




Stefan Strengert



**Methodik zur ganzheitlichen, dynamischen  
Analyse sicherheitsrelevanter, verteilter  
Kraftfahrzeugsysteme unter dem Aspekt  
des Fehlerverhaltens**



 Cuvillier Verlag Göttingen

Methodik zur ganzheitlichen, dynamischen Analyse  
sicherheitsrelevanter, verteilter Kraftfahrzeugsysteme  
unter dem Aspekt des Fehlerverhaltens

Vom Fachbereich Elektrotechnik und Informatik  
der Universität Kassel  
zur Erlangung des akademischen Grades

**Doktor-Ingenieur**

genehmigte

**Dissertation**

von

Dipl.-Ing. Stefan Strengert

geboren am 19. Oktober 1975 in Stuttgart

Referent: Prof. Dr.-Ing. Jürgen Lehold

Koreferent: Prof. Dr.-Ing. Heinz Theuerkauf

Tag der Disputation: 18.01.2008

### **Bibliografische Information der Deutschen Nationalbibliothek**

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.ddb.de> abrufbar.

1. Aufl. - Göttingen : Cuvillier, 2008

Zugl.: Kassel, Univ., Diss., 2008

978-3-86727-537-8

© CUVILLIER VERLAG, Göttingen 2008

Nonnenstieg 8, 37075 Göttingen

Telefon: 0551-54724-0

Telefax: 0551-54724-21

[www.cuvillier.de](http://www.cuvillier.de)

Alle Rechte vorbehalten. Ohne ausdrückliche Genehmigung des Verlages ist es nicht gestattet, das Buch oder Teile daraus auf fotomechanischem Weg (Fotokopie, Mikrokopie) zu vervielfältigen.

1. Auflage, 2008

Gedruckt auf säurefreiem Papier

978-3-86727-537-8

# Vorwort

Die hier vorgestellte Arbeit ist während meiner Tätigkeit im Zentralbereich Forschung und Voraentwicklung der Robert Bosch GmbH in Schwieberdingen entstanden.

Mein besonderer Dank gilt den Herren Prof. Dr.-Ing. J. Leohold und Prof. Dr.-Ing. H. Theuerkauf. Durch ihre fachliche und persönliche Unterstützung, die hilfreichen Anregungen, ihre Diskussionsbereitschaft und ihre Fähigkeit, die richtigen Fragen zu stellen, haben sie zum Gelingen dieser Arbeit beigetragen.

Der Robert Bosch GmbH, die mir die Erstellung dieser Arbeit im Rahmen ihres Doktorandenprogramms ermöglicht hat, bin ich zu besonderem Dank verpflichtet. Allen Mitarbeitern der Voraentwicklung der Robert Bosch GmbH und der Vorentwicklung der ZF Lenksysteme GmbH, die mir während der Erstellung mit Rat und Tat behilflich waren, danke ich für ihre Unterstützung, die zahlreichen anregenden Diskussionen und die sehr angenehme Arbeitsatmosphäre. Dies gilt insbesondere für Herrn Werner Harter, der durch die vorbildliche Betreuung, sein stetes Engagement und den mir zur Verfügung gestellten Freiraum, maßgeblich zu dieser Arbeit beigetragen hat. Den Herren Alexander Krautstrunk und Michael Sprinzel bin ich für die vielfältigen Informationen und ihre zahlreichen spontanen Antworten zum Steer-by-wire System dankbar. Besonders erwähnen möchte ich auch meine Doktoranden-Kollegen Thomas Kottke, Stephan Stabrey und Marcus Wagner, die mich nicht nur in fachlicher Hinsicht, sondern auch durch den allgemeinen Gedankenaustausch unter Gleichgesinnten unterstützt haben.

Mein Dank gilt auch allen Studenten, die mit ihren Praktika und Diplomarbeiten einen nützlichen und gewinnbringenden Beitrag zu dieser Arbeit geleistet haben.

Schließlich möchte ich auch Meike Ullrich erwähnen, die mich in der unangenehmen Phase der schriftlichen Ausarbeitung immer wieder aufs Neue motiviert hat, und danke ihr für ihre Unterstützung im Kampf mit der deutschen Sprache.



---

# Inhaltsverzeichnis

<b>1</b>	<b>Einführung</b>	<b>1</b>
1.1	Motivation und Problemstellung . . . . .	1
1.2	Zielsetzung der Methodik zur Systemanalyse . . . . .	3
1.3	Gliederung der Arbeit . . . . .	4
<b>2</b>	<b>Charakteristika mechatronischer, verteilter, sicherheitsrelevanter Kraftfahrzeugsysteme</b>	<b>6</b>
2.1	Eigenschaften mechatronischer Systeme . . . . .	6
2.2	Eigenschaften sicherheitsrelevanter Systeme . . . . .	7
2.3	Eigenschaften verteilter Systeme . . . . .	9
2.4	Beispiele und Konzeptionen für heutige und zukünftige mechatronische, verteilte, sicherheitsrelevante Kraftfahrzeugsysteme . . . . .	9
<b>3</b>	<b>Anforderungen an eine modellbasierte, ganzheitliche Systemanalyse</b>	<b>16</b>
3.1	Anforderungen an die Analysemethode . . . . .	16
3.2	Anforderungen an das Modellierungskonzept . . . . .	17
<b>4</b>	<b>Vergleichende Analyse und Bewertung bekannter Methoden im Themenkomplex der Systemanalyse</b>	<b>19</b>
4.1	Methoden der System- und Sicherheitsanalyse . . . . .	19
4.1.1	HAZOP-Verfahren . . . . .	21
4.1.2	Fehlermöglichkeits- und Einflussanalyse . . . . .	22
4.1.3	Fehlerbaum-Analyse . . . . .	25
4.1.4	ETA-Verfahren . . . . .	27
4.1.5	Markov-Analyse . . . . .	28
4.1.6	Formale Methoden . . . . .	30
4.1.7	Zusammenfassung der diskutierten Methoden . . . . .	32

4.2	Modellierungskonzepte . . . . .	33
4.2.1	Graphen als Modellelemente eines zustandsorientierten Konzepts . . . . .	34
4.2.2	Zustandsautomaten . . . . .	35
4.2.3	Petri-Netze als Elemente eines zustandsorientierten Konzepts . . . . .	36
4.2.4	CARTRONIC als objektorientierte Modellierungstechnik . . . . .	37
4.2.5	Qualitative Modellierung in einem prozessorientierten Konzept . . . . .	38
4.2.6	Quantitative Modellierung in einem prozessorientierten Konzept . . . . .	39
4.2.7	Hybride Modellierung . . . . .	40
4.2.8	Zusammenfassung . . . . .	41
4.3	Methoden zur Automatisierung von System- und Sicherheitsanalysen . . . . .	43
<b>5</b>	<b>Konzept der ganzheitlichen, dynamischen Systemanalyse</b>	<b>46</b>
5.1	Methodik der ganzheitlichen, dynamischen Analyse . . . . .	46
5.2	Einbettung in einen Entwicklungsprozess . . . . .	50
<b>6</b>	<b>Quantitative hybride Gesamtsystemmodellierung</b>	<b>52</b>
6.1	Klassifizierung von Systemen und Systemkomponenten . . . . .	52
6.2	Modellierung unter funktionalen Aspekten . . . . .	55
6.2.1	Modell des zu untersuchenden Kraftfahrzeugsystems . . . . .	55
6.2.2	Fahrzeugmodell . . . . .	55
6.2.3	Fahrermodell . . . . .	56
6.2.4	Umweltmodellierung . . . . .	57
6.3	Modellierung unter strukturellen Aspekten . . . . .	58
6.3.1	Erweiterung des Kraftfahrzeugsystemmodells . . . . .	59
6.4	Modellierung unter Sicherheitsaspekten . . . . .	60
6.4.1	Fehlermodelle . . . . .	61
6.4.2	Erweiterung des Kraftfahrzeugsystemmodells . . . . .	63
6.4.3	Fehlerinjektion . . . . .	64
6.4.4	Objektiv-quantifizierbare Bewertungsfunktion . . . . .	64

---

<b>7</b>	<b>Identifikation signifikanter Fehler</b>	<b>66</b>
7.1	Algorithmen zur globalen Suche . . . . .	66
7.1.1	Monte Carlo Methode . . . . .	66
7.1.2	Simulated Annealing . . . . .	66
7.1.3	Genetische Algorithmen . . . . .	67
7.2	Evolutionäre Programme . . . . .	67
7.2.1	Aufbau und Funktionsweise . . . . .	68
7.2.2	Selektionsverfahren . . . . .	69
7.2.3	Evolutionäre Operatoren . . . . .	70
7.2.4	Erweiterungen zum Umgang mit Nebenbedingungen . . . . .	72
7.3	Identifikation von Einzelfehlern . . . . .	75
7.4	Erweiterungen . . . . .	76
7.4.1	Identifikation von Fehlerkombinationen . . . . .	76
7.4.2	Identifikation von Fehlersequenzen . . . . .	77
7.4.3	Verknüpfung mit Fehlerauftrittswahrscheinlichkeiten . . . . .	78
7.5	Visualisierung . . . . .	79
7.6	Parallele und skalierbare Anwendung . . . . .	82
7.7	„What happened“-Analyse . . . . .	85
7.8	Iterative Durchführung . . . . .	86
7.9	Anmerkungen . . . . .	87
<b>8</b>	<b>Anwendung am Beispiel eines steer-by-wire Systems</b>	<b>88</b>
8.1	Systemarchitektur der Beispielanwendung . . . . .	88
8.2	Gesamtsystemmodellierung . . . . .	90
8.2.1	Überblick zur Simulationsumgebung . . . . .	90
8.2.2	Modellierung unter funktionalen Aspekten . . . . .	92
8.2.3	Modellierung der Systemumwelt . . . . .	95
8.2.4	Modellierung unter strukturellen Aspekten . . . . .	96
8.2.5	Modellierung unter Sicherheitsaspekten . . . . .	97
8.3	Identifikation signifikanter Fehlerszenarien . . . . .	102
8.3.1	Objektiv-quantifizierbare Bewertungsfunktion . . . . .	103



8.3.2	Parametrierung, Fehlerinjektion und Ergebnisse . . . . .	105
8.3.3	What happened - Analyse . . . . .	109
8.3.4	Iterative Anwendung . . . . .	110
8.3.5	Reproduzierbarkeit des Identifikationsergebnisses . . . . .	111
8.3.6	Einfluss der Parameter des evolutionären Algorithmus auf das Identifikationsergebnis . . . . .	112
<b>9</b>	<b>Zusammenfassung und Ausblick</b>	<b>115</b>
9.1	Zusammenfassung . . . . .	115
9.2	Bewertung und Erkenntnisse . . . . .	116
9.3	Ausblick . . . . .	118
<b>A</b>	<b>Ein redundantes, synchronisiertes TTCAN-Kommunikationsnetz</b>	<b>121</b>
A.1	Einführung in das TTCAN-Kommunikationsprotokoll . . . . .	121
A.2	Synchronisationsalgorithmus . . . . .	123
A.3	Analyse des Fehlerverhaltens . . . . .	126
A.3.1	Fehler, die das zeitliche Verhalten beeinflussen . . . . .	127
A.3.2	Fehler, die die Nachrichteninhalte beeinflussen . . . . .	133
	<b>Literaturverzeichnis</b>	<b>144</b>

# Nomenklatur

ABS	Antiblockiersystem
AFS	Aktivlenkung (engl.: active front steering)
AMR	Anisotroper Magnetwiderstandseffekt (engl.: anisotropic magnetoresistance)
BbW	brake-by-wire
CAN	controller area network
ECU	Steuergerät (engl.: electronic control unit)
EHB	Elektrohydraulische Bremse
ESP	Elektronisches Stabilitätsprogramm
ETA	Ereignisbaumanalyse (engl.: event tree analysis)
FMEA	Fehlermöglichkeits- und Einflussanalyse (engl.: failure modes and effects analysis)
FPK	freely-programmable instrument cluster
FTA	Fehlerbaumanalyse (engl.: fault tree analysis)
HAZOP	hazard and operability studies
HiL	hardware-in-the-loop
MiL	model-in-the-loop
PCU	Leistungselektronik (engl.: power control unit)
RPS	Rotorposition-Sensor (engl.: rotor position sensor)
RTOS	Echtzeit-Betriebssystem (engl.: real-time operating system)
SAS	Lenkwinkel-Sensor (engl.: steering angle sensor)
SG	Steuergerät

SiL	software-in-the-loop
TTCAN	zeitgesteuertes CAN-Kommunikationssystem (engl.: time-triggered CAN)
UML	unified modeling language
$\sigma_{\dot{\Psi},\delta}$	fahrerbezogenes Gefahrenmaß für unerwartete Fahrzeugbewegung
$\Delta\dot{\Psi}_\delta$	Gierratendifferenz
$i_n^t$	$n$ -tes Individuum einer Population $P$ zur Generation $t$
$P(t)$	Population zur Generation $t$
<i>popsiz</i> e	Anzahl Individuen in einer Population
$d$	Anzahl Individuen, die von einer Generation zur nächsten aussterben
$\sigma_{fahrzeug}$	Bewertungsmaß in Bezug auf die Fahrzeugbewegung
$\sigma_{gesamt}$	objektiv-quantifizierbares Bewertungsmaß für die Fehlerauswirkung
$\sigma_{system}$	Bewertungsmaß in Bezug auf den Systemzustand
$t$	Indexierung der Generation
$wp_1, wp_2, wp_3$	Gewichtungsparameter für objektiv-quantifizierbare Bewertungsfunktion



# Kurzfassung

Seit einigen Jahren ist in der Automobilindustrie ein klarer Trend zu mehr Elektronik, hochgradig vernetzten Strukturen und komplexen Assistenz- und Sicherheitssystemen zu erkennen. Diese Systeme unterstützen den Fahrer in kritischen Situationen und entlasten ihn bei der verantwortungsvollen Aufgabe des Führens eines Kraftfahrzeugs. Die immer weitreichendere Übertragung von Verantwortung vom Menschen hin zu technischen Systemen stellt hohe Anforderungen an die Systemsicherheit, um das Risiko einer vom Kraftfahrzeugsystem ausgehenden Gefährdung zu begrenzen.

Mit Hilfe von Sicherheitsanalysen kann das vorhandene Risiko im Betrieb von Kraftfahrzeugsystemen analysiert werden. Klassische Sicherheitsanalysen betrachten in der Regel nur einzelne Bestandteile eines Kraftfahrzeugsystems, welches aber im Allgemeinen aus folgenden Hauptbestandteilen besteht: dem Kraftfahrzeugsystem selbst, dem Fahrzeug, der Fahrzeugumgebung und dem Fahrer. Was passiert jedoch, wenn im Kraftfahrzeugsystem eine Komponente ausfällt, die Software Fehler enthält und zur gleichen Zeit der Fahrer ein kritisches Fahrmanöver einleitet? Solche Fragen können mit klassischen Sicherheitsanalysen nur unzureichend beantwortet werden. Es ist darüber hinaus zu beachten, dass bei den meisten klassischen Sicherheitsanalysen die eigentliche Analyse in Form von Brainstorming-Prozessen mit Expertenteams durchgeführt wird. Dabei kann aber kein Experte alle möglichen Fehlerauswirkungen und -kombinationen im Zusammenspiel der komplexen Bestandteile überblicken und bewerten.

In der vorliegenden Arbeit wird ein modellbasierter Ansatz zur Durchführung einer ganzheitlichen Sicherheitsanalyse vorgestellt, welcher alle Bestandteile eines Kraftfahrzeugsystems berücksichtigt. Dieser Ansatz erweitert die in der Automobilindustrie verwendeten Methoden zur modellbasierten Systementwicklung auf Sicherheitsaspekte und kombiniert diese mit klassischen Sicherheitsanalysen. Die Modellierung des Gesamtsystems basiert auf einem quantitativen, dynamischen und kontinuierlichen Ansatz wie er auch bei der Systementwicklung zu Rapid Prototyping Zwecken eingesetzt wird. Die Durchführung der vorgestellten Analyse erfolgt rechnergestützt und kann mit Hilfe eines Identifikationsalgorithmus bereits während der Systementwicklung selbständig Einzelfehler, Fehlerkombinationen und Fehlersequenzen entdecken, die zu signifikanten Auswirkungen im Gesamtsystem führen. Durch iterative Anwendung dieses Ansatzes ist es möglich, das Kraftfahrzeugsystem kontinuierlich zu analysieren, Verbesserungspotential frühzeitig zu erkennen und entsprechende Änderungen zielgerichtet durchzuführen.

# Abstract

In the automotive industry there is a clear trend to an increase in the number of electronic systems, distributed and interconnected system architectures and complex driver assistant and safety functions in a vehicle. All these systems assist the driver in critical driving situations and relieve him of his responsible job of driving a motor vehicle. The transfer of responsibility from the human being to technical systems makes high demands on system safety in order to limit the risk of operation.

By use of safety analysis it is possible to evaluate the existing risk of operating the automotive system. Classical safety analysis are normally able to examine only single elements of an automotive technical system, but in general an automotive system consists of the following principle elements: the automotive technical system itself, the vehicle, the vehicle environment and the driver. But what happens, if a component of the technical system fails, the software has bugs and the driver starts a critical driving manoeuvre at the same time? Using classical safety analysis methods such question can be answered insufficiently only. An additional lack of classical safety analysis is that the analysis process is done by brainstorming. But as the automotive systems are becoming more and more complex there is hardly any system expert who is able to take all possible failure modes, effects and their combinations into account.

In this work a model based concept for an integrated safety analysis is presented which takes all principle elements of an automotive system into account. This approach extends the well applied methods of model based system development to the aspects of safety and combines them with classical safety analysis. The modeling of the over-all system is based on a quantitative and dynamic description language as it is known from rapid prototyping or controller design. The evaluation is computer-based and a presented algorithm based on evolution programs is used to identify significant system failure effects autonomously. This enables an iterative application of the presented method to analyse the automotive system continuously while development phase, identify improvement potential in an early design phase and realise system modifications to decrease the risk of system operation.



# 1 Einführung

## 1.1 Motivation und Problemstellung

Seit einigen Jahren nimmt der Anteil der Elektrik und Elektronik im Automobilbau stark zu [BFS<sup>+</sup>02, Dai02]. Es werden zunehmend komplexe Assistenz- und Sicherheitssysteme realisiert, die, wie zum Beispiel das Antiblockiersystem (ABS) oder das elektronische Stabilitätsprogramm (ESP), den Fahrer in kritischen Situationen unterstützen und damit die Sicherheit aller Teilnehmer im Straßenverkehr erhöhen. Schon bei der Entwicklung und Einführung dieser Generation von Fahrzeugsystemen hat sich gezeigt, dass die Frage nach der Systemsicherheit eine wichtige Rolle spielt [GBW01, RKR05, EPK<sup>+</sup>02]. Die Sicherheitsrelevanz dieser Systeme für das Gesamtprodukt Fahrzeug hat sich durch den Umstand des möglicherweise fehlerhaften, aktiven Eingriffs ins Fahrgeschehen signifikant erhöht. Das Sicherheitskonzept der meisten heutigen Assistenz- und Sicherheitssysteme beruht im Wesentlichen auf den drei Grundsäulen: Fehlererkennung, Funktionsdegradation und Vorhandensein einer mechanischen Rückfallebene. Damit ist garantiert, dass die Grundfunktionen eines Fahrzeugs, wie Lenken, Bremsen und Beschleunigen, gewährleistet sind.

Die Bandbreite möglicher zukünftiger Kraftfahrzeugsysteme geht jedoch weit über die bereits realisierten Produkte hinaus und reicht bis hin zu Systemen zum autonomen Fahren [ISS02]. Diese Konzepte erfordern jedoch noch weitreichendere Eingriffsmöglichkeiten der elektronischen Systeme in die Grundfunktionen des Fahrzeugs und sind damit in ihrer Sicherheitsrelevanz höher einzustufen. Zudem können diese erweiterten Funktionalitäten oft nur durch den Wegfall der mechanischen Kopplung und damit einer mechanischen Rückfallebene realisiert werden. Dies muss zwangsläufig zu noch höherer Sicherheitsrelevanz führen.

An sicherheitsrelevanten Kraftfahrzeugsystemen ohne mechanische Kopplung, den so genannten x-by-wire Systemen [XBWT98, ISS02], wird seit einigen Jahren geforscht und diverse Ansätze zur Beherrschung dieser Problematik wurden in den einzelnen Fachgebieten entwickelt. Redundante Systemkomponenten [Ech90], verteilte Systemarchitekturen [KKN95], zeitgesteuerte Kommunikations- und Betriebssysteme [Hed01], modellbasierte Fehlererkennungsmechanismen [Höf96], robuste Regelungskonzepte und Methoden der Softwaretechnik [Sie03] können dazu beitragen, die Komponenten eines Systems noch sicherer gegenüber Fehlern und Ausfällen zu machen. Letztendlich wird die Sicherheitsrelevanz sowohl der Einzelkomponenten als auch



des Gesamtsystems durch die Interaktion der Teilkomponenten, das Verhalten des Systemverbunds im Fehlerfall und die Verknüpfung mit der Systemumgebung, wie Fahrer, Fahrzeug oder anderer Verkehrsteilnehmer, bestimmt.

Mit Hilfe von Sicherheitsanalysen kann das vorhandene Risiko im Betrieb eines Systems untersucht werden. Klassische Analysemethoden - dazu zählen z.B. die Fehlermöglichkeits- und Einflussanalyse (FMEA) [MT00] oder die Fehlerbaumanalyse (FTA) [Sch99] - haben sich in vielfacher Hinsicht über Jahre hinweg bewährt. Gemein ist fast allen klassischen Analysemethoden die Durchführung in Form von Brainstorming-Prozessen durch Expertenteams. Dabei erörtern die Experten in Teamsitzungen die Fehlermöglichkeiten in den einzelnen Systemkomponenten und deren Fehlerauswirkungen auf das Gesamtsystem aus ihrer persönlichen Sicht. Es hat sich allerdings gezeigt, dass diese Vorgehensweise einige Nachteile mit sich bringt [Lev95]. Können die Experten die Fehlermöglichkeiten der Einzelkomponenten mit Hilfe von strukturierten Vorgehensweisen noch sehr detailliert und meist auch vollständig beschreiben, so sind hingegen die Fehlerauswirkungen im Zusammenspiel der Einzelkomponenten oft nicht eindeutig und vollständig zu identifizieren. Diese Problematik verschärft sich bei den sicherheitsrelevanten Systemen durch den Einsatz von Redundanz- und Verteiltheitskonzepten zudem erheblich. Eine größere Anzahl an Komponenten, die in einer deutlich komplexeren Art und Weise interagieren, zeigen die Grenzen der klassischen Analysen auf. Die bisher grundsätzlich rein statischen Analysen der Systeme lassen zudem eine zeitliche Betrachtung nicht zu. Fehlerausbreitungen, Folgefehler oder Fehlersequenzen können bisher nur unzureichend untersucht werden.

Das notwendige Systemwissen der Experten bildet sich meist erst im Laufe des Entwicklungsprozesses. Es ist damit einem ständigen Erweiterungs- und Veränderungsprozess unterworfen und lässt sich deshalb schwer dokumentieren. Damit sind die Einschätzungen der Experten bezüglich der Systemsicherheit diesem Veränderungsprozess unterworfen und von ihren aktuellen Erfahrungen und Herausforderungen während der Systementwicklung geprägt. Studien haben gezeigt, dass die Analyseergebnisse in starker Abhängigkeit weiterer Einflussgrößen stehen, die von der Teamzusammensetzung bis hin zur Tagesverfassung einzelner Experten reichen [Lev95, Bis90]. Die Reproduzierbarkeit der Analyseergebnisse ist nicht gegeben.

Der enorme Zeit- und Kostenaufwand für eine klassische Analyse verhindert deren iterative Durchführung während des Entwicklungsprozesses. Viele Fragestellungen, die die Relevanz von Komponenteneigenschaften für die Gesamtsystemsicherheit betreffen, können in einem frühen Stadium der Entwicklung mangels geeigneter Identifikations- und Bewertungsverfahren nicht beantwortet werden. Entscheidungen, wie z.B. die Festlegung der Systemarchitektur, müssen jedoch zu einem frühen Zeitpunkt der Systementwicklung getroffen werden, ohne deren Auswirkungen auf die Sicherheit des Gesamtsystems dynamisch und quantitativ bewerten zu können.

Eine parallele Entwicklung sowohl der funktionalen Charakteristika als auch der Sicherheitseigenschaften eines Systems erscheint mit heute bekannten Prozessen nur unzureichend realisierbar.

## 1.2 Zielsetzung der Methodik zur Systemanalyse

Gegenwärtig ist in der Automobilindustrie ein starkes Bestreben zur modellbasierten Systementwicklung festzustellen. Die in den letzten Jahren entwickelten Methoden eignen sich aber hauptsächlich für die Entwicklung funktionaler Systemeigenschaften, wie z.B. dem Entwurf eines Regelungsalgorithmus.

Ziel dieser Arbeit ist es, die Methoden der modellbasierten Systementwicklung auf Sicherheitsaspekte zu erweitern und mit den klassischen Verfahren zur Sicherheitsanalyse zu kombinieren. Dadurch sollen die eingangs dargelegten Schwachstellen heutiger Analysemethoden reduziert werden. Ergebnis dieser Arbeit sind neue Vorgehensweisen und Algorithmen, die vor allem im Hinblick auf folgende Eigenschaften Vorteile bieten:

**dynamisch:** Mathematische Modelle, die das zeitliche Verhalten eines Systems beschreiben, werden heute bereits bei der Funktionsentwicklung eines neuen Kraftfahrzeugsystems erzeugt. Diese Modelle sollen die Grundlage für eine dynamische Analyse des Systemverhaltens im Fehlerfall bilden. Die zeitliche Fehlerausbreitung und die Auswirkungen von Folgefehlern und Fehlersequenzen können damit untersucht und bewertet werden.

**ganzheitlich:** Durch die modulare Struktur solcher dynamischer Komponentenmodelle ist es möglich, sehr große Systemmodelle zu bilden, die das gesamte zu entwickelnde Kraftfahrzeugsystem und dessen Umgebung aus Fahrzeug, Fahrer und Straße beschreiben. Damit kann die Systemanalyse interdisziplinär und über das ganze System hinweg durchgeführt werden.

**objektiv quantifizierbar:** Mit Hilfe der modellbasierten Verhaltensbeschreibung für das System selbst als auch für dessen Umgebung können Auswirkungen von Fehlern berechnet werden. Die subjektiven Einschätzungen der Experten in den klassischen Sicherheitsanalysen werden ersetzt durch eine objektiv quantifizierbare Berechnungsvorschrift.

**reproduzierbar:** Diese Berechnungen sind jederzeit wiederholbar und zu einem späteren Zeitpunkt nachvollziehbar.

**iterativ anwendbar:** Mathematische Systembeschreibungen eröffnen die Möglichkeit zur rechnerunterstützten Durchführung der Analyse und versprechen eine schnellere und damit mehrmalige Anwendung während des Entwicklungsprozesses. Nach jeder Änderung am System im Laufe des Entwicklungsprozesses kann die Auswirkung dieser Änderung auf die Systemsicherheit überprüft und analysiert werden.

**automatisierbar:** Mit Hilfe komplexer Suchalgorithmen ist es möglich, die Sicherheitsanalyse zu automatisieren. Rechnergestützte Verfahren sind in der Lage, viele Fehlermöglichkeiten zu analysieren und selbständig nach den Fehlermöglichkeiten zu suchen, die besonders signifikante Fehlerauswirkungen im System hervorrufen.

**strukturiert und gut dokumentiert:** Viele der wichtigsten Informationen und Erfahrungen, die während einer Systementwicklung gewonnen werden, sind in den dynamischen Modellen gespeichert und dokumentiert.

## 1.3 Gliederung der Arbeit

Das zweite Kapitel fasst zunächst die Charakteristika mechatronischer, verteilter, sicherheitsrelevanter Kraftfahrzeugsysteme zusammen. Die folgende Gegenüberstellung von Beispielen für heutige und Konzeptionen für zukünftige mechatronische, verteilte, sicherheitsrelevante Kraftfahrzeugsysteme macht die Herausforderungen bei deren Entwicklung deutlich und bildet die Grundlage zur Erarbeitung der Anforderungen an eine modellbasierte, ganzheitliche System- und Sicherheitsanalyse im dritten Kapitel.

Die Untersuchung und Bewertung unterschiedlicher Verfahren zur Sicherheitsanalyse, heute bekannter Modellierungsarten und verschiedener Verfahren zur Automatisierung von Analysen hinsichtlich deren Anwendbarkeit unter den erstellten Anforderungen an eine modellbasierte, ganzheitliche Systemanalyse, ist Bestandteil des vierten Kapitels.

Im Kapitel 5 wird auf Basis der bisher erlangten Erkenntnisse das entwickelte Konzept zur ganzheitlichen, dynamischen Systemanalyse vorgestellt. Es ermöglicht neben der Analyse von Fahrzeugsystemen unter funktionalen Aspekten vor allem die Untersuchung des Verhaltens im Fehlerfall. Auf die Frage nach der Einbettung der Methodik in einen möglichen Entwicklungsprozess für sicherheitsrelevante Kraftfahrzeugsysteme wird anschließend eingegangen.

Neben der notwendigen hybriden Modellierung des Gesamtsystems, die in Kapitel 6 beschrieben wird, ist vor allem das Verfahren zur Identifikation signifikanter Fehlerauswirkungen ein elementarer Bestandteil der vorgestellten Analysemethode. Die entwickelten Algorithmen und deren

Verknüpfung mit dem Gesamtsystemmodell zur rechnergestützten und damit automatisierten Analyse sind Bestandteil von Kapitel 7.

Die praktische Anwendung der ganzheitlichen, dynamischen Analyse sicherheitsrelevanter, verteilter Fahrzeugsysteme erfolgt beispielhaft an einem Steer-by-wire System. Der Beschreibung der Systemarchitektur, der Systemfunktionalität und des Sicherheitskonzepts folgt eine Vorstellung der erstellten Simulationsmodelle. Anschließend wird der Einsatz der entwickelten Methodik erläutert. Die Identifikation sicherheitsrelevanter Fehlerszenarien, deren Bewertung und die daraus abgeleitete Ergreifung von Sicherheitsmaßnahmen bilden zusammen mit einer Aufstellung der gewonnenen Ergebnisse und Erkenntnisse der Analysemethodik das Kapitel 8.

Das neunte Kapitel beinhaltet eine zusammenfassende Beschreibung und Bewertung der Ergebnisse der Arbeit sowie einen Ausblick auf weiterführende Aspekte.

## **2 Charakteristika mechatronischer, verteilter, sicherheitsrelevanter Kraftfahrzeugsysteme**

Schon bei der verbalen Beschreibung und Analyse von Systemen hinsichtlich ihrer Funktionsweise und insbesondere hinsichtlich ihrer Sicherheit stößt man auf das Problem eines nicht einheitlichen Verständnisses aufgrund unterschiedlicher Betrachtungsweisen und Hintergründe. Deshalb werden im ersten Teil dieses Kapitels die Charakteristika mechatronischer, verteilter, sicherheitsrelevanter Kraftfahrzeugsysteme und ihre Konzeptionen herausgearbeitet. Ziel ist eine kompakte, auf wesentliche Aspekte konzentrierte Analyse der Systemeigenschaften und eine Zusammenfassung der daraus resultierenden Herausforderungen. Der sich daran anschließende, zweite Teil dieses Kapitels befasst sich mit Beispielen für heutige, aber auch mit Konzeptionen für zukünftige mechatronische, verteilte, sicherheitsrelevante Kraftfahrzeugsysteme. Der Vergleich der verschiedenen Konzepte, der Systemarchitekturen und der zur Realisierung eingesetzten Technologien dient zur Verdeutlichung der besonderen Herausforderungen und Schwierigkeiten bei der Entwicklung von mechatronischen, verteilten, sicherheitsrelevanten Fahrzeugsystemen.

### **2.1 Eigenschaften mechatronischer Systeme**

Das Kunstwort Mechatronik ist ursprünglich auf die Begriffe Mechanik und Elektronik zurückzuführen. Inzwischen umfasst der Begriff Mechatronik jedoch weit mehr und so kann heute festgestellt werden, dass ein mechatronisches System seine Funktionalität durch die enge Verknüpfung von mechanischen, elektronischen und datenverarbeitenden Komponenten erzielt. Der grundlegende Aufbau eines mechatronischen Systems ist in Abbildung 2.1 dargestellt.

Die Integration verschiedenster Teilsysteme und Technologien zu einem gemeinsamen Ganzen hat nicht nur Auswirkungen auf das mechatronische System selbst, sondern auch auf die Entwicklung solcher Systeme. Die Notwendigkeit zur interdisziplinären Kooperation der beteiligten Entwickler erfordert eine angepasste Entwicklungsweise und das Verständnis für die Herausforderungen der unterschiedlichen Ingenieurdisziplinen. Vor allem bei der Analyse der System-

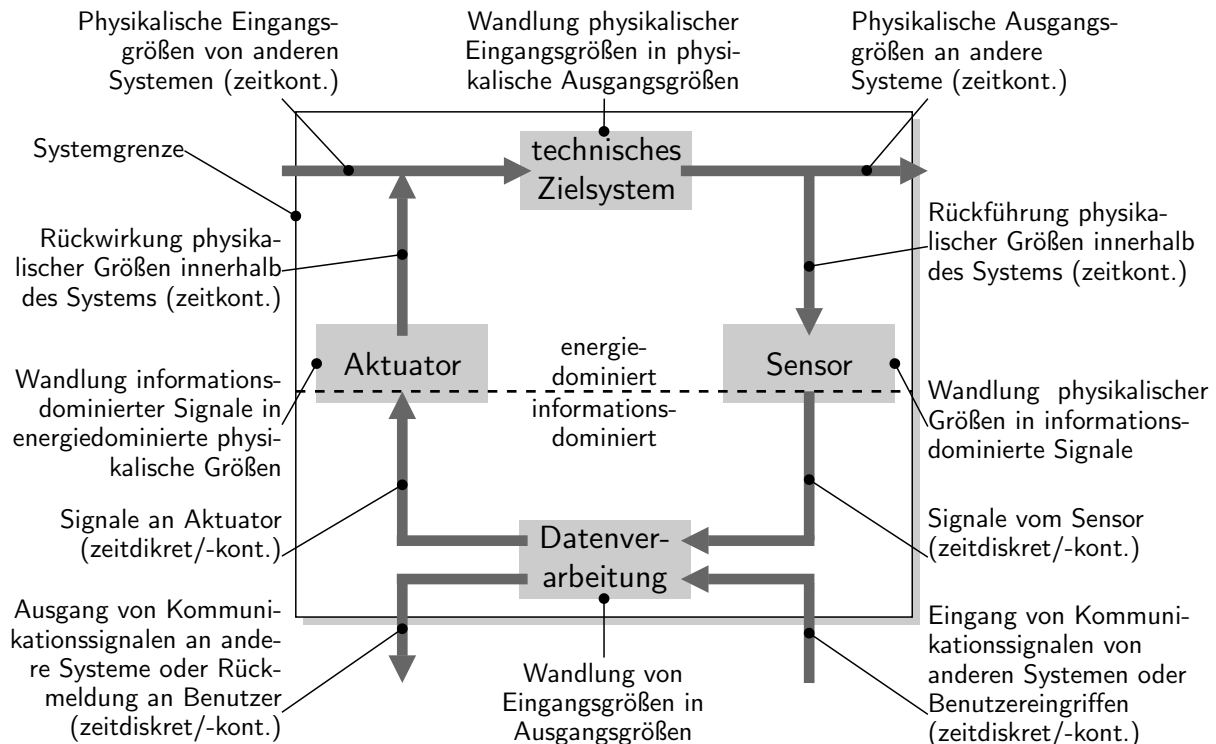


Bild 2.1: Typischer, prinzipieller Aufbau eines mechatronischen Systems

funktionalität hinsichtlich deren Einhaltung funktionsbezogener Anforderungen sind Methoden erforderlich, die sowohl das komplexe und interdisziplinäre Zusammenwirken wie auch die daraus resultierenden Herausforderungen berücksichtigen und beherrschen. Einen noch höheren Stellenwert kommt dieser Analyse zu, wenn es sich beim technischen System um einen Vorgang handelt, der potentiell Schaden an Menschen, Tieren oder Sachwerten verursachen kann.

## 2.2 Eigenschaften sicherheitsrelevanter Systeme

Die Sicherheit eines Systems kann beeinträchtigt werden, wenn Fehler auftreten. Als Fehler bezeichnet man die Abweichung des tatsächlichen Systemverhaltens von einem vorgegebenen oder erwarteten Sollverhalten. Fehler können ursächlich innerhalb einer Komponente entstehen und lassen sich entsprechend ihrer Fehlerursache nach [Ech90] klassifizieren:

**Entwurfsfehler** führen dazu, dass schon vor der Inbetriebnahme Fehler im System vorhanden sind. Die Klasse der Entwurfsfehler lässt sich weiter unterteilen in Spezifikationsfehler, Implementierungsfehler und Dokumentationsfehler.

**Produktionsfehler** können verhindern, dass aus einem korrekten Entwurf ein fehlerfreies Produkt entsteht. Zu hohe Fertigungstoleranzen aufgrund nicht eingehaltener Herstellungsprozesse können z.B. dafür verantwortlich sein, dass die Umsetzung des Entwurfs in ein fehlerfreies Produkt misslingt.

**Betriebsfehler** entstehen im Gegensatz zu den Entwurfs- und Herstellungsfehlern erst durch die Inbetriebnahme des Fahrzeugsystems. Sie lassen sich weiter detaillieren in störungsbedingte Fehler, Verschleißfehler, Bedienungsfehler und Wartungsfehler.

Im Bereich der Sicherheitstechnik werden Systeme entsprechend einem von ihnen ausgehenden Gefährdungspotential im Nominal- wie auch im Fehlerfall klassifiziert. Anhand dieser Klassen leiten sich die Anforderungen an die Systemarchitektur ab. Besondere Bedeutung für die Auslegung der verschiedenen Elektronikarchitekturen erlangt vor allem das notwendige Systemverhalten im Vorhandensein von Fehlern. Man unterscheidet dabei zwischen ausfallsicherem (fail-safe oder fail-silent) und ausfalloperationalem (fail-operational) Systemverhalten:

**ausfallsicher (fail-safe oder fail-silent)** bezeichnet ein System, das nach dem Eintreten eines Fehlers unmittelbar in einen sicheren Zustand übergehen und auch nach weiteren Ausfällen in einem - eventuell auch anderen - sicheren Zustand verbleiben kann.

**ausfalloperational (fail-operational)** wird ein System genannt, wenn im Fehlerfall das System eine Grundfunktionalität solange erbringen kann, bis ein sicherer Systemzustand erreicht wird. Diese Eigenschaft wird von Systemen verlangt, die nicht unmittelbar einen sicheren Systemzustand erreichen können. Dazu gehört z.B. das Flugregelsystem moderner Passagierflugzeuge, das trotz eines Fehlers einen Weiterflug und eine sichere Landung ermöglichen muss.

Bei der Entwicklung von Systemarchitekturen, die das beschriebene ausfalloperationale Verhalten aufweisen müssen, wird oft auf den Mechanismus der Fehlertoleranz zurückgegriffen. Fehlertoleranz bezeichnet die Fähigkeit eines Systems auch mit einer begrenzten Anzahl an fehlerhaften Komponenten seine geforderte Funktion zu erbringen. Meist ist das jedoch nur durch redundante Strukturen zu erreichen, so dass eine Funktionalität von mehreren unabhängigen Komponenten angeboten wird. Die erhöhte Anzahl an Komponenten bedeutet jedoch eine deutliche Steigerung der Systemkomplexität und damit eine potentielle Reduktion der Systemsicherheit. Die augenscheinliche Herausforderung bei der Entwicklung sicherheitsrelevanter Systeme besteht also in der Beherrschung dieses Teufelskreises.

## 2.3 Eigenschaften verteilter Systeme

Der Zusammenschluss unabhängiger Komponenten, die gemeinsam eine Funktion erbringen und dazu ausschließlich über Nachrichten miteinander kommunizieren, wird als verteiltes System bezeichnet. Damit einhergehend entsteht unweigerlich ein steigender Bedarf an Nachrichtenaustausch und Abstimmung unter den beteiligten Systemkomponenten selbst, so dass dem Kommunikationssystem die Bedeutung eines entscheidenden Systembestandteils zukommt. Ähnlich den Schwierigkeiten in einer menschlichen Arbeitsgruppe, die ebenfalls eine gemeinsame Aufgabe zu erfüllen hat, ergeben sich durch die komplexen Interaktionsstrukturen neue Herausforderungen und Fehlermöglichkeiten. Anhand der im folgenden Abschnitt dargelegten Beispiele werden diese Herausforderungen konkretisiert.

## 2.4 Beispiele und Konzeptionen für heutige und zukünftige mechatronische, verteilte, sicherheitsrelevante Kraftfahrzeugsysteme

Seit einigen Jahren nimmt der Anteil der Elektrik und Elektronik im Automobilbau stark zu. In unterschiedlichsten Bereichen des Fahrzeugbaus werden unter stark differierenden Rahmenbedingungen immer mehr Systeme realisiert, die vorwiegend auf Elektronik basieren. Bild 2.2 zeigt eine Unterteilung der elektrischen und elektronischen Systeme im Automobilbau und führt Systembeispiele der jeweiligen Klasse an. Wie zu erkennen ist, lassen sich die Systeme insbesondere in Bezug auf ihre Sicherheitsanforderungen den verschiedenen Systemklassen zuordnen. Zur ersten Klasse, der „consumer electronics“, gehören alle Systeme, die so genannte Infotainment- oder Entertainment-Dienste erbringen. Sie unterhalten den Fahrer oder stellen ihm Informationen jeglicher Art zur Verfügung. Fehlerhafte Informationen, wie z.B. der Routenvorschlag eines Navigationssystems entgegen einer Einbahnstraße, müssen durch den Fahrer überprüft, erkannt und verworfen werden. Die tatsächliche Verwertung und Umsetzung der angebotenen Informationen obliegt allein dem Fahrer.

Die nächste Klasse bilden die Systeme der Karosserieelektronik, wie z.B. die Alarmanlage, die Klimaanlage oder die Tür- und Beleuchtungselektronik. Sie erbringen nicht nur Informationsleistungen, sondern können über Aktuatoren ihnen zugeordnete Prozesse steuern. Ihre Eingriffsmöglichkeiten wirken sich jedoch grundsätzlich nicht auf die Stabilität oder die Grundfunktionen eines Fahrzeugs, wie dem Lenken, Bremsen oder Beschleunigen, aus. Um Fehlfunktionen zu beherrschen, sind diese Systeme nach dem fail-safe Prinzip aufgebaut. Sie verfügen über



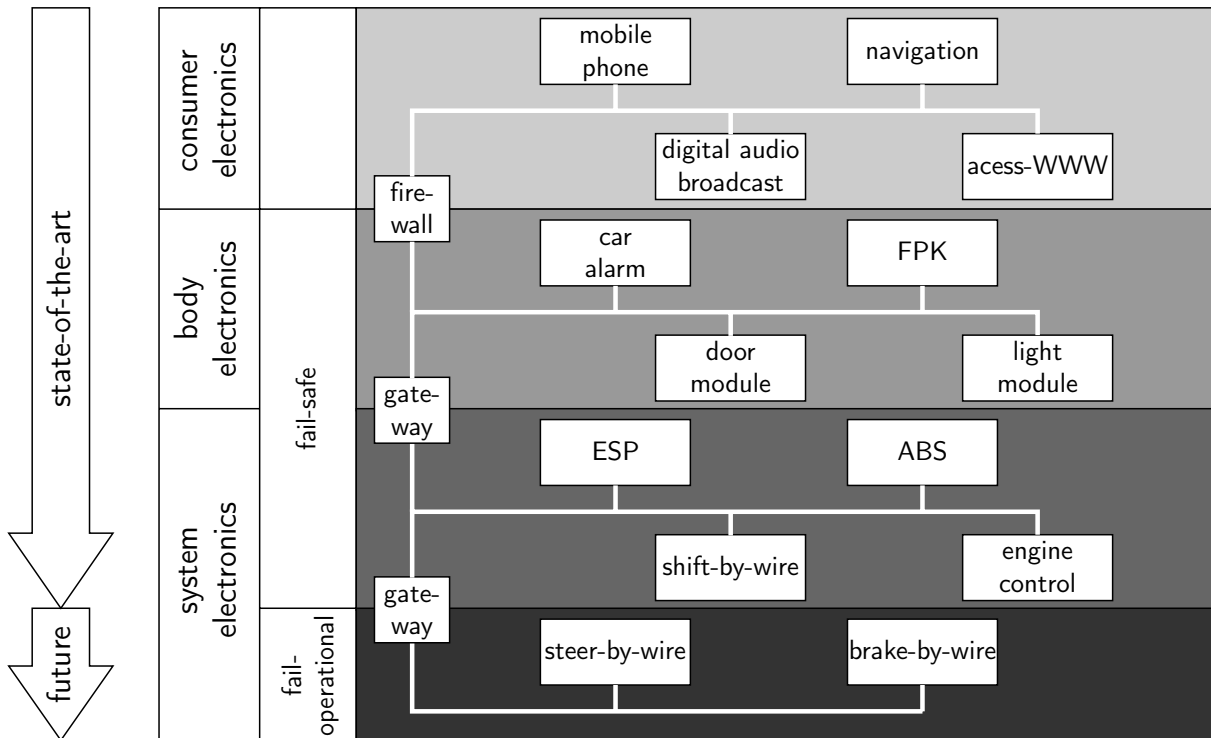


Bild 2.2: Systemklassen in der Automobilindustrie nach [Di199]

ein Sicherheitskonzept, das auf verschiedenen Fehlererkennungsmechanismen und einem Degradationsmanagement beruht. Je nach Art des Fehlers wird die Funktion des Systems entweder eingeschränkt oder deaktiviert. Die Sicherheit steht über der Zuverlässigkeit bzw. Verfügbarkeit.

Zunehmend werden aber auch komplexe Assistenz-, Komfort- und Sicherheitssysteme realisiert, die, wie z.B. das ABS, das ESP oder die Aktivlenkung (AFS), der Gruppe der „system electronics“ angehören. Sie unterstützen oder entlasten den Fahrer in den unterschiedlichsten Situationen durch einen aktiven Eingriff in die Grundfunktionen seines Fahrzeugs. Der Fahrer kann erstmalig nicht allein für das Verhalten des Fahrzeugs verantwortlich gemacht werden, sondern ist auf eine korrekte und sichere Funktion des Systems angewiesen. Er ist meist sowohl aus technischen als auch physischen Gründen nicht in der Lage, fehlerhafte Eingriffe in das Fahrgeschehen zu korrigieren. Schon bei der Entwicklung und Serieneinführung dieser Klasse von elektronischen Fahrzeugsystemen hat sich gezeigt, dass die Frage nach der Systemsicherheit eine immer wichtigere Rolle einnimmt. Die Relevanz dieser Systeme für die Sicherheit des Gesamtprodukts Kraftfahrzeug hat sich durch den Umstand des möglicherweise fehlerhaften aktiven Eingriffs ins Fahrgeschehen signifikant erhöht.

Dieser Problematik muss mit einem entsprechenden Sicherheitskonzept begegnet werden, das am Beispiel der Aktivlenkung kurz erläutert werden soll. Bei AFS handelt es sich um ein elek-

tronisch geregeltes Lenksystem, das situationsabhängig eine Überlagerung eines Winkels zum Lenkradwinkel erlaubt und somit eine Variation der Lenkübersetzung ermöglicht. Die mechanische Kopplung zwischen Lenkrad und Vorderachse muss dabei nicht aufgetrennt werden. Die Funktionsweise ist in Bild 2.3 dargestellt.

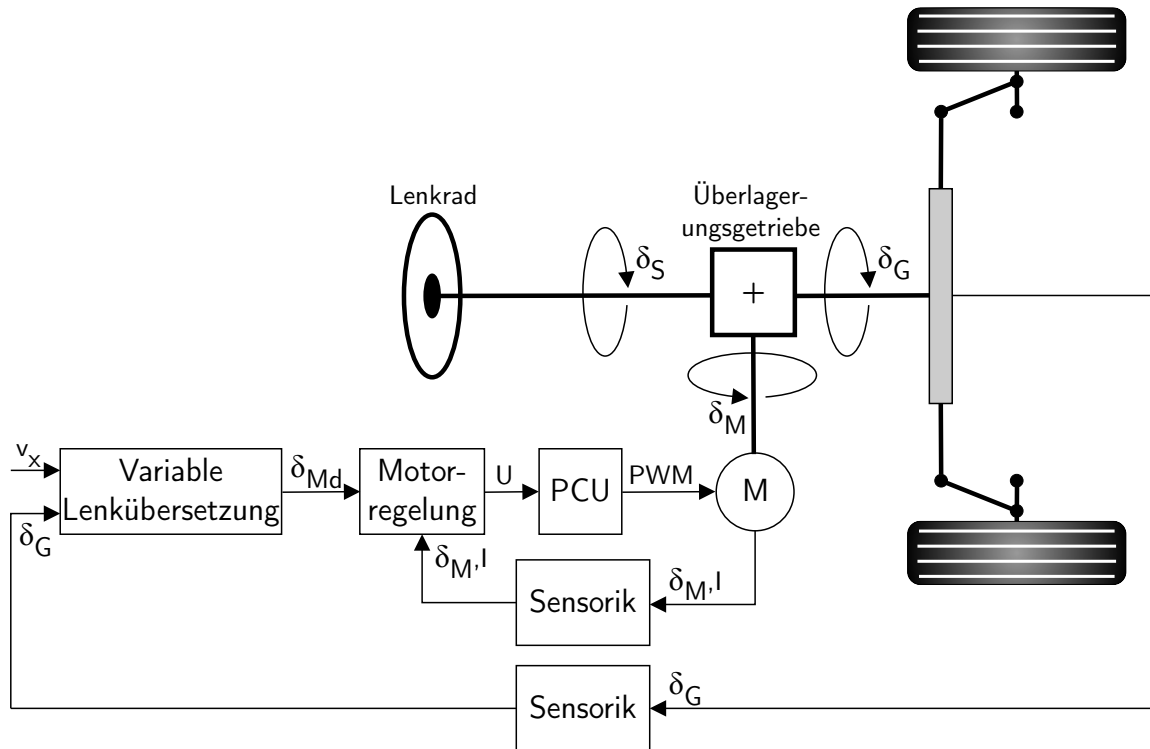


Bild 2.3: Prinzipaufbau des Active Front Steering nach [RKR05]

Mit Hilfe einer Sicherheitsanalyse wurden die Fehlfunktionen auf Komponentenebene ermittelt, die Auswirkungen auf die Systemfunktion haben [RKR05]. Für jede dieser Fehlfunktionen sieht das Sicherheitskonzept mindestens eine der folgenden Überwachungsfunktionen vor:

- Applikationsunabhängige Überwachung der elektronischen Komponenten wie Sensorik, Steuergerät, etc. in Form von Prüfung analoger Signale, Speichertests, Watchdog, usw.
- (Applikationsabhängiges) diversitäres Rechnen
- Plausibilisierung der Nutzsignale gegeneinander und gegen die Fahrsituation
- Absicherung der Kommunikationswege zum und vom Steuergerät
- Applikationsabhängige Überwachung der Aktuatorik

Das Sicherheitskonzept steht damit auf den folgenden drei Grundsäulen: Fehlererkennung, Funktionsdegradation und Vorhandensein einer mechanischen Rückfallebene. Ein fehlerhaftes Eingreifen wird durch die Fehlererkennungsmechanismen verhindert. Das Vorhandensein der mechanischen Kopplung garantiert, auch im Falle eines Systemausfalls, die Grundfunktionalität des Fahrzeugs. Damit sind auch bei dieser Systemklasse die Anforderungen an die Sicherheit deutlich stärker gewichtet als die Anforderungen an die Zuverlässigkeit.

Die Bandbreite möglicher zukünftiger Kraftfahrzeugsysteme und -funktionen geht jedoch weit über die bereits realisierten Produkte hinaus und reicht bis hin zu Systemen zum autonomen Fahren. Alle diese Konzepte haben jedoch gemeinsam, dass sie noch weitreichendere Eingriffsmöglichkeiten ins Fahrgeschehen erfordern als bisher realisiert. Dafür benötigen sie elektronische Schnittstellen, mit deren Hilfe auch die Grundfunktionen eines Fahrzeugs beeinflussbar sind. Der damit einhergehenden Gefahr einer fehlerhaften Beeinflussung und deren negativen Auswirkungen muss deshalb Rechnung getragen werden.

Ein Konzept zur Erweiterung der Eingriffsmöglichkeiten stellen die so genannten x-by-wire Systeme dar. Sie sind gekennzeichnet durch einen vollständigen Verzicht auf eine mechanische Kopplung bei der Erfassung und Umsetzung eines Fahrerwunsches. Stattdessen erfolgt sowohl die Energie- als auch die Informationsübertragung rein elektrisch bzw. elektronisch. Die weiteren Vorteile von x-by-wire Systemen sind vielfältig [DFM<sup>+</sup>97]. Durch den weitgehenden Ersatz der Mechanik durch Elektronik kann neben völlig neuen konstruktiven Freiheiten Bauraum gewonnen, die Montage vereinfacht, die Modularität gesteigert und eine bisher nicht gekannte Flexibilität bezüglich Applikation und Funktionalität erzielt werden. Der Umweltverträglichkeit wird durch den Verzicht auf Hydraulik-Flüssigkeit gedient. Ferner kann die passive Sicherheit der Fahrgastzelle erhöht werden. So reduziert sich beispielsweise durch den Einsatz von steer-by-wire Systemen das Verletzungsrisiko bei einer Kollision im Lenkradbereich deutlich.

Von diesen Vorteilen lässt sich nur dann profitieren, wenn die x-by-wire Systeme auch mindestens ebenso sicher und zuverlässig funktionieren wie ihre mechanischen Pendanten. Dies stellt jedoch aufgrund der steigenden Komponentenzahl und einer deutlich komplexeren Interaktion zwischen den Systembestandteilen eine erhebliche Herausforderung dar. Meist wird versucht, dieser Problematik mit Hilfe von Fehlertoleranzmechanismen zu begegnen. Sie bringen jedoch oft noch komplexere und aufwändigere Systemstrukturen mit sich. Erschwerend wirkt sich zudem aus, dass im Zuge der Lösung der mechanischen Kopplung die Sicherheit und die Verfügbarkeit eines Fahrzeugsystems untrennbar miteinander verknüpft werden. Könnte die Entwicklung aller bisher klassifizierten, mechatronischen Fahrzeugsysteme noch auf ihre Sicherheit oder ihre Zuverlässigkeit hin optimiert werden, so ist mit der Einführung von x-by-wire Systemen zum ersten Mal das Fahrzeug nur noch dann sicher, wenn das x-by-wire System auch zuverlässig ist.

Eine Möglichkeit zur Realisierung des damit notwendigen fail-operational Prinzips ist die Kombination des elektronischen x-by-wire Systems mit einer zusätzlichen mechanischen Rückfallebene. Das hat den Vorteil, dass im nominalen Betriebsfall herkömmliche und damit kostengünstige Elektronikarchitekturen zur Realisierung der Systemfunktion ausreichen. Im Fehlerfall wird allerdings auf die meist funktional deutlich eingeschränkte und zusätzliche Kosten verursachende Rückfallebene umgeschaltet, um die Grundfunktionen des Fahrzeugs zu gewährleisten. Ein Vertreter dieser Art von x-by-wire Systemen stellt die elektrohydraulische Bremse (EHB) dar. Ihre Systemarchitektur ist im Bild 2.4 gezeigt.

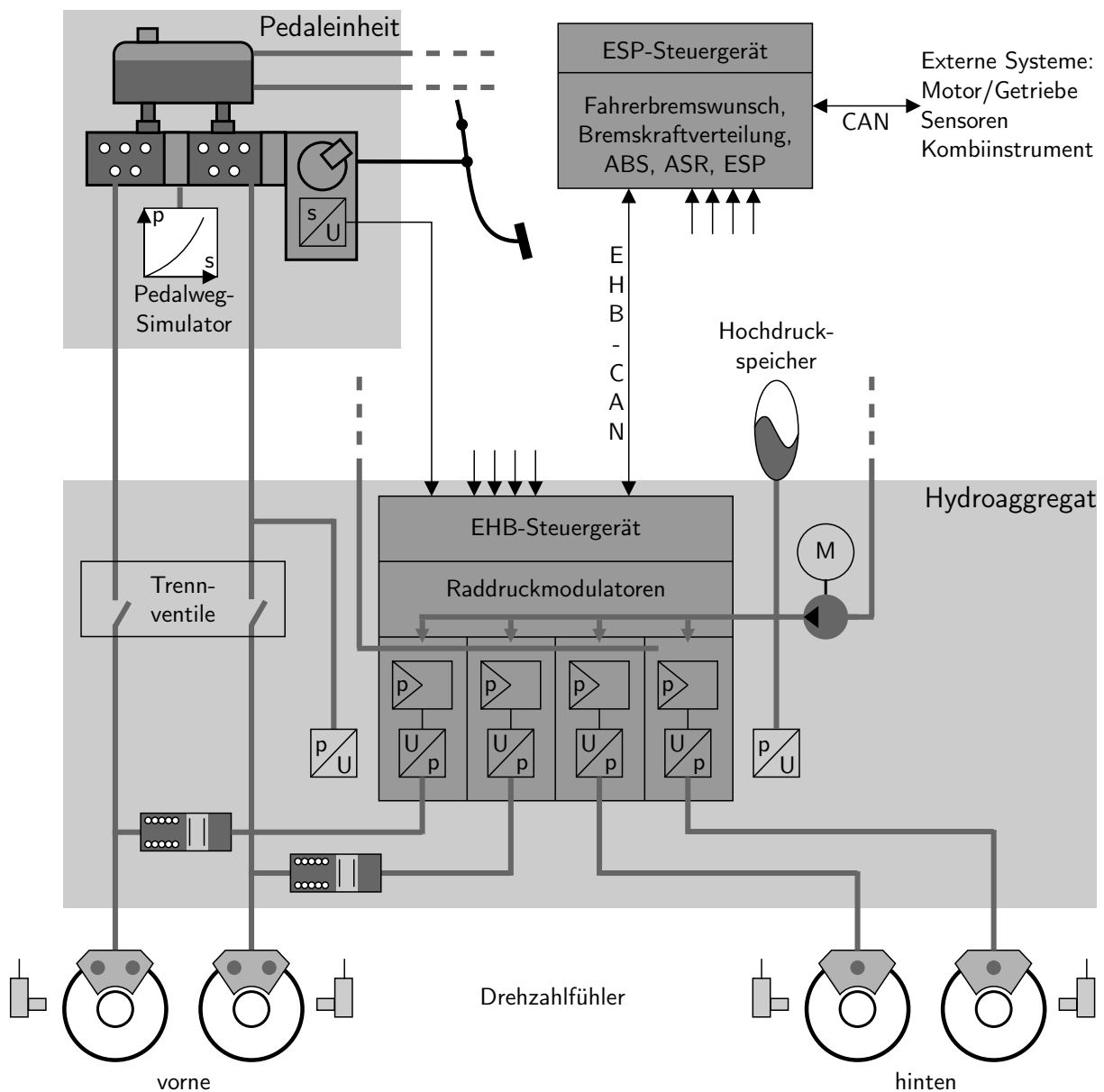


Bild 2.4: Prinzipschaubild der elektrohydraulischen Bremse nach [GBW01]

Oft wird diese erste Realisierungsvariante nur als Wegbereiter für eine Lösung ohne mechanische Rückfallebene bezeichnet. Damit kann man auf die zusätzliche Kosten verursachende und nur im Fehlerfall benötigte Mechanik verzichten. Denn neben dem Kostenfaktor haben Studien gezeigt, dass insbesondere bei Lenksystemen die Kombination von mechanischen und elektronischen Systemen einen nicht unerheblichen Nachteil mit sich bringt [FMH<sup>+</sup>01, NK03]. Der im Fehlerfall sehr plötzliche, ungewohnte und deutlich spürbare Übergang von einem komfortablen elektronischen Lenksystem zu einem funktionell erheblich eingeschränkten, mechanischen System wird vom Normalfahrer nicht beherrscht bzw. als sehr störend empfunden. Es darf folglich nicht mehr davon ausgegangen werden, dass durch Aktivierung eines mechanischen Backupsystems automatisch ein sicherer Betriebszustand des Fahrzeugs erreicht wird.

Für eine rein elektrische/elektronische Lösung bedarf es einer bisher im Kraftfahrzeug nicht bekannten Elektronik-Architektur. Diese gründet sowohl aus technologischer, struktureller, als auch methodischer Sicht auf neuen Konzeptionen. Der Trend, im Automobilbereich einzelne Fahrzeugkomponenten miteinander zu verknüpfen, um neue Funktionen zu realisieren, führt aus übergeordnetem Blickwinkel zu einer wachsenden Anzahl verteilter Systeme. Die Gründe für diese Entwicklung wurden ausführlich in der Fachliteratur untersucht und veröffentlicht (z.B. in [Kop97]). Damit wird der Datenaustausch zwischen den Komponenten einer verteilten Architektur immer wichtiger und das Kommunikationssystem selbst zu einem zentralen Systembestandteil. Dies ist insbesondere dann der Fall, wenn, wie bei x-by-wire Systemen üblich, die gemeinschaftlich erbrachte Funktion harten Echtzeitanforderungen unterliegt. Es also eine bedeutende Rolle spielt, dass das Ergebnis dieser verteilten Funktionen sowohl im Wertebereich als auch im Zeitbereich korrekt ist, da diesbezügliche Fehler zu katastrophalen Konsequenzen (Verlust von Menschenleben und Sachwerten) führen können. Herkömmliche Kommunikationssysteme im Kraftfahrzeug, wie z.B. das Controller-Area-Network (CAN), können diese Anforderungen nicht erfüllen. Aus diesem Grund werden für die x-by-wire Systeme zeitgesteuerte Kommunikationsnetze [Fle05, ISO04, TTT05] entwickelt, die durch a priori Definition eines Kommunikationsfahrplans und der Bildung eines einheitlichen Zeitverständnisses, einen zeitlichen Determinismus bezüglich jeder Art von Ausführungszeit gewährleisten.

Damit lässt sich stets vorhersagen, was zu welchem Zeitpunkt in der Zukunft geschehen wird und eine Systemdimensionierung finden, die auch im Fehlerfall eine Überlastung sowohl der Kommunikationsnetze als auch der Rechenknoten verhindert. Die Systemintegration findet somit bereits während der Entwicklung statt. Die Systementwicklung muss methodisch neue Wege gehen, denn jede Schnittstelle, jede Kommunikationsnachricht und jeder zeitliche Ablauf ist schon während der Entwicklung zu definieren. Darüber hinaus kann die Auslegung eines x-by-wire Systems als fehlertolerantes, verteiltes Echtzeitsystem nur dann gelingen, wenn durch einen konsequenten und systematischen Ansatz vom Entwurf bis zur Realisierung die Sicherheits-

und Verlässlichkeitsanforderungen berücksichtigt werden. Die Einführung neuer Entwicklungsprozesse und die Etablierung einer veränderten Sicherheitsphilosophie ist zwingend erforderlich, um den Nachweis der Sicherheit eines rein elektrischen/elektronischen x-by-wire Systems führen zu können.

Neben den notwendigen strukturellen und methodischen Neuerungen spielt auch der Einsatz neuer Technologien eine wichtige Rolle bei der Realisierung von x-by-wire Systemen. Sichere und vor allem zuverlässige Elektronik und Rechnerarchitekturen, fehlervermeidende Konzepte aus der Softwaretechnik, fehlertolerante Betriebssysteme, energiesparende Aktuatorkonzepte und intuitiv bedienbare Mensch-Maschine-Schnittstellen sind nur eine kleine Auswahl neuer Technologien und Konzepte, die es zu beherrschen und zu verbessern gilt. Doch auch im Bereich des Bordnetzes sind noch intensive Forschungsaktivitäten zu leisten, da eine sichere Energieversorgung das Rückgrad eines jeden rein elektrischen/elektronischen Kraftfahrzeugsystems darstellt. Insbesondere da durch die mechanische Trennung auf eine mögliche Nutzung der menschlichen Betätigungsenergie im Fehlerfall verzichtet wird.

Unter Berücksichtigung dieser zukunftsweisenden Konzeptionen und der Beherrschung der dargelegten Herausforderungen könnte nach [Hed01] eine mögliche Architektur für ein rein elektrisches/elektronisches Bremssystem, wie in Bild 2.5 gezeigt, aussehen.

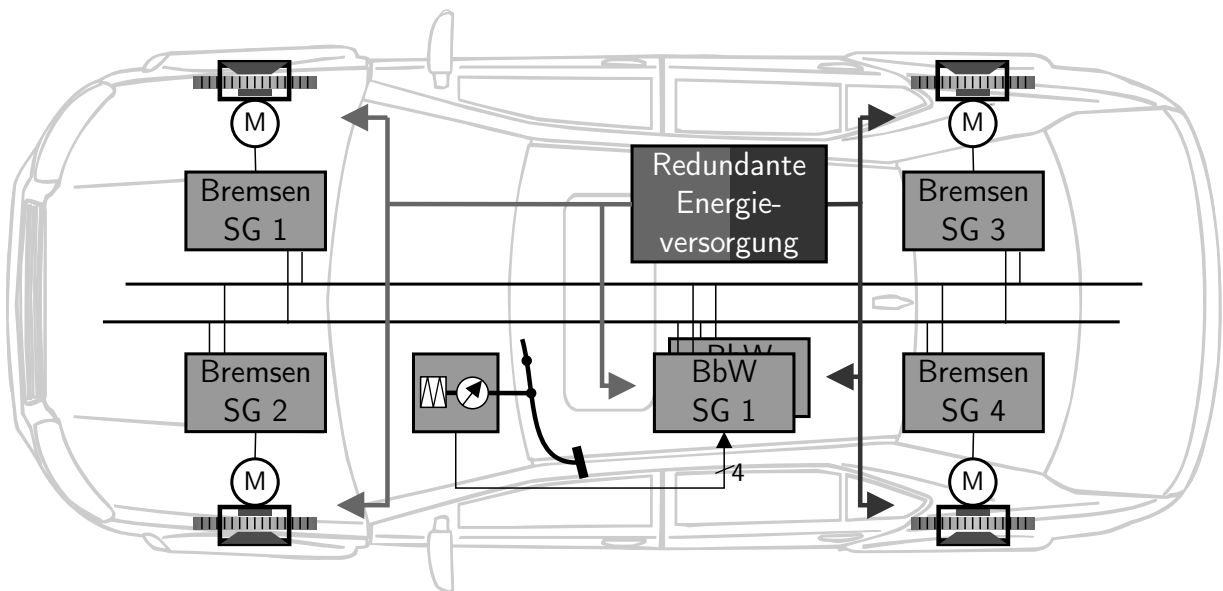


Bild 2.5: Prototypischer Konzeptvorschlag eines brake-by-wire Systems nach [Hed01]

## 3 Anforderungen an eine modellbasierte, ganzheitliche Systemanalyse

Das vorherige Kapitel hat deutlich gemacht, welche Herausforderungen bei der Entwicklung mechatronischer, sicherheitsrelevanter, verteilter Kraftfahrzeugsysteme zu bewältigen sind. Insbesondere im Zuge des Wandels der Entwicklungsmethodik hin zu einem modellbasierten Entwicklungsprozess und dem Bestreben, sicherheitsrelevante Grundfunktionen im Fahrzeug mechatronisch zu realisieren, ergibt sich neben zahlreichen Chancen für innovative Funktionen auch der Bedarf für eine Methodik zur ganzheitlichen, dynamischen Analyse solcher Fahrzeugsysteme insbesondere unter dem Aspekt des Fehlerverhaltens. Mit Hilfe dieses Analyseverfahrens muss es möglich sein, ein mechatronisches Gesamtsystem hinsichtlich dessen funktionalen, strukturellen und sicherheitstechnischen Aspekten auf Basis eines Systemmodells zu untersuchen. Die detaillierten Anforderungen an ein derartiges Analyseverfahren und die damit in Verbindung stehende Systemmodellierung werden im Folgenden erläutert.

### 3.1 Anforderungen an die Analysemethode

Nachfolgend werden zunächst die Anforderungen für eine derartige Methodik formuliert:

- Das komplexe Zusammenspiel aller Komponenten eines sicherheitsrelevanten, verteilten Fahrzeugsystems muss analysierbar sein. Insbesondere die Interaktion mit dem Fahrer, dem Fahrzeug und einer Systemumgebung muss bei einer ganzheitlichen Systemanalyse berücksichtigt werden.
- Die Analyse soll rechnergestützt und automatisiert durchgeführt werden. Hierzu ist ein modellbasierter Ansatz erforderlich. Das Modellierungskonzept muss den im folgenden Abschnitt aufgeführten Anforderungen genügen.
- Die zu entwerfende Methodik muss in einen modellbasierten Entwicklungsprozess integrierbar sein. Das daraus hervorgehende funktionale Modell soll als Grundlage für die notwendige Modellbildung zur Sicherheitsbetrachtung dienen. Diese einheitliche Modellbasis gewährleistet einerseits einen erheblich geringeren Aufwand bei der Modellerstellung

und reduziert andererseits die Möglichkeit von Fehlern durch unterschiedliche Modellierungen zur Funktions- und Sicherheitsanalyse.

- Mögliche Fehler in den Komponenten des Fahrzeugsystems, deren Ausbreitungswege im System, die daraufhin ausgelösten Auswirkungen und die entsprechenden, möglicherweise ungünstigen Reaktionen sowohl des Fahrzeugs als auch des Fahrers sind zu berücksichtigen.
- Die automatisierte Bewertung des Systemverhaltens soll objektiv-quantifizierbar sein, um ein nachvollziehbares und reproduzierbares Analyseergebnis zu erhalten.
- Die Untersuchung darf sich - insbesondere bei der Analyse unter sicherheitstechnischen Aspekten - nicht auf ausgewählte Zeitpunkte, Einzelfehler oder bestimmte Systemzustände beschränken.
- Die zu entwickelnde Methodik muss schon während der Entwicklung einsetzbar sein und durch iterative Anwendung zur kontinuierlichen Verbesserung und parallelen Entwicklung von funktions- und sicherheitsorientierten Systemaspekten beitragen.

## 3.2 Anforderungen an das Modellierungskonzept

Aus den Anforderungen an eine modellbasierte, ganzheitliche Analyse in der Entwicklung befindlicher Kraftfahrzeugsysteme unter funktionalen wie auch sicherheitstechnischen Aspekten gehen folgende Anforderungen an das zu Grunde liegende Modellierungskonzept hervor:

- Die Eigenschaften und Funktionen aller Bestandteile eines Fahrzeugsystems müssen beschreibbar sein. Dazu zählen neben Abläufen in sequentieller und paralleler Form auch zeit-kontinuierliche und zeit-diskrete Verhaltensprozesse.
- Die zeitlich korrekte Nachbildung der tatsächlichen Abläufe muss durch die Modellierungsart gewährleistet sein. Insbesondere die Analyse von Kommunikationsbeziehungen, zulässigen Fehlererkennungszeiten und Reaktionszeiten zwischen Mensch und System bedürfen einer Berücksichtigung von Ausführungs-, Übertragungs- und Totzeiten.
- Die Architektur eines Systems bestimmt die Art und Weise des Zusammenspiels der Komponenten in schwerwiegendem Maße. Aus diesem Grund müssen auch die strukturellen Aspekte eines Systems mit Hilfe der zum Einsatz kommenden Modellierungsmittel darstellbar sein.



- Das Modellierungskonzept muss neben dem Nominal-Systemverhalten auch die Verhaltensbeschreibung unter Annahme verschiedener Fehler ermöglichen.
- Für jede Komponente ist neben deren funktionalem Modell auch ein Fehlermodell anzulegen. Der Wechsel zwischen den beiden Beschreibungen der Funktions- und der Fehlerebene muss zu jedem Betrachtungszeitpunkt möglich sein. Dabei spielt weniger die Modellierung der exakten Fehlerabläufe im Inneren einer Komponente eine Rolle als vielmehr deren Schnittstellenverhalten im Fehlerfall.
- Eine Fehlerausbreitung über die Schnittstellen der fehlerhaften Systemkomponente hinweg in angrenzende Systembestandteile muss umsetzbar sein.
- Die Herstellung einer eindeutigen Verbindung zwischen Fehlerursache und deren Auswirkungen muss gewährleistet sein. Die Art und Weise der entsprechenden Auswirkungen sind bei der Quantifizierung und Bewertung eines Gefährdungsgrades zu verwenden.
- Die Reaktion eines Systems auf Fehler muss abgebildet werden. Dazu sind Fehlererkennungsmechanismen und -behandlungsroutinen zu implementieren, die einen Wechsel zwischen unterschiedlichen Betriebsmodi hervorrufen können.
- Die ganzheitliche Betrachtung bringt unwillkürlich eine hohe Komplexität und eine große Menge an zu analysierenden Einzelkomponenten mit sich. Deshalb ist schon bei der Wahl eines Modellierungskonzepts darauf zu achten, dass Mechanismen zur strukturierten, hierarchischen und übersichtlichen Modellerstellung vorgesehen sind. Eine modularisierte und komponentenbasierte Modellierung ist wünschenswert, da sie die Wiederverwendbarkeit fördert und zur Reduktion der Komplexität beiträgt.

---

## 4 Vergleichende Analyse und Bewertung bekannter Methoden im Themenkomplex der Systemanalyse

Im vorigen Kapitel wurden die Anforderungen an eine Analysemethodik für die Entwicklung von mechatronischen, sicherheitsrelevanten, verteilten Kraftfahrzeugen herausgearbeitet. Nachfolgend sind bekannte Techniken auf den für diese Arbeit relevanten Themengebieten zusammengefasst und im Hinblick auf die definierten Anforderungen vergleichend bewertet.

Der erste Teil des folgenden Kapitels beinhaltet die Vorstellung bekannter Methoden der System- und Sicherheitsanalysen. Die jeweiligen Vor- und Nachteile der einzelnen Methoden sind hinsichtlich der Anforderungen zusammengefasst. Meist verwenden die System- und Sicherheitsanalysen Modelle zur Dokumentation der Ergebnisse oder zur Nachbildung des tatsächlichen Systemverhaltens. Eine Gegenüberstellung verschiedener Modellierungskonzepte zeigt die Eigenschaften und Unterschiede heute bekannter Verfahren auf. Die Möglichkeit zur Automatisierung ist heute weniger im Bereich der System- und Sicherheitsanalysen gegeben als viel mehr im Bereich des Testens. Eine Zusammenfassung von Verfahren und Methoden aus dem Bereich der Testautomatisierung, wie z.B. Hardware-in-the-Loop Tests oder Hardware-Fehlerinjektionen, schließen deshalb die Ausführungen ab.

### 4.1 Methoden der System- und Sicherheitsanalyse

Die bestehenden Methoden der System- und Sicherheitsanalysen können unter verschiedenen Aspekten in Gruppen unterteilt werden. Eine mögliche Klassifizierung der Methoden kann nach ihrem Anwendungszweck erfolgen. Die identifizierenden Verfahren haben zum Ziel, mögliche Fehlerquellen und deren Auswirkungen zu bestimmen. Im Unterschied zu den identifizierenden Analysen setzen die bewertenden Methoden die Kenntnis aller Fehlerquellen voraus. Sie ermöglichen jedoch auf Basis von qualitativen oder quantitativen Analysen die Bewertung der einzelnen Fehlermöglichkeiten hinsichtlich ihrer Auftretswahrscheinlichkeit oder ihrer Tragweite für das Systemverhalten.

Des Weiteren lassen sich die bestehenden Konzepte auch nach ihrer Vorgehensweise einteilen. Während die induktiven Verfahren nach dem Bottom-Up-Prinzip, ausgehend von den einzelnen Fehlerquellen in Richtung Fehlerauswirkungen arbeiten, gehen die deduktiven Methoden mit dem Top-Down-Ansatz von einem unerwünschten Systemereignis aus und analysieren die dazu notwendigen Fehlerquellen.

Das dritte und letzte Unterscheidungskriterium ist die Zielsetzung der Analyse. Die Verfügbarkeitsanalysen setzen ein zum Zeitpunkt der Inbetriebnahme korrekt arbeitendes System voraus. Lediglich Ausfälle können zu Unkorrektheiten führen. Die Wahrscheinlichkeiten für das Auftreten solcher Ausfälle bzw. deren Auswirkungen sind Ziel der Analyse. Bei der Betrachtung der Sicherheit geht es hingegen um die Vermeidung der Gefahr, die, sowohl im Normal- als auch im Fehlerfall, von dem System für Mensch und Umwelt ausgehen.

Tabelle 4.1 bietet eine Übersicht zu den bestehenden Verfahren der System- und Sicherheitsanalyse. Die am häufigsten zum Einsatz kommenden Methoden werden im Folgenden näher erläutert.

Tabelle 4.1: Übersicht über bestehende Sicherheits- und Verfügbarkeitsanalysen

Anwendungszweck	Ziel	Arbeitsprinzip	Bestehende Methode
Identifikation von Gefahrenquellen	Vollständigkeit des Sicherheitskonzepts	Vermittlung von Denkanstößen	Anwendung von Checklisten
			Matrixdarstellung von Wechselwirkungen
		Verwendung von Suchhilfen und tabellarische Dokumentation	Fehlermöglichkeits- und Einflussanalyse (FMEA)
			HAZOP / PAAG
Rechnerbasierte Auswertung bestehenden Wissens	Bedienungsfehleranalyse		
Bewertung von Gefahrenquellen nach ihrer Auftrittswahrscheinlichkeit	Optimierung von Sicherheitssystemen hinsichtlich Zuverlässigkeit und Verfügbarkeit	Graphische Darstellung von Fehlerverknüpfungen und Wahrscheinlichkeitsbewertung	Wissensbasierte Systeme
			Ereignisbaumanalyse (ETA)
			Fehlerbaumanalyse (FTA)
Bewertung von Gefahrenquellen nach ihrer Tragweite	Minimierung des Gefahrenpotentials und optimale Planung von Schutzmaßnahmen	Mathematische Analyse physikalischer Vorgänge	Markov-Analyse
			Störungs-Auswirkungs-Analyse (CCA)

### 4.1.1 HAZOP-Verfahren

Die HAZOP-Analyse (engl.: hazard and operability studies) stammt ursprünglich aus der Chemischen Industrie zur Analyse verfahrenstechnischer Anlagen. Sie basiert auf dem Grundgedanken, dass Gefahren ausschließlich durch Abweichungen von Systemanforderungen oder vorgeschriebenen Bedieneingriffen verursacht werden.

#### 4.1.1.1 Aufbau und Durchführung der HAZOP-Analyse

Bei der Durchführung des Verfahrens erfasst ein interdisziplinäres Team im Detail alle möglichen und denkbaren Abweichungen des Systemablaufs vom nominalen Betrieb. Um die Gefahr zu minimieren, dass dabei etwas übersehen wird, ist die Vorgehensweise im HAZOP-Verfahren systematisch und formell angelegt. Zunächst wird das System nach und nach in seine Bestandteile zerlegt und die Beziehungen (z.B. Druck, Fluss, Geschwindigkeit oder Spannung) zwischen den einzelnen Komponenten erfasst. Alle Elemente des Systems und deren mögliche Abweichungen vom gewünschten Verhalten werden anhand einer Reihe von Leitwörtern untersucht. Die beim HAZOP-Verfahren angewendeten Leitworte sind in Tabelle 4.2 zusammengestellt.

Tabelle 4.2: Leitworte der HAZOP-Analyse

Leitwort	Bedeutung
Nein / Nicht / Kein	Die erwünschte Funktionalität ist nicht erfüllt und es treten keine Nebeneffekte auf
Mehr	Überschreitung eines Wertebereichs durch einen Parameter
Weniger	Unterschreitung eines vorgegebenen Minimalwertes durch einen Parameter
Sowohl / Als auch	Gewünschte Funktionalität wird erfüllt, dennoch gibt es Nebeneffekte
Teilweise	Die erwünschte Funktionalität wird bis zu einem bestimmten Grad erbracht
Umkehrung	Die Umkehrung der gewünschten Funktionalität tritt auf
Anders als	Die erwünschte Funktionalität wird nicht erfüllt und es treten Nebeneffekte auf

Das Experten-Team stellt sich anhand der erfassten Systemstruktur und der Leitwortliste Fragen nach den lokalen und globalen Auswirkungen der Abweichungen. Bei der Beantwortung

dieser Fragen analysieren die Experten das gesamte System und entdecken unerwünschte Abweichungen und die damit verbundenen Gefahrenquellen. Alle Ergebnisse werden in Formblättern dokumentiert. Die Analysemethode ist durchaus geeignet bereits in frühen Phasen der Entwicklung zum Einsatz zu kommen.

#### **4.1.1.2 Bewertung der HAZOP-Analyse**

Die Stärken des HAZOP-Verfahrens liegen vor allem in dessen einfacher Anwendbarkeit und der sehr frühzeitigen Einsetzbarkeit, um schon während der Design Phase eine Identifizierung von Gefahrenquellen und -potential zu ermöglichen. Der Einsatz der HAZOP-Methode wird jedoch als sehr zeitintensiv und aufwändig bewertet [Bis90]. Die Qualität der Analyseergebnisse hängt, wie bei allen Brainstorming basierten Methoden, sehr stark vom Fachwissen und der Erfahrung der Anwender ab.

### **4.1.2 Fehlermöglichkeits- und Einflussanalyse**

Die Fehlermöglichkeits- und Einflussanalyse (FMEA) (engl.: failure mode and effect analysis) stellt eine systematische Vorgehensweise bei der Untersuchung von sicherheitsrelevanten Systemen dar. Schon im Entwicklungsstadium erlaubt es die FMEA, die Fehlerquellen und deren Auswirkungen auf das System zu identifizieren. Die FMEA gehört damit zur Gruppe der identifizierenden Methoden und ist in der Norm zur Ausfalleffektanalyse [DIN85b] standardisiert. Sie kommt neben vielen anderen Industriezweigen auch im Automobilbereich zum Einsatz [Ver96].

Das Ziel einer FMEA ist es, alle möglichen Fehler, Schwachstellen und Mängel, die ein Produkt hat, haben oder noch bekommen könnte, im Vorfeld zu entdecken und zu beseitigen. Die FMEA kann sowohl dem Entwickler als auch dem Auftraggeber die Gewissheit vermitteln, dass bei der Realisierung eines Produkts eine etablierte Methodik angewendet wurde, um mögliche Fehler zu entdecken, zu bewerten und gegebenenfalls zu behandeln.

Man unterscheidet je nach Aufgabenstellung und Einsatzzeitpunkt zwischen: System-FMEA, Konstruktions-FMEA und Prozess-FMEA. Die System-FMEA wird bei der Konzepterstellung und Entwurfsphase angewendet. Mit einer System-FMEA werden die Fehlermöglichkeiten der einzelnen Systemkomponenten identifiziert und das Zusammenwirken der Komponenten hinsichtlich ihrer Funktionstüchtigkeit und Wechselwirkungen untersucht. Die Konstruktions-FMEA wird in der Regel parallel zur Entwicklung eines Produktes durchgeführt und untersucht alle möglichen Auslegungsfehler in der Konstruktion von Komponenten einschließlich deren Auswirkungen. Die Prozess-FMEA wird im allgemeinen während der Vorbereitungsphase der

Fertigung eines Produktes angewendet. Sie untersucht, aufbauend auf die vorangegangenen FMEA-Schritte, die Prozessplanung und -ausführung im Fertigungsprozess.

#### 4.1.2.1 Aufbau und Durchführung der System-FMEA

Die Erstellung einer System-FMEA wird in die fünf Schritte: Strukturanalyse, Funktionsanalyse, Fehleranalyse, Risikoabschätzung und Konzeptoptimierung eingeteilt. Den Ablauf einer System-FMEA zeigt Bild 4.1.

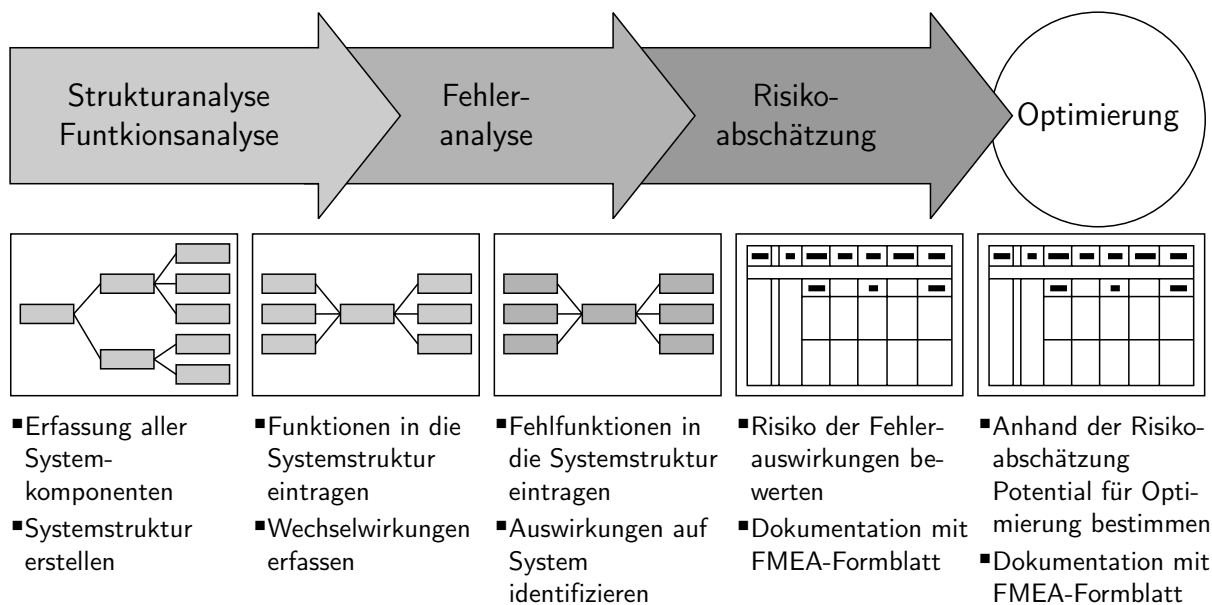


Bild 4.1: Ablauf einer System-FMEA [Deu01]

Zu Beginn der Strukturanalyse werden zunächst die Systemgrenzen festgelegt und anschließend die Systembestandteile bestimmt. Diese Zerlegung erfolgt hierarchisch in Form einer Baumstruktur. Der Grad der Zerlegung hat erheblichen Einfluss auf die Genauigkeit und den Aufwand zur Erstellung einer FMEA.

Das zu entwickelnde System muss bestimmte Anforderungen bzw. Funktionen erfüllen. Um diese Systemfunktionen zu gewährleisten, müssen die in der Strukturanalyse identifizierten Systemkomponenten ihrerseits entsprechende Teilfunktionen erbringen. Die systematische Zuordnung der Funktionen zu den Systemkomponenten und die Beschreibung der dazu notwendigen Schnittstellen zwischen den Komponenten wird Funktionsanalyse genannt.

Die Nichterfüllung oder die teilweise Erfüllung von Funktionen ist genau die Art von Fehler, die es gilt aufzufinden und zu analysieren. Die systematische und zielgerichtete Vorgehensweise in den vorangegangenen Analyseschritten gewährleistet die Vollständigkeit der Fehleranalyse.

Nach erfolgter Identifikation aller Fehlermöglichkeiten einer Systemfunktion, müssen die Auswirkungen jedes einzelnen Fehlers untersucht werden. Dabei wirken sich Fehler grundsätzlich nur auf die im Strukturbaum höheren Ebenen aus.

Die letzten Schritte bei der Durchführung einer System-FMEA bestehen in der Risikoabschätzung und einer eventuellen Optimierung des Systemkonzepts. Dabei wird das Risiko eines Fehlers und seiner Fehlerfolgen nach den Kriterien: Bedeutung, Auftritts- und Entdeckungswahrscheinlichkeit bewertet. Es werden allerdings nicht die tatsächlichen Wahrscheinlichkeiten bestimmt, sondern die Beurteilungen erfolgen in Abstufungen zwischen den Werten 1 und 10. Die Bewertung erfolgt damit quasi quantitativ, ist somit vergleichbar und erlaubt die Bildung einer Rangfolge für die Systemoptimierung.

Die gesamte Analyse wird in mehreren Brainstorming-Sitzungen unter Einbeziehung von Experten durchgeführt. Das so genannte FMEA-Formblatt ist dabei zum einen ein Leitfaden für das FMEA-Expertenteam, um eine systematische und vollständige Vorgehensweise zu garantieren, und zum anderen ein Hilfsmittel zur Dokumentation der erfassten Ergebnisse. Zur Unterstützung der Experten-Teams stehen bereits einige kommerzielle Software-Werkzeuge zur Verfügung, die sowohl die Überwachung der methodischen Vorgehensweise, wie auch die Datenerfassung, -speicherung und -auswertung bieten.

#### **4.1.2.2 Bewertung der FMEA-Methode**

Die besondere Stärke der FMEA ist zweifelsfrei ihre vollständige und systematische Vorgehensweise bei der Identifikation von Fehlerquellen der Systemkomponenten. Mit steigender Systemkomplexität kann diese Stärke zugleich ein entscheidender Nachteil der FMEA werden. Sehr komplexe Systeme machen naturgemäß auch eine sehr umfangreiche und damit zeitintensive Analyse notwendig und führen oft zu nur unübersichtlich darstellbaren Ergebnissen [PPG04]. Da die FMEA auf Brainstorming von menschlichen Experten basiert, ist eine Automatisierung der Analyse durch rechnergestützte Software-Werkzeuge und damit eine Reduktion des zur Erstellung notwendigen Zeitaufwands nur selten zu realisieren. Verschiedene Studien haben gezeigt, dass aufgrund der Vorgehensweise die Qualität der FMEA-Ergebnisse sehr stark schwankt und von vielen Einflussfaktoren, die von der Teamzusammensetzung bis hin zur Tagesform der Experten reichen, abhängt [Lev95].

Zudem wird die Analyse für das Experten-Team in den FMEA-Sitzungen mit zunehmender Systemkomplexität immer schwieriger. Ist es theoretisch noch vorstellbar, dass alle Fehlermöglichkeiten der Einzelkomponenten identifiziert werden können, so erscheint dies in Bezug auf die Fehlerauswirkungen fast unmöglich. Allzu oft sind die Folgen eines Fehlers aufgrund der

dynamischen Eigenschaften und der zunehmenden Vernetzung von Funktion auf Systemebene nicht mehr zu überblicken. Dies führt zu einer oft mangelhaften, unvollständigen und nicht reproduzierbaren Bestimmung der Fehlerauswirkungen und deren Schwerebewertung. Damit wird klar, dass die FMEA-Methode in ihrer herkömmlichen Form nur auf Einzelfehler anwendbar ist. Die enorme Menge an möglichen Fehlerkombinationen oder gar Fehlersequenzen macht eine dies bezügliche Analyse nicht möglich.

### 4.1.3 Fehlerbaum-Analyse

Die Fehlerbaumanalyse, kurz FTA (engl.: fault tree analysis) ist eine graphische Methode zur Analyse der Zuverlässigkeit eines Systems [Sch99]. Ziel ist es, ausgehend von einem so genannten Top-Ereignis den Zusammenhang mit den jeweiligen Ursachen aufzuzeigen.

#### 4.1.3.1 Aufbau und Durchführung der Fehlerbaumanalyse

In der Regel handelt es sich bei dem Top-Ereignis, das an der Spitze des Graphen steht, um einen unerwünschten Vorgang bzw. eine Gefahr. Die hierarchisch untergeordneten Ebenen repräsentieren die Ursachen bzw. Kombinationen von Ursachen, die zu diesem Top-Ereignis führen können. Die Verknüpfung mehrerer unabhängiger Fehlerursachen erfolgt mit Hilfe boolescher Operatoren. Die Ursachenanalyse kann so lange fortgeführt werden, bis an den Enden des Graphen nur noch Elementarereignisse stehen. Die so entwickelte Baumstruktur wird Fehlerbaum genannt und ist in Bild 4.2 beispielhaft dargestellt. Die Vorgehensweise zur Erstellung eines Fehlerbaums wird auf Grund ihres Top-Down Ansatzes als deduktiv bezeichnet und ist in [DIN81] standardisiert.

Ist es möglich, den Elementarereignissen Auftrittswahrscheinlichkeiten zuzuordnen, so lässt sich unter Anwendung der Booleschen Algebra die Auftrittswahrscheinlichkeit des Top-Ereignisses angeben. Die quantitative Vorgehensweise erlaubt einerseits die Bewertung der Zuverlässigkeit eines Systems und andererseits eine konkrete Bestimmung von Verbesserungspotentialen und -maßnahmen.

Falls eine Fehlerbaumanalyse mangels genauer Kenntnis der entsprechenden Ausfallwahrscheinlichkeiten nicht quantitativ durchgeführt werden kann, so ist sie doch ein wertvolles Werkzeug bei der Systemanalyse. Bei der rein qualitativen Auswertung des Fehlerbaums werden insbesondere die so genannten Minimalschnitte ermittelt. Ein Schnitt ist definiert als eine Menge von Ereignissen, die zum Top-Ereignis führen [MP03]. Aus dieser Menge leiten sich die Minimalschnitte ab, die nur die unbedingt notwendige Anzahl von Ereignissen zum Erreichen



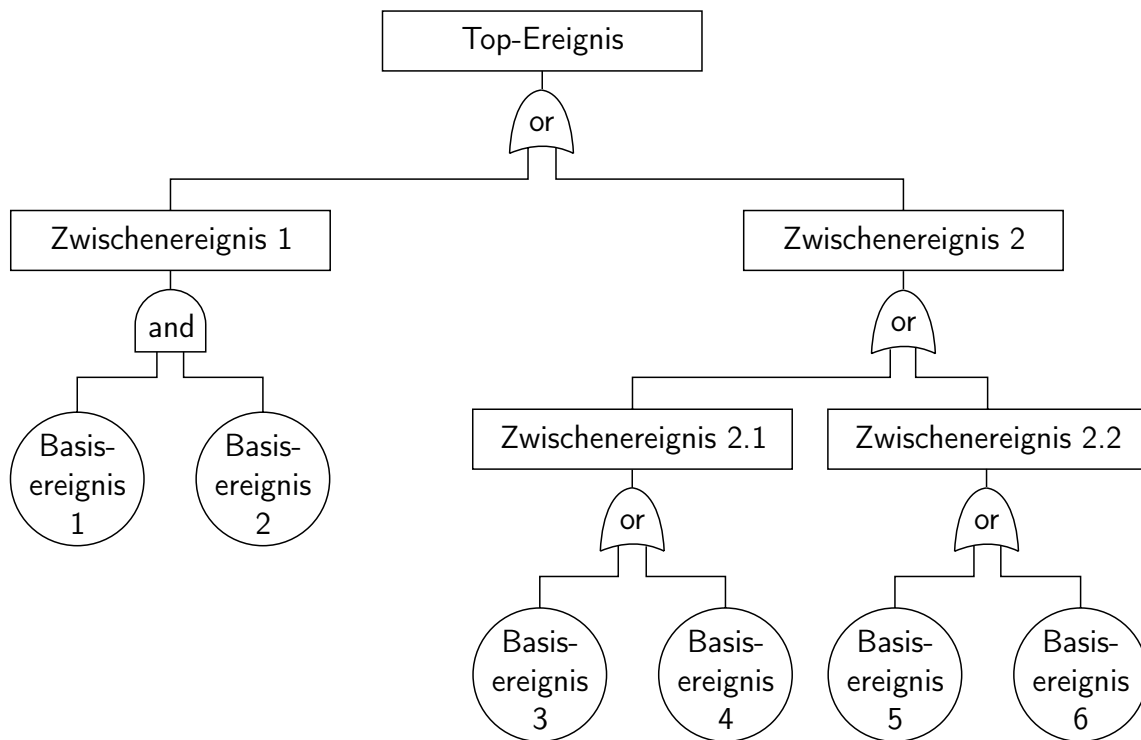


Bild 4.2: Beispielhafte Darstellung eines Fehlerbaums

des Top-Ereignisses enthalten. Der kritische Pfad ist der Minimalschnitt mit den wenigsten Elementarereignissen und stellt damit die Schwachstelle im System dar.

#### 4.1.3.2 Bewertung der Fehlerbaumanalyse

Obwohl die Fehlerbaumanalyse ursprünglich zur Berechnung der quantitativen Auftretswahrscheinlichkeiten entwickelt wurde, wird sie heute mehr und mehr zur qualitativen Analyse verwendet. Dies ist hauptsächlich in der mangelhaften Verfügbarkeit von verlässlichen Auftretswahrscheinlichkeiten für die Basisereignisse begründet [Lev95]. Aber auch qualitative Fehlerbäume und speziell die Minimalschnitte haben sich als geeignete Hilfen für den Systemanalysten bei der Identifizierung von Gefahren, Schwachstellen und Verbesserungspotential erwiesen.

Bei der Durchführung einer Fehlerbaumanalyse können allerdings nur die Top-Ereignisse berücksichtigt werden, die bereits als solche bekannt sind. Zur systematischen und vollständigen Bestimmung dieser unerwünschten Ereignisse sind zusätzliche vorgelagerte Systemanalysen, wie z.B. die FMEA, erforderlich.

Das komplexe Zusammenspiel zwischen allen Bestandteilen des Fahrzeugsystems muss dem Systemanalysten bei der Erstellung des Fehlerbaums genau bekannt sein. Darüberhinaus muss er in

der Lage sein, dieses Verhalten mit den vorgesehenen Modellierungsmöglichkeiten abzubilden. Dies fällt aufgrund der nur sehr einfachen Verknüpfungsmöglichkeiten mit Hilfe von booleschen Logikfunktionen vor allem bei dynamischen Vorgängen schwer [Bis90]. Deshalb wird bei einem Fehlerbaum häufig von einer Momentaufnahme des Systemzustands zu einem ganz bestimmten Zeitpunkt gesprochen.

#### 4.1.4 ETA-Verfahren

Während die Fehlerbaumanalyse Ursachen für ein bestimmtes Ereignis ermittelt, verfolgt die Ereignisablaufanalyse (engl.: event tree analysis) exakt den umgekehrten Analyseweg. Für ein spezielles Ereignis werden die Auswirkungen bzw. die Folgen analysiert. Die Vorgehensweise bei der Erstellung einer Ereignisablaufanalyse ist in [DIN85a] standardisiert.

##### 4.1.4.1 Aufbau und Durchführung einer ETA-Analyse

Die Ergebnisse einer ETA-Analyse werden in einem so genannten Ereignisablaufdiagramm dargestellt. Ein Beispieldiagramm zeigt Bild 4.3.

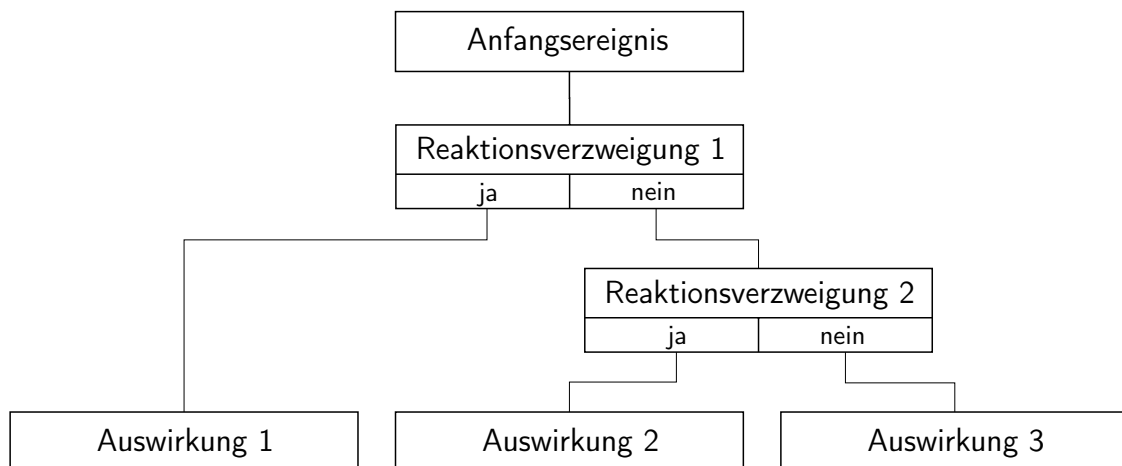


Bild 4.3: Beispielhafte Darstellung eines Ereignisablaufdiagramms

Ausgehend von einem Anfangsereignis, das z.B. ein Ausfall einer Systemkomponente sein kann, werden dessen Auswirkungen auf das betrachtete System ermittelt. Die Auswirkungen werden im Ereignisablaufdiagramm durch Linien symbolisiert und rufen ihrerseits eventuell weitere kausale Folgewirkungen hervor. Die Folgen auf weitere Systemelemente werden in Form von Verzweigungen dargestellt und bilden so das Eingangsereignis für das nächste Systemelement.

Die Analyse wird so lange fortgeführt, bis alle Systemkomponenten auf deren Folgewirkungen abgefragt wurden.

Soll statt der bisher rein qualitativen Betrachtung eine quantitative Analyse des Systems erfolgen, so kann für jeden Zweig im Ereignisablaufdiagramm eine Wahrscheinlichkeitsbewertung vorgenommen werden. Die Berechnung der Wahrscheinlichkeit eines Pfades erfolgt dann über die Multiplikation aller Wahrscheinlichkeitswerte der betroffenen Zweige.

#### **4.1.4.2 Bewertung der Ereignisablaufanalyse**

Trotz der unterschiedlichen Zielsetzungen von ETA und FTA-Analyse sind die zur Erstellung notwendigen Schritte sehr ähnlich. Die Baumstruktur des Ereignisablaufdiagramms erleichtert auch bei der ETA-Analyse die eigentliche Erstellung und die anschließende Auswertung der Analyseergebnisse. Eine Berechnung von Auftrittswahrscheinlichkeiten der sicherheitsrelevanten Ereignisse kann auf Basis des Ereignisablaufdiagramms bei Kenntnis aller Wahrscheinlichkeiten der Einzelereignisse ohne große Schwierigkeiten erfolgen.

Deutlich schwieriger gestaltet sich allerdings die korrekte und vor allem vollständige Erfassung des Ereignisablaufs. Die meist sehr komplexen Interaktionen der Systemkomponenten untereinander als auch deren Auswirkungen im Fehlerfall müssen dem Experten-Team bekannt und mit Hilfe der Baumstruktur modellierbar sein. Dies fällt insbesondere bei Abhängigkeiten oder Kombinationen einzelner Fehlermöglichkeiten und bei Fehlern gemeinsamer Ursache in den meisten Fällen schwer.

### **4.1.5 Markov-Analyse**

Die Markov-Analyse ist eine mathematische Vorgehensweise zur Ermittlung der Aufenthaltswahrscheinlichkeit eines betrachteten Systems in einem bestimmten Systemzustand. Dabei ist es möglich sowohl dynamische Vorgänge als auch Reparaturen bei einer Markov-Analyse zu berücksichtigen.

#### **4.1.5.1 Aufbau und Durchführung von Markov-Analysen**

Die Markov-Analyse basiert auf dem gewöhnlichen Markov-Prozess, einem stochastischen Prozess  $Z(t)$  mit endlich vielen Zuständen  $Z_1, Z_2, \dots, Z_n$ . Die Übergänge zwischen den einzelnen Zuständen sind nur vom gegenwärtig eingenommenen Systemzustand und von der Zeit  $t$  abhängig. Im einfachsten Fall werden bei der Markov-Analyse Systeme betrachtet, deren Komponenten

konstante Ausfall- und Reparaturraten aufweisen und damit auch über konstante Übergangsraten  $\alpha_{ij}$  von Zustand  $Z_i$  in den Zustand  $Z_j$  verfügen. Wird ein System mit  $m$  Komponenten und  $2^m$  Zuständen der Analyse zu Grunde gelegt, so ergeben sich, für die Wahrscheinlichkeit  $P_i(t)$  zum Zeitpunkt  $t$  im Zustand  $Z_i$  zu sein,  $n$  Differentialgleichungen der Form

$$\frac{dP_i(t)}{dt} = - \sum_{j=1, j \neq i}^n \alpha_{ij} P_i(t) + \sum_{j=1, j \neq i}^n \alpha_{ji} P_j(t) \quad \forall \quad i = 1..n \quad (4.1)$$

mit der Normierungsbedingung

$$\sum_{i=1}^n P_i(t) = 1. \quad (4.2)$$

Gleichung 4.1 lässt sich einfacher in Matrixschreibweise darstellen:

$$\frac{d\mathbf{P}(t)}{dt} = \mathbf{A} \cdot \mathbf{P}(t) \quad (4.3)$$

Die Matrix  $\mathbf{A}$  besteht aus den Übergangsraten  $\alpha_{ij}$  zwischen den einzelnen Zuständen, wobei die Summe einer Spalte der Matrix immer gleich 0 sein muss. Es gilt also:

$$\mathbf{A} = \begin{pmatrix} \beta_1 & \alpha_{21} & \alpha_{31} & \cdots & \alpha_{n1} \\ \alpha_{12} & \beta_2 & \alpha_{32} & \cdots & \alpha_{n2} \\ \alpha_{13} & \alpha_{23} & \beta_3 & \cdots & \alpha_{n3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \alpha_{1n} & \alpha_{2n} & \alpha_{3n} & \cdots & \beta_n \end{pmatrix} \quad (4.4)$$

mit der Bedingung

$$\beta_i = \sum_{k=1, k \neq i}^n \alpha_{ik}. \quad (4.5)$$

Zur Analyse der Sicherheit eines Systems ist es nun möglich,  $\Gamma_S$  als die Menge aller Zustände zu definieren, in denen die Systemfunktionalität keinen sicherheitsrelevanten Einschränkungen unterliegt. Die Gesamtwahrscheinlichkeit für einen sicheren Betrieb des betrachteten Systems ergibt sich dann zu:

$$P_S(t) = \sum_{Z_i \in \Gamma_S} P_i(t). \quad (4.6)$$

Ebenso kann verfahren werden, um die Verfügbarkeit eines Systems zu ermitteln. Sei  $\Gamma_V$  die Menge aller Zustände, in der das System funktionsfähig ist, so kann die Verfügbarkeit wie folgt dargestellt werden:

$$P_V(t) = \sum_{Z_i \in \Gamma_V} P_i(t). \quad (4.7)$$

#### 4.1.5.2 Bewertung der Markov-Analysen

Mithilfe der Markov-Analyse kann bereits in einem sehr frühen Stadium des Entwicklungsprozesses eine erste quantitative Aussage zur Auftrittswahrscheinlichkeit eines sicherheitsrelevanten bzw. unverfügbaren Systemzustands gemacht werden. Damit sind zu einem sehr frühen Zeitpunkt Architektur- und Konzeptoptimierungen möglich.

Diese offensichtlich vorteilhafte Eigenschaft der Markov-Analyse wird allerdings zumeist durch die zu diesem Zeitpunkt zumeist mangelhafte Kenntnis der exakten Ausfall- und Reparaturraten kompensiert. Wie bereits erwähnt, sind die Zustandsübergänge ausschließlich vom gegenwärtigen Zustand und von der Zeit abhängig. Vorhergehende Zustände bzw. Zustandsübergänge können keinen Einfluss auf das spätere Übergangsverhalten nehmen. Um trotzdem mit Systemarchitekturen und -konzepten umgehen zu können, die exakt diese Eigenschaft aufweisen, ist eine besondere Vorgehensweise bei der Modellierung zu befolgen. Jeder Übergang, der von vorhergehenden Zuständen abhängt, muss in einem separaten Pfad mit eben diesen Zuständen modelliert werden. Die Anzahl der Zustände und Übergänge steigt dadurch jedoch an.

Die Markov-Gleichungen sind in ihrer Grundform zudem nicht zur Berücksichtigung von zeitabhängigen Übergangsraten vorgesehen. Damit ist es nicht möglich, Alterungsprozesse oder Ermüdungserscheinungen in die Analyse miteinzubeziehen. Dies gelingt nur mithilfe von Semi-Markov-Prozessen, die jedoch mathematisch sehr viel komplexer beschreibbar sind. Daher sei an dieser Stelle auf die entsprechende Fachliteratur verwiesen [MP03].

#### 4.1.6 Formale Methoden

Im Zusammenhang mit der Sicherheit, insbesondere mit deren Nachweis, werden häufig die so genannten formalen Methoden erwähnt. Ihr Konzept weicht jedoch in Bezug auf dessen Zielsetzung deutlich von den bisher vorgestellten Verfahren ab. Stand bisher das Auffinden bzw. die Bewertung von Gefahrenquellen im Vordergrund der Analysen, so wird mit den formalen Methoden das Ziel verfolgt, aus gegebenen Sicherheitsanforderungen eine vollständige, konsistente, eindeutige und korrekte Systemspezifikation zu erstellen und deren nachweisbar fehlerfreie Realisierung zu gewährleisten.

##### 4.1.6.1 Aufbau und Durchführung formaler Methoden

Die Vorgehensweise der formalen Methoden basiert auf dem Vergleich einer Systembeschreibung und der Beschreibung des tatsächlich implementierten Systemverhaltens. Werden diese

Beschreibungen in natürlicher Sprache verfasst, so eröffnet sich eine Vielzahl an Interpretationsspielräumen. Dies birgt die Gefahr von Missverständnissen, Inkonsistenzen und Realisierungsfehlern. Deshalb gründen die formalen Methoden auf formalen Sprachen mit einer präzise festgelegten Syntax und Semantik. Die erlangten Beschreibungen für die Systemspezifikation zum einen und das aus dem Entwurf abgeleitete Systemverhalten zum anderen, ermöglichen eine mathematische Beweisführung im Sinne einer Überprüfung auf deren Konsistenz. Dieser Beweis wird auch formale Verifikation genannt.

Neben der reinen Beschreibung ist zur Beweisführung darüber hinaus ein mathematisches Modell von Nöten, das die Regeln für die logischen Verknüpfungen der einzelnen Beschreibungen definiert. Ein solches Modell wird Kalkül oder Logik genannt. Zu den wichtigsten Vertretern gehören die Aussagen-, Prädikaten- und Temporallogik [Moi01, Sch05].

Als ein Teil der Temporallogik wird im Folgenden beispielhaft auf die Computation Tree Logic (CTL) eingegangen. Sie verfügt über eine formale Sprache mit einer festen Menge an definierten Operatoren:

<b>X</b> .. next time	<b>E</b> .. exists
<b>F</b> .. in the future	<b>A</b> .. always
<b>G</b> .. globally	
<b>U</b> .. until	

Als Anwendungsbeispiel dient ein sehr einfaches System bestehend aus einem Behälter mit Ein- und Auslass-Ventil. Ein Steuerprogramm regelt die Öffnung der einzelnen Ventile. Dabei soll für alle möglichen Zustände garantiert sein, dass die Steuerung nach der Öffnung des Einlass-Ventils zu einem beliebigen späteren Zeitpunkt ebenfalls das Auslass-Ventil öffnet. Diese Anforderung kann mithilfe der CTL in einer formalen Spezifikation der Form:

$$\mathbf{AG}(p=\text{Einlass offen} \Rightarrow \mathbf{AF} q=\text{Auslass offen}) \quad (4.8)$$

ausgedrückt werden.

Eine mögliche Realisierung dieses Systems ist in Bild 4.4 dargestellt. Im rechten Teil der Abbildung ist das Verhalten in Form eines Baumes ausgehend von Zustand 1 dargestellt. Anhand dieser Visualisierung lässt sich leicht nachvollziehen, dass bei der Implementierung die Spezifikation eingehalten wurde. Nach jedem Öffnen des Einlass-Ventils erfolgt die Öffnung des Auslass-Ventils.

Durch die eindeutige, mathematische Formulierung kann mit Hilfe so genannter Model-Checking-Verfahren [Sch05] die Einhaltung der Spezifikation formal bewiesen werden. Dieser Vorgang ist dann die formale Verifikation der Realisierung gegenüber der Spezifikation.

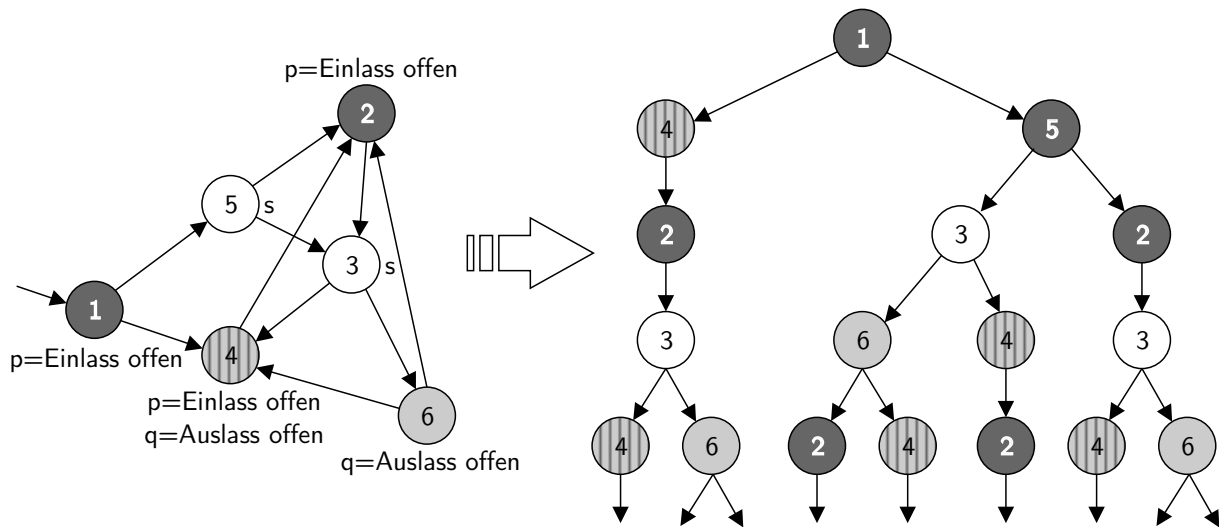


Bild 4.4: Beispiel-Netz mit resultierendem Verhaltensbaum ausgehend von Zustand 1

#### 4.1.6.2 Bewertung der formalen Methoden

Der Einsatz von formalen Methoden gewährleistet eine mathematisch korrekte Überprüfung einer Systemimplementierung auf deren Sicherheitsanforderungen hin. Die dazu notwendige Systembeschreibung in komplexen mathematischen Sprachen erscheint aus heutiger Sicht schwierig bzw. nicht „ingenieurgerecht“ [Moi01]. Dies führt im Vergleich zu den anderen vorgestellten Verfahren leichter zu Spezifikations- und Modellierungsfehlern. Der Aufwand für eine formale Verifikation ist schon bei relativ kleinen, einfachen Systemen enorm hoch. Die ganzheitliche Anwendung auf komplexe, vernetzte und dynamische Systeme wie sie heute im Automobilbereich zum Einsatz kommen, erscheint damit nicht möglich. Der entscheidende Nachteil der formalen Methoden besteht aber darin, dass es für eine ganzheitliche Sicherheitsanalyse nicht ausreicht, die formale Korrektheit einer Spezifikation und deren konsistente Umsetzung in ein Systemverhalten nachzuweisen, da viele sicherheitsrelevante Fehler bereits in den Anforderungen enthalten sind [Lev95].

#### 4.1.7 Zusammenfassung der diskutierten Methoden

Aus der vorangegangenen Zusammenfassung der Methoden zur System- und Sicherheitsanalyse geht hervor, dass keines der Verfahren allein für eine ganzheitliche Analyse geeignet ist. Nur durch die kombinierte Anwendung verschiedener Methoden ist eine Gesamtbetrachtung vom Fahrerverhalten, über das Fahrzeugsystem selbst, bis hin zum Fahrzeug möglich. Die hochkomplexen Zusammenhänge und Abhängigkeiten zwischen den Bestandteilen des Kraftfahrzeugsystems, seiner Umgebung und dem Menschen können auf diese Weise nur unzureichend

erfasst werden. Insbesondere Fehler, die sich aufgrund der komplexen Strukturen im ganzen System ausbreiten und dessen Funktion grundlegend beeinträchtigen können, oder aber Kombinationen von Fehlern, die in den unterschiedlichen Systembestandteilen auftreten, stellen die Hauptursache vieler schwerer Unfälle dar [Lev95].

Diese Problematik verschärft sich zusätzlich durch die Tatsache, dass bei den meisten Verfahren Brainstorming als Analysetechnik zum Einsatz kommt. Die Komplexität vernetzter Systeme mit immer größeren Software-Anteilen übersteigt jedoch die Leistungsfähigkeit der menschlichen Experten im Brainstorming-Prozess. Deshalb müssen im Laufe einer Analyse Abstraktionen hingenommen, der Betrachtungsumfang reduziert und das Untersuchungsziel auf ausgesuchte Zeitpunkte beschränkt werden.

Diese Einschränkungen spiegeln sich auch in den Systemmodellen wider, die im Laufe der Analyse der Sicherheit herangezogen werden. Wie bereits dargelegt, kommen bei der Mehrzahl der klassischen Methoden zur Sicherheitsanalyse Graphen und Baumstrukturen zur Modellierung der Systemabläufe zum Einsatz. Sie sind aber meist weniger Hilfsmittel bei der Durchführung der Sicherheitsanalyse als vielmehr Dokumentationsmittel zur Darstellung der erlangten Ergebnisse. Zudem sind ihre statischen Strukturen und ihre auf diskrete Zustände begrenzten Modellelemente nicht zur Nachbildung dynamischer Abläufe geeignet.

Wünschenswert ist hingegen, ein Modellierungsmittel schon bei der Erstellung der Analyse einzusetzen, um mit dessen Hilfe zuverlässige und realitätsnahe Aussagen über die dynamische Interaktion der Systembestandteile zu erhalten. Aus diesem Grund stellen sich folgende Fragen: Welche Modellierungskonzepte sind geeignet, die Erstellung der Sicherheitsanalyse zu unterstützen? Welche Modellierungsarten und -techniken gibt es und welche werden davon den im vorherigen Kapitel definierten Anforderungen gerecht?

Zur Beantwortung dieser Fragen wird im nächsten Abschnitt auf unterschiedliche Modellierungsarten eingegangen, deren Eigenschaften und Zielsetzungen erörtert und deren Eignung für Sicherheitsanalysen und dynamische Systembeschreibungen analysiert.

## 4.2 Modellierungskonzepte

Grundsätzlich ist ein Modell eine Abbildung der Wirklichkeit. Grundlage für die Erstellung eines Modells bildet das Modellierungskonzept. Es definiert Modellelemente, mit denen die Eigenschaften der Realität abgebildet werden und legt die Modellierungstechnik fest, die die Art und Weise der Verknüpfung der einzelnen Modellelemente beschreibt. Die im folgenden beschriebenen, bekannten Modellierungskonzepte lassen sich grob in vier Gruppen gliedern [LG99]:



**Prozessorientierte Modellierungskonzepte** sind dadurch gekennzeichnet, dass die in der Realität ablaufenden Prozesse modelliert werden. In den häufigsten Fällen handelt es sich dabei um Vorgänge, bei denen die Eingangsgrößen im Sinne einer mathematischen Funktion zu Ausgangsgrößen verknüpft werden.

**Produktorientierte Modellierungskonzepte** gehen nicht von den auszuführenden Vorgängen, sondern von den in diesen Vorgängen verwendeten bzw. erzeugten Produkten aus. Oft handelt es sich bei den Produkten um Informationen bzw. um Daten, weshalb auch von einem datenorientierten Modellierungskonzept gesprochen wird.

**Zustandorientierten Modellierungskonzepten** liegt die Idee zugrunde, dass sich in einer zu modellierenden Realität diskrete Zustände und deren Übergänge definieren lassen.

**Objektorientierte Modellierungskonzepte** basieren auf der Idee, Verarbeitungsprodukte und die darauf angewandten Verarbeitungsprozesse zu „gekapselten Objekten“ zusammenzufassen.

#### 4.2.1 Graphen als Modellelemente eines zustandsorientierten Konzepts

Graphen bestehen aus Knoten und Kanten. Die Knoten dienen als Platzhalter für Aktionen, Ereignisse oder Komponenten. Sie werden über Kanten miteinander verbunden. Dabei unterscheidet man gerichtete und ungerichtete Graphen. Die Baumstruktur, wie in Bild 4.5, stellt dabei einen Sonderfall eines gerichteten Graphen dar.

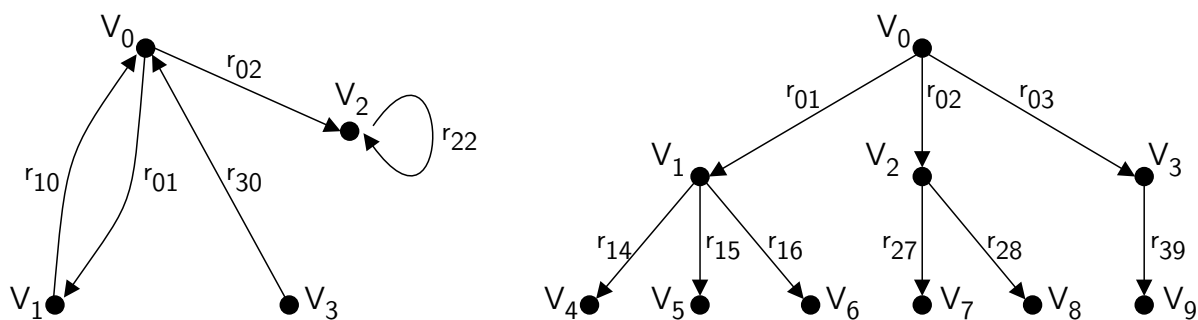


Bild 4.5: Beispiele eines gerichteten Graphen und dessen Sonderform in Baumstruktur

Bei vielen Graphen werden die Knoten und Kanten quantitativ gewichtet. Damit lassen sich unterschiedliche Wege durch den Graphen bewerten und vergleichen. Typische Beispiele dieser Art der Modellierung sind Wahrscheinlichkeitsbäume oder Markov-Ketten. Eine zusätzliche

Erweiterung stellen die so genannten Logik-Bäume dar. Sie verwenden Logikelemente, wie UND, ODER, NICHT, um Abhängigkeiten zwischen den Knoten detaillierter beschreiben zu können. Dieser Gruppe gehören z.B. die Fehlerbäume und Ereignisablaufdiagramme an.

Baumdarstellungen und Graphen eignen sich hervorragend, um Zusammenhänge strukturiert wiederzugeben. Sowohl die prinzipielle Erstellung als auch das Verständnis eines Graphen fällt dem Menschen aufgrund der beschränkten Zahl an Modellierungselementen meist recht leicht. Ein Graph ist auch formal beschreibbar, so dass Maschinen auf Basis mathematischen Algorithmen mit ihm umgehen können.

Bei der Erstellung eines Systemmodells in Form eines Graphen ist darauf zu achten, dass der Anwender das abzubildende System sehr gut kennt. Welche Systemelemente in welcher Form miteinander in Verbindung stehen, muss dem Anwender bereits bekannt sein. Das Modellierungskonzept bietet bei der Erlangung dieses Wissens keine Hilfestellung. Die begrenzte Anzahl an Modellierungselementen birgt darüber hinaus die Gefahr von Abstraktionsfehlern. Es liegt in der Verantwortung des Anwenders, zu entscheiden, wie und unter welchen Annahmen er das meist komplexe Systemverhalten auf die recht einfachen Graphen überträgt.

### 4.2.2 Zustandsautomaten

Ein Automat stellt ein mathematisches Modell einer idealen Rechenmaschine dar, die zu einer Eingabe ein bestimmtes Ergebnis ausgibt. Dabei bestehen die endlichen Automaten aus einer abzählbaren Menge an Grundelementen, die jeweils durch die drei Größen: Zustand, Ereignis und Aktion definiert sind. Ereignisse lösen einen Zustandsübergang aus und haben dabei ihrerseits die Ausführung einer Aktion zur Folge (vgl. Bild 4.6).

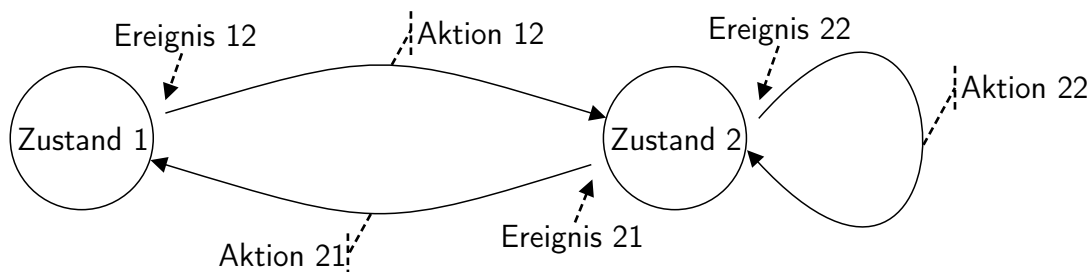


Bild 4.6: Grafische Darstellung eines Zustandsautomaten

Um den unterschiedlichen Einsatzgebieten von Zustandsautomaten gerecht zu werden, haben sich viele Varianten entwickelt. Die wichtigsten Vertreter sind die Moore-Automaten, Mealy-Automaten, Statecharts und zeitbehaftete Zustandsautomaten.

Die Automatentheorie umfasst die mathematische Grundlage zur Berechnung und Analyse von Zustandsautomaten. Ähnlich dem Einsatz von Graphen muss der Anwender das Modell gedanklich entwickelt haben und das System sehr gut kennen, bevor er Zustandsautomaten zur Beschreibung eines abzubildenden Systems einsetzen kann. Diskrete Abläufe sind besonders zur Modellierung mittels Zustandsautomaten geeignet. Kontinuierliche Prozesse können hingegen nur über den meist aufwändigen Weg einer Diskretisierung abgebildet werden. Die physikalische Struktur eines Systems lässt sich anhand von Zustandsautomaten nicht beschreiben.

### 4.2.3 Petri-Netze als Elemente eines zustandsorientierten Konzepts

Ein Petri-Netz ist ein gerichteter Graph mit zwei unterschiedlichen Arten von Knoten: Stellen und Transitionen. Die Stellen dienen zur Beschreibung von Zuständen und Bedingungen während die Transitionen in der Regel Aktionen oder Ereignisse darstellen. Kanten dürfen nur zwischen unterschiedlichen Knotenarten bestehen. Zur Beschreibung von Vorgängen in einem Petri-Netz werden die Stellen mit Marken belegt. Die Marken zeigen den aktuellen Systemzustand an und wechseln beim Eintreten von Ereignissen, dem so genannten Schalten der Transitionen, auf andere Stellen im Netz. Bild 4.7 zeigt die graphische Struktur eines Petri-Netzes.

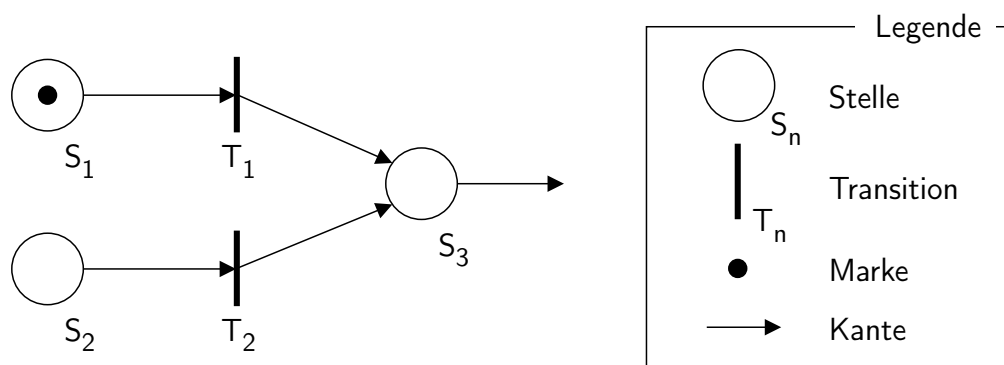


Bild 4.7: Graphische Darstellung eines Petri-Netzes

Neben dem hier dargestellten klassischen Petri-Netz gibt es eine Vielzahl von abgewandelten Modellierungskonzepten, die für spezielle Anwendungszwecke angepasst sind. Zu den wichtigsten Vertretern dieser Gruppe gehören die gewichteten, die farbigen, die zeitbewerteten Petri-Netze und die unscharfen (Fuzzy) Netze.

Petri-Netze besitzen wie Zustandsautomaten eine fundierte mathematische Beschreibung. Mit Hilfe einer Matrix, der Inzidenzmatrix, kann die gesamte Struktur eines Petri-Netzes wiedergegeben werden. Matrix-Operationen ermöglichen die Berechnung des Systemzustandes nach

beliebig vielen Schaltvorgängen. Die besondere Stärke liegt dabei in der Modellierung von Synchronisationsproblemen und der damit verbundenen Analyse der Verklemmungsproblematik (engl.: deadlock).

Ähnlich wie Zustandsautomaten eignen sich die Petri-Netze für die Modellierung von sequentiellen Abläufen, die durch diskrete Zustandsgrößen charakterisiert sind. Informationsflüsse und Regelungsaufgaben können mit Petri-Netzen hingegen nur abstrakt und eingeschränkt beschrieben werden. Keine Berücksichtigung finden strukturelle Eigenschaften des abzubildenden Systems bei der Umsetzung in ein Petri-Netz.

#### 4.2.4 CARTRONIC als objektorientierte Modellierungstechnik

CARTRONIC [BSDV97] ist ein Strukturierungskonzept für alle Steuer- und Regelungssysteme eines Kraftfahrzeugs basierend auf einem objektorientierten Ansatz. Das System Kraftfahrzeug wird dabei in seine logischen Funktionen unterteilt, die über standardisierte, funktionale Schnittstellen miteinander kommunizieren. Gemäß diesem Konzept existieren folgende Modellelemente: funktionale Komponenten als Basiselement zur Modellierung der Funktion und deren mögliche Schnittstellenbeziehungen bestehend aus Aufträgen, Anforderungen und Anfragen. In Bild 4.8 ist ein hierarchisches Kraftfahrzeug-Modell dargestellt, dem das Strukturierungskonzept CARTRONIC zugrunde liegt.

Um eine rechnergestützte Analyse des CARTRONIC-Funktionsmodells zu ermöglichen, wird es mit Hilfe der Unified Modeling Language (UML) in einem Klassendiagramm abgebildet [LFS<sup>+</sup>01]. Dabei wird jede funktionale Komponente in mindestens zwei Klassen transformiert - einem Interface und einer Klasse, die eine Realisierung dieses Interface darstellt.

Die CARTRONIC-Funktionsstruktur ist besonders für die Systembetrachtung in frühen Entwicklungsphasen konzipiert. Zu diesem Zeitpunkt steht eine abstrakte, auf die funktionalen Aspekte beschränkte Sichtweise im Vordergrund. Die tatsächliche Realisierung der geforderten Funktionalität ist noch nicht bekannt. Nicht-funktionale Systemeigenschaften, wie z.B. strukturelle Systemaspekte, sind deshalb nicht Bestandteil des Modellierungskonzepts.

In seiner Dissertation „Formale Anwendung von Sicherheitsmethoden bei der Entwicklung verteilter Systeme“ [Län03] und in [LLF<sup>+</sup>02] befasst sich Wolfgang Längst mit der Durchführung einer Sicherheitsanalyse für verteilte Kraftfahrzeugsysteme auf Basis eines CARTRONIC-Funktionsmodells. Zielsetzung ist dabei, die sicherheitsrelevanten Funktionsbereiche zu identifizieren und Unterstützung bei der Auswahl der optimalen Realisierungsform zu leisten.

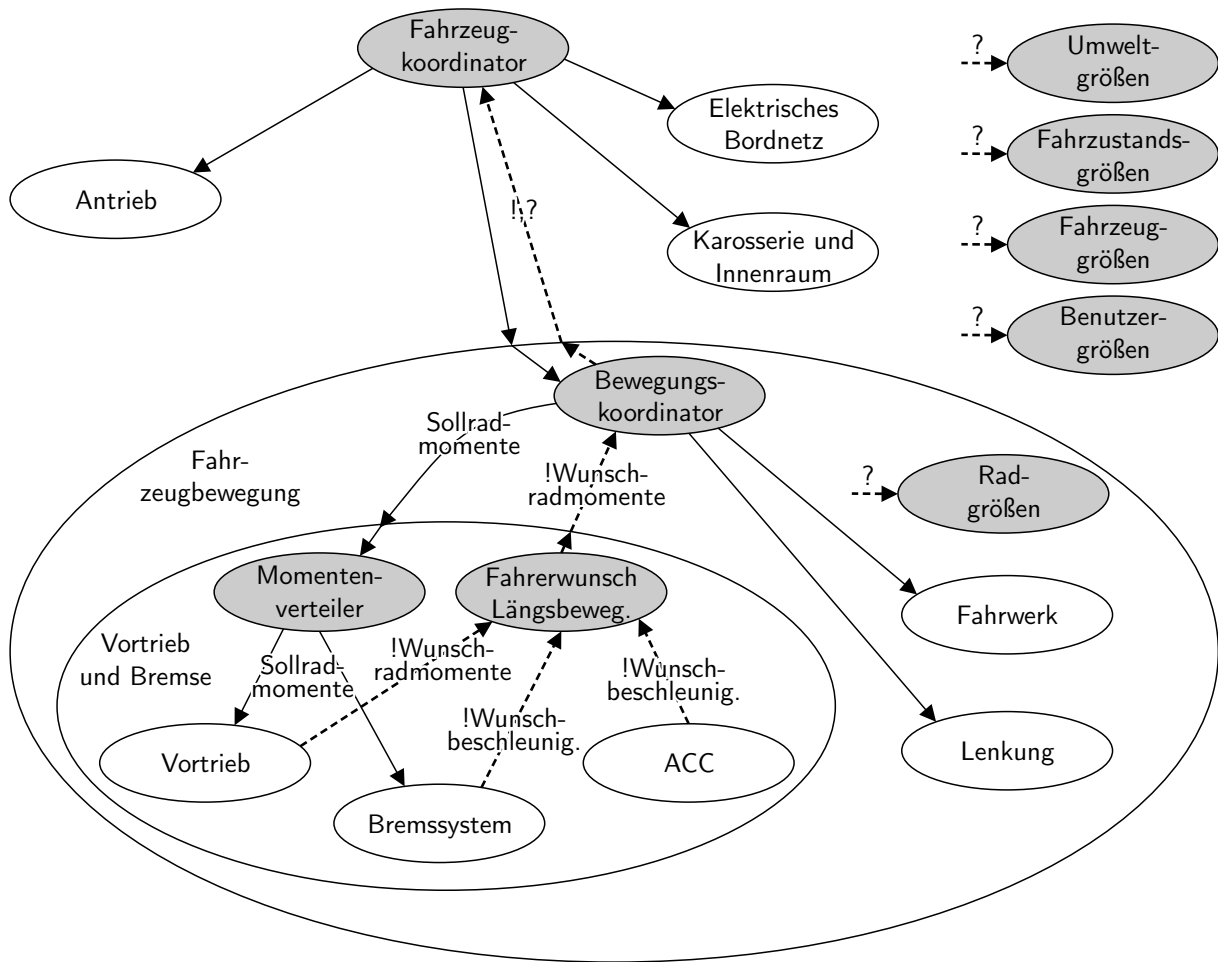


Bild 4.8: CARTRONIC-Funktionsstruktur

#### 4.2.5 Qualitative Modellierung in einem prozessorientierten Konzept

Bei den qualitativen Modellen wird zwar von einem zeit-kontinuierlichen Prozess ausgegangen, die Zusammenhänge und die Verläufe der Prozessgrößen werden jedoch nur prinzipiell und qualitativ betrachtet. Wichtige interne Prozessgrößen werden mit Hilfe von qualitativen Intervallvariablen, wie in Bild 4.9 beispielhaft dargestellt, beschrieben. Aus den einzelnen Intervallen und vor allem aus deren Übergängen lassen sich die unterschiedlichen Verhaltensmuster und die Situationen der abzubildenden Komponente ableiten. Für jede dieser Situationen wird in einer Tabelle das zugehörige Schnittstellenverhalten und die Übergangsmöglichkeiten in andere Situationen festgehalten [LG99].

Die Kopplung der Komponentenmodelle zu einem System findet durch Angabe einer Netzliste statt. Die so genannte Intervall-Arithmetik bildet die mathematische Grundlage für die Verknüpfung der Einzelkomponenten und die Berechnung des Gesamtsystemverhaltens.

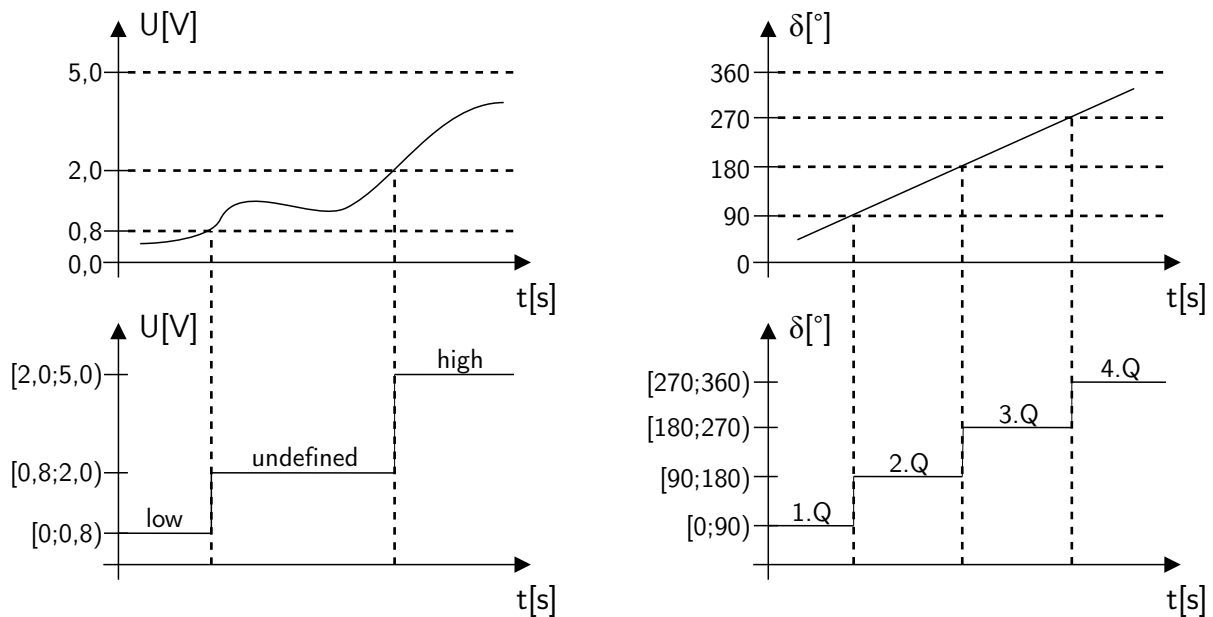


Bild 4.9: Qualitative Modellierung von Spannungs- und Winkelgrößen mit Intervallen

Qualitative Modelle haben den Vorteil, dass eine Beschreibung schon zu einer sehr frühen Entwicklungsphase auf der Basis von ungenauen Informationen möglich ist. Es gibt darüber hinaus einige Vorgänge in der Realität, die quantitativ nur sehr schwer oder gar nicht beschreibbar, qualitativ aber durchaus nachzubilden sind. Qualitative Modelle sind meist einfach zu interpretieren, da die häufig schwierige Bewertung der quantitativen Fakten bereits im Modell impliziert ist. Die physikalische Struktur des Systems lässt sich durch die komponentenbasierte Vorgehensweise realitätsnah abbilden. Allerdings sind mit einem qualitativen Modell aufgrund des hohen Abstraktionsgrads nur grobe, das prinzipielle Verhalten betreffende Aussagen möglich.

Uwe Biegert entwickelt in seiner Dissertation „Ganzheitliche modellbasierte Sicherheitsanalyse von Prozessautomatisierungssystemen“ [Bie03] auf der Basis eines qualitativen Modells eine Sicherheitsbewertung für den Bereich der Prozessautomatisierungstechnik. Die zur Anwendung kommenden Modellierungselemente sind Teil des SQMA-Verfahrens nach [Lau96].

#### 4.2.6 Quantitative Modellierung in einem prozessorientierten Konzept

Oft bilden qualitative Beschreibungen die Vorstufe zu den quantitativen Modellen. Die quantitative Beschreibung von zeit-kontinuierlichen Prozessen ist die Grundlage aller systemdynamischen bzw. regelungstechnischen Untersuchungen. Die sich ergebenden Prozessmodelle werden in verschiedensten Formen angegeben. Eine Variante stellen die Eingangs-/Ausgangs-Modelle dar. Bei diesem Modellierungskonzept wird das Verhalten eines Systems durch die Abhängig-

keit seiner Ausgangsgrößen von den Eingangsgrößen beschrieben (vgl. Bild 4.10). Man spricht dabei von der Verknüpfung von Ursache und Wirkung. Bei den Eingangs- und Ausgangsgrößen handelt es sich meistens um zeitabhängige physikalische Größen, deren Zusammenhang mathematisch, z.B. durch Differentialgleichungen oder Übertragungsfunktionen, beschrieben wird.

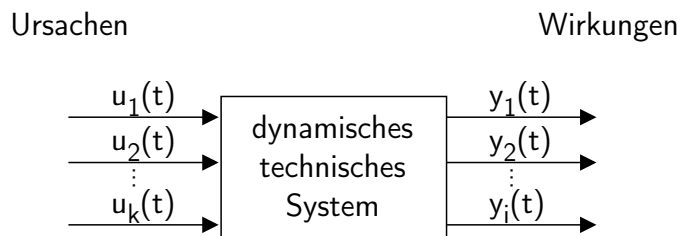


Bild 4.10: Quantitative zeit-kontinuierliche Modellierung mittels Eingangs-/Ausgangs-Modell

### 4.2.7 Hybride Modellierung

Eine andere, erweiterte Form zur quantitativen Modellierung stellen hybride Modelle aus kontinuierlichen und diskreten Systemen dar. Ihr Verhalten kann mit Hilfe eines zustandsorientierten und gleichermaßen gleichungsbezogenen wie graphischen Beschreibungsmittel, dem so genannten Netz-Zustands-Modell [Nen01], abgebildet werden. Die allgemeine Darstellungsform eines Netz-Zustands-Modells zeigt Bild 4.11. Es besteht aus einem Petri-Netz oder Zustandsautomaten für den diskreten Modellanteil und einem erweiterten Zustandsraummodell für den kontinuierlichen Modellanteil. Beide Modellteile sind über Transformationskomponenten miteinander verbunden. Die diskreten und kontinuierlichen Ein- und Ausgangsgrößen  $\underline{u}_D(t)$  und  $\underline{y}_D(t)$  bzw.  $\underline{u}_K(t)$  und  $\underline{y}_K(t)$  stellen die Verbindung zur Außenwelt dar. Die inneren Systemzustände werden durch  $\underline{x}_D(t)$  und  $\underline{x}_K(t)$  repräsentiert.

Generell gilt, dass zur Erstellung quantitativer Modelle bedeutend mehr Informationen und Systemkenntnisse erforderlich sind als bei den bisher vorgestellten Modellierungskonzepten. Damit kann eine Modellierung im Vergleich z.B. zu den qualitativen Modellen erst zu einem späteren Zeitpunkt im Entwicklungsprozess starten.

Trotzdem ist durch eine geeignete Wahl des anfänglichen Abstraktionsgrades und einer sukzessiven Erweiterung eine Anpassung des Modells an den Entwicklungsstand möglich. Das Modellierungskonzept kann sowohl mit abstrakten Vorstellungen zu einem recht frühen Entwicklungszeitpunkt als auch später mit immer detaillierteren Verhaltenbeschreibungen umgehen. Die vollständige Berücksichtigung der Zeit ermöglicht sowohl die Modellierung von sequentiellen und parallelen Abläufen als auch den Umgang mit kontinuierlichen Vorgängen. Darüber

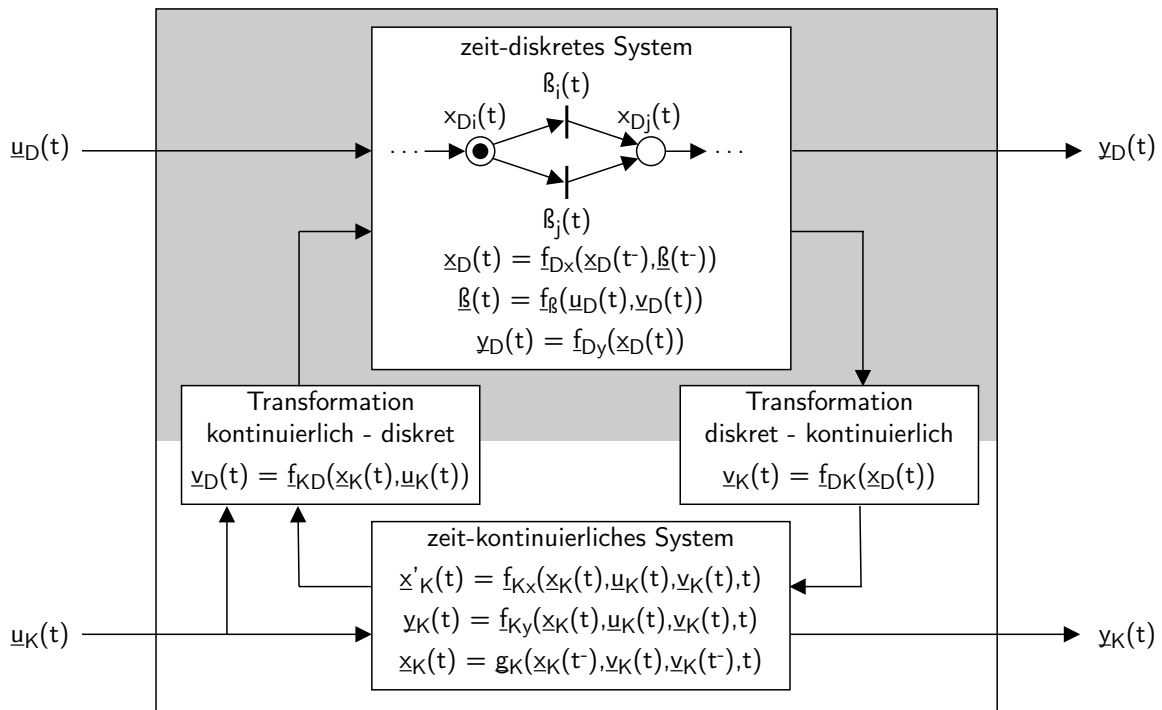


Bild 4.11: Netz-Zustands-Modell zur Modellierung hybrider Systeme

hinaus erlaubt es präzise Aussagen zum dynamischen Systemverhalