung des Radträgers bzw. des Rades bez. der Karosserie. Darüber hinaus stellt sie die Schnittstellen für Achsfeder, Achsdämpfer und Wankstabilisator und zum Lenksystem bereit. Das Rad, als äußerstes Element der Baumstruktur, ist am Radträger drehbar gelagert. Zwischen Rad und System \mathcal{F}_0 wirken die Reifenkräfte. Eine kinematische Bindung existiert nicht. Die Kopplung Fahrzeug-Umwelt findet ausschließlich auf Basis eingprägter Kräfte statt. Im Modell sind zusätzliche Beladungen (loads) vorgesehen. Sie sind gegenüber dem Fahrzeugaufbau beweglich. Mit ihrer Hilfe lassen sich ausgehend von einem Fahrzeug in Basis-Konfiguration unterschiedliche Beladungszustände parametrieren oder Veränderungen an der Massegeometrie des Fahrzeugs über der Zeit realisieren, wie z.B. ein sich leerender Tank oder verrutschende Ladungen.

Die funktionalen Einheiten Lenksystem, Antriebsstrang, Bremsanlage, Aerodynamik, Verbraucher und Elektrik werden vom Fahrzeug getrennt modelliert und an dieses angekoppelt.

Ein so modelliertes "Fahrzeug-Skelett" bildet das Grundgerüst für flexibel erweiterbare oder einfach integrierbare, Fahrdynamiksimulation. Dem Ingenieur oder Wissenschaftler wird es mit seiner Hilfe ermöglicht, sich auf die Fragestellungen aus seinem Spezialgebiet zu konzentrieren: Für die komplette Dynamik und räumliche Kinematik des Fahrzeugs und alle hierzu gehörenden Signale kann er auf ein robustes, über klar definierte, einfache Schnittstellen nutzbares, schnelles Fahrzeugmodell zurückgreifen und seine eigenen Subsystem-Modelle auf diese bereitgestellte Fahrdynamik-Plattform integrieren.

4.3.3 Fahrdynamikmodell eines Pkw – “Car”

Die Topologie des Pkw-Fahrzeugmodells “Car” ist in Abbildung 4.3.3 skizziert. Seine

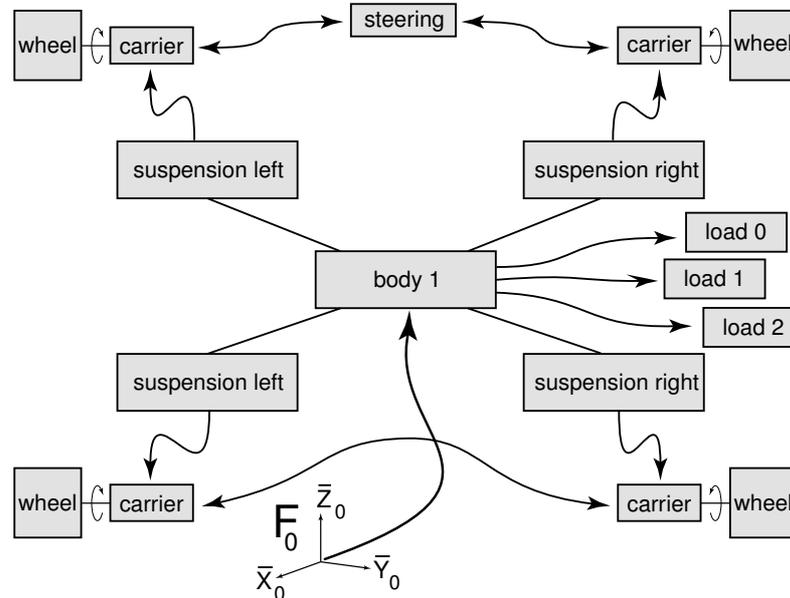


Abbildung 4.3.3: Fahrdynamikmodell “Car”

wesentlichen Elemente sind: der Fahrzeugaufbau, System \mathcal{F}_1 und Körper \mathcal{B}_1 , bewegt bez. des stets ruhenden Systems \mathcal{F}_0 , das Fahrwerk mit einer Vorder- und einer Hinterachse, bestehend aus linker und rechter Radaufhängung, parametrisiert gegenüber \mathcal{F}_1 und vier Rädern. Die vorderen Radaufhängungen sind über das *Lenksystem* (steering system) miteinander gekoppelt: einfach kinematisch abhängig vom Lenkradwinkel oder dynamisch auf Beschleunigungsebene oder über Kräfte abhängig vom Lenkmoment und/oder den Aktuatoren des Lenksystems. Details hierzu wurden in Abschnitt 3.5.3 besprochen. Die hinteren Aufhängungen sind, im Falle einer Einzelradaufhängung voneinander unabhängig, bei einer Aufhängung als Verbundlenker-Konstruktion miteinander gekoppelt. Das Fahrzeug verfügt über mehrere bez. des Fahrzeugaufbaus frei positionierbare und bewegliche Beladungs-Körper mit veränderlicher Massengeometrie.

Das Modul *Antriebsstrang* treibt die Räder über Antriebsmomente an. Es stützt sich am Fahrzeugaufbau ab. Das Modul *Bremsanlage* generiert Bremsmomente zwischen Rad und Radträger. Die Aerodynamik greift über Kräfte und Momente (*Aerodynamik-Modul*) am Fahrzeugaufbau an. Sie wirken im System \mathcal{F}_1 . Ein Anhänger kann an das Fahrzeug angehängt werden. Er generiert in der Anhängerkupplung Kräfte und Momente auf den Fahrzeugaufbau. Äußere Kraft- und Momenten-Einwirkungen auf das Fahrzeug, wie z.B. Windböen, Stöße durch Kollision oder Antriebskräfte, die den fahrzeugeigenen Antrieb unterstützen, lassen sich mittels zweier auf den Fahrzeugaufbau wirkender “virtuelle”

Kraftwinder, der eine im ruhenden Galileischen System \mathcal{F}_0 und der andere im aufbaufesten System \mathcal{F}_1 , realisieren.

4.3.4 Fahrdynamikmodell eines Pkw-Anhängers – “Trailer”

Die Topologie von Pkw-Anhängern ist in Abbildung 4.3.4 skizziert. Sie besitzen im Gegensatz zum Pkw keine aktive Lenkung und keinen Antrieb. Der Anhänger wird über

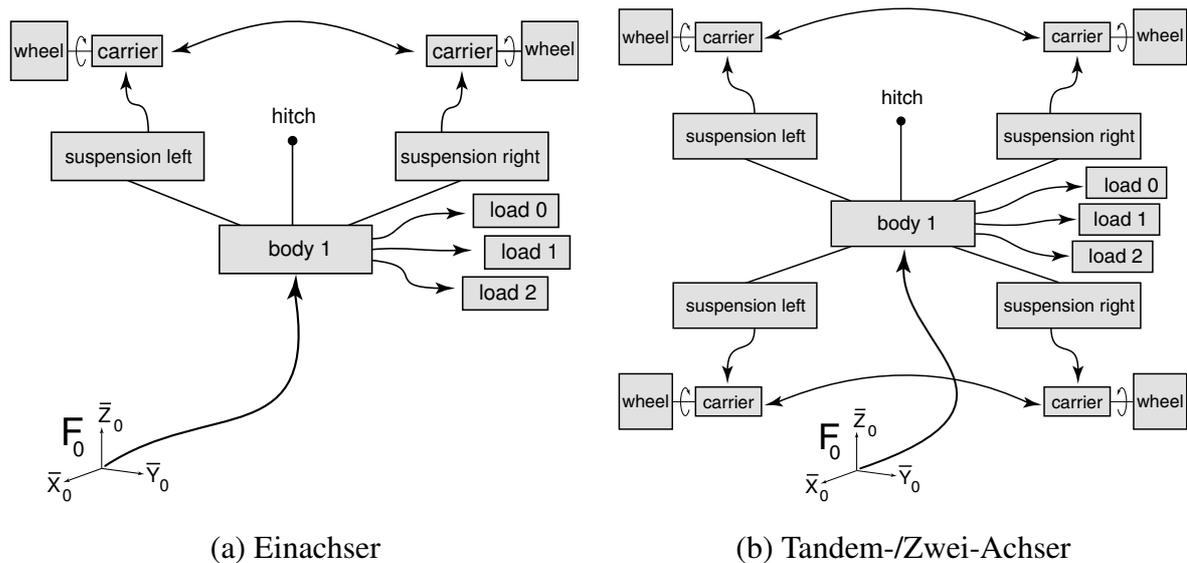


Abbildung 4.3.4: Fahrdynamikmodell “Trailer”

eine am Ende einer Deichsel angebrachte Anhängerkupplung an das Fahrzeug angehängt. Das Fahrwerk ist ausgeführt als Einzel-Achse oder als Tandem-Achse mit zwei einzelnen, nahe beieinander liegenden Achsen. Linke und rechte Radaufhängung können unabhängig voneinander oder miteinander gekoppelt sein. Eine Bremsanlage ist bei kleineren Pkw-Anhängern nicht immer vorhanden. Bei größeren bestimmt meist eine Auflaufbremse die Bremswirkung, die hierbei von der in Längsrichtung der Anhänger-Deichsel wirkenden Kraft abhängt. Aerodynamik und virtuelle Kraftwinder entsprechen denen des Pkw-Modells.

4.3.5 Fahrdynamikmodell eines Motorrads – “Motorcycle”

Die Topologie des Motorrad-Modells ist in Abbildung 4.3.5 skizziert. Beim Motorrad sind die zwei Spuren des Pkws auf eine Spur reduziert. Es besitzt nur eine Aufhängung pro Achse. Der Fahrer ist beim Motorrad im Verhältnis zum Fahrzeug schwer. Seine Positions- und Orientierungsänderung während der Fahrt, d.h. die Bewegung und Sitzhaltung, haben deutlichen Einfluss auf die Fahrdynamik des Gesamtsystems. Der Fahrer ist deshalb als beweglicher Körper bez. des Fahrzeugsystems \mathcal{F}_1 modelliert.

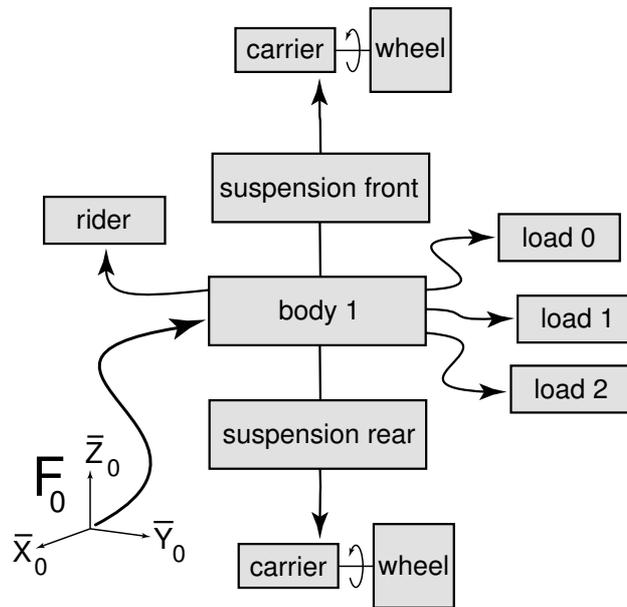


Abbildung 4.3.5: Fahrdynamikmodell "Motorcycle"

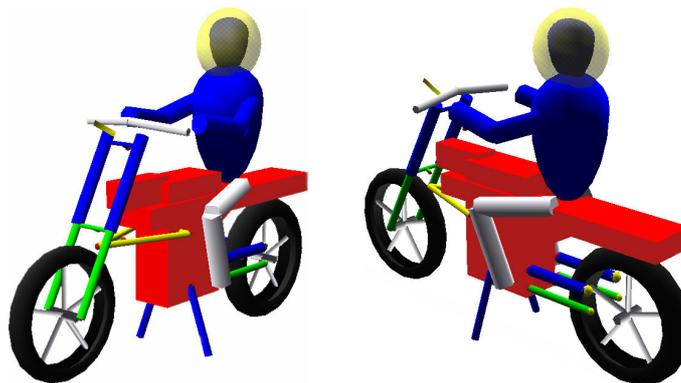


Abbildung 4.3.6: Motorrad-Modell

4.4 Koordinaten des Fahrzeugs

Ausgangspunkt der Modellierung dynamischer Systeme bilden die Differentialgleichungen (DGL)

$$\mathbf{A}(\mathbf{q}) \dot{\boldsymbol{\omega}} = \mathbf{b}(\mathbf{q}, \boldsymbol{\omega}) \quad (\text{dynamische DGL}) \quad (4.4.1)$$

$$\mathbf{H} \dot{\mathbf{q}} = \boldsymbol{\omega} \quad (\text{kinematische DGL}) \quad (4.4.2)$$

Hierbei ist $\mathbf{A}(\mathbf{q})$ die System- oder Trägheitsmatrix, $\mathbf{b}(\mathbf{q}, \boldsymbol{\omega})$ der Vektor der eingepprägten Kräfte und $\dot{\boldsymbol{\omega}}$ die Spaltenmatrix der ersten Ableitung der (Quasi)-Geschwindigkeiten $\boldsymbol{\omega}$. Bei der Wahl der Koordinaten und Quasigeschwindigkeiten wird ein besonderes Augenmerk darauf gerichtet, ein möglichst einfaches DGL-System zu erhalten:

- Das Gesamtsystem Fahrzeug wird in funktional getrennte, dynamisch möglichst unabhängige Subsysteme und Submodelle untergliedert. Die Systemmatrix $\mathbf{A}(\mathbf{q})$ soll im wesentlichen nur auf der Hauptdiagonalen und nur möglichst dünn besetzt werden.
- Die Trennung physikalisch unabhängiger Effekte wird durch die Wahl der Koordinaten und Quasigeschwindigkeiten erhalten (siehe Abschnitt 4.4.1).
- Die Koeffizienten der Systemmatrix \mathbf{A} sollen jeweils nur von wenigen Koordinaten q_i abhängig sein.

Die Auswahl geeigneter (Quasi)-Koordinaten und -Geschwindigkeiten ist ein entscheidender Beitrag zu einem effizient lösbar Differentialgleichungssystem.

Die für das Fahrzeugmodell benötigten Koordinaten lassen sich in drei Aspekte gliedern:

- externe Kinematik,
- interne Kinematik,
- Rotation der Räder.

Nachfolgend werden diese drei Bereiche nacheinander beleuchtet. Eine vollständige Liste der gewählten Koordinaten findet sich in Abschnitt 4.4.4.

4.4.1 Externaler Aspekt – ein kinematisch ungebundenes System

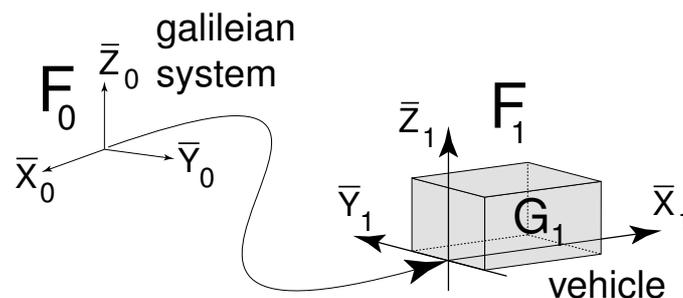


Abbildung 4.4.1: Kinematisch ungebundenes System \mathcal{F}_1

Die Interaktion der hier betrachteten Fahrzeuge und der Umwelt findet ausschließlich über die eingepprägten Kräfte der Reifen, die Aerodynamik etc. statt. Es handelt sich deshalb – wie bereits in Abschnitt 4.3.2 angedeutet – bei den Fahrzeugen um kinematisch ungebundene Systeme. Sie legen somit dem Modellierer keinerlei besonders ausgezeichnete Parameter zu ihrer Beschreibung nahe und geben ihm die Freiheit, ohne

Einschränkung genau jene Koordinaten und Geschwindigkeiten zu wählen, mit denen ihm die Modellbeschreibung besonders anschaulich oder effizient gelingt.

Das System Fahrzeug bewegt sich frei im Raum. Gerade bei aktuellen Fragestellungen an die Simulation aus dem Bereich Unfall- oder Überschlagerkennung (pre crash, roll over detection) existieren keine Lagen oder Orientierungen, die von vornherein ausgeschlossen werden können; die Parametrierung darf deshalb keine Singularitäten beinhalten.

Der kinematische Zustand eines Systems ist vollständig beschrieben durch seine Lage und Orientierung und den Vektor seiner absoluten Translations- und Rotationsgeschwindigkeiten bez. des Galileischen Systems \mathcal{F}_0 . Das System Fahrzeug wird beim hier vorgestellten Verfahren external wie folgt parametriert:

- Lage und Translationsgeschwindigkeit in kartesischen Koordinaten,
- Orientierung über eine Euler-Quaternion bez. \mathcal{F}_0 und
- Rotationsgeschwindigkeit über den Vektor der Quasigeschwindigkeiten $\omega_i, i = 4, 5, 6$ bez. \mathcal{F}_1 .

Die zur Beschreibung von Orientierungen weit verbreiteten Kardan-/Euler-Winkel sind nicht frei von Singularitäten. Es wird hier deshalb eine Euler-Quaternion verwendet.

Bisweilen erweist es sich als vorteilhaft, die Bewegungsgleichungen in einem nichtgalileischen Bezugssystem aufzustellen: Die Geschwindigkeiten werden – eher unüblich – im gegenüber \mathcal{F}_0 beschleunigten Fahrzeugsystem \mathcal{F}_1 formuliert. Durch diese Koordinatenwahl ergeben sich für die externe Kinetik einfache Gleichungen. Zusätzlich wird erreicht, dass die kinematischen Vorgänge im Inneren des Fahrzeugs unabhängig von den Koordinaten sind, die die absolute Lage und Orientierung des Fahrzeug-Systems \mathcal{F}_1 beschreiben.

Die externale Kinematik beschreibt mit Hilfe des fahrzeugaufbaufesten Koordinatensystems \mathcal{F}_1 die Bewegung des generalisierten Gesamtfahrzeug-Körpers \mathcal{G}_1 bez. des stets ruhenden Galileischen Systems \mathcal{F}_0 .

Die Quasigeschwindigkeiten ω_1, ω_2 und ω_3 beschreiben die Translationsgeschwindigkeit von O_1 relativ zu \mathcal{F}_0 . Sie sind im System \mathcal{F}_1 längs körperfester Achsen aufgespannt [Ham27][Lur68b]

$$\begin{bmatrix} \omega_1 = \bar{v}^0(O_1) \cdot \bar{X}_1 \\ \omega_2 = \bar{v}^0(O_1) \cdot \bar{Y}_1 \\ \omega_3 = \bar{v}^0(O_1) \cdot \bar{Z}_1 \end{bmatrix} \equiv [\bar{v}^0(O_1)]_1 \quad , \quad (4.4.3)$$

und nicht etwa im ruhenden System \mathcal{F}_0

$$[\dot{q}_1, \dot{q}_2, \dot{q}_3]_0 = [\bar{v}^0(O_1)]_0 \quad . \quad (4.4.4)$$

Dieser Koordinatenwahl ist die effiziente Struktur der Gleichungen zu verdanken, bei der die Systemmatrix frei von der absoluten Position und Orientierung des Bezugssystems \mathcal{F}_1 ist.

Der Übergang von den Quasigeschwindigkeiten $(\omega_1, \omega_2, \omega_3)$ auf generalisierte Geschwindigkeiten $(\dot{q}_1, \dot{q}_2, \dot{q}_3)$ im System \mathcal{F}_0 wird über eine Hamel-Transformation mit der Transformationsmatrix $\mathbf{T}_{1 \rightarrow 0}$ hergestellt [Lur68b].

$$[\bar{v}^0(O_1)]_0 = [\mathbf{T}_{1 \rightarrow 0}] [\bar{v}^0(O_1)]_1 \quad (4.4.5)$$

Die absolute Lage des Systempunktes O_1 liefert die Integration der Translationsgeschwindigkeit $[\bar{v}^0(O_1)]_0$ im System \mathcal{F}_0 .

Die Rotation des Fahrzeugs wird mit den Quasi-Geschwindigkeiten, mit ω_4, ω_5 und ω_6 um die körperfesten Basisvektoren von \mathcal{F}_1 beschrieben.

$$[\bar{\Omega}_1^0]_1 \equiv \begin{bmatrix} \omega_4 = \bar{\Omega}_1^0 \cdot \bar{X}_1 \\ \omega_5 = \bar{\Omega}_1^0 \cdot \bar{Y}_1 \\ \omega_6 = \bar{\Omega}_1^0 \cdot \bar{Z}_1 \end{bmatrix} \quad (4.4.6)$$

Die Orientierung des Fahrzeugs bez. \mathcal{F}_0 beschreibt das 4-Tupel der Euler-Quaternion $\bar{\eta}^0$ [Wit77] [Shu93].

$$\bar{\eta}^0 \equiv \begin{bmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \\ \eta_4 \end{bmatrix} \quad (4.4.7)$$

Berechnet wird η durch Integration des in das System \mathcal{F}_0 transformierten $\bar{\Omega}_1^0$ -Vektors .

Die Quasi-Geschwindigkeiten $\dot{\eta}$ werden durch Hamel-Transformation der Rotationsgeschwindigkeit $\bar{\Omega}_1^0$ mit einer von den generalisierten Koordinaten η berechnet (siehe Gleichung (4.2.50)):

$$\dot{\eta} = \frac{1}{2} \begin{bmatrix} \eta_4 & \eta_3 & -\eta_1 \\ -\eta_3 & \eta_4 & \eta_1 \\ \eta_2 & -\eta_1 & \eta_3 \\ -\eta_1 & -\eta_2 & \eta_3 \end{bmatrix} \bar{\Omega}_1^0 \quad (4.4.8)$$

Die Transformationsmatrix $\mathbf{T}_{1 \rightarrow 0}$ als Funktion der Quaternion η lautet

$$\mathbf{T}_{1 \rightarrow 0} = \begin{bmatrix} 2(\eta_4^2 + \eta_1^2) - 1 & 2(\eta_1\eta_2 + \eta_4\eta_3) & 2(\eta_1\eta_3 - \eta_4\eta_2) \\ 2(\eta_1\eta_2 - \eta_4\eta_3) & 2(\eta_4^2 + \eta_2^2) - 1 & 2(\eta_2\eta_3 + \eta_4\eta_1) \\ 2(\eta_1\eta_3 + \eta_4\eta_2) & 2(\eta_2\eta_3 - \eta_4\eta_1) & 2(\eta_4^2 + \eta_3^2) - 1 \end{bmatrix} \quad (4.4.9)$$

4.4.2 Internaler Aspekt – Modellierung der Radaufhängungen

Mit dem Begriff “internal” werden die von der externalen, globalen Bewegung des Fahrzeugs als Ganzes separierten Bewegungen materieller Systeme \mathcal{B}_2 “im Inneren des

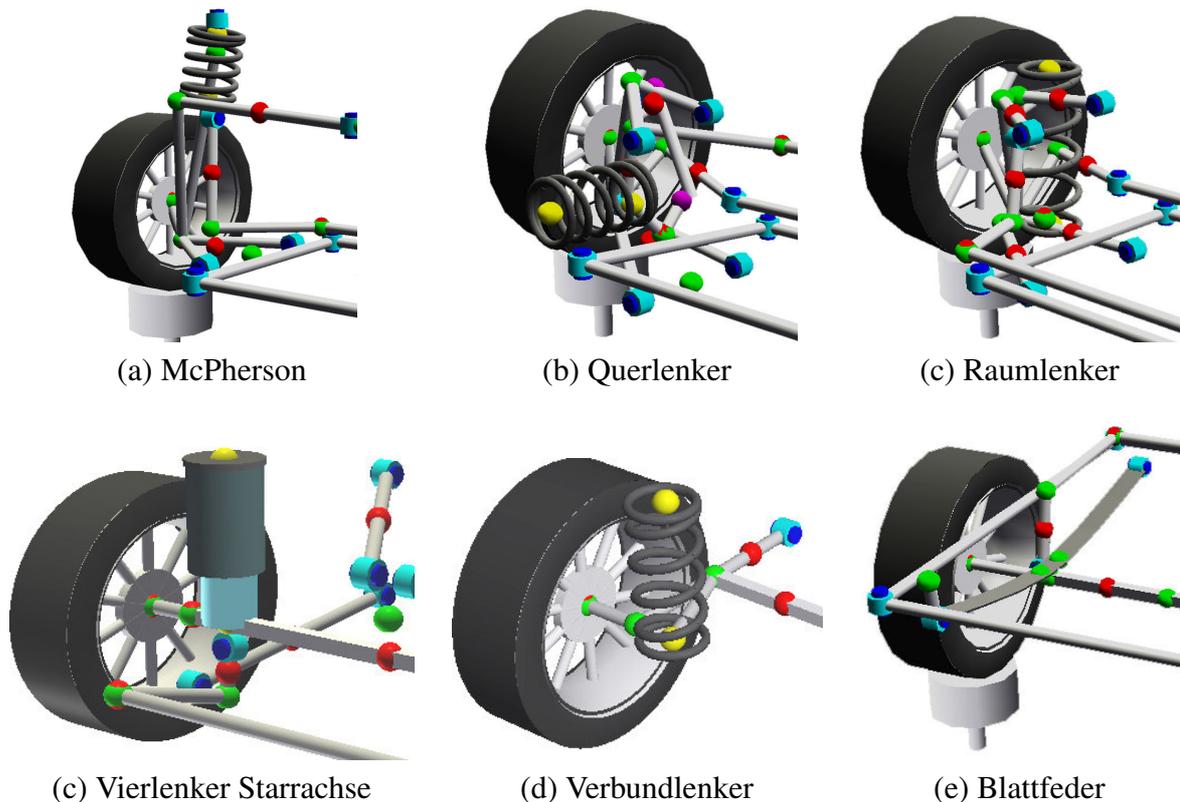


Abbildung 4.4.2: Pkw-Radaufhängungskonstruktionen (Quelle: IPG Automotive GmbH, IPG-KINEMATICS)

Fahrzeugs” bezeichnet – die Bewegungen der Einzelkörper der Radaufhängungen oder der beweglichen Beladungen.

Eine Radaufhängung besitzt “eine im wesentlichen vertikal ausgerichtete Beweglichkeit, um Fahrbahnunebenheiten auszuweichen” und es “werden bei schnellen Strassenfahrzeugen die Vorderräder gelenkt” [Mat98]. Die Aufgabe jeder Radaufhängung ist es, Radlast, Antriebs-, Brems- und Seitenkräfte des Reifens im Fahrzeug abzustützen, indem dem Radträger eine bestimmte Kinematik aufgezwungen wird. Design-Kriterien für Konstruktion und Abstimmung sind hierbei Fahrsicherheit, Fahrverhalten und Fahrkomfort.

Radaufhängungen beim Automobil und Motorrad bestehen aus einer Vielzahl von Körpern, die meist als starre Körper angesehen werden können. Abbildung 4.4.3 zeigt eine Doppel-Querlenker-Achse eines Pkws. Am Radträger (carrier) ist das Rad (wheel) montiert. Die Radbremse stützt sich (typischerweise) zwischen Rad und Radträger ab. Das Antriebsmoment wirkt im wesentlichen zwischen Fahrzeugaufbau und Rad. Der Radträger bildet zusammen mit den Lenkern (control arm), Feder (spring), Dämpfer (damper) und dem Aufbau (vehicle body) einen räumlichen Mechanismus.

Für die Modellierung wird folgender, allgemeiner, einheitlicher, von der Konstruktion der Aufhängung unabhängiger Ansatz gewählt: Fahrzeugaufbaufeste Körper heißen \mathcal{B}_1 ; die

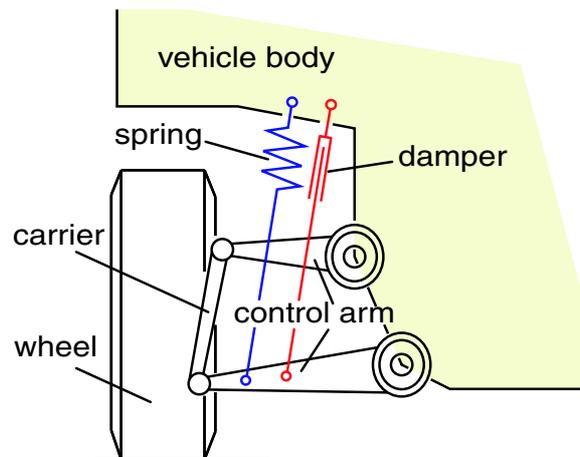


Abbildung 4.4.3: Prinzipskizze einer Doppel-Querlenker Einzelradaufhängung: Rad, Radträger, Lenker, Feder, Dämpfer, Aufbau

relativ zu \mathcal{B}_1 beweglichen Körper heißen \mathcal{B}_2 . Die Bewegung des räumlichen “Kontinuums Radaufhängung” wird durch die Bewegung $N \geq 1$ einzelner (starrer) \mathcal{B}_2 modelliert. Ihre Bewegung wird jeweils einzeln, relativ zu \mathcal{F}_1 beschrieben. Geschlossene kinematische Schleifen sind nicht mehr vorhanden. Die Bewegung eines \mathcal{B}_2 wird eindeutig durch die Position eines auf ihm festen Punktes $\overline{O_1 M}$ und die Orientierung ϕ bez. des Fahrzeugaufbaus \mathcal{F}_1 festgelegt. Sie wird auf eine von zwei Freiheitsgraden abhängige Führungsbewegung reduziert (siehe Abschnitt 4.4.2). Parametriert wird die Bewegung über Kennfelder (siehe Abschnitt 4.4.2.1). Die Freiheitsgrade der \mathcal{B}_2 -Bewegung sind *Radhub links* $q_1 \equiv q_{\text{whl travel}}$ und *Radhub rechts* $q_2 \equiv q_{\text{whl travel}}$ oder – bei gelenkten Achsen – *Lenken* $q_2 \equiv q_{\text{steer}}$. Die nachfolgenden Erläuterungen nehmen Bezug auf eine gelenkte Vorderradaufhängung.

$$\begin{aligned} \text{Position:} \quad \overline{O_1 M} &\equiv \overline{O_1 M}(q_{\text{whl travel}}, q_{\text{steer}}) \quad , \\ \text{Orientierung:} \quad \phi &\equiv \phi(q_{\text{whl travel}}, q_{\text{steer}}) \quad . \end{aligned}$$

Mit der Kondensation der räumlichen Bewegung von 6 Koordinaten auf 2 generalisierte Koordinaten (Maggi-Transformation, Gleichung (4.2.52)) werden die Bewegungen der einzelnen Bauteile der Radaufhängung miteinander gekoppelt. Voneinander unabhängige, höherfrequente ($f > 30\text{Hz}$) Schwingungen vor allem kleiner und leichter Körper in den Gummilagern realer Radaufhängungen haben hauptsächlich Einfluß auf den Fahrkomfort. Die Fahrdynamik wird von ihnen nur nachrangig beeinflusst. Durch den Ansatz, diese Einzelkörper-Bewegungen von nur zwei gemeinsamen Freiheitsgraden abhängig zu machen, sind diese Schwingungen im Modell eliminiert. Das für die Fahrdynamik relevante elastokinematische Eigenlenkverhalten einer Radaufhängung, bei dem die Radstellung vor allem durch die wirkenden Seiten- und Umfangskräfte verändert wird, wird der kinematischen Führungsbewegung überlagert. Die Parametrierung erfolgt über Steifigkeitsbeiwerte und -kennlinien. Die Bewegung der Körper der Radaufhängung und ihre Massen- und Trägheitsverteilung ist hiermit bekannt.

Die Kräfte der Bauteile Achsfeder, Achsdämpfer und Wankstabilisator werden über die Bauteillänge bzw. beim Stabilisator über Auslenkung oder Verdrehung parametrisiert

$$l_{\text{spring}} \equiv l_{\text{spring}}(q_{\text{whl travel}}, q_{\text{steer}}) \quad (4.4.10)$$

$$l_{\text{damp}} \equiv l_{\text{damp}}(q_{\text{whl travel}}, q_{\text{steer}}) \quad (4.4.11)$$

$$l_{\text{stabi}} \equiv l_{\text{stabi}}(q_{\text{whl travel}}, q_{\text{steer}}) \quad (4.4.12)$$

Die Bauteilkräfte werden als generalisierte Kräfte berücksichtigt. Sie treten als in Richtung der freien Bewegung der Freiheitsgrade $q_{\text{whl travel}}$ und q_{steer} projizierte Kräfte auf.

Die internale Kinematik wird achsweise modelliert, beim Pkw als Vorder- und als Hinterachse. Jede Achse gliedert sich in eine linke und eine rechte Radaufhängung mit jeweils zwei generalisierten Koordinaten. An der Vorderachse ist üblicherweise die Lenk-Koordinate links und rechts kinematisch gekoppelt und auf beiden Seiten identisch. Insgesamt besitzt die Vorderachse somit drei unabhängige Koordinaten. An der Hinterachse sind zwei Koordinaten für die gesamte Achse ausreichend, da die Koordinate "Einfederung" der einen Radaufhängung der zweiten Koordinate der gegenüberliegenden, gekoppelten Aufhängung entspricht.

$$\begin{bmatrix} \dot{q}_7 = \dot{q}_{\text{whl travel, fl}} & q_7 = q_{\text{whl travel, fl}} \\ \dot{q}_8 = \dot{q}_{\text{steer}} & q_8 = q_{\text{steer}} \end{bmatrix} \begin{bmatrix} \dot{q}_9 = \dot{q}_{\text{whl travel, fr}} & q_9 = q_{\text{whl travel, fr}} \\ \dot{q}_{10} = \dot{q}_{\text{steer}} & q_{10} = q_{\text{steer}} \end{bmatrix} \quad (4.4.13)$$

$$\begin{bmatrix} \dot{q}_{11} = \dot{q}_{\text{whl travel, rl}} & q_{11} = q_{\text{whl travel, rl}} \\ \dot{q}_{12} = \dot{q}_{\text{whl travel, rr}} & q_{12} = q_{\text{whl travel, rr}} \end{bmatrix} \begin{bmatrix} \dot{q}_{13} = \dot{q}_{\text{whl travel, rr}} & q_{13} = q_{\text{whl travel, rr}} \\ \dot{q}_{14} = \dot{q}_{\text{whl travel, rl}} & q_{14} = q_{\text{whl travel, rl}} \end{bmatrix} \quad (4.4.14)$$

Der Körper Radträger ist stets modelliert. Er ist immer "der erste" Körper vom Typ \mathcal{B}_2 . Wird er über den Punkt auf der Raddrehachse, der mit dem Radmittelpunkt zusammenfällt, parametrisiert, wird die Berechnung besonders effizient: Die Schnittstelle zwischen Radaufhängung und Rad/Reifen liegt im Radmittelpunkt auf der Raddrehachse, einem auf dem Radträger festen Punkt. Die Kinematik des Radmittelpunktes wird benötigt. Steht sie direkt aus der Parametrierung des Körpers Radträger zur Verfügung, entfallen zusätzliche Transformationen in den und vom Radmittelpunkt.

Dem Umstand, dass die Bewegungen der Einzelkörper \mathcal{B}_2 beim Einfedern und Lenken voneinander verschieden sind, wird bei der Modellierung durch eine variable Anzahl an \mathcal{B}_2 mit jeweils eigener Bewegung, Masse und Trägheit Rechnung getragen. Besonders anschaulich ist die unterschiedliche Bewegung am Motorrad bei der "Telelever"-Vorderradaufhängung oder der "Paralever"-Hinterradaufhängung (siehe Abbildung 4.4.4). Bei der Telelever-Aufhängung bewegt sich die (leichte) Gabel und der Radträger translatorisch, während der (schwere) A-Lenker um seine quer zur Fahrtrichtung angeordneten Lagerpunkte am Rahmen rotiert. Bei der Paralever-Hinterradschwinge ist es offensichtlich, dass für exakte Modellierung der schweren (Gewichtskraft) und der trägen (Trägheitstensor) Massen kein reduziertes Ein-Körper-Modell existiert.

Im übergeordneten System *Fahrwerks-Achse* treten alle Einzelkörper einer Radaufhängung zusammengefasst als ein generalisierter Körper in Erscheinung und werden dort als solcher berücksichtigt. Gegenüber dem *Gesamt-Fahrzeug* sind die einzelnen Achsen selbst

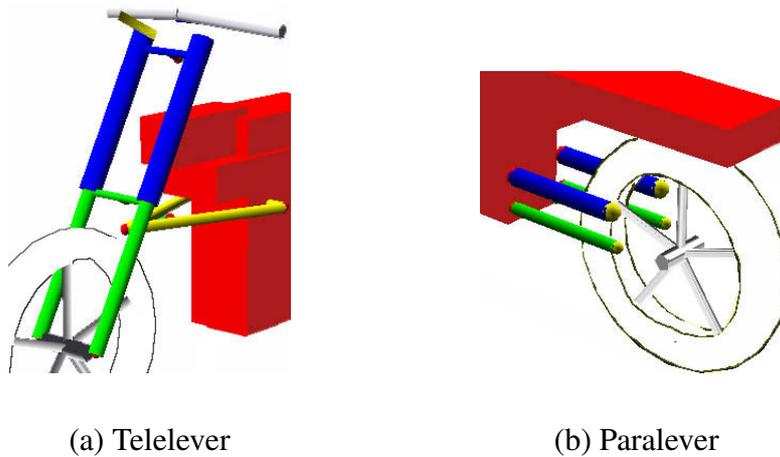


Abbildung 4.4.4: Motorrad Radaufhängungen (Quelle: IPG Automotive GmbH [Rei04b])

wiederum generalisierte Körper. Der Formalismus fordert keine speziellen Eigenschaften der materiellen Subsysteme \mathcal{B}_2 . Es kann sich bei ihnen sowohl um starre als auch um flexible Körper handeln.

In der frühen Entwicklungsphase eines Fahrzeugs ist häufig nur die angestrebte Raderhebungskurve (Trajektorie der Radbewegung) bekannt. Sie entstammt im CAD-System entwickelten Fahrzeugkonzepten oder Prüfstandsmessungen von anderen Fahrzeugen aus der gleichen oder höheren Fahrzeug-Klasse mit sehr guten Fahreigenschaften. Weitere Details werden während der Konstruktionsphase nach und nach ergänzt. Für die Modellierung bedeutet dies, dass die Radaufhängung zuerst nur als einzelner Radführungs-Körper “Radträger” beschrieben wird. Ist die Aufhängung fertig konzeptioniert, konstruiert und ausgelegt, stehen aus der Konstruktionsabteilung umfangreiche Daten aus den FEM- und MKS-Modellen für Lenker, Federn- und Dämpfer bereit, die für die Bauteilparametrierung der Fahrdynamikmodelle direkt übernommen werden können. Zu den “ \mathcal{B}_2 ” einer Radaufhängung gehört auch der Teil des Körpers Rad, der sich “beim Einfedern und Lenken” bewegt. Nicht dazu gehört seine Rotor-Trägheit um die Raddrehachse (siehe Abschnitt 4.4.3), die für die Raddrehung wesentlich ist.

Eine gelenkte Vorderachse kann zusätzlich durch ein (dynamisches) Lenksystem ergänzt werden. “Dynamisch” meint in diesem Zusammenhang die Kopplung Radaufhängung--Lenksystem auf Beschleunigungsebene im Gegensatz zu einer “kinematischen” Kopplung, bei der die Lenkbewegung rein kinematisch ohne Betätigungskräfte erfolgt. Die dynamische Kopplung Radaufhängung/Lenksystem wurde in Abschnitt 3.5.3 besprochen.

4.4.2.1 Kinematik Handling

Während der Simulation wird für jeden Zeitschritt auf die Parametrierung aller Achsen, Radaufhängungen und deren Bauteile zugegriffen. In Abhängigkeit der generalisierten Koordinaten q_1 und q_2 (siehe Abschnitt 4.4.2) werden Position, Orientierung, Massen- und Trägheitsverteilung aller Einzelkörper \mathcal{B}_2 und die aktuellen Längen und Längenänderung-

en der Achsfedern, Achsdämpfer und Wankstabilisatoren ermittelt. Es ist deshalb wichtig, die Modellparameter effizient zu verwalten und während der Simulation rechenzeit-optimiert bereitzustellen.

Die Parametrierung der Kinematik erfolgt durch die Vorgabe von Stützstellen. Der minimale Kinematik-Parametersatz einer Radaufhängung besteht aus folgenden, von den generalisierten Koordinaten q_1 und q_2 abhängigen Größen (2-dimensionale Kennfelder):

- Position des Rades über den 3×1 Translationsvektor des auf der Raddrehachse im Radmittelpunkt liegenden Punktes C .
- Orientierung des Radträgers und der Raddrehachse (3×3 Orientierungs- oder Transformationsmatrix) über 3 Winkel eines Drehgelenkes mit der Drehreihenfolge $\bar{Z} - \bar{X} - \bar{Y}$.
- 3 Abstände der Anlenkpunkte Achsfeder, Achsdämpfer und Stabilisator.

Bei dieser bauteilorientierten Parametrierung der Radaufhängung sind somit mindestens $3(\text{trans}) + 3(\text{rot}) + 3(\text{length}) = 9$ und für jeden weiteren Körper zusätzlich $3(\text{trans}) + 3(\text{rot}) = 6$ Größen zu verwalten. Jeder Körper wird zusätzlich durch konstante Parameter wie Masse, Schwerpunkt und Trägheit charakterisiert. Die Kraffelemente besitzen ebenfalls weitere Parameter.

Die generalisierten Koordinaten q_1 und q_2 können für die Simulation beschränkt werden, sofern dies nicht bereits durch die Aufhängungskonstruktion erfolgt ist. Über diesem "Grundgitter" werden die Kennfeld-Größen abgetragen. Der Zugriff auf ein Kennfeld oder eine Kennlinie unterteilt sich in zwei Abschnitte:

1. Intervallermittlung: Ermittlung der zur Abfragestelle nächstliegenden Stützstellen
2. Berechnung: Interpolation des Kennfelds zwischen den vorgegebenen Stützstellen

Im Hinblick auf geringen Rechenzeit- und Ressourcen-Bedarf während der Simulation stehen folgende Maßnahmen zur Verfügung:

Bei beliebigen, inäquidistanten Kennfeldern erfolgt die Intervallermittlung effizient über binäre Suche. Optimieren lässt sich der Zugriff dadurch, dass über einem äquidistanten Grundgitter aufgetragene Kennfelder verwendet werden: Das passende Intervall ist dann nicht mehr zu suchen, es kann direkt über die – konstante – Intervallbreite bestimmt werden.

Die Kennlinien und Kennfelder der einzelnen Größen werden zu Vektor-Kennlinien und Vektor-Kennfeldern zusammengefasst. D.h. über einem Grundgitter wird nicht nur eine Größe abgelegt. Es werden vielmehr über einem gemeinsamen Grundgitter alle M Größen angeordnet.

Hierdurch muss die Intervallermittlung nur ein einziges Mal pro Rechenschritt durchgeführt werden – und nicht für jede Größe extra. Der Speicherbedarf für das Grundgitter wird ebenfalls reduziert.

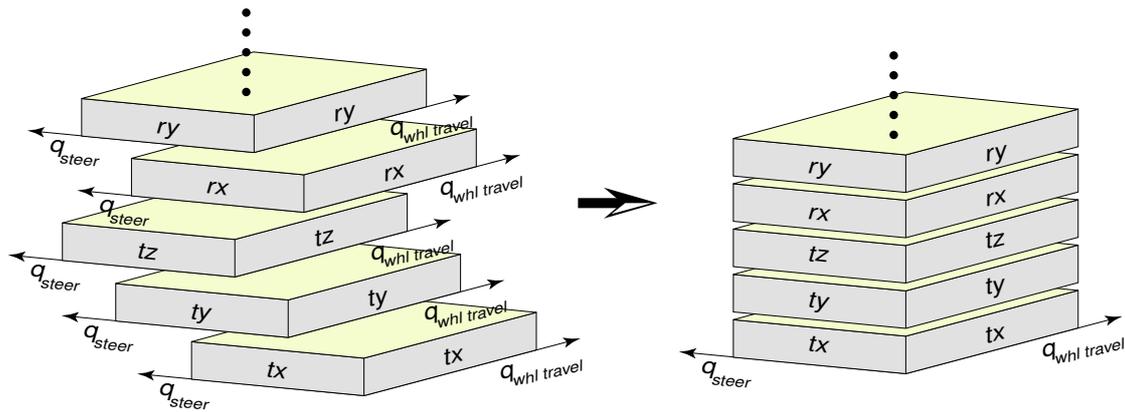


Abbildung 4.4.5: Übergang von Einzel- auf Vektor-Kennfelder

Bei rechts-links-gekoppelten Achskonstruktionen (z.B. Verbundlenker-Hinterachse) werden beide Radaufhängungen über einem gemeinsamen Grundgitter parametrisiert. Hierdurch reduziert sich die Anzahl der Intervallermittlung um die Hälfte, von je einmal pro Radaufhängung auf insgesamt einmal pro Achse. Der Speicherbedarf für das Grundgitter wird ebenfalls reduziert.

Bei den in dieser Arbeit eingesetzten Modellen kommen optimierte kubische Polynomfunktionen zum Einsatz. Die für den MKS-Formalismus ebenfalls benötigten Ableitungen an das Kennfeld werden über symbolische Differentiation ermittelt.

Der numerische Aufwand lässt sich noch weiter reduzieren, indem die Modellierungsfunktionen an den abzubildenden Parameter adaptiert werden. Vielfach ist ein Parameter schon ausreichend genau durch ein Polynom 1. oder 2. Grades darstellbar. Die zu verwendenden Berechnungsfunktionen werden während der Initialisierungsphase, zu Beginn eines Simulationslaufs, festgelegt und zur Laufzeit direkt verwendet.

Weiteres Augenmerk wird auf die Strukturierung der Parameter gerichtet: Daten, die zur Berechnung der Kennfelder zum gleichen Zeitpunkt benötigt werden, sind in den Datenstrukturen im Arbeitsspeicher des Computers kompakt und nahe beieinander liegend anzuordnen. Der schnelle Cache-Speicher des Prozessors wird hierdurch effizient eingesetzt und verbessert vor allem bei großen Kennfeldern das Laufzeitverhalten des Simulationsprogramms.

4.4.3 Drehbewegung der Räder – Sonderrolle des drehenden Rades

Das Rad bildet topologisch den äußersten Körper einer Gelenkkette. Hierdurch lässt sich seine Rotation um die Raddrehachse von der restlichen Bewegung des Radträgers trennen. Die Rotation des Rades j , $j = 1, 2, 3, 4$, wird bez. des Eigensystems \mathcal{F}_{E_j} um die \bar{Y}_{E_j} -Achse parametrisiert. Der Drehwinkel wird durch Integration der Raddrehgeschwindigkeit um \bar{Y}_{E_j} berechnet.

Hinweis: Das Eigensystem des Rades ist das System, dessen Y -Achse stets mit der Raddrehachse – typischerweise auch der Y -Achse des Radträgers – zusammenfällt, das

jedoch niemals eine Drehung um diese Achse ausführt. Siehe hierzu auch den Abschnitt 5.4.2.

Unter dem Begriff “Rad” werden die “sich drehenden” Teile des Fahrwerks verstanden: die Felge, die rotierenden Teile der Radbremse und der Reifen. Das Rad ist am Radträger über ein Drehgelenk befestigt. Die Radaufhängung verbindet die Lagerstelle des Rades, den Radträger, mit dem Fahrzeugaufbau. *Anmerkung:* An angetriebenen Rädern führt zusätzlich die Antriebswelle zum Rad.

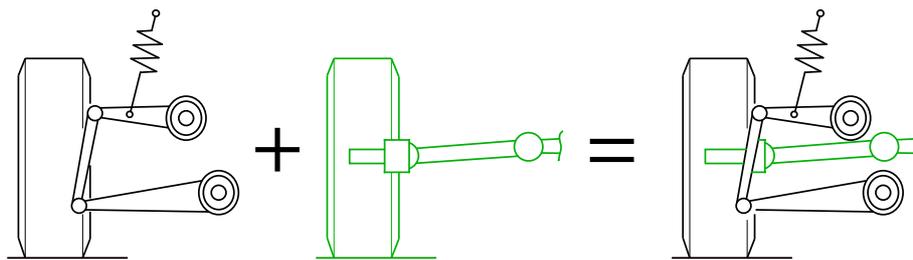


Abbildung 4.4.6: Das Rad als Bestandteil von Antriebsstrang und Radaufhängung

Das Rad ist Bestandteil zweier funktional getrennter Teilsysteme und nimmt insofern eine Sonderrolle ein:

- Das Rad ist ein Bauteil der Radaufhängung: Beim Einfedern und Lenken ändert das Rad seine Position und Orientierung relativ zum Fahrzeugaufbau. Konstruktives Ziel der Einfederungs- und Lenk-Kinematik ist die optimale Abstützung der eingepprägten Kräfte und Momente zwischen Fahrbahnoberfläche und Gummistollen des Reifens im Reifen-Fahrbahn-Kontaktpunkt.
- Das Rad ist ein Rotor des Antriebsstranges: Es ist das letzte Glied einer Kette von Rotoren: Beginnend mit dem Motor, über Kupplung, Getriebe und Triebstrang mit Wellen, Differentialen und den Abtriebswellen zu den Rädern. Die Momentenbilanz aus Motormoment, Bremsmoment zwischen Rad und Radaufhängung, Reifenmoment und Radträgheit führen zu einer Beschleunigung des Rades.

Bei der Modellierung wird diese Trennung in die funktionalen Einheiten Fahrwerk und Antriebsstrang üblicherweise beibehalten: Das Fahrwerk beschreibt die Bewegung aller Aufhängungskomponenten relativ zum sich im Raum frei bewegenden Fahrzeug. Der Antriebsstrang beschreibt die Übertragung des Antriebsmomentes vom Motor auf das Rad. Der Reifen setzt die Raddrehung in Kräfte im Reifen-Fahrbahn-Kontaktpunkt zwischen Fahrzeug und Umwelt um und generiert hierdurch die Bewegung des Fahrzeugs. Aus kinematischer Sicht stellt die Kopplung des Antriebsstranges und des Fahrzeugs und dessen Radaufhängung eine kinematische Einkopplung einer Rotorkette in ein sich dreidimensional bewegendes System dar. Die Orientierungsänderung des rotierenden Rades beim Einfedern und Lenken bewirkt Coriolis-Effekte. Die dreidimensionale Bewegung

der Raddrehachse kann Verdrehungen der Antriebswelle und Verspannungen innerhalb der Radaufhängung verursachen.

Die Größe ‘‘Raddrehung’’ will korrekt und prazise definiert sein. Man bedenke die Situation beim Durchfahren eines Loopings, bei der das Fahrzeug einen Nickwinkel (Drehung um die Querachse, pitch) von 2π iberstreicht, und den Einfluss auf den Drehwinkel der Antriebswellen, die meist als eine ‘‘im Labor feststehende’’ Rotorkette modelliert sind.

4.4.4 Koordinaten des Fahrzeugmodells – bersicht

Geschwindigkeit	Lage	Erlauerung
$\omega_1 = \bar{v}^0(O_1) \cdot \bar{X}_1$	$p_x(O_1)$	Translationsgeschwindigkeit $[\bar{v}^0(O_1)]_1$ von O_1 bez. \mathcal{F}_0 ; Translation von O_1 bez. \mathcal{F}_0
$\omega_2 = \bar{v}^0(O_1) \cdot \bar{Y}_1$	$p_y(O_1)$	
$\omega_3 = \bar{v}^0(O_1) \cdot \bar{Z}_1$	$p_z(O_1)$	
$\omega_4 = \bar{\Omega}_1^0 \cdot \bar{X}_1$	η_1	Rotationsgeschwindigkeit $[\bar{\Omega}_1^0]_1$ von \mathcal{F}_1
$\omega_5 = \bar{\Omega}_1^0 \cdot \bar{Y}_1$	η_2	
$\omega_6 = \bar{\Omega}_1^0 \cdot \bar{Z}_1$	η_3	Orientierung $\bar{\eta}^0$ (Euler-Quaternion) von \mathcal{F}_1 bez. \mathcal{F}_0
	η_4	
\dot{q}_7	$q_7 = q_{0fl}$	‘‘Einfedern’’ $q_{\text{whl travel}}$ vorne links
\dot{q}_8	$q_8 = q_{1fl}$	‘‘Lenken’’ q_{steer} vorne links
\dot{q}_9	$q_9 = q_{0fr}$	‘‘Einfedern’’ $q_{\text{whl travel}}$ vorne rechts
\dot{q}_{10}	$q_{10} = q_{1fr}$	‘‘Lenken’’ q_{steer} vorne rechts
\dot{q}_{11}	$q_{11} = q_{0rl}$	‘‘Einfedern’’ $q_{\text{whl travel}}$ hinten links
\dot{q}_{12}	$q_{12} = q_{1rl}$	‘‘Einfedern’’ $q_{\text{whl travel}}$ hinten links
\dot{q}_{13}	$q_{13} = q_{0rr}$	‘‘Einfedern’’ $q_{\text{whl travel}}$ hinten rechts
\dot{q}_{14}	$q_{14} = q_{1rr}$	‘‘Einfedern’’ $q_{\text{whl travel}}$ hinten rechts
$\omega_{15} = \bar{\Omega}_{E_{fl}}^0 \cdot \bar{Y}_{E_{fl}}$	$q_{15} = q_{fl}$	vorne links Raddrehung bez. des Eigensystems $\mathcal{F}_{E_{fl}}$ Drehwinkel von $\mathcal{F}_{W_{fl}}$ bez. \mathcal{F}_{fl} (wheel front left)
$\omega_{16} = \bar{\Omega}_{E_{fr}}^0 \cdot \bar{Y}_{E_{fr}}$	$q_{16} = q_{fr}$	vorne rechts Raddrehung bez. des Eigensystems $\mathcal{F}_{E_{fr}}$ Drehwinkel von $\mathcal{F}_{W_{fr}}$ bez. \mathcal{F}_{fr} (wheel front right)
$\omega_{17} = \bar{\Omega}_{E_{rl}}^0 \cdot \bar{Y}_{E_{rl}}$	$q_{17} = q_{rl}$	hinten links Raddrehung bez. des Eigensystems $\mathcal{F}_{E_{rl}}$ Drehwinkel von $\mathcal{F}_{W_{rl}}$ bez. \mathcal{F}_{rl} (wheel rear left)
$\omega_{18} = \bar{\Omega}_{E_{rr}}^0 \cdot \bar{Y}_{E_{rr}}$	$q_{18} = q_{rr}$	hinten rechts Raddrehung bez. des Eigensystems $\mathcal{F}_{E_{rr}}$ Drehwinkel von $\mathcal{F}_{W_{rr}}$ bez. \mathcal{F}_{rr} (wheel rear right)

4.5 Massenmatrix des Fahrzeugs

4.5.1 Bemerkungen zur direkten Auswertung des Prinzips von D'Alembert

Das D'Alembert'sche Prinzip (siehe Abschnitt 4.2.7) ist auf beliebige, materielle Systeme anwendbar. Der einzelne starre Körper ist keine Voraussetzung.

An dieser Stelle wird zum ersten Mal auf das Objekt "starrer Körper" zurückgegriffen. Das verwendete Verfahren setzt dies nicht voraus. Das Aufstellen und Berechnen der Trägheiten wird damit jedoch deutlich erleichtert.

Bei den Elementen der Trägheitsmatrix A

$$a_{ij} = \int_{\Sigma} \frac{{}^0\partial\overline{OM}}{\partial\pi_i} \cdot \frac{{}^0\partial\overline{OM}}{\partial\pi_j} dm \quad (4.5.1)$$

handelt es sich um Terme, die ausschließlich in Kombination mit zweiten Ableitungen von Lagen bzw. ersten Ableitungen von Geschwindigkeiten auftreten. Sie werden deshalb auch als "reine" Trägheiten oder "reine" Beschleunigungen bezeichnet.

Mit *Restbeschleunigungen* werden Trägheitswirkungen aufgrund von Geschwindigkeitsprodukten oder Produkten erster Ableitungen bezeichnet. Sie werden zu den eingepprägten Kräften auf die rechte Seite des Gleichungssystems geschrieben. Ermittelt werden sie basierend auf einem Verfahren von Fayet [Fay90], das an die Berechnung der Restbeschleunigungen und an die hier vorliegende Systemtopologie adaptiert wurde.

4.5.2 Struktur der Massenmatrix, Übersicht

Zur Beschreibung der Dynamik des Fahrzeugs wurden 18 Koordinaten in Abschnitt 4.4 eingeführt. Sie führen auf eine klar strukturierte, zur Hauptdiagonalen symmetrischen Systemmatrix A . In Abbildung 4.5.1 ist die Systemmatrix A skizziert. Die Quadrate stehen jeweils für ein einzelnes Matrix-Element. Die farblich hinterlegten Quadrate sind i.Allg. von Null verschieden. Bei den hell hinterlegten Elementen handelt es sich um Koppelemente zwischen zwei Koordinaten. Die dunkler hinterlegten Elemente sind die "Eigentragheiten" der betreffenden Koordinaten (Diagonalterme). Die weißen Quadrate mit schwarzer Umrandung sind stets Null.

Im Folgenden wird die Bedeutung der verschiedenen Bereiche diskutiert. Eine detaillierte, formelmäßige Herleitung der einzelnen Matrixelemente findet sich in Abschnitt 4.6.

Zur Bezeichnung einer $n \times m$ -Submatrix von A mit den Elementen $a_{i,j}$, $i = k + n - 1$, $j = l + m - 1$ wird im folgenden die abkürzende Schreibweise $A_{[i,j/k,l]}$ verwendet.

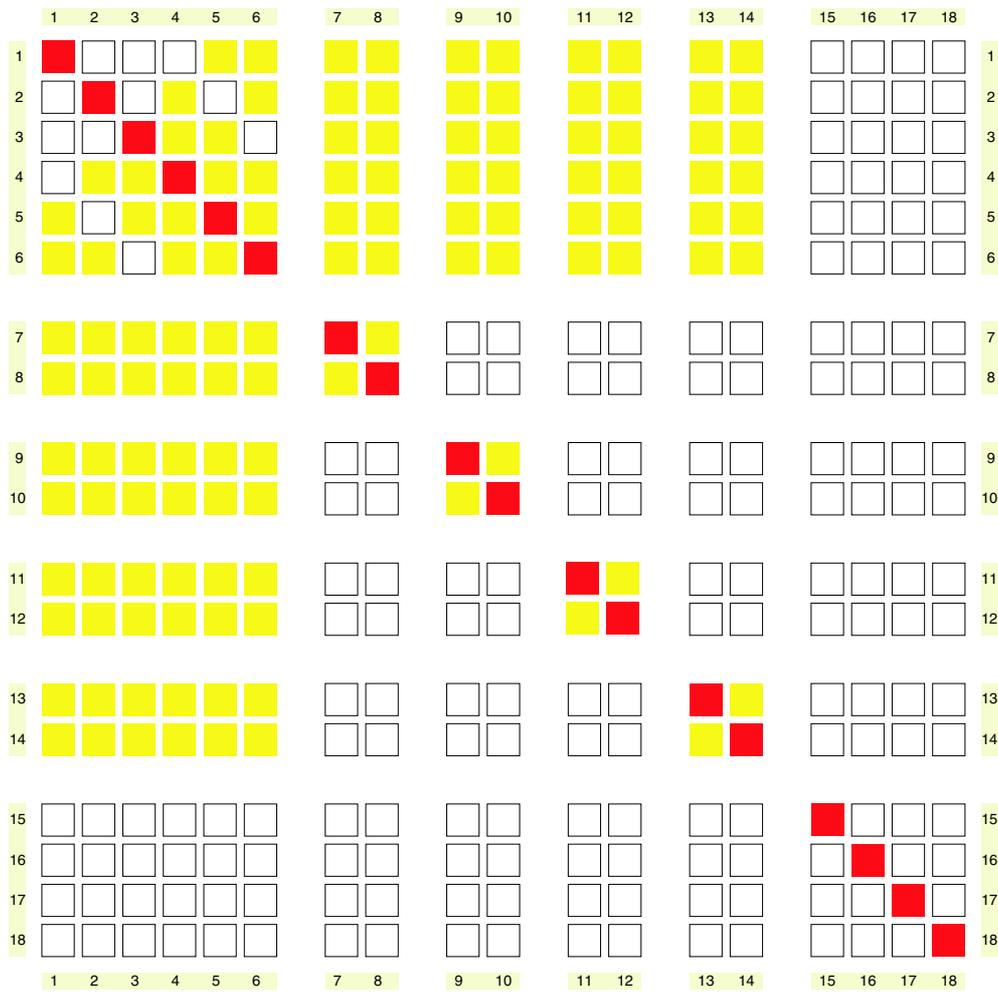
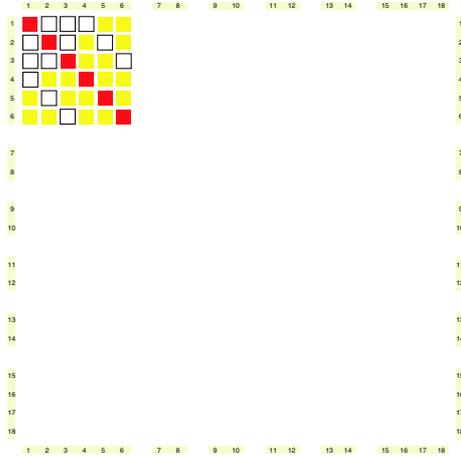


Abbildung 4.5.1: Systemmatrix A. Dunkel, rot sind Diagonalelemente ungleich Null; hell, gelb sind Koppелеlemente ungleich Null; schwarz umrandete Elemente sind stets Null. Detailliert vorgestellt werden die einzelnen Elemente in Abschnitt 4.5.6.

4.5.3 Generalisiertes Fahrzeug-System \mathcal{F}_1 : Externale Bewegung



Die 6×6 -Submatrix $\mathbf{A}_{[1,1/6,6]}$ repräsentiert die Trägheiten des generalisierten Körpers für die externe Bewegung von \mathcal{F}_1 bez. \mathcal{F}_0 .

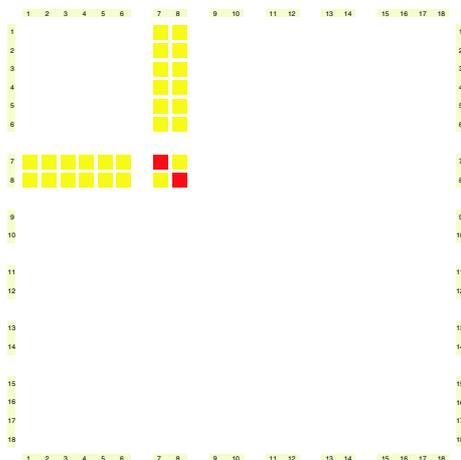
Die 3×3 -Submatrix $\mathbf{A}_{[1,1/3,3]}$ ist die Systemmatrix der Translation des Gesamtsystems. Die Diagonalelemente sind gleich der generalisierten Systemmasse. Alle anderen Elemente sind gleich Null.

Die 3×3 -Submatrix $\mathbf{A}_{[4,4/6,6]}$ beschreibt die Rotation von \mathcal{F}_1 . Die Kopplung von Translation und Rotation ist über die zwei 3×3 -Submatrizen realisiert:

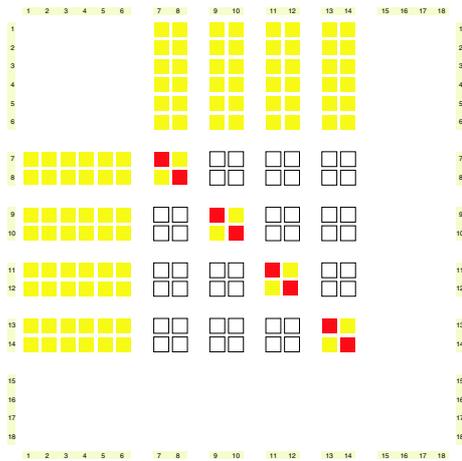
$\mathbf{A}_{[1,4/3,6]}$ repräsentiert den Einfluss der Drehung auf die Translation von \mathcal{F}_1 und $\mathbf{A}_{[4,3/6,3]}$ den Einfluss der Translation auf die Drehung von \mathcal{F}_1 .

Durch die Parametrierung im System \mathcal{F}_1 sind diese Koppelmatrizen jeweils antisymmetrisch.

4.5.4 Radaufhängungen: Einfedern, Lenken



Die radaufhängungsinterne Dynamik wird jeweils durch eine 2×2 -Submatrix auf der Hauptdiagonalen der Systemmatrix \mathbf{A} modelliert. Die Kopplungswirkung der Bewegung der Radaufhängung auf die Bewegung des Fahrzeugsystems \mathcal{F}_1 bildet jeweils eine 2×6 -Submatrix im oberen Dreieck von \mathbf{A} nach. Den Einfluss der Fahrzeugbewegung auf die Bewegung der Radaufhängung wird jeweils durch eine 6×2 -Submatrix im unteren Dreieck repräsentiert. Die Elemente dieser Matrizen sind i.Allg. stets von Null verschieden. Die Dynamik der einzelnen Radaufhängungen ist voneinander unabhängig.

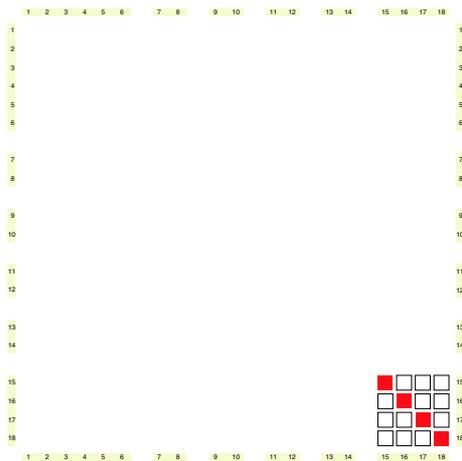


Im Bereich $\mathbf{A}_{[7,7/14,14]}$ der ‘reinen’ Radaufhängungsdynamik sind deshalb nur die 2×2 -Submatrizen für die interne Dynamik einer Radaufhängung besetzt. Die anderen Elemente zur Kopplung der Radaufhängungen untereinander sind stets Null.

Bei einer als unabhängige Einzelradaufhängung realisierten Hinterradaufhängung reduzieren sich die zwei Freiheitsgrade der Aufhängung auf einen einzigen. Die zugehörigen Matrix-Elemente sind stets Null.

Eine als rechts/links-gekoppelt ausgeführte Hinterachse besitzt insgesamt zwei Freiheitsgrade. Jeweils die zweite Koordinate einer Radaufhängung ist kinematisch an die Koordinate der gegenüberliegenden Aufhängung gekoppelt.

4.5.5 Räder: Rotation



Die Raddrehung wird unabhängig von der Fahrzeug- und Radaufhängungsbewegung formuliert.

Die Raddrehungen sind untereinander und von den anderen Freiheitsgraden des Fahrzeugs entkoppelt.

In der 4×4 -Submatrix $\mathbf{A}_{[15,15/18,18]}$ der Raddrehungen sind die Diagonalelemente stets von Null verschieden. Alle anderen Terme sind Null.

Kopplungen der Raddrehung an die Radaufhängung und das Fahrzeug werden über eingeprägte Kräfte modelliert. Sie treten deshalb in der Systemmatrix \mathbf{A} nicht auf und stehen auf der rechten Seite des DGL-Systems.

Die Bereiche $\mathbf{A}_{[1,15/14,18]}$ und $\mathbf{A}_{[15,1/18,14]}$ sind stets Null.

4.5.6 Elemente der Massenmatrix

Die Elemente der Massenmatrix \mathbf{A} werden durch direktes Auswerten des Prinzips von D’Alembert/Lagrange berechnet (siehe Abschnitt 4.2.7 und 4.5.1). Folgende Elemente von \mathbf{A} werden nachfolgend exemplarisch vorgestellt:

- Die Elemente der ersten Zeile von \mathbf{A} für die Bewegung von \mathcal{G}_1
- Die Elemente der ersten und zweiten Zeile von \mathbf{A} für die Bewegung eines \mathcal{G}_2
- Die Elemente von \mathbf{A} für die Drehung der Räder

Abkürzend gilt: $\bar{X} = \bar{X}_1$, $\bar{Y} = \bar{Y}_1$, $\bar{Z} = \bar{Z}_1$. Nicht aufgeführte Elemente berechnen sich analog oder sind gleich Null.

Kopplung Translation \mathcal{B}_1 an Translation \mathcal{B}_1

$$a_{11} = \int_{\mathcal{G}_1} \bar{X} \cdot \bar{X} \, dm = m^1 \quad (4.5.2)$$

$$a_{12} = \int_{\mathcal{G}_1} \bar{X} \cdot \bar{Y} \, dm = 0 \quad (4.5.3)$$

$$a_{13} = \int_{\mathcal{G}_1} \bar{X} \cdot \bar{Z} \, dm = 0 \quad (4.5.4)$$

Kopplung Translation \mathcal{G}_1 an Rotation \mathcal{G}_1

$$a_{14} = \int_{\mathcal{G}_1} \underbrace{\bar{X} \cdot (\bar{X} \times \overline{O_1 M})}_{=0} \, dm = 0 \quad (4.5.5)$$

$$a_{15} = \int_{\mathcal{G}_1} \bar{X} \cdot (\bar{Y} \times \overline{O_1 M}) \, dm = \bar{Z} \cdot \int_{\mathcal{G}_1} \overline{O_1 M} \, dm \quad (4.5.6)$$

$$a_{16} = \int_{\mathcal{G}_1} \bar{X} \cdot (\bar{Z} \times \overline{O_1 M}) \, dm = -\bar{Y} \cdot \int_{\mathcal{G}_1} \overline{O_1 M} \, dm \quad (4.5.7)$$

Kopplung Translation \mathcal{G}_1 an Bewegung \mathcal{G}_2

$$a_{17} = \bar{X} \cdot \int_{\mathcal{G}_2} \frac{\partial \overline{O_1 M}}{\partial q_{\text{whl travel}}} \, dm = \bar{X} \cdot (\bar{p}_{\text{whl travel}} M^2 + \bar{r}_{\text{whl travel}} \times \bar{U}^2) \quad (4.5.8)$$

$$a_{18} = \bar{X} \cdot \int_{\mathcal{G}_2} \frac{\partial \overline{O_1 M}}{\partial q_{\text{steer}}} \, dm = \bar{X} \cdot (\bar{p}_{\text{steer}} M^2 + \bar{r}_{\text{steer}} \times \bar{U}^2) \quad (4.5.9)$$

Kopplung Rotation \mathcal{G}_1 an Rotation \mathcal{G}_1

$$a_{44} = \int_{\mathcal{G}_1} (\bar{X} \times \overline{O_1 M}) \cdot (\bar{X} \times \overline{O_1 M}) \, dm = \bar{X} \cdot \bar{I}^1 \cdot \bar{X} \quad (4.5.10)$$

$$a_{45} = \bar{X} \cdot \bar{I}^1 \cdot \bar{Y} \quad (4.5.11)$$

$$a_{46} = \bar{X} \cdot \bar{I}^1 \cdot \bar{Z} \quad (4.5.12)$$

Kopplung Rotation \mathcal{B}_1 an Bewegung $\mathcal{G}_{2,fl}$

$$\begin{aligned} a_{47} &= \bar{X} \cdot \int_{\mathcal{G}_2} \overline{O_1 M} \times \frac{\partial \overline{O_1 M}}{\partial q_{\text{whl travel}}} dm \\ &= \bar{X} \cdot \overline{O_1 C_2} M^2 - \bar{X} \cdot \bar{I}^1 \cdot \bar{r}_{\text{whl travel}} \end{aligned} \quad (4.5.13)$$

$$\begin{aligned} a_{48} &= \bar{X} \cdot \int_{\mathcal{G}_2} \overline{O_1 M} \times \frac{\partial \overline{O_1 M}}{\partial q_{\text{steer}}} dm \\ &= \bar{X} \cdot \overline{O_1 C_2} M^2 - \bar{X} \cdot \bar{I}^1 \cdot \bar{r}_{\text{steer}} \end{aligned} \quad (4.5.14)$$

Kopplung Bewegung \mathcal{G}_2 an Bewegung \mathcal{G}_2

$$\begin{aligned} a_{77} &= \bar{p}_{\text{whl travel}} \cdot \bar{p}_{\text{whl travel}} M^2 + 2 \bar{p}_{\text{whl travel}} \cdot \bar{r}_{\text{whl travel}} \times \bar{U}^2 \\ &\quad + \bar{r}_{\text{whl travel}} \cdot \bar{I}^2 \cdot \bar{r}_{\text{whl travel}} \end{aligned} \quad (4.5.15)$$

$$\begin{aligned} a_{78} &= \bar{p}_{\text{whl travel}} \cdot \bar{p}_{\text{steer}} M^2 + 2 \bar{p}_{\text{whl travel}} \cdot \bar{r}_{\text{steer}} \times \bar{U}^2 \\ &\quad + \bar{r}_{\text{whl travel}} \cdot \bar{I}^2 \cdot \bar{r}_{\text{steer}} \end{aligned} \quad (4.5.16)$$

$$a_{88} = \bar{p}_{\text{steer}} \cdot \bar{p}_{\text{steer}} M^2 + 2 \bar{p}_{\text{steer}} \cdot \bar{r}_{\text{steer}} \times \bar{U}^2 + \bar{r}_{\text{steer}} \cdot \bar{I}^2 \cdot \bar{r}_{\text{steer}} \quad (4.5.17)$$

Rotation der Räder

$$\begin{aligned} a_{1515} &= \int_{W_{fl}} (\bar{Y}_{E_{fl}} \times \overline{O_{W_{fl}} M}) \cdot (\bar{Y}_{E_{fl}} \times \overline{O_{W_{fl}} M}) dm \\ &= \bar{Y}_{E_{fl}} \cdot \bar{I}^{W_{fl}} \cdot \bar{Y}_{E_{fl}} \end{aligned} \quad (4.5.18)$$

4.5.7 Optimierung der Transformation eines Tensors 2-ter Stufe

Bei bauteilorientierter Parametrierung wird der Trägheitstensor eines Körpers im körperfesten System angegeben, üblicherweise bez. des Körperschwerpunkts.

Die Berechnung des Gesamt-Trägheitstensors eines Systems, das aus mehreren Einzelkörpern aufgebaut ist, setzt sich aus zwei Schritten zusammen: Der Transformation auf einen neuen Bezugspunkt und der Änderung der Darstellungsbasis.

Transformation eines Trägheitstensors auf einen neuen Bezugspunkt ohne Änderung des Bezugssystems

Die Einzelkörper \mathcal{B}_i besitzen die Masse m_i . Ihr Trägheitstensor $\bar{I}_i(O_i)$ liegt jeweils im Koordinatensystem \mathcal{F}_i , bezogen auf einen Punkt O_i , vor. Bevor zwei Trägheitstensoren

aufaddiert werden können, müssen sie auf einen gemeinsamen Bezugspunkt O_j umgerechnet werden (Steinerscher Satz):

$$\bar{\bar{I}}_i(O_j) = \bar{\bar{I}}_i(O_i) + m_i \overline{O_j O_i} \otimes \overline{O_j O_i} \quad (4.5.19)$$

Transformation eines Trägheitstensors in ein neues Bezugssystem ohne Änderung des Bezugspunktes

Mit Hilfe der Transformationsmatrix $\mathbf{T}_{i \rightarrow j}$ (“von i in das Ziel-System j ”) werden die Trägheitstensoren der Einzelkörper nach folgender Berechnungsvorschrift vom Koordinatensystem \mathcal{F}_i in das Ziel-System \mathcal{F}_j transformiert:

$$\left[\bar{\bar{I}}_j(O_j) \right] = \mathbf{T}_{i \rightarrow j} \cdot \left[\bar{\bar{I}}_i(O_i) \right] \cdot \mathbf{T}_{i \rightarrow j}^T \quad (4.5.20)$$

Für diese Transformation sind – vollständig berechnet – zwei komplette $[3 \times 3]$ -Matrix-Multiplikationen notwendig.

Ein Trägheitstensor besitzt folgende, für eine Optimierung interessante Eigenschaften:

- Der Tensor ist symmetrisch: $a_{ij} = a_{ji}$.
- Seine Spur ist unabhängig von der Basis, in der er dargestellt wird:
 $\text{tr}(\bar{\bar{I}}) = a_{11} + a_{22} + a_{33} = \text{const}$ und kann somit vorab berechnet werden.

Werden diese Eigenschaften bei der Tensor-Transformation berücksichtigt, lässt sich der numerische Aufwand um über 36 Prozent reduzieren: Statt 54 Multiplikationen und 36 Additionen müssen nur noch 33 und 24 ausgeführt werden. Dies führt vor allem bei Modellen, bei denen die Radaufhängung mit vielen Einzelkörpern detailliert modelliert ist, zu einer deutlichen Einsparung an Rechenzeit.

4.6 Aufstellen der Bewegungsgleichungen des Fahrzeugs (in analytischer Behandlung)

Dieser Abschnitt behandelt die Formalismen zur Beschreibung der Bewegungsgleichungen des Systems “Kraftfahrzeug”. Dafür notwendige Grundlagen und Teilaspekte der Modellierung und der Koordinatenwahl wurden bereits zuvor betrachtet.

Die Bewegungsgleichungen des Systems werden in folgender Form (siehe Gleichung (4.4.1)) aufgestellt. Die kinematischen Basisdaten werden hierfür aufsteigend entlang der kinematischen Kette relativkinematisch ermittelt: von der Wurzel beim Galileischen System \mathcal{F}_0 , über das fahrzeugaufbaufeste Bezugssystem \mathcal{F}_1 und den einzelnen Fahrwerksachsen zu den Körpern \mathcal{B}_2 der Radaufhängung, den Ästen. Die Raddrehung wird hiervon getrennt behandelt. In umgekehrter Richtung wieder zurück, von den Blättern zur Wurzel, wird die Massengeometrie der Einzelkörper, der generalisierten Körper und

darüber die Massengeometrie des gesamten Systems bestimmt. Grundlage der Berechnung bildet ein Formalismus von Fayet [Fay90]. Ursprünglich zur Berechnung notwendiger Antriebskräfte in reinen Dreh- und Schubgelenken von Robotern gedacht, wurde es auf die im Fahrzeugmodell verwendeten allgemeinen Gelenke mit 6 Koordinaten (3 translatorisch, 3 rotatorisch) erweitert.

Die Elemente der Massenmatrix werden in analytischer Behandlung aufgestellt. Hierzu wird das D'Alembert'sche Prinzip direkt ausgewertet, siehe Abschnitt 4.5.1, und vollständig algebraisiert.

Auf der linken Seite des Gleichungssystems stehen ausschließlich Terme reiner Beschleunigungen. Die Restbeschleunigungen, d.h. Beschleunigungen oder Koppelterme, die durch Produkte von Geschwindigkeiten entstehen, werden auf der rechten Seite des Gleichungssystems mit den äußeren, eingepprägten Kräften und Momenten im Vektor \mathbf{b} zusammengefasst. Zur Ermittlung der Restbeschleunigungen wird das bereits zuvor erwähnte Verfahren von Fayet erneut speziell adaptiert. Wird der Formalismus mit modifizierten Eingangsbedingungen, bei denen alle zweiten Ableitungen von Lagen und ersten von Geschwindigkeiten zu Null gesetzt wurden, ausschließlich mit Geschwindigkeiten durchgeführt, liefert er statt der vollständigen Trägheitskräfte die gesuchten Restbeschleunigungsgrößen.

4.6.1 Rekursive Verfahren der Kinetik und das Fischer-Koordinatensystem

Das hier vorgestellte rekursive Verfahren wird besonders übersichtlich durch die Definition eines neuen Beobachtersystems, das "Fischer Frame" \mathcal{F}_i^F [Fay90]:

$$\mathcal{F}_i^F \equiv \{O_i, \bar{X}_0 \bar{Y}_0 \bar{Z}_0\} \quad . \quad (4.6.1)$$

Das Fischer-Koordinatensystem \mathcal{F}_i^F hat seinen Ursprung stets im Gelenkpunkt O_i . Die Achsen von \mathcal{F}_i^F sind stets parallel zum Galileischen Koordinatensystem \mathcal{F}_0 , d.h. \mathcal{F}_i^F ergibt sich aus \mathcal{F}_0 durch reine Translation.

Mit Hilfe von \mathcal{F}_i^F lassen sich Translations- und Rotationsanteile einer Bewegung voneinander trennen. Die Absolutbeschleunigung $\bar{a}^0(P_i)$ eines Punktes P_i ergibt sich aus der absoluten Beschleunigung des Gelenkpunktes O_i und der Beschleunigung von P_i bez. des zugehörigen Fischer-Systems \mathcal{F}_i^F . Der Coriolissterm $2 \bar{\Omega}_{\mathcal{F}_i^F}^0 \times \bar{v}^{\mathcal{F}_i^F}(P_i)$ entfällt, da $\bar{\Omega}_{\mathcal{F}_i^F}^0 = \bar{0}$:

$$\bar{a}^0(P_i) = \bar{a}_{\mathcal{F}_i^F}^0(P_i) + \bar{a}^{\mathcal{F}_i^F}(P_i) = \bar{a}^0(O_i) + \bar{a}^{\mathcal{F}_i^F}(P_i) \quad . \quad (4.6.2)$$

Im Rahmen des Formalismus zur Berechnung der Bewegungsgleichungen des Fahrzeugmodells werden nachfolgende iterative Formeln verwendet. Sie bedienen sich hierfür der Strukturen generalisierter und erweiterter Körper (siehe Abschnitt 4.2.1.3 und Abschnitt

4.2.3.2).

Die dynamische Summe $\bar{\Sigma}_{\mathcal{G}_i}^0$ auf ein System \mathcal{G}_i bez. \mathcal{F}_0 lässt sich zerlegen in

$$\bar{\Sigma}_{\mathcal{G}_i}^0 = \bar{\Sigma}_{\mathcal{G}_i}^{\mathcal{F}_i^F} + M^{\mathcal{G}_i} \bar{a}^0(O_i) \quad . \quad (4.6.3)$$

Für $\bar{\Sigma}_{\mathcal{G}_k}^{\mathcal{F}_i^F}$, $k \geq i + 1$, auf ein System \mathcal{G}_k bez. des Fischer-Systems \mathcal{F}_i gilt hiermit

$$\bar{\Sigma}_{\mathcal{G}_k}^{\mathcal{F}_i^F} = \bar{\Sigma}_{\mathcal{G}_k}^{\mathcal{F}_k^F} + M^{\mathcal{G}_k} \bar{a}^{\mathcal{F}_i^F}(O_k) \quad . \quad (4.6.4)$$

Die dynamische Summe $\bar{\Sigma}_{\mathcal{G}_i}^{\mathcal{F}_i^F}$ bez. \mathcal{F}_i^F auf ein System \mathcal{G}_i lässt sich mit dem erweiterten Körper \mathcal{B}_i^+ rekursiv formulieren:

$$\bar{\Sigma}_{\mathcal{G}_i}^{\mathcal{F}_i^F} \equiv \int_{\mathcal{G}_i} \bar{a}^{\mathcal{F}_i^F}(M) dm = \int_{\mathcal{B}_i} \bar{a}^{\mathcal{F}_i^F}(M) dm + \sum_{k|D_{jk}=1} \int_{\mathcal{G}_k} \bar{a}^{\mathcal{F}_i^F}(M) dm \quad (4.6.5)$$

$$= \bar{\Sigma}_{\mathcal{B}_i}^{\mathcal{F}_i^F} + \sum_{k|D_{jk}=1} \left\{ \bar{\Sigma}_{\mathcal{G}_k}^{\mathcal{F}_k^F} + M^k \bar{a}^{\mathcal{F}_i^F}(O_k) \right\} \quad (4.6.6)$$

$$= \bar{\Sigma}_{\mathcal{B}_i^+}^{\mathcal{F}_i^F} + \sum_{k|D_{jk}=1} \bar{\Sigma}_{\mathcal{G}_k}^{\mathcal{F}_k^F} \quad (4.6.7)$$

Das dynamische Moment $\bar{\delta}_{\mathcal{G}_i}^0(O_i)$ im Punkt O_i bez. \mathcal{F}_i^F auf ein System \mathcal{G}_i lautet

$$\bar{\delta}_{\mathcal{G}_i}^0(O_i) = \bar{\delta}_{\mathcal{B}_i}^0(O_i) + \sum_{k|D_{ik}=1} \left\{ \bar{\delta}_{\mathcal{G}_k}^0(O_k) + \overline{O_i O_k} \times \bar{\Sigma}_{\mathcal{G}_k}^0 \right\} \quad (4.6.8)$$

Mit Hilfe des erweiterten Körpers lässt sich dies auf folgende rekursive Darstellung umformen:

$$\bar{\delta}_{\mathcal{G}_i}^0(O_i) = \bar{\delta}_{\mathcal{B}_i^+}^0(O_i) + \sum_{k|D_{ik}=1} \left\{ \bar{\delta}_{\mathcal{G}_k}^0(O_k) + \overline{O_i O_k} \times \bar{\Sigma}_{\mathcal{G}_k}^{\mathcal{F}_k^F} \right\} \quad (4.6.9)$$

Für $\bar{\delta}_{\mathcal{B}_i^+}^0(O_i)$ gilt die Berechnungsvorschrift

$$\bar{\delta}_{\mathcal{B}_i^+}^0(O_i) = \bar{\delta}_{\mathcal{B}_i^+}^{\mathcal{F}_i^F}(O_i) + \bar{u}_+^i(O_i) \times \bar{a}^0(O_i) \quad . \quad (4.6.10)$$

4.6.2 Kinematische Basisdaten ($\mathcal{F}_0 \rightarrow \mathcal{F}_1 \rightarrow \mathcal{F}_2$, “aufwärts”)

Kinematische Basisdaten: $\mathcal{F}_1, \mathcal{B}_1$

Aus dem zurückliegenden Zeitschritt stehen folgende Größen zur Verfügung:

\bar{t}_1^0 Translation von \mathcal{F}_1 bez. \mathcal{F}_0

\bar{v}_1^0	Translationsgeschwindigkeit von \mathcal{F}_1 bez. \mathcal{F}_0
\bar{a}_1^0	Translationsbeschleunigung von \mathcal{F}_1 bez. \mathcal{F}_0
$\bar{\eta}_1^0$	Orientierung von \mathcal{F}_1 bez. \mathcal{F}_0 , ausgedrückt über eine Euler-Quaternion
$\bar{\Omega}_1^0$	Rotationsgeschwindigkeit von \mathcal{F}_1 bez. \mathcal{F}_0
$\bar{\alpha}_1^0$	Rotationsbeschleunigung von \mathcal{F}_1 bez. \mathcal{F}_0

Kinematische Basisdaten: $\mathcal{F}_2, \mathcal{B}_2$

Die einzelnen Körper \mathcal{B}_2 der Radaufhängungen werden alle bez. des fahrzeugaufbaufesten Koordinatensystems \mathcal{F}_1 parametrisiert. Es werden für jeden \mathcal{B}_2 folgende Größen, abhängig von den (ein oder zwei) Freiheitsgraden \mathbf{q} bereitgestellt:

$\bar{t}_2^1(\mathbf{q})$	Translation von \mathcal{F}_2 bez. \mathcal{F}_1
$\bar{\eta}_2^1(\mathbf{q})$	Orientierung von \mathcal{F}_2 bez. \mathcal{F}_1 , ausgedrückt über Euler-Winkel mit der Drehreihenfolge $\bar{Z} - \bar{X} - \bar{Y}$

$$\frac{\partial \bar{t}_2^1(\mathbf{q})}{\partial q_i}, \frac{\partial \bar{\eta}_2^1(\mathbf{q})}{\partial q_i}$$

Erste Ableitung der Lage und Orientierung

$$\frac{\partial^2 \bar{t}_2^1(\mathbf{q})}{\partial q_i^2}, \frac{\partial^2 \bar{t}_2^1(\mathbf{q})}{\partial q_i \partial q_j}, \frac{\partial^2 \bar{\eta}_2^1(\mathbf{q})}{\partial q_i^2}, \frac{\partial^2 \bar{\eta}_2^1(\mathbf{q})}{\partial q_i \partial q_j}$$

Zweite Ableitungen der Lage und Orientierung

Hiermit werden folgende Größen berechnet:

$\mathbf{M}(\mathbf{q})$ Maggi-Matrix

$\bar{\mathbf{p}}(\mathbf{q})$ Translationsachsenvektor

$\bar{\mathbf{r}}(\mathbf{q})$ Rotationsachsenvektor von $\dot{\mathbf{q}}$

$\bar{\mathbf{r}}_{zxy}(\mathbf{q})$

Rotationsachsenvektor abh. von den von \mathbf{q} abhängigen Euler-Winkelgeschwindigkeiten \dot{r}_z, \dot{r}_x und \dot{r}_y

Die Rotationsgeschwindigkeit $\bar{\Omega}_2^0$ aller Koordinatensysteme \mathcal{F}_2 werden über eine kinematische Kette berechnet:

$$\bar{\Omega}_2^0 = \bar{\Omega}_1^0 + \bar{\Omega}_2^1 \quad (4.6.11)$$

$$= \bar{\Omega}_1^0 + \sum_{q_2} \bar{\mathbf{r}}_{2,q_2}^1 \cdot \dot{q}_2 \quad (4.6.12)$$

$$= \bar{\Omega}_1^0 + \sum_{q_2} \bar{\mathbf{r}}_{zxy,q_2}^1 \cdot \mathbf{M}_{q_2} \cdot \dot{q}_2 \quad (4.6.13)$$

4.6.3 Massengeometrie ($\mathcal{F}_2 \rightarrow \mathcal{F}_1 \rightarrow \mathcal{F}_0$, “abwärts”)

Trägheitstensor zweiter Ordnung $\bar{\bar{I}}$ eines Körpers \mathcal{B}_2 bez. O_1 :

$$\bar{\bar{I}}^{1\ 2} = -\bar{\bar{I}}^{2\ 2} - \hat{u}^2 \cdot \overline{O_1 O_2} - \overline{O_1 O_2} \cdot \overline{O_1 O_2} m^2 \quad . \quad (4.6.14)$$

Mit $\mathcal{G}_2 = \mathcal{B}_2$ gilt hierbei: $\bar{U}^2 = \bar{u}^2$.

Trägheitstensor (-Hilfsgröße) $\bar{\bar{I}}_{12}$ eines Körpers \mathcal{B}_2

$$\bar{\bar{I}}_{12} = \overline{O_1 O_2} \cdot \bar{U}^2 + \bar{\bar{I}}^{2\ 2} = \int_{\mathcal{B}_2} \overline{O_1 M} \cdot \overline{O_2 M} dm \quad . \quad (4.6.15)$$

Trägheitstensor erster Ordnung \bar{U}^1 des generalisierten (generalized) Körpers \mathcal{G}_1 bez. O_1

$$\bar{U}^1 = \bar{u}_+^1 + \sum_{\mathcal{B}_2} \bar{u}^{1\ 2} \quad (4.6.16)$$

$$= \overline{O_1 C_1} m^1 + \sum_{\mathcal{B}_2} \{ \overline{O_1 O_2} m^2 + \bar{u}^2 \} \quad . \quad (4.6.17)$$

Trägheitstensor zweiter Ordnung $\bar{\bar{I}}^{1+}$ des erweiterten (augmented) Körpers \mathcal{B}_1^+ bez. O_1

$$\bar{\bar{I}}^{1+} = \bar{\bar{I}}^1 - \sum_{\mathcal{B}_2} \overline{O_1 O_2} \cdot \overline{O_1 O_2} m^2 \quad . \quad (4.6.18)$$

Trägheitstensor zweiter Ordnung des generalisierten (generalized) Körpers \mathcal{G}_1 bez. O_1

$$\bar{\bar{I}}_{1,\text{gen}}^1(O_1) = \bar{\bar{I}}_{1,\mathcal{B}_1}^1(O_1) + \sum_{\mathcal{B}_2} \bar{\bar{I}}_{1,\mathcal{B}_2}^1(O_1) \quad (4.6.19)$$

$$= \bar{\bar{I}}_{1,\mathcal{B}_1}^1(O_1) - \sum_{\mathcal{B}_2} \left\{ \overline{O_1 O_2} \cdot \overline{O_1 O_2} \cdot m^2 + \hat{u}^2 \cdot \overline{O_1 O_2} + \bar{\bar{I}}_{12} \right\} \quad (4.6.20)$$

4.6.4 Kinematik und Dynamik

Der zugrunde liegende Algorithmus [Fay90] ermittelt bei vorgegebener Systembewegung die sie generierenden Trägheitskräfte. In der hier eingesetzten Form sind die “reinen” Beschleunigungsterme alle zu Null gesetzt. Die Beschleunigungsterme einer vollständigen Berechnung der Trägheitskräfte sind in eckigen Klammern zusätzlich mit angegeben.

Der Algorithmus ist jeweils nur für einen Körper \mathcal{B}_2 bzw. den generalisierten Körper \mathcal{G}_2 ausformuliert. Zur vollständigen Berechnung ist über alle \mathcal{B}_2 bzw. \mathcal{G}_2 zu summieren.

Rotationsgeschwindigkeit:

$$\bar{\Omega}_2^1 = \bar{r}_1 \dot{q}_1 + \bar{r}_2 \dot{q}_2 = \bar{r}_j \dot{q}_k \quad , \quad (4.6.21)$$

$$\bar{\Omega}_2^0 = \bar{\Omega}_1^0 + \bar{\Omega}_2^1 \quad . \quad (4.6.22)$$

Rotationsbeschleunigungen: Summation über die Freiheitsgrade des \mathcal{B}_2

$$\overline{\alpha}_1^0 = \overline{0} + [\overline{\alpha}_1^0] \quad , \quad (4.6.23)$$

$$\overline{\alpha}_2^1 = \left[\frac{\partial \overline{r}_j}{\partial q_l} \dot{q}_l \right] + \overline{r}_j \dot{q}_k \dot{q}_l \quad . \quad (4.6.24)$$

Beschleunigungs-Distributor:

$$\overline{\overline{A}}_1^0 = \hat{\overline{\Omega}}_1^0 \cdot \hat{\overline{\Omega}}_1^0 + \hat{\overline{\alpha}}_1^0 \quad , \quad (4.6.25)$$

$$\overline{\overline{A}}_2^0 = \hat{\overline{\Omega}}_2^0 \cdot \hat{\overline{\Omega}}_2^0 + \hat{\overline{\alpha}}_2^0 \quad . \quad (4.6.26)$$

Mix-Beschleunigung:

$$\overline{a}_{1,\text{mix}}(O_1) = \overline{0} + [\overline{a}_1^0(O_1)] \quad , \quad (4.6.27)$$

$$\overline{a}_{2,\text{mix}}(O_2) = 2 \overline{\Omega}_1^0 \times \overline{v}_2^1(O_2) + [\overline{a}_2^1(O_2)] \quad . \quad (4.6.28)$$

Beschleunigung der Gelenkpunkte O_i bez. \mathcal{F}_1^F :

$$\overline{a}^0(O_1) = \overline{a}^{\mathcal{F}_1^F}(O_1) = \overline{\overline{A}}_1^0 \cdot \overline{O_0 O_1} + \overline{a}_{1,\text{mix}}(O_1) \quad , \quad (4.6.29)$$

$$\overline{a}^{\mathcal{F}_1^F}(O_2) = \overline{\overline{A}}_2^0 \cdot \overline{O_1 O_2} + \overline{a}_{2,\text{mix}}(O_2) \quad . \quad (4.6.30)$$

Dynamische Summe bez. \mathcal{F}_i^F :

$$\overline{\Sigma}_{\mathcal{B}_1^+}^{\mathcal{F}_1^F} = \overline{\overline{A}}_1^0 \cdot \overline{u}^1 + m_2 \overline{a}_{2,\text{mix}}(O_2) \quad , \quad (4.6.31)$$

$$\overline{\Sigma}_{\mathcal{B}_2^+}^{\mathcal{F}_2^F} = \overline{\overline{A}}_2^0 \cdot \overline{u}^2 \quad . \quad (4.6.32)$$

Dynamisches Moment bez. \mathcal{F}_i^F :

$$\overline{\delta}_{\mathcal{B}_1^+}^{\mathcal{F}_1^F}(O_1) = \overline{\overline{A}}_1^0 \cdot \overline{\pi}_+^1 + \overline{O_1 O_2} \cdot m_2 \overline{a}_{2,\text{mix}}(O_2) \quad , \quad (4.6.33)$$

$$\overline{\delta}_{\mathcal{B}_2^+}^{\mathcal{F}_2^F}(O_2) = \overline{\overline{A}}_2^0 \cdot \overline{\pi}_+^2 \quad . \quad (4.6.34)$$

Beschleunigung der Gelenkpunkte O_i

$$\overline{a}_1^0(O_1) = \overline{0} + [\overline{a}_1^0(O_1)] \quad , \quad (4.6.35)$$

$$\overline{a}_2^0(O_2) = \overline{a}_1^0(O_1) + \overline{\overline{A}}_2^0 \cdot \overline{O_1 O_2} + 2 \overline{\Omega}_1^0 \times \overline{v}_2^1(O_2) + [\overline{a}_2^1(O_2)] \quad . \quad (4.6.36)$$

Dynamische Summe bez. \mathcal{F}_i^F :

$$\overline{\Sigma}_{\mathcal{G}_2}^{\mathcal{F}_2^F} = \overline{\Sigma}_{\mathcal{B}_2}^{\mathcal{F}_2^F} \quad , \quad (4.6.37)$$

$$\overline{\Sigma}_{\mathcal{G}_1}^{\mathcal{F}_1^F} = \overline{\Sigma}_{\mathcal{B}_1^+}^{\mathcal{F}_1^F} + \overline{\Sigma}_{\mathcal{B}_2}^{\mathcal{F}_2^F} \quad . \quad (4.6.38)$$

Dynamische Summe bez. \mathcal{F}_0 :

$$\overline{\Sigma}_{\mathcal{G}_2}^0 = \overline{\Sigma}_{\mathcal{B}_2}^0 = \overline{\Sigma}_{\mathcal{B}_2}^{\mathcal{F}_2^F} + m_2 \overline{a}_2^0(O_2) \quad , \quad (4.6.39)$$

$$\overline{\Sigma}_{\mathcal{G}_1}^0 = \overline{\Sigma}_{\mathcal{G}_1}^{\mathcal{F}_1^F} + M^1 \overline{a}^{\mathcal{F}_0^F}(O_1) \quad . \quad (4.6.40)$$

Dynamisches Moment bez. \mathcal{F}_0 :

$$\overline{\delta}_{\mathcal{G}_2}^0(O_2) = \overline{\delta}_{\mathcal{B}_2^+}^{\mathcal{F}_2^F}(O_2) + \overline{u}^2 \times \overline{a}^0(O_2) \quad , \quad (4.6.41)$$

$$\overline{\delta}_{\mathcal{G}_1}^0(O_1) = \overline{\delta}_{\mathcal{G}_2}^0(O_2) + \overline{O_1 O_2} \times \overline{\Sigma}_{\mathcal{G}_2}^{\mathcal{F}_2^F} + \overline{\delta}_{\mathcal{B}_1^+}^{\mathcal{F}_1^F}(O_1) + \overline{u}^1 \times \overline{a}^0(O_1) \quad . \quad (4.6.42)$$

Generalisierte Kräfte: Mit den externen Kräften \overline{F}_i^0 und den externen Momenten \overline{T}_i^0 auf die Systeme \mathcal{G}_i im Punkt O_i der Translationsachse $\overline{p}_j = \overline{p}_j(O_i)$

$$Q_2 = \overline{p}_2 \cdot \overline{F}_2^0 + \overline{r}_2 \cdot \overline{T}_2^0(O_2) \quad , \quad (4.6.43)$$

$$Q_1 = \overline{p}_1 \cdot \overline{F}_1^0 + \overline{r}_1 \cdot \overline{T}_1^0(O_1) \quad . \quad (4.6.44)$$

Trägheitskräfte Γ_i

$$\Gamma_2 = - \left(\overline{p}_2 \cdot \overline{\Sigma}_{\mathcal{G}_2}^0 + \overline{r}_2 \cdot \overline{\delta}_{\mathcal{G}_2}^0(O_2) \right) \quad , \quad (4.6.45)$$

$$\Gamma_1 = - \left(\overline{p}_1 \cdot \overline{\Sigma}_{\mathcal{G}_1}^0 + \overline{r}_1 \cdot \overline{\delta}_{\mathcal{G}_1}^0(O_1) \right) \quad . \quad (4.6.46)$$

Dynamik-Gleichungen

$$a_2 \ddot{q}_2 = Q_2 + \Gamma_2 \quad , \quad (4.6.47)$$

$$a_1 \ddot{q}_1 = Q_1 + \Gamma_1 \quad . \quad (4.6.48)$$

4.7 Zusammenfassung

Den Kern der vorgestellten Simulationsplattform bildet die funktional strukturierte Modellierung der “mechanischen Grundstruktur” der Fahrzeuge. Die Bewegungsgleichungen sind nach dem Prinzip von D’Alembert durch Projektion virtueller Verrückungen auf freie Bewegungsrichtungen direkt in den Beschleunigungen aufgestellt. Vereinfachende Annahmen, wie z.B. Symmetrieeigenschaften des Fahrzeugs, werden nicht getroffen. Das Objekt des “starrten Körpers” ist keine Grundvoraussetzung; es wird erst in einem der letzten Modellierungsschritte zur Vereinfachung und besseren Nachvollziehbarkeit der Terme darauf zurückgegriffen. Deformierbare Strukturen können bei Bedarf integriert werden. Die Dynamik der Räder wird entkoppelt von der Fahrzeugbewegung nach Euler aufgestellt. Corioliseffekte sind über die rechte Seite \mathbf{b} berücksichtigt.

Durch intelligente Wahl der Koordinaten wird Externales von Internalem getrennt. D.h. die System-Massenmatrix \mathbf{A} ist stets frei von der Position und Orientierung des Bezugssystems \mathcal{F}_1 bez. des Galileischen Systems. Genauigkeit und Stabilität des Modells

bleiben erhalten, unabhängig davon, ob sich das Fahrzeug in der Nähe des Ursprungs oder sehr weit davon entfernt befindet. Lage und Orientierung des Fahrzeugs sind über eine Euler-Quaternion frei von singulären Lagen parametrisiert und unterstützen den gesamten Bereich der geforderten Szenarien. Sowohl die Gesamtstruktur der Bewegungsgleichungen (Systemmatrix A und rechte Seite b) als auch deren einzelne Terme sind aus physikalisch anschaulichen Ausdrücken aufgebaut.

Das Modell des Fahrwerks abstrahiert von der konkreten Konstruktion der Aufhängung. Die Bewegung aller Körper der Radaufhängung wird auf eine Führungsbewegung abhängig von den zwei generalisierten Koordinaten Einfedern und Lenken kondensiert. Die Massenverteilung innerhalb der Radaufhängung (Radträger, Querlenker, Feder, Dämpfer, Stabilisator, ...) wird vollständig berücksichtigt. Elastokinematische Effekte werden dieser Führungsbewegung überlagert. Die Parametrisierung der dreidimensionalen Bauteilbewegungen erfolgt über optimierte Kennlinien- und Kennfeld-Funktionen. Die Anzahl der modellierten Einzelkörper ist über die Parametrisierung frei wählbar und reicht von einem auf "High Performance" ausgelegten Parametersatz mit minimierter Rechenzeit und minimiertem Ressourcenbedarf mit nur einem modellierten Körper Radträger bis zu einem "High Quality"-Datensatz, bei dem sich jedes reale Bauteil auch im Modell als Körper wiederfindet. Eine Parametrisierung durch Vermessung eines vorhandenen Fahrzeugs ist direkt möglich.

Durch seine Modularität, seine Schnittstellen und die verfügbaren und anpassbaren Beobachtergrößen ist diese Fahrdynamik-Simulationsplattform sehr effizient nach dem Baukastenprinzip durch anwendungsspezifische Teilmodule erweiterbar, wie z.B. ein Lenksystem oder aktive Fahrwerkselemente. Die Parameter, Beobachtergrößen und Schnittstellen sind für die verschiedenen Typen von Fahrzeugen übersichtlich und identisch strukturiert.

Das Ergebnis dieser auf die Struktur des Kraftfahrzeugs und die angestrebten Möglichkeiten zur Parametrisierung hin ausgerichteten Modellierung sind ressourceneffiziente, stabile und echtzeitfähige Fahrdynamik-Simulationsmodelle für Pkw, Pkw-Anhänger und Motorrad. Sie sind ohne "proprietäre Simulationsumgebung" und spezielle Programm-Bibliotheken lauffähig. Sie liegen direkt im Quellcode vor und konnten dadurch auf alle gängigen Plattformen portiert und leicht in die Simulationsumgebungen zur Standalone-Simulation, in Simulink-Blockdiagramme und zur HIL-Simulation integriert werden.

Kapitel 5

Reifen-Kinematik auf gewölbter Fahrbahn

5.1 Motivation und Aufgabenstellung

Durch die Ergebnisse dieses Abschnitts werden Reifenparametrierungen und Modelle, die für ebene Fahrbahnen gewonnen wurden, auf gewölbte Fahrbahnen anwendbar.

Definition *Gewölbte Fahrbahn*: Die Krümmungsradien der Fahrbahnoberfläche S sind größer als r . Hierbei ist r der Reifenradius.

Entsprechend den Modellvorstellungen der Fahrdynamik sind Reifenkräfte sogenannte eingeprägte Kräfte. In der Terminologie der klassischen Mechanik sind *eingeprägte Kräfte* $\overline{F}^{(E)}$ durch ein von *vornherein* bekanntes Kraftgesetz als Funktion der kinematischen Größen “Ort” und “Geschwindigkeit” gegeben. Im Gegensatz dazu sind *Zwangskräfte* $\overline{F}^{(Z)}$ Kräfte, die für einen Punkt oder Körper eine geometrische Bindung erzwingen.

Die “eingepprägten” Reifenkräfte werden bestimmt durch die kinematischen Verhältnisse im Reifen-Fahrbahn-Kontakt, insbesondere durch die Größen *Schlupf* [Jah18] [Sac24] [Fro27] und *Schräglaufwinkel*. Ihre Berechnung ist einfach im Falle ebener Fahrbahnen. Gegenstand dieses Kapitels ist eine Verallgemeinerung auf den Fall gewölbter Fahrbahnen. Es werden eine Notation und Definitionen eingeführt, auf deren Grundlage sich die Kinematik des Reifen-Fahrbahn-Kontaktes präzise formulieren lässt. Mit dem eingeführten Formelapparat werden die reifenkinematischen Referenzgrößen *Schlupf* und *Schräglaufwinkel* zunächst präzise definiert. Die “Karlsruher Kinematische Kette K3”, eine Kette kinematischer Beziehungen, wird aufgebaut. K3 ist das zentrale Element zur Berechnung der Reifen- und Rad-Kinematik auf gewölbter Fahrbahn. Dieser Abschnitt schließt mit einer Zusammenstellung des Auswertungsablaufs zur Algebraisierung der Referenzgrößen.

5.2 Kontaktkraft-Modell

Zum besseren Verständnis dessen, wofür die Ergebnisse des hier vorgestellten Verfahrens eingesetzt werden, sei hier kurz die für die Fahrdynamik- und Echtzeitsimulation typische Modellierung der Reifen-Fahrbahn-Kontaktkräfte erläutert (Abbildung 5.2.1).

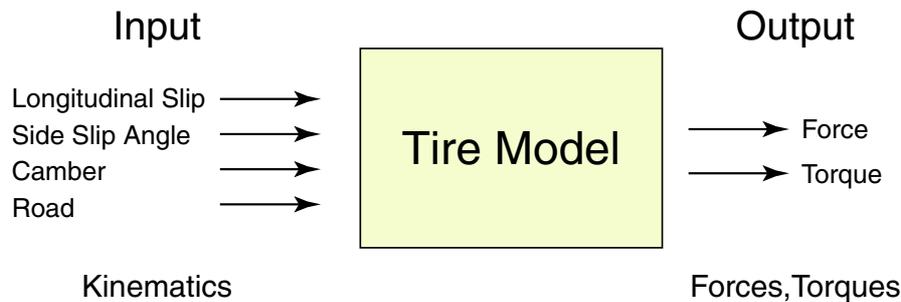


Abbildung 5.2.1: Reifenmodell

5.2.1 Kinematik

Alle Reifenmodelle – soweit sie für die Echtzeitsimulation von Bedeutung sind – stellen einen Zusammenhang her zwischen dem kinematischen Zustand im Latsch (an der Kontaktstelle P von Reifen und Fahrbahn) und den daraus resultierenden Kräften und Momenten.

Der kinematische Zustand im Latsch wird durch folgende Größen beschrieben:

- $\vec{v}^0(P)$ = die Geschwindigkeit des Kontaktpunktes P gegenüber der Fahrbahn \mathcal{F}_0 (ruhesystem)
- $\vec{v}^B(P)$ = die Geschwindigkeit des Kontaktpunktes P gegenüber dem Latsch-(Belt)-festen System \mathcal{F}_B
- $\vec{v}_B^0(P)$ = die Geschwindigkeit gegenüber der Fahrbahn \mathcal{F}_0 desjenigen latschfesten Punktes, der zum Beobachtungszeitpunkt mit dem Kontaktpunkt P koinzidiert
- $\vec{\Omega}_B^0$ = die Winkelgeschwindigkeit der Latschfläche gegenüber der Fahrbahnoberfläche
- \vec{OP}^0 bzw. \vec{OC}^0 = Ortsvektor zum Kontaktpunkt P bzw. zum Radmittelpunkt C

Die dem Reifenmechaniker geläufigen Größen *Schlupf* [Fro27] und *Schräglaufwinkel* sind aus den obengenannten Größen bestimmbar (siehe Abschnitt 5.5.1). Für ebene Fahrbahnen lässt sich der kinematische Zustand mit Methoden der ‘Schulphysik’

vergleichsweise leicht lösen. Im Falle einer gewölbten Fahrbahnoberfläche \mathcal{S} zeigt sich schnell, dass diese auf den ersten Blick anscheinend so einfache Aufgabe nur durch geschicktes Einführen geeigneter Bezugssysteme elegant gelöst werden kann. Reifenkinematik wird so zu einem interessanten Problem der räumlichen Kinematik.

Die nachfolgend beschriebene Methode erlaubt die Berechnung der genannten Größen im Fall gewölbter (siehe Abschnitt 5.1), zweidimensionaler Fahrbahnoberflächen. Ebenso erlaubt das im Folgenden beschriebene Verfahren eine Verallgemeinerung auf $i > 1$ Kontaktpunkte (Bürstenmodelle, ...).

5.2.2 Kräfte und Momente

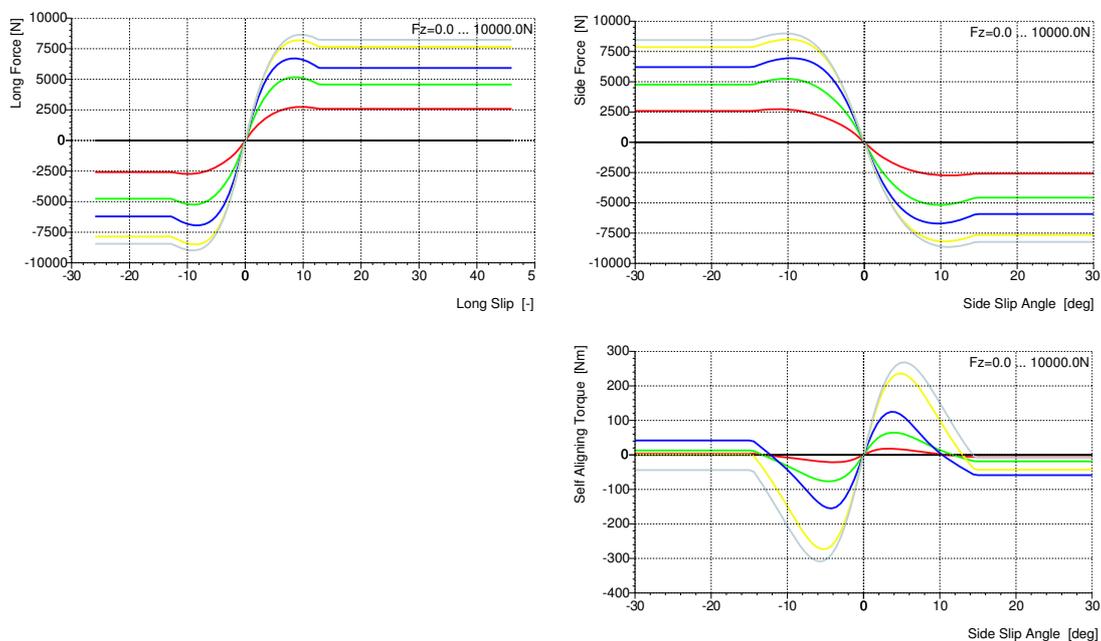


Abbildung 5.2.2: Typische “Reifenkennfelder”

Zur Modellierung statischer und dynamischer Reifeneigenschaften sind umfangreiche Messungen notwendig. Hierfür kommen unterschiedliche Verfahren zum Einsatz, u.a. folgende:

- Prüfstandsmessungen: Der Reifen ist in einer Laborhalle auf eine Kraft/Moment-Messnabe montiert und läuft auf einer sich bewegenden Fahrbahn (Innentrommel, Außentrommel, ebenes Band). Messungen sehr hoher Qualität auf unterschiedlichen Strassenoberflächen können z.B. auf dem Innentrommelprüfstand des Instituts für Fahrzeugtechnik der Universität Karlsruhe durchgeführt werden [Aug02].

- Vermessung mittels Messanhänger: Der Reifen ist auf einem Messanhänger montiert und wird von einem Zugfahrzeug über eine Teststrecke bewegt.
- Vermessung direkt am Fahrzeug: Der Reifen ist auf eine spezielle Messnabe montiert.

Unabhängig vom Messverfahren, werden die Reifenmessungen meist im TYDEX-Format [UZ97] – einem vom internationalen TYDEX-Arbeitskreis standardisierten Format – für die Weiterverarbeitung aufbereitet. Zur Charakterisierung der stationären Eigenschaften eines Reifens werden daraus die folgenden Größen extrahiert:

- Kraft in Abhängigkeit vom “Verspannungszustand” im Latsch in Richtung der Spur: Umfangskraft,
- Kraft und Moment in Abhängigkeit vom Verspannungszustand im Latsch senkrecht zur Spur: Seitenkraft und Rückstellmoment,
- Fahrbahnnormalkraft in Abhängigkeit von der Einfederung.

Folgende Zusammenhänge werden ermittelt:

- Umfangskraft in Abhängigkeit vom Längsschlupf s , Fahrbahnnormalkraft und Sturz,
- Seitenkraft und Rückstellmoment in Abhängigkeit vom Querschlupf bzw. vom Schräglaufwinkel β , Fahrbahnnormalkraft und Sturz.

Diese werden tabellarisch als die sogenannten “Reifenkennfelder” (siehe Abbildung 5.2.2) abgelegt und anschließend während der Simulation über Einfederung bzw. Normalkraft, Sturz, Längs- und Querschlupf abgefragt. Die Diskussion und Parametrierung von Reifenkennfeldern ist nicht Bestandteil der vorliegenden Arbeit. Diskutiert wird vielmehr die Bestimmung des kinematischen Zustandes im Reifen-Fahrbahn-Kontakt.

5.2.3 Dynamik – Seitenwandtorsion und Querelastizität

Heutige Fahrdynamikregelung bedeutet nicht mehr einfach nur die “Lenkfähigkeit beim Bremsen” sicherzustellen wie es zu Zeiten der Einführung von ABS der Fall war. Sie greift vielmehr aktiv während hochdynamischer, sicherheitskritischer Fahrsituationen in das System Fahrer-Fahrzeug ein. Innerhalb der Grenzen, die durch die Physik gezogen sind, versucht die Fahrdynamikregelung Fahrfehler auszugleichen und Unfälle zu vermeiden. Die Fahrdynamikregler greifen aktiv in den Antrieb, die Bremsanlage, die Lenkung und das Fahrwerk ein, auf Basis von Modellen, die mit aktuellen Messgrößen von unterschiedlichsten Sensoren versorgt werden. Zur Verbesserung der Regelung der Brems- und Antriebskräfte laufen z.Z. Versuche, mit “intelligenten Reifen” sich weiter an die physikalischen Grenzen heran zu tasten: Durch direkte Messung des Verspannungszustandes in der Lauffläche und der Seitenwand wird die Kinematik im Latsch präziser erfasst – die Regelung kann verfeinert werden.

Reifenkennfelder beschreiben den statischen bzw. “eingeschwungenen” Betriebszustand, bei dem die Kinematik der Lauffläche direkt aus der der Felge abgeleitet werden kann. Bei einer ABS-Regelung z.B. wird der Reifen ständig an der Kraftschlussgrenze bzw. im Übergangsbereich zwischen Gleiten und Haften betrieben. Durch folgende Modellierungsansätze lässt sich das dynamische Verhalten des Reifenkraftauf- bzw. abbaus beschreiben:

- Seitenwandtorsion: Lauffläche (Latsch \mathcal{F}_B , Belt) und Felge (\mathcal{F}_R , Rim) sind um die Raddrehachse gegeneinander über ein dynamisches System verdrehbar. Ein Raddrehzahlsensor kann nicht die Drehgeschwindigkeit $\overline{\Omega}_B^C$ des Latsches bez. des Radträgers (\mathcal{F}_C , Carrier) messen, sondern misst die Drehgeschwindigkeit $\overline{\Omega}_R^C$ der Felge gegenüber dem Radträger. Bei der Regelung der maximal übertragenen Antriebs- und Bremskräfte wird dies berücksichtigt.
- Querverschiebung und Querdynamik: Lauffläche (Latsch \mathcal{F}_B) und Felge (\mathcal{F}_R) sind in Richtung der Raddrehachse gegeneinander über ein dynamisches System verschieblich. Seitenführungskräfte werden verzögert aufgebaut. Z.B. bei aktiven Lenkeingriffen zur Stabilisierung des Fahrzeugs wird dieser Effekt berücksichtigt.

Häufig werden Seitenwandtorsion und Querverschiebung auch nur als Freiheitsgrad erster Ordnung modelliert.

5.3 Notation und Formelapparat

Die zugrunde liegende allgemeine Kinematik-Notation wurde bereits in Abschnitt 1 eingeführt. Hier wird deshalb ausschließlich auf das darüber Hinausgehende eingegangen.

5.3.1 Differentialgeometrie gekrümmter Flächen

In diesem Abschnitt werden Begriffe der Differentialgeometrie gekrümmter Flächen eingeführt, sofern sie für den Rad-Fahrbahn-Kontakt relevant sind.

\overline{P} bezeichnet den Ortsvektor des Punktes P bez. \mathcal{F}_0 . Ist ein Ortsvektor \overline{P} im Raum \mathbb{R}^3 Funktion von zwei Veränderlichen u und v , so wird durch $\overline{P}(u, v) = \overline{O_0P}(u, v)$ eine gekrümmte, im Folgenden als Fahrbahnoberfläche \mathcal{S} bezeichnete Fläche aufgespannt:

$$\mathcal{S} = \overline{P}(u, v) = \sum_{i=1}^3 P_i(u, v) \overline{e}_i \quad (5.3.1)$$

Die Richtungsvektoren

$$\overline{u}(u, v) = \overline{P}_{,u}(u, v) = \partial \overline{P}(u, v) / \partial u \quad \text{und} \quad (5.3.2)$$

$$\overline{v}(u, v) = \overline{P}_{,v}(u, v) = \partial \overline{P}(u, v) / \partial v \quad (5.3.3)$$

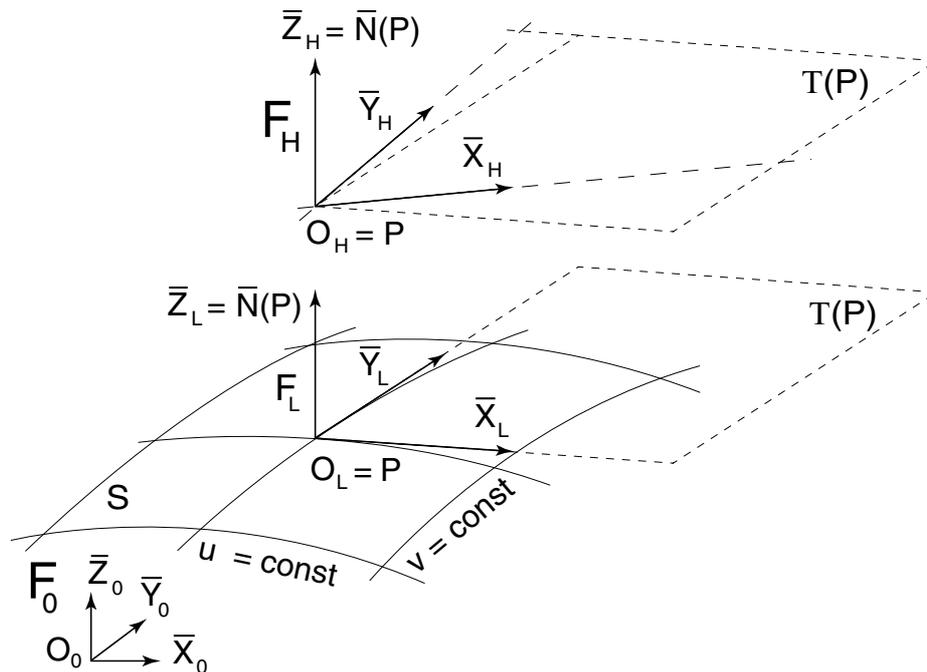


Abbildung 5.3.1: Oberfläche \mathcal{S} , Punkt P , Normalen-Einheitsvektor $\bar{N}(P)$, Tangentialebene $\tau(P)$ und die Koordinatensysteme \mathcal{F}_{LC} und \mathcal{F}_H

bilden nach Abbildung 5.3.1 ein Koordinatennetz und spannen im Punkt $P(u, v)$ die Tangentialebene $\tau(P)$. Ihr Kreuzprodukt beschreibt den Normalen-Einheitsvektor $\bar{N}(P)$ im Punkt P , der senkrecht zur Tangentialfläche $\tau(P)$ steht:

$$\bar{N}(P) = \frac{\bar{u} \times \bar{v}}{\|\bar{u} \times \bar{v}\|} \quad (5.3.4)$$

Die Bezeichnung der Koordinaten u, v sei ohne Beschränkung der Allgemeinheit so gewählt, dass $\bar{N}(P)$ aus der Ebene \mathcal{S} heraus zeigt.

5.3.2 Levi-Civita Bewegung

Das Levi-Civita-Koordinatensystem $\mathcal{F}_{LC} = (O_{LC}, \bar{X}_{LC}, \bar{Y}_{LC}, \bar{Z}_{LC})$ beschreibt relativ zur \mathcal{F}_0 -festen Strassenoberfläche \mathcal{S} eine sogenannte Levi-Civita-Bewegung (L-Bewegung).

Definition: Levi-Civita-Bewegung nach Pfister [Pfi02]

1. Der Ursprung O_{LC} des Systems \mathcal{F}_{LC} koinzidiert mit dem Punkt P , einem Punkt auf der Fahrbahnoberfläche \mathcal{S} , $O_{LC} \equiv P \in \mathcal{S}$,
2. \bar{Z}_{LC} weist in Richtung des Normalenvektors $\bar{N}(O_{LC})$ der Tangentialebene $\tau(O_{LC})$,
3. die Winkelgeschwindigkeit $\bar{\Omega}_{LC}^0$ von \mathcal{F}_{LC} relativ zu \mathcal{F}_0 hat keinen Anteil um $\bar{N}(O_{LC})$, d.h. die Winkelgeschwindigkeit $\bar{\Omega}_{LC}^0$ liegt stets in der Tangentialebene

oder als mathematische Formeln:

$$O_{LC} = P(u_1, v_1) \in \mathcal{S}, \quad \bar{Z}_{LC} = \bar{N}(O_{LC}), \quad \bar{\Omega}_{LC}^0 \cdot \bar{N}(O_{LC}) = 0 \quad (5.3.5)$$

Die beiden ersten “Bindungsgleichungen” sind geometrischer Natur. Die dritte Bindungsgleichung (Zwangsbedingung) ist eine nicht integrierbare (nicht holonome Zwangsbedingung) auf Geschwindigkeitsebene.

Bei Pfister [Pfi02] findet sich die folgende Formulierung auf Geschwindigkeitsebene:

$$\bar{v}_{LC}^0(P) = \bar{v}^0(P) = \bar{P}_u \dot{u} + \bar{P}_v \dot{v} \quad (5.3.6)$$

$$\bar{\Omega}_{LC}^0 = \bar{N} \times \bar{N}_u \dot{u} + \bar{N} \times \bar{N}_v \dot{v} \quad (5.3.7)$$

Die aktuelle Lage des \mathcal{F}_C -Systems ergibt sich durch Integration der kinematischen Gleichungen Gleichung (5.3.6) und Gleichung (5.3.7).

Mit Hilfe des Levi-Civita-Koordinatensystems \mathcal{F}_{LC} lässt sich die allgemeine Roll- und Gleitbewegung von punktkontaktierenden starren Körpern in tangentiale und normale Anteile zerlegen. Das Levi-Civita-Koordinatensystem übernimmt für den Fall gewölbter Fahrbahnen die Rolle, die dem horizontalen, mitbewegten Koordinatensystem für die ebene Fahrbahn zukommt.

Die durch die Gleichung (5.3.6) und Gleichung (5.3.7) eindeutig definierte Bewegung ist eng verwandt mit dem aus der Differentialgeometrie bekannten Begriff “Parallelverschiebung nach Levi-Civita”, siehe [Pfi02]. Die sog. Parallelverschiebung nach Levi-Civita verallgemeinert den Begriff der Parallelität zweier Vektoren auf den Fall gekrümmter “Mannigfaltigkeiten” (Flächen).

5.4 Geometrie-Mapping: Von Radmitte C zum Kontaktpunkt P

5.4.1 Geometrische Begriffe

Spur

Die Gerade, an der die Mittenebene des Rades die Fahrbahn-Tangentialebene τ durchstößt, heißt Spur. Ihr Richtungsvektor ist \bar{X}_H .

Radmittelpunkt C

Das Rad dreht sich um die Drehachse \bar{Y}_C . Auf ihr liegt der Radmittelpunkt C .

Reifen-Fahrbahn-Kontaktpunkt P

Der Kontaktpunkt P liegt auf der Spur und minimiert den Abstand zur Radmitte C .

Einfederung h und Einfederungsgeschwindigkeit \dot{h}

Länge des Vektors \bar{CP} und dessen zeitliche Ableitung.

5.4.2 Bezugssysteme

Die zur exakten Berechnung der Kontaktpunktkinematik auf gewölbten Fahrbahnen notwendigen Punkte, Vektoren und Koordinatensysteme werden in den folgenden Absätzen – sofern nicht schon bereits weiter oben geschehen – eingeführt.

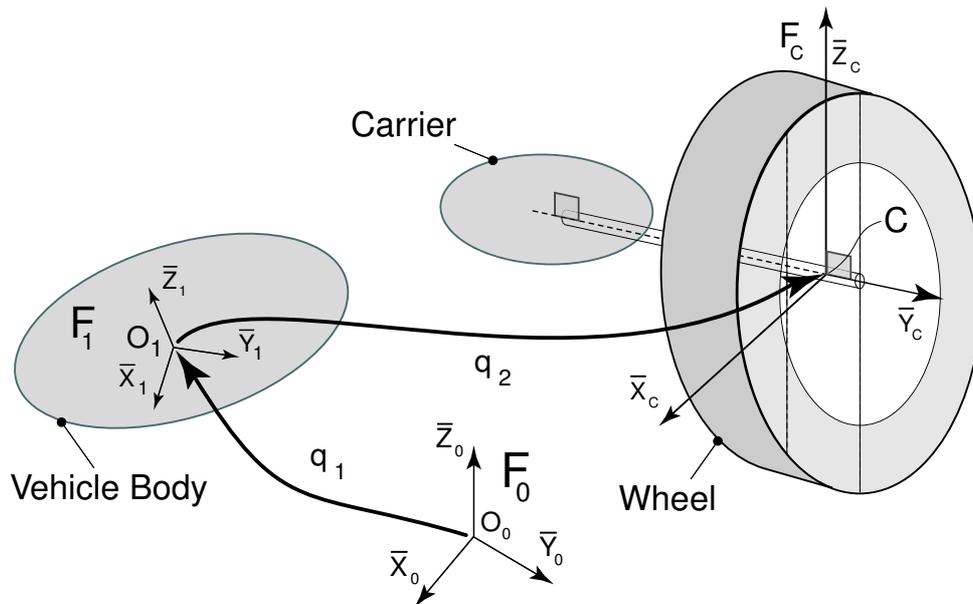


Abbildung 5.4.1: Koordinatensysteme \mathcal{F}_0 , \mathcal{F}_1 und \mathcal{F}_C

Fahrzeug (Vehicle Body) festes Koordinatensystem $\mathcal{F}_1 \equiv (O_1, \bar{X}_1, \bar{Y}_1, \bar{Z}_1)$

Es wird gegenüber dem Inertialsystem \mathcal{F}_0 über 6 Koordinaten $q1[i]$, $i = 0 \dots 5$, drei translatorische $q1[i]$, $i = 0, 1, 2$ und drei rotatorische $q1[i]$, $i = 3, 4, 5$, parametrisiert. Die Lage und Orientierung sowie die Ableitungen $\dot{q}1$ von \mathcal{F}_1 werden typischerweise von einem Mehrkörperformalismus geliefert und können im Folgenden als bekannte Funktionen der Zeit vorausgesetzt werden. Siehe Abbildung 5.4.1.

Radträger-(Carrier)-Koordinatensystem $\mathcal{F}_C \equiv (O_C, \bar{X}_C, \bar{Y}_C, \bar{Z}_C)$

O_C liegt im Radmittelpunkt C ($O_C \equiv C$). Der Vektor \bar{Y}_C zeigt in Richtung der Raddrehachse. \mathcal{F}_C wird gegenüber \mathcal{F}_1 über 6 Koordinaten parametrisiert – meist abhängig von der Einfederung und an der Vorderachse zusätzlich vom Zahnstangenweg oder Lenkwinkel. Die Bewegung des Radträgers \mathcal{F}_C ist im Allgemeinen eine gebundene Bewegung mit den 2 Freiheitsgraden $q2[i]$, $i = 0, 1$ (z.B. “Lenken” und “Einfederung”). Diese beiden Parameter $q2[0]$ und $q2[1]$ (und somit die Lage und Orientierung sowie Ableitungen von \mathcal{F}_C) können im Folgenden als bekannte Funktionen der Zeit vorausgesetzt werden. Siehe Abbildung 5.4.1.

Ferrers-Woronetz-Koordinatensystem $\mathcal{F}_{FW} \equiv (O_{FW}, \bar{X}_{FW}, \bar{Y}_{FW}, \bar{Z}_{FW})$

O_{FW} liegt im Radmittelpunkt C ($O_{FW} = O_C$). Die \bar{Y}_{FW} -Achse zeigt in Richtung der Radachse \bar{Y}_C . Die Projektion der absoluten Winkelgeschwindigkeit $\bar{\Omega}_{FW}^0$ auf die Radachse \bar{Y}_C ist Null. Dies ist eine nicht-holonome Bindung! Die Orientierung von \mathcal{F}_{FW} wird gegenüber \mathcal{F}_C durch den Winkel θ um $\bar{Y}_{FW} = \bar{Y}_C$ im Gelenkpunkt C festgelegt.

$$\begin{aligned} O_{FW} &:= C, \\ \bar{Y}_{FW} &:= \bar{Y}_C, \quad \bar{\Omega}_{FW}^0 \cdot \bar{Y}_C := 0, \quad \theta = -\angle(\bar{X}_{FW}, \bar{X}_C) \end{aligned} \quad (5.4.1)$$

Bemerkung: Das Ferrers-Woronetz Koordinatensystem \mathcal{F}_{FW} erhält seine Bedeutung durch die Möglichkeit der Verknüpfung der Winkelgeschwindigkeit $\bar{\Omega}_{LC}^0$ der Fahrbooberfläche S , der Winkelgeschwindigkeit $\bar{\Omega}_R^0$ der Felge und Winkelgeschwindigkeit $\bar{\Omega}_C^0$ des Radträgers. Anschaulich ausgedrückt ist das \mathcal{F}_{FW} -Koordinatensystem “ein in der Radebene ruhendes Bezugssystem”. Relativ zu diesem System wird die Rotation der anderen Koordinatensysteme, deren Y -Achse mit \bar{Y}_C zusammenfällt, beschrieben.

Die Bezeichnung “Ferrers-Woronetz”-Koordinatensystem geschieht zu Ehren des Mathematikers Ferrers und des Mechanikers Woronetz. Ferrers war der Erste, der mit Methoden der Analytischen Mechanik das Problem der rollenden Kreisscheibe untersuchte [1870]. Woronetz war Wegbereiter Hamels. Er verallgemeinerte die Arbeiten Ferrers auf den Fall beliebiger Körper mit gewölbter Fläche [1901]. Die Arbeiten von Ferrers und Woronetz sind ein gutes Beispiel dafür, wie bedingt durch Fragestellungen aus der Praxis (“rollende Kreisscheibe”) neue theoretische Erkenntnisse und Methoden entwickelt werden. Eine ausführliche Darstellung der Geschichte non-holonomer mechanischer Systeme und der Arbeiten von Ferrers und Woronetz findet sich in [Pap02].

Levi-Civita-Koordinatensystem $\mathcal{F}_{LC} \equiv (O_{LC}, \bar{X}_{LC}, \bar{Y}_{LC}, \bar{Z}_{LC})$

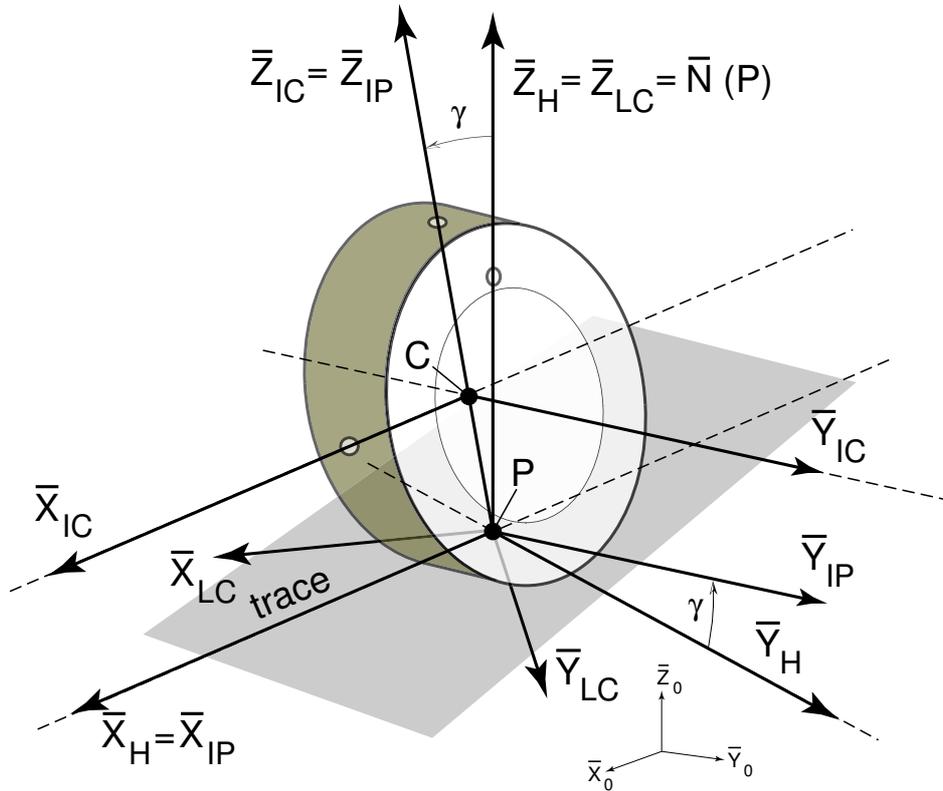
Sein Ursprung O_{LC} ist der Reifen-Fahrbahn-Kontaktpunkt P ($O_{LC} \equiv P, O_{LC} \in S$). Seine Orientierung (genauer: seine Rotation um die \bar{Z}_{LC} -Achse) braucht für die Bedürfnisse der vorliegenden Arbeit nicht näher parametrisiert werden. Sie muss einzig der Bedingung genügen, sich nicht um $\bar{N}(P)$ zu drehen. Siehe hierzu auch Abschnitt 5.3.2.

$$\begin{aligned} O_{LC} &:= P \in S, \quad \bar{N}(P) = \bar{Z}_{LC}, \\ \bar{\Omega}_{LC}^0 \cdot \bar{N}(P) &= 0 \end{aligned} \quad (5.4.2)$$

Die Bezeichnung L -Koordinatensystem und die Definition dieser Bewegung stammt von Pfister [Pfi02].

Horizontiertes Koordinatensystem $\mathcal{F}_H \equiv (O_H, \bar{X}_H, \bar{Y}_H, \bar{Z}_H)$

Sein Ursprung O_H ist der Kontaktpunkt P . \bar{Z}_H zeigt in Richtung des Normalenvektors $\bar{N}(P)$. \bar{X}_H zeigt in Richtung der Spurgeraden der Radebene und der Fahrbooberflächentangentialebene τ . Das Koordinatensystem \mathcal{F}_H kann durch die Drehung $\psi = \angle(\bar{X}_{LC}, \bar{X}_H)$ um $\bar{Z}_{LC} = \bar{N}(P)$ in \mathcal{F}_{LC} überführt werden. Der Winkel ψ ist für die vorliegende Arbeit ohne

Abbildung 5.4.2: Koordinatensysteme \mathcal{F}_{LC} , \mathcal{F}_H , \mathcal{F}_{IP} , \mathcal{F}_{IC}

Bedeutung.

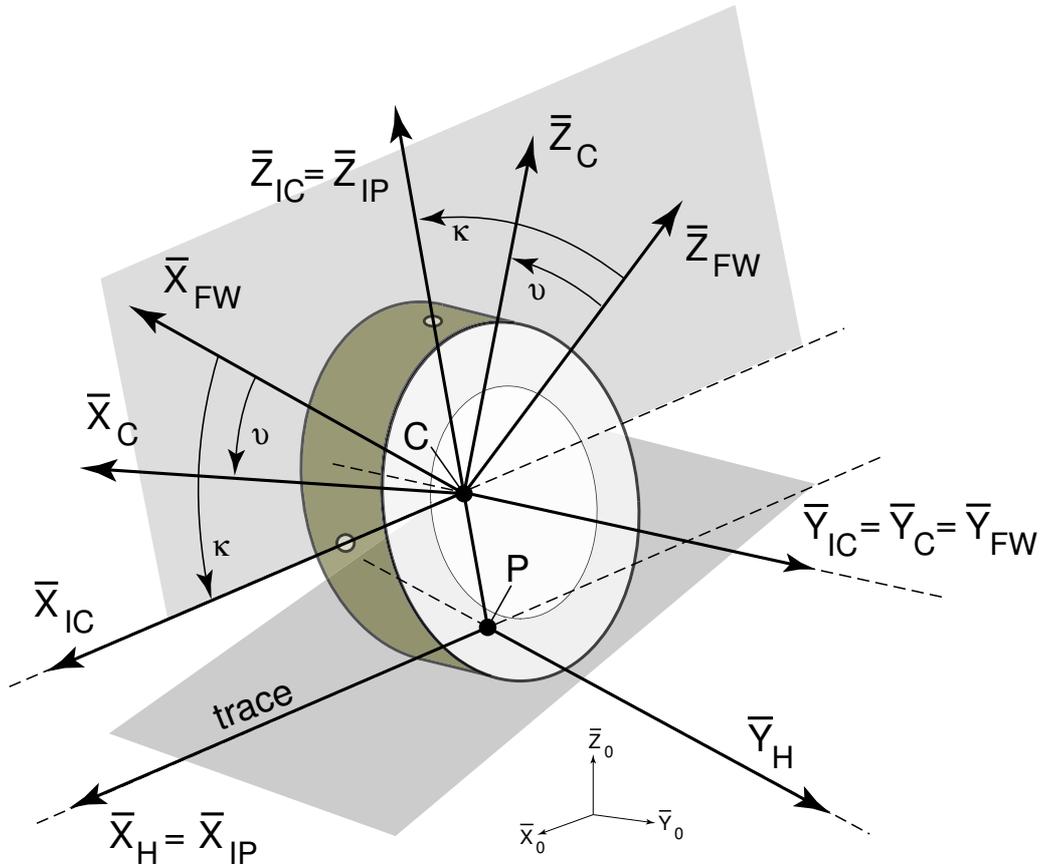
$$\begin{aligned} O_H &:= O_{LC} := P \in S, \\ \overline{X}_H &\parallel \text{Spur}, \quad \overline{Z}_H := \overline{N}(P), \quad \psi = \angle(\overline{X}_{LC}, \overline{X}_H) \end{aligned} \quad (5.4.3)$$

Horizontiertes, verschobenes Koordinatensystem $\mathcal{F}_{HS} \equiv (O_{HS}, \overline{X}_{HS}, \overline{Y}_{HS}, \overline{Z}_{HS})$
 Das Koordinatensystem \mathcal{F}_{HS} ist parallel zu \mathcal{F}_H orientiert ($\overline{X}_{HS} = \overline{X}_H$, $\overline{Y}_{HS} = \overline{Y}_H$, $\overline{Z}_{HS} = \overline{Z}_H$). Durch die Translation $d = \overline{PP}_S \cdot \overline{Y}_H$ in \overline{Y}_H -Richtung geht \mathcal{F}_{HS} aus \mathcal{F}_H hervor. Die Translation d dient dazu, eine seitliche Auslenkung des Latsches aus der Radebene aufgrund der Elastizität der Reifen-Seitenwand zu berücksichtigen. Siehe Abbildung 5.4.4.

$$\begin{aligned} O_{HS} &\equiv P_S = P + \delta \overline{Y}_H, \\ \mathcal{F}_{HS} &\parallel \mathcal{F}_H, \quad d = \overline{PP}_S \cdot \overline{Y}_H \end{aligned} \quad (5.4.4)$$

Gekipptes Koordinatensystem $\mathcal{F}_{IP} \equiv (O_{IP}, \overline{X}_{IP}, \overline{Y}_{IP}, \overline{Z}_{IP})$ in P

Sein Ursprung O_{IP} ist der Kontaktpunkt P . Seine X -Achse ist die Spur \overline{X}_H , Y -Achse ist

Abbildung 5.4.3: Koordinatensysteme $\mathcal{F}_C, \mathcal{F}_{FW}, \mathcal{F}_{IC}$

die Radachse \bar{Y}_C . Eine Drehung um die Spur \bar{X}_H mit dem Sturzwinkel $\gamma = \angle(\bar{Y}_H, \bar{Y}_C)$ überführt \mathcal{F}_H in \mathcal{F}_{IP} .

$$\begin{aligned} O_{IP} \equiv O_H \equiv O_{LC} \equiv P \in S, \\ \bar{X}_{IP} = \bar{X}_H, \quad \bar{Y}_{IP} = \bar{Y}_C, \quad \gamma = \angle(\bar{Y}_H, \bar{Y}_{IP} = \bar{Y}_C) \end{aligned} \quad (5.4.5)$$

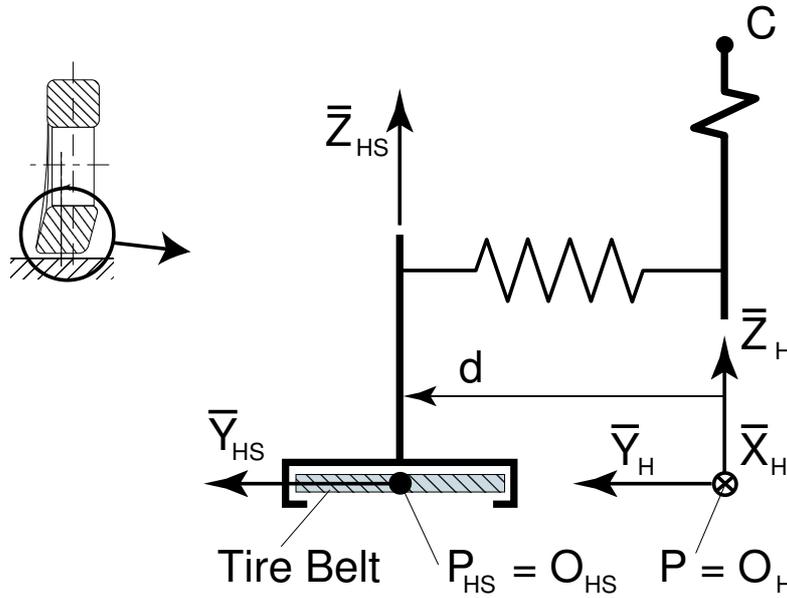
Gekipptes Koordinatensystem $\mathcal{F}_{IC} \equiv (O_{IC}, \bar{X}_{IC}, \bar{Y}_{IC}, \bar{Z}_{IC})$ in C

Sein Ursprung O_{IC} ist der Radmittelpunkt C . Seine Y -Achse ist die Radachse \bar{Y}_C . Aus Koordinatensystem \mathcal{F}_{IP} geht \mathcal{F}_{IC} durch die Translation $h = \overline{PC} \cdot \bar{Z}_{IC}$ in Richtung \bar{Z}_{IC} hervor. Gegenüber Koordinatensystem \mathcal{F}_{FW} ist die Orientierung von \mathcal{F}_{IC} durch den Drehwinkel $\kappa = \angle(\bar{X}_{FW}, \bar{X}_H)$ um die Radachse $\bar{Y}_{FW} \equiv \bar{Y}_C$ festgelegt.

$$\begin{aligned} O_{IC} \equiv O_{FW} \equiv C, \quad \bar{Y}_{IC} \equiv \bar{Y}_{IP} \equiv \bar{Y}_C, \\ \mathcal{F}_{IC} \parallel \mathcal{F}_{IP}, \quad h = \overline{PC} \cdot \bar{Z}_{IC}, \quad \kappa = \angle(\bar{X}_{FW}, \bar{X}_H) \end{aligned} \quad (5.4.6)$$

Felgen-Koordinatensystem $\mathcal{F}_R \equiv (O_R, \bar{X}_R, \bar{Y}_R, \bar{Z}_R)$ in C

Sein Ursprung O_R ist der Radmittelpunkt C . Gegenüber Koordinatensystem \mathcal{F}_{FW} ist die

Abbildung 5.4.4: Koordinatensystem \mathcal{F}_{HS} , Querverschiebung d

Orientierung von \mathcal{F}_R durch den Drehwinkel $\phi_R = \sphericalangle(\bar{X}_{FW}, \bar{X}_R)$ um die Radachse \bar{Y}_C festgelegt.

$$\begin{aligned} O_R &\equiv O_W \equiv C, \\ \bar{Y}_R &\equiv \bar{Y}_{FW} \equiv \bar{Y}_C, \quad \phi_R = \sphericalangle(\bar{X}_{FW}, \bar{X}_R) \end{aligned} \quad (5.4.7)$$

Latsch-Koordinatensystem $\mathcal{F}_B \equiv (O_B, \bar{X}_B, \bar{Y}_B, \bar{Z}_B)$ in C

Sein Ursprung O_B ist der Radmittelpunkt C . Gegenüber Koordinatensystem \mathcal{F}_R ist die Orientierung von \mathcal{F}_B durch den Drehwinkel $\phi_B = \sphericalangle(\bar{X}_R, \bar{X}_B)$ um die Radachse \bar{Y}_C festgelegt.

Der Winkel ϕ_B modelliert die Verdrehung der Reifen-Lauffläche gegenüber der Felge um die Raddrehachse. Diese Verdrehung ist auch unter dem Begriff "Seitenwandtorsion" bekannt. Siehe Abbildung 5.4.5.

$$\begin{aligned} O_B &\equiv O_R \equiv O_W \equiv C, \\ \bar{Y}_B &\equiv \bar{Y}_R \equiv \bar{Y}_{FW} \equiv \bar{Y}_C, \quad \phi_B = \sphericalangle(\bar{X}_R, \bar{X}_B) \end{aligned} \quad (5.4.8)$$

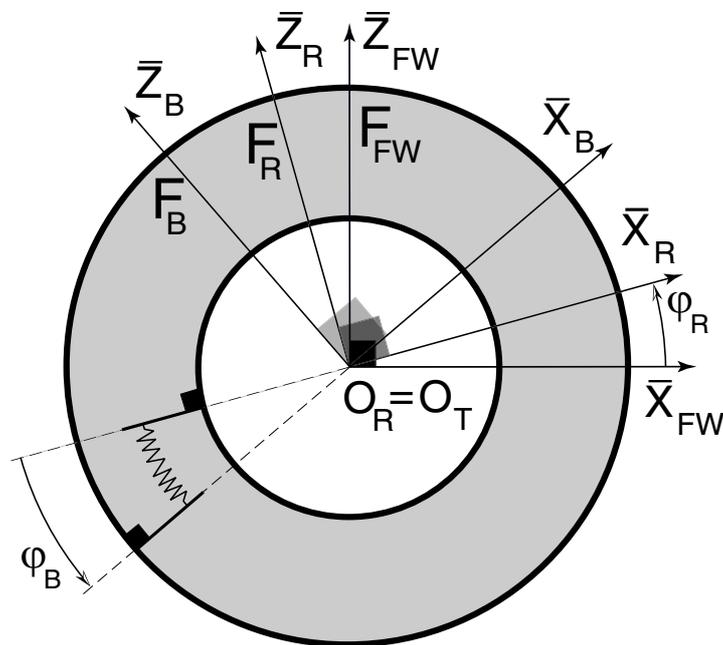


Abbildung 5.4.5: Koordinatensysteme: Seitenwandtorsion, \mathcal{F}_B , \mathcal{F}_R , \mathcal{F}_{FW}

5.4.3 Karlsruher Kinematische Kette – K3

Die Berechnung der in Abschnitt 5.5 eingeführten Reifenkinematischen Referenzgrößen Längsschlupf s , Querschlupf s_{\perp} und Schräglaufwinkel β im allgemeinen Fall muss als Problem einer geschlossenen kinematischen Kette verstanden werden. Bei der Bestimmung der Größen dieser kinematischen Kette, im Folgenden mit ‘‘Karlsruher Kinematische Kette K3’’ bezeichnet, wird auf numerische Differentiationen weitestgehend verzichtet. Differentiationen werden nur an den Stellen vorgenommen, an denen sie unvermeidlich sind. An allen anderen Stellen werden die unbekannt kinematischen Größen durch Projektionen ermittelt.

Die Karlsruher Kinematische Kette und die Beziehung der verschiedenen Kettenglieder untereinander ist in Abbildung 5.4.6 skizziert.

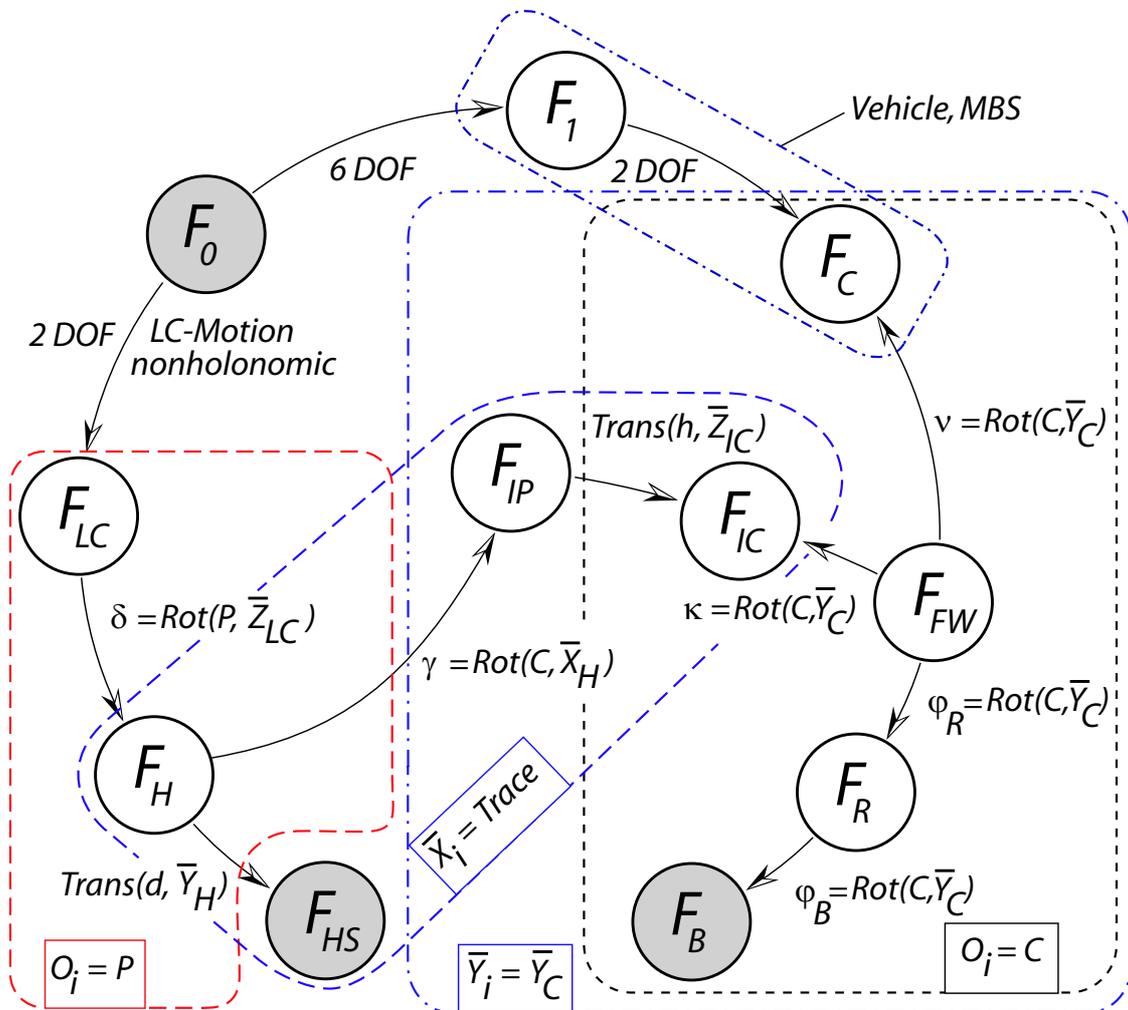


Abbildung 5.4.6: Karlsruher Kinematische Kette K3: Koordinatensysteme und Relativbewegungen

5.5 Reifenkinematische Referenzgrößen

In der Simulation gilt es, die aktuelle Situation über die Referenzgrößen Schlupf und Schräglaufwinkel zu bewerten, und so die zugehörige Messsituation und damit auch die entsprechende Reifenreaktion (Messwerte) zuzuordnen. Nachfolgend werden diese Referenzgrößen exakt und allgemein in vektorieller Schreibweise definiert.

5.5.1 Längsschlupf s , Querschlupf s_{\perp} und Schräglaufwinkel β

Der Längsschlupf s wird definiert als das Verhältnis zweier in Richtung der Spur \overline{X}_H projizierter Geschwindigkeiten relativ zum Koordinatensystem \mathcal{F}_0 :

$$s \stackrel{\text{Def.}}{=} -\frac{\overline{v}_B^0(P) \cdot \overline{X}_H}{\overline{v}^0(P) \cdot \overline{X}_H} \quad (5.5.1)$$

In Worten: Der Längsschlupf s wird definiert als das Verhältnis der Geschwindigkeit des zum Beobachtungszeitpunkt mit dem Kontaktpunkt P koinzidierenden Reifen-/Stollenteilchens und der Fortschrittggeschwindigkeit des Kontaktpunktes P , projiziert in Richtung der Spur \overline{X}_H (siehe Abbildung 5.5.1).

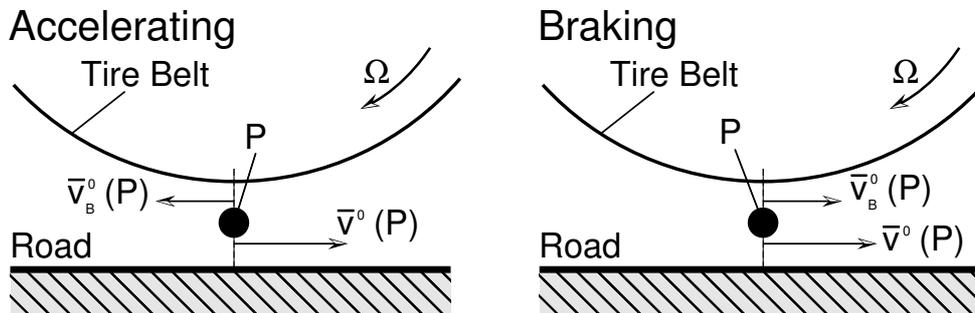


Abbildung 5.5.1: Längsschlupf für Antreiben und Bremsen

Diese Definition des Schlupfes stützt sich ausschließlich auf Kinematikgrößen im Kontaktpunkt P . Die in der Literatur im Zusammenhang mit der Definition von Längsschlupf häufig anzutreffenden Größen wie ‐Drehgeschwindigkeit Ω ‐, Reifenradius r und $\overline{\Omega} \times \overline{r}$ beinhalten Hypothesen über die Modellierung des Reifens. Zu einer allgemeinen Definition von Längsschlupf s sollten sie deshalb nicht herangezogen werden. Das in Abbildung 5.5.2 skizzierte Modell soll helfen, sich von dieser Denkweise zu lösen und sich auf die Kinematik im Kontaktpunkt P zu konzentrieren.

Analog zum Längsschlupf s wird der Querschlupf s_{\perp} eingeführt:

$$s_{\perp} = -\frac{\overline{v}_B^0(P) \cdot \overline{Y}_H}{\overline{v}^0(P) \cdot \overline{X}_H} \quad (5.5.2)$$

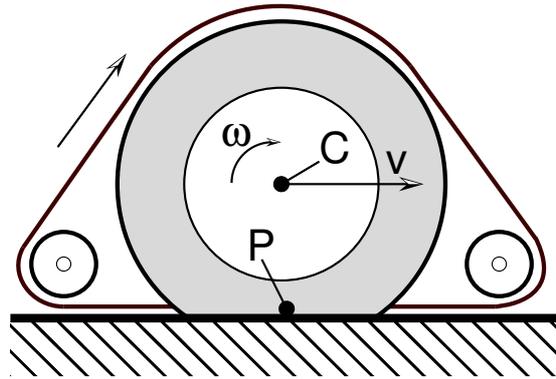


Abbildung 5.5.2: Das rollende Rad als “Bandmechanismus” [Amm97]

Der Schräglaufwinkel β wird definiert als

$$\beta = \sphericalangle(\bar{v}^0(P), \bar{X}_H) \quad (5.5.3)$$

und kann über die Arcus-Tangens-Funktion berechnet werden: $\beta = \arctan(s_{\perp})$. Begründung: $\bar{v}_B^0(P) \cdot \bar{Y}_H = \bar{v}^0(P) \cdot \bar{Y}_H$. Hierbei ist zu beachten, dass Winkel β außerhalb der Quadranten *I* und *IV* ($-\pi/2 < \beta < +\pi/2$) gesondert zu berücksichtigen sind. Programmiertechnisch geschieht dies typischerweise durch Verwenden der Funktion $\text{atan2}(y, x)$ statt der Funktion $\text{atan}(y/x)$.

5.5.2 Bemerkungen

Eine häufig anzutreffende Berechnungsformel für die “Relativgeschwindigkeit” im Latsch ist:

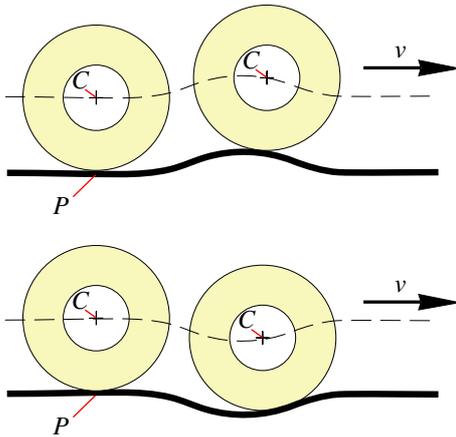
$$v_{\text{slip}} = v_C - \omega r \quad (5.5.4)$$

Hierbei sind die Größen v_s , v_C , ω und r meist gar nicht oder nicht exakt definiert. \bar{v}_C wird als “Fahrzeuggeschwindigkeit” oder “translatorische Radgeschwindigkeit” bezeichnet; $\omega \cdot r$ wird als Umfangsgeschwindigkeit “des Rades” bezeichnet. Hierbei bleibt unklar, ob ω die auf die Drehachse des Rades projizierte Winkelgeschwindigkeit Rad/Fahrbahn oder die entsprechende Projektion der relativen Winkelgeschwindigkeit Radträger/Rad darstellt. Für den allgemeinen Fall gewölbter Fahrbahnen wird die “Begrenztheit” dieser schulphysikalischen Ansätze deutlich.

Ebenso wird die Größe v_C nicht exakt definiert: Auf welche Achse wurde welcher Vektor projiziert, um diese skalare Geschwindigkeitsgröße zu erhalten? Wird hier auf eine “radträgerfeste” Achse projiziert? Was gilt, wenn sich der Bewegung des Radträgermittelpunktes eine Sturz- und/oder Vorspurgeschwindigkeit überlagert?

Nicht selten wird der Antriebsstrang getrennt vom Fahrzeugaufbau, dem sogenannten “Chassis”, modelliert. Bei der Kopplung der beiden Modelle wird die Drehung der Antriebswelle gegenüber dem “Chassis” gleichgesetzt mit der Drehung gegenüber dem Radträger, und diese wiederum wird in obiger Formel unbesehen übernommen.

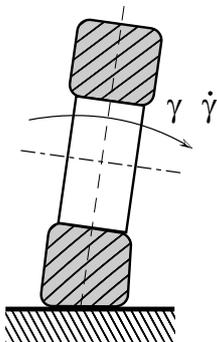
Typische, weit verbreitete Modellierungsvereinfachungen sind im Folgenden zusammengestellt.



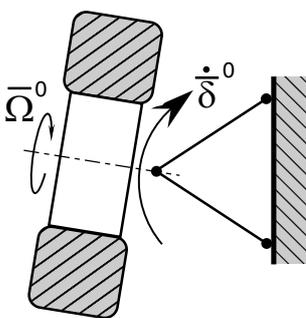
Während der Fahrt über eine konvexe oder konkave Fahrbahnoberfläche (Schwelle oder Senke) eilt der Kontaktpunkt P dem Radmittelpunkt C voraus oder hinterher. Die Winkelgeschwindigkeit der Tangentialebene für $\bar{v}_B^0(P)$ ist von Belang.

Geschwindigkeit im Radmittelpunkt C ist ungleich der Geschwindigkeit im Kontaktpunkt P ($\bar{v}^0(C) \neq \bar{v}^0(P)$).

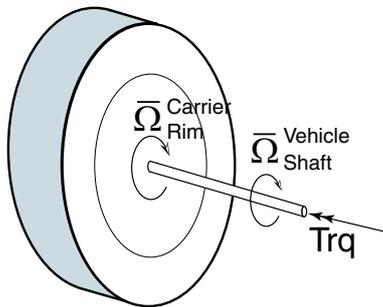
(Fehlerabschätzung: Bei Überfahrt eines Reifens mit Radius $r = 0.3m$ über eine Schwelle mit Radius $r_1 = 3m$ bewegt sich der Radmittelpunkt C entsprechend des Radienverhältnisses r/r_1 um $\approx 10\%$ schneller als die Fortschreitgeschwindigkeit des Kontaktpunktes P .



Die Sturzgeschwindigkeit $\dot{\gamma}$ kann von Null verschieden sein. Sie liefert einen Beitrag zur Quergeschwindigkeit des Kontaktpunktes P und somit zur Seitenkraft.



Die Lenkgeschwindigkeit $\dot{\delta} \neq 0$ verändert die kinematischen Grundgrößen im Kontaktpunkt P , da der Durchstoßpunkt der sog. Lenkachse im Allgemeinen nicht mit dem Radaufstandspunkt P übereinstimmt.

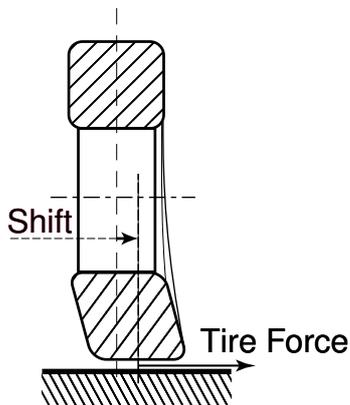


Raddrehzahlsensoren “messen” die Relativgeschwindigkeit der Felge um die Radachse relativ zum Radträger und nicht zum Aufbau oder zu absolut.

Die Drehgeschwindigkeit $\bar{\Omega}_R^V$ der Antriebswelle gegenüber dem Fahrzeugaufbau unterscheidet sich von der Drehgeschwindigkeit $\bar{\Omega}_R^C$ gegenüber dem Radträger ($\bar{\Omega}_R^V \cdot \bar{Y}_C \neq \bar{\Omega}_R^C \cdot \bar{Y}_C$).

Die Raddrehzahl ist ungleich der Antriebswellendrehzahl ($\bar{\Omega}_R^0 \cdot \bar{Y}_C \neq \bar{\Omega}_R^C \cdot \bar{Y}_C$).

Folgende Drehgeschwindigkeiten werden unterschieden: Drehgeschwindigkeit am Ausgang des Differentials gegenüber dem Differentialgehäuse, d.h. gegenüber dem Fahrzeugaufbau, und Drehgeschwindigkeit der Felge des Rades gegenüber dem Radträger.



Kontaktpunkt P befindet sich nicht immer in der Radebene. Die Querverschiebung d des Latsches gegenüber dem Radmittelpunkt C beeinflusst die Hebelverhältnisse für Seiten- und Umfangskraft und damit das Rückstellmoment des Reifens.

5.6 Organisation des Auswertungsablaufs

Gegeben

Folgende Größen liegen der Algebraisierung der K3 zugrunde:

\mathcal{F}_1	Fahrzeug/Vehicle: Position, Orientierung und Geschwindigkeit	MKS-Formalismus
\mathcal{F}_C	Radträger/Carrier: Position, Orientierung und Geschwindigkeit	MKS-Formalismus
\mathcal{S}	Fahrbahnoberfläche, $z(x, y)$, Fahrbahnnormale $\bar{N}(x, y)$	Fahrbahndefinition
$\bar{C}\bar{P}$	Vektor Radmitte-Fahrbahnoberfläche, Einfederung h	Reifenmodell
$\bar{\Omega}_{LC}^0$	Winkelgeschwindigkeit von \mathcal{F}_{LC}	Gleichung (5.3.7)
$\bar{\Omega}_R^C$	Felge/Rim: Drehgeschwindigkeit der Felge gegenüber dem Radträger	Antriebsstrang

Gesucht

Die Größen $\bar{v}^0(P)$, $\bar{v}_B^0(P)$ bzw. $\bar{v}^B(P)$

Lösungsweg

Ermittlung des Kontaktpunktes P

Der Kontaktpunkt P ist definiert als derjenige fahrbahnfeste Punkt, der den Abstand zur Radmitte C minimiert (vgl. Abschnitt 5.4.1). Aufgrund der getroffenen Annahme einer sog. gewölbten Fahrbahn ist er bei gegebenem C und \mathcal{S} stets eindeutig definiert.

Bei ebenen Fahrbahnen existiert folgende explizite Lösung zur Bestimmung von P :

$$\bar{P} = \bar{C} + \bar{C}\bar{R} + \alpha \bar{X}_H + \beta \bar{Y}_H, \quad (5.6.1)$$

wobei der Punkt $R \in \mathcal{S}$ ein beliebiger Punkt der Fahrbahnoberfläche ist, sich die Vektoren $\bar{X}_H = \bar{Y}_C \times \bar{N}$ und $\bar{Y}_H = \bar{N} \times \bar{X}_H$ und die Faktoren $\alpha = -\bar{C}\bar{R} \cdot \bar{X}_H$ und $\beta = -\frac{\bar{C}\bar{R} \cdot \bar{Y}_C}{\bar{Y}_H \cdot \bar{Y}_C}$ berechnen.

Im Falle einer gewölbten Fahrbahn muss \bar{P} iterativ berechnet werden, da der Fahrbahnnormalenvektor $\bar{N} = \bar{N}(P)$ vom Punkt P abhängt. Der rechenaufwändigste Teil dieses iterativen Prozesses ist der Aufruf des Fahrbahnmodells $z = f(x, y)$ mit Schätzwerten $P^*(x, y)$. Der Bestimmung von Schätzwerten $P^*(x, y)$ kommt daher eine erhebliche Bedeutung zu. Sie wird hier nicht im Detail besprochen. Vielmehr genüge es darauf hinzuweisen, dass hierfür folgende Größen zur Verfügung stehen:

- aktueller Radmittelpunkt \bar{C} ,

- Vektor \overline{CP} vom letzten Zeitschritt,
- aktuelle Radträgerbewegung und Drehung um die Spin-Achse \overline{Y}_C ,
- Einfederung h und Einfederungsgeschwindigkeit \dot{h} vom letzten Zeitschritt,
- Krümmungsverlauf der Fahrbahnoberfläche \mathcal{S} .

Bei Berücksichtigung dieser Größen lässt sich der Kontaktpunkt \overline{P} für typische Fahrsituationen bei einer akzeptierten Toleranz von $\|\overline{P^*P}\| = 1.0 \cdot 10^{-5}m$ meist ohne bzw. mit nur einer Iteration ermitteln.

Somit stehen die geometrischen Größen $C, \overline{Y}_C, P, N(P), \overline{X}_H = \text{Spur}$ zur Verfügung.

Berechnen der Koordinatensysteme und ihrer Relativbeziehungen

Im Folgenden werden die Koordinatensysteme der K3 (siehe Abschnitt 5.4.3) erzeugt und ihre Relativbeziehungen ausgewertet:

1. Im Kontaktpunkt P werden die Koordinatensysteme $\mathcal{F}_{LC}, \mathcal{F}_H$ und \mathcal{F}_{IP} erzeugt.
2. \mathcal{F}_{IC} geht aus \mathcal{F}_{IP} durch die Translation $-\overline{CP}$ von P nach C hervor.
3. Aus der Kinematik des Radträgers \mathcal{F}_C wird im Radmittelpunkt C das Koordinatensystem \mathcal{F}_{FW} abgeleitet.
4. Das Reifenmodell liefert den Verdrehwinkel ϕ_R des Koordinatensystems der Felge \mathcal{F}_R gegenüber \mathcal{F}_{FW} und – bei Modellierung einer Seitenwandtorsion – den Verdrehwinkel ϕ_B der Lauffläche \mathcal{F}_B gegenüber \mathcal{F}_R und deren Ableitungen $\dot{\phi}_R$ und $\dot{\phi}_B$.
5. Berücksichtigt das Reifenmodell eine Querverschiebung des Latsches, geht aus dem Koordinatensystem \mathcal{F}_H durch Translation d in Richtung \overline{Y}_H das Koordinatensystem \mathcal{F}_{HS} hervor.
6. Die Seitenwandtorsion kann fast ohne Zusatzaufwand berücksichtigt werden. Für Reifenmodelle mit Querelastizität, d.h. wenn das Koordinatensystem \mathcal{F}_{HS} von \mathcal{F}_H verschieden ist, entsteht zusätzlicher Aufwand für die Transformation der für den Kontaktpunkt P und das Koordinatensystem \mathcal{F}_H hergeleiteten Kinematik in den verschobenen Punkt P_S und das Koordinatensystem \mathcal{F}_{HS} .

Die Algebraisierung der K3 liefert die algebraischen Ausdrücke der gesuchten Terme, aus denen die reifenkinematischen Referenzgrößen Schlupf und Schräglaufwinkel berechnet werden.

Es ist bemerkenswert, dass nur die Winkelgeschwindigkeit des L -Koordinatensystems durch numerische Differentiation bestimmt werden muss, während die Geschwindigkeit $\overline{v}^0(O_{LC})$ differentiationsfrei durch vektorielle Projektion gewonnen werden kann.

Kapitel 6

Zusammenfassung und Ausblick

Der Forderung nach neuen, verbesserten und zuverlässig arbeitenden Fahrdynamik-Regelsystemen lässt sich nur durch angepasste Entwicklungsprozesse und geeignete Entwicklungswerkzeuge nachkommen. Die Analyse und Optimierung des einzelnen Regelsystems kann erfolgreich nur in einer Closed-Loop-Entwicklungsumgebung im Kontext des Gesamtsystems mit allen beteiligten Systemkomponenten erfolgen. Die vorliegende Arbeit stellt hierfür im ersten Teil (Kapitel 3) eine Entwicklungsumgebung vor, die den gesamten Entwicklungsprozess von der Konzeptphase bis hin zur Erprobung von Seriensteuergeräten unterstützt. Sie ist in allen Bereichen modular strukturiert und auf die Entwicklung von Regelsystemen für die Fahrzeuge Pkw und Motorrad ausgerichtet. Sie ist auf PCs unter Windows und Linux und in einer Hardware-in-the-Loop-Version unter dem Echtzeit-Betriebssystem LynxOS verfügbar. Als Echtzeit-Hardware wird auf Industrie-Standards wie VMEbus, Prozessor-Platinen mit Motorola-PowerPCs und Input/Output -Karten nach dem Common Mezzanine Card Standard zurückgegriffen. Auf Software-Seite kommen Standards wie ANSI-C, die Scriptsprache Tcl/Tk und TCP/UDP Netzwerkprotokolle zum Einsatz. In einer Matlab/Simulink-Version und einer Standalone-Offline-Version steht sie dem Entwickler zur Verfügung, um Regelkonzepte zu entwickeln und Regelalgorithmen zu optimieren. Die Erprobung von Hardware-Steuergeräten mit elektrischem Signal- und Aktuatorinterface erfolgt anschließend mit der Hardware-in-the-Loop-Version. Die Forderung nach intuitiver Bedienbarkeit der Simulationsumgebung wurde durch ein an der Versuchspraxis angelehntes Bedienkonzept berücksichtigt.

Durchgängig während der verschiedenen Entwicklungsphasen kommen die gleichen, modular strukturierten Simulationsmodelle zum Einsatz. Die Fahrzeugmodelle für Pkw, Pkw-Anhänger und Motorrad sind intern identisch strukturiert und verwenden einheitliche Schemata für Parameter, Beobachtersignale und Schnittstellengrößen. Die Parametrierung erfolgt bauteil- bzw. baugruppenorientiert. Die Modelle verfügen über eine Vielzahl von Schnittstellen zur projektspezifischen Anpassung. Vorhandene Modelle können erweitert oder durch eigene, neue Modelle ersetzt werden. Der beschrittene Weg richtet sich jeweils nach den Anforderungen der zu entwickelnden Regelsysteme. Als Untermodule vorhanden sind u.a. die Fahrwerkskinematik, die Fahrwerks-Federn und -Dämpfer, das Lenksystem, die Bremsanlage, der Motor, die Kupplung, das Getriebe, der Triebstrang, der Antriebsstrang als übergeordnete Einheit, die Reifen und die Aerodynamik. Die

Parameter für Modelle, Versuche, Testszenarien und die Signalkonditionierung sind in einer Datenbank abgelegt. Zur prozesssicheren Parametrierung greifen alle Versionen auf die gleiche Parameter-Datenbank zu.

Nachdem der erste Teil der Arbeit auf die Modularisierung und die Schnittstellen der Fahrzeugmodelle aus Anwendersicht eingeht, steht im Mittelpunkt des zweiten Teils (Kapitel 4) die Entwicklung der Dynamikgleichungen der zugrunde liegenden Fahrzeugmodelle. Ziel sind kompakte und schnelle Gleichungen für das eigentliche Fahrzeugmodell, die speziell auf dem Hardware-in-the-Loop-Simulator ausreichend Ressourcen für notwendige Aktuator-, Sensor- oder weitere Streckenmodelle frei lassen. Alle Fahrzeugvarianten sind topologisch einheitlich strukturiert. Die externe Bewegung des generalisierten Fahrzeug-Systems im Raum erfolgt ohne Singularitäten über Euler-Parameter (Quaternionen). Die Kinematik innerhalb des Fahrzeugs wird bez. eines mitbewegten Systems formuliert. Durch diese Koordinatenwahl bleibt die Systemmatrix (Massenmatrix) stets frei von der Position und Orientierung des mitbewegten Bezugssystems. Das Fahrwerk wird aus einzelnen Achsen und diese wiederum aus einer linken und einer rechten Radaufhängung aufgebaut. Beim Motorrad entfällt die rechte Radaufhängung. Die Radaufhängung wird jeweils über zwei generalisierte Freiheitsgrade beschrieben. Die Bewegung der n Einzelkörper einer Radaufhängung ist meist über Kennfelder parametrisiert. Die Modellierung der Radaufhängung erfolgt hierbei einheitlich, unabhängig von ihrer konkreten Konstruktion. Der erste Einzelkörper jeder Radaufhängung ist der Radträger. An ihn ist das Rad drehbar gekoppelt. Für weitere Elemente, wie z.B. den Motor oder die Beladung, sind im Modell zusätzliche, frei positionierbare Einzelkörper vorgesehen. Das Ergebnis sind kompakte, mit minimaler Rechenleistung auswertbare Bewegungsgleichungen für die Fahrzeugmodelle.

Der dritte Teil der Arbeit (Kapitel 5) widmet sich dem Thema Reifen-Kinematik auf gewölbter Fahrbahn. Mit dem hier entwickelten Formalismus der "Karlsruher Kinematischen Kette K3" werden Reifenmessungen unter Prüfstandsbedingungen auf allgemeine, gewölbte Fahrbahnoberflächen anwendbar. Die Kopplung Antriebsstrang--Rad und die für Fahrdynamikregler interessanten Signale der Raddrehung lassen sich hiermit ebenfalls exakt ausdrücken. Die allgemein gebräuchlichen Größen Schlupf und Schräglaufwinkel werden in diesem Zusammenhang nochmals präzise formuliert.

Zusammenfassend kann festgehalten werden, dass die Fahrzeugmodelle präzise die Dynamik des Fahrzeugs beschreiben. Sie bieten dem Entwicklungsingenieur über viele durchdachte Schnittstellen die Möglichkeit, die ihn speziell interessierenden und für seine Aufgabe notwendigen Modellkomponenten zu integrieren. Zukünftig wird die Palette verfügbarer Fahrzeugmodelle, basierend auf den vorgestellten Modellierungsprinzipien und unterstützt durch die verfügbaren Funktionsmodule, um spezielle Modelle für Sattelschlepper, Zugmaschinen und Omnibusse erweitert werden. Eine "Entwicklungsplattform", die obige Konzepte, Modelle und Lösungsstrategien kombiniert, stellt einen wichtigen Beitrag zur Entwicklung, Erprobung und Verbesserung von Fahrdynamikreglern in der Automobilindustrie dar.

Literaturverzeichnis

- [Amm97] AMMON, DIETER: *Modellbildung und Systementwicklung in der Fahrzeugdynamik*. Leitfäden der angewandten Mathematik und Mechanik. Teubner Verlag Stuttgart, 1997. Habilitations-Schrift.
- [Aug02] AUGUSTIN, M.: *Entwicklung einer Mess-, Steuer- und Regel-Einrichtung für einen Reifenprüfstand zur Durchführung realer Messprozeduren in Echtzeit*. Dissertation, Universität Karlsruhe, 2002. Shaker Verlag, Aachen.
- [BAK02] BÖSCH, DR. P., DR. D. AMMON und F. KLEMAPAU: *Reifenmodelle – Wunsch und Wirklichkeit aus der Sicht der Fahrzeugentwicklung*. 4. Darmstädter Reifenkolloquium, 2002. Fortschritts-Berichte VDI Reihe 12 Nr. 511, VDI Verlag Düsseldorf.
- [Bro94] BROSSARD, JEAN-PIERRE: *Mécanique générale – Cinématique générale*. Techniques de l'Ingénieur, 11 1994.
- [Bro95] BROSSARD, JEAN-PIERRE: *Mécanique générale – Dynamique générale. Forme vectorielle*. Techniques de l'Ingénieur, 1995.
- [Czi00] CZICHOS, HORST: *Hütte: Die Grundlagen der Ingenieurwissenschaften*. Springer Verlag, 31. Auflage, 2000.
- [ETA02] ETAS: *LabCar – Hardware-in-the-Loop-Testsysteme für Steuergeräte*. www.etas.de, Stuttgart, Germany, 08 2002.
- [Ets00] ETSCHBERGER, K.: *CAN – Controller Area Network. Grundlagen, Protokolle, Bausteine, Anwendungen*. Carl Hanser Verlag, München, 2000.
- [Fay90] FAYET, MICHEL: *Contributions à l'étude algorithmique du comportement des systèmes de solides rigides, Apports dus à la Robotique*. Thèse de Grade de Doctorat, L'Institut National des Sciences Appliquées de Lyon, 1990.
- [FGU01] FISCHLEIN, H., R. GNADLER und H.-J. UNRAU: *Der Einfluss der Fahrboflächenstruktur auf das Kraftschlussverhalten von Pkw-Reifen bei trockener und nasser Fahrbahn*. ATZ Automobiltechnische Zeitschrift, 10, 2001.

- [FP94] FAYET, MICHEL und FELIX PFISTER: *Analysis of Multibody Systems with Indirect Coordinates and Global Inertia Tensors*. European Journal of Mechanics, 13(3):431–457, 1994.
- [Fro27] FROMM, HANS: *Berechnung des Schlupfes beim Rollen deformierbarer Scheiben*. Zeitschrift für angewandte Mathematik und Mechanik, 7(1):27–58, 2 1927. Dissertation.
- [Gor02] GORONCY, JÜRGEN: *Hardware-in-the-Loop-Simulation und Testautomatisierung, Einsatz von Simulationstools zur Funktionserprobung*. ATZ Sonderheft Phaeton, 7 2002.
- [Ham04] HAMEL, GEORG: *Die Lagrange-Eulerschen Gleichungen der Mecahnik (Habilitationsschrift 1903)*. Zeitschrift Math. Physik, 40(1):1–57, 1904.
- [Ham27] HAMEL, GEORG: *Grundlagen der Mechanik, Mechanik der Punkte und starren Körper*. Handbuch der Physik, 5(1):1–42, 1927.
- [Ham49] HAMEL, GEORG: *Theoretische Mechanik*. Springer Verlag, Heidelberg, 1949.
- [Heu02] HEUN, K.: *Die Bedeutung des D’Alembertschen Prinzips für starre Systeme und Gelenkmechanismen*. Nummer 2 in *Dritte Reihe*. Archiv der Mathematik und Physik, 1902. 57-77, 298-326.
- [Hub04] HUBER, PROF. DR. FRANK: *17 Prozent der Autofahrer haben mit Elektronikproblemen ihres Fahrzeugs zu kämpfen*. <http://www.uni-protokolle.de/nachrichten>, 2004. 30.04.2004 - (idw) Johannes Gutenberg-Universität, Studie.
- [ISO03] ISO14230:1999: *Road vehicles – Diagnostic systems – Keyword protocol 2000*, 11 2003.
- [Jah18] JAHN, J.: *Die Beziehungen zwischen Rad und Schiene hinsichtlich des Kräftespiels und der Bewegungsverhältnisse*. Zeitschrift des Vereines deutscher Ingenieure, 62(11+12):121–125, 145–150, 160–163, 3 1918.
- [Kif02] KIFMANN, ALFONS: *Antiebskräfte*. Auto & Elektronik, 03 2002.
- [KN00] KIENCKE, UWE und LARS NIELSEN: *Automotive Control Systems*. Springer-Verlag Berlin Heidelberg New York, 2000.
- [Lan97] LANG, HANS-PETER: *Kinematik-Kennfelder in der objektorientierten Mehrkörpermodellierung von Fahrzeugen mit Gelenkelastizitäten*. Fortschritts-Berichte VDI Reihe 12 Nr. 323. VDI Verlag Düsseldorf, 1997.
- [Law94] LAWRENZ, W.: *CAN – Controller Area Network. Grundlagen und Praxis*. Hüthig Buch Verlag, Heidelberg, 1994.

- [LN27] L. NORDHEIM, GÖTTINGEN: *Die Prinzipie der Dynamik*. Handbuch der Physik – Grundlagen der Mechanik, Mechanik der Punkte und starren Körper, 5(2):43–90, 1927.
- [Lur68a] LUR'É, L.: *Mécanique Analytique – Cinématique et déplacements, Équation de Lagrange et d'Euler-Lagrange*, Band I. Masson & Cie, Paris, 1968.
- [Lur68b] LUR'É, L.: *Mécanique Analytique – Mouvement relatif et théorie des perturbations, Équations canoniques et principes variationnels*, Band II. Masson & Cie, Paris, 1968.
- [Mag96] MAGGI, G. A.: *Principii della Teoria Matematica del Movimento dei Corpi – Corso di Meccanica Razionale*. U. Hoepli Editore Libraio della real Casa, Milano, 1896.
- [Mag03] MAGGI, G. A.: *Principii di Stereodinamica – Corso sulla Formazione, l'Interpretazione e l'Integrations delle equazioni del movimento dei solidi*. U. Hoepli Editore Libraio della real Casa, Milano, 1903.
- [Mat98] MATSCHINSKY, WOLFGANG: *Radführungen der Straßenfahrzeuge – Kinematik, Elasto-Kinematik und Konstruktion*. Springer Verlag Berlin Heidelberg, 2. Auflage, 1998.
- [ML86] MCINNIS, BAYLISS C. und CHEN-KANG FRANK LIU: *Kinematics and Dynamics in Robotics: A Tutorial Based Upon Classical Concepts of Vectorial Mechanics*. IEEE Journal of Robotics and Automation, RA-2, No. 4, 12-1986.
- [NG01] NÜSSLE, M. und R. GNADLER: *Messeinrichtung zur Bestimmung von Reifenkennlinien im Fahrbetrieb*. ATZ Automobiltechnische Zeitschrift, 7+8, 2001.
- [Orl87] ORLANDEA, N.: *ADAMS (theory and application)*. Vehicle System Dynamics, 16:121–166, 1987.
- [Pac02] PACEJKA, HANS B.: *Tyre and Vehicle Dynamics*. Butterworth-Heinemann, 1. Auflage, 2002.
- [Pap02] PAPA STRAVRIDIS, JOHN G.: *Analytical Mechanics – A Comprehensive Treatise on the Dynamics of Constrained Systems; for Engineers, Physicists, and Mathematicians*. Oxford University Press, 1st Auflage, 2002.
- [Pes88] PESTEL, EDUARD: *Technische Mechanik Band 3: Kinematik und Kinetik*. BI Wissenschaftsverlag, 2. Auflage, 1988.
- [Pfi95] PFISTER, FELIX: *Contributions à la Mécanique Analytique des Systèmes Multicorps*. Thèse de Grade de Doctorat, L'Institut National des Sciences Appliquées de Lyon, 1995.

- [Pfi02] PFISTER, FELIX: *Spatial point contact kinematics and parallel transport*. Journal of Mechanical Engineering Science, 216 Part C:33–45, 2002.
- [PR99] PFISTER, FELIX und CLEMENS REITZE: *Hardware in the Loop – The Technologie for Development and Test of Vehicle Control Systems*. In: *Conférence: La dynamique du véhicule*. Société des Ingénieur Automobile (SIA), 6 1999.
- [PRS03] PFISTER, FELIX, CLEMENS REITZE und CHRISTIAN STEINLE: *What is Steering? Some common misconceptions regarding suspension measures (SDMs)*. In: *SIA - Congrès dynamique du véhicule*. SIA, 6 2003.
- [Rei04a] REITZE, CLEMENS: *Car Reference Manual – One of CarMakers Vehicle Models*. IPG Automotive GmbH, 2004.
- [Rei04b] REITZE, CLEMENS: *Motorcycle Reference Manual – One of CarMakers Vehicle Models*. IPG Automotive GmbH, 2004.
- [Ren80] RENAUD, M.: *Contribution à la modélisation et à la commande dynamique des robots manipulateurs*. Thèse de Doctorat d’Etat, Université Paul Sabatier, Toulouse, 1980.
- [Rie94] RIEDEL, ANDREAS: *Koordinatensysteme, kinematische Kenngrößen, charakteristische Punkte und Transformationen bei Reifen auf ebener und räumlicher/unebener Fahrbahn*. Internal Communication, 1994.
- [Rie02] RIEDEL, ANDREAS: *ABS für Zweiräder - Schnelle und sichere Entwicklung mit HIL*. AEP Automotive Engineering Partners, 5, 10 2002.
- [Ril94] RILL, GEORG: *Simulation von Kraftfahrzeugen*. Grundlagen und Fortschritte der Ingenieurwissenschaften. Vieweg Verlagsgesellschaft mbH Braunschweig, 1994.
- [RW66] ROBERSON, R.E. und J. WITTENBURG: *A dynamical formalism for an arbitrary number of interconnected rigid bodies. With reference to the problem of satellite attitude control*, Band 46D. IFAC Congress, London, 1966.
- [RW03] RIEDEL, ANDREAS und UWE WURSTER: *Zur Relevanz von Fahrermodellen bei Entwicklung und Test von Fahrwerksregelsystemen*. AEP Automotive Engineering Partners, 3, 6 2003.
- [Sac24] SACHS, G.: *Versuche über die Reibung fester Körper*. Zeitschrift für angewandte Mathematik und Mechanik, 4(1):1–33, 2 1924.
- [Sch92] SCHNEIDER, MANFRED: *Himmelsmechanik*, Band 1: Grundlagen, Determinierung. BI Wissenschaftsverlag, 3. neu bearbeitet und erweitert Auflage, 1992.

- [Shu93] SHUSTER, MALCOM D.: *A Survey of Attitude Representation*. Journal of the Astronautical Sciences, 41(4):439–517, 1993.
- [STJ99] SVEN T.H. JANSEN, PETER W.A. ZEGELAAR, HANS B. PACEJKA: *The Influence of In-Plane Tyre Dynamics on ABS Braking of a Quarter Vehicle Model*. Vehicle System Dynamics, 32:249–261, 1999.
- [SW88] SCHIESCHKE und WURSTER: *IPG-TIRE – Ein flexibles, umfassendes Reifenmodell für den Einsatz in Simulationsumgebungen*. 5 1988.
- [tcl04] TCL.TK: *Tcl Developer Xchange*. www.tcl.tk, 2004.
- [TES03] TESIS: *ve-DYNA 3.7 Model Overview*. www.thesis.de, München, Germany, 10 2003.
- [UZ97] UNRAU, H.-J. und J. ZAMOW: *TYDEX-Format: Description and Reference Manual, Release 1.3*. TYDEX Workshop, 1997.
- [WH03] WEILACHER, VOLKER und GERHARD HICKEL: *Automatische Fehlergenerierung*. ATZ Automobiltechnische Zeitschrift, 3, 6 2003.
- [Wit77] WITTENBURG, J.: *Dynamics of systems of rigid bodies*. Teubner Verlag, Stuttgart, 1977.
- [Wor11] WORONETZ, P.: *Über die Bewegung eines starren Körpers, der ohne Gleitung auf einer beliebigen Fläche rollt*. Math. Annalen, 70:410–453, 1911.
- [WWS90] WITTENBURG, J., U. WOLZ und A. SCHMIDT: *MESA VERDE: A General-purpose Program Package for the Symbolical Dynamics Simulation of Multibody Systems*. Springer Verlag, 1990.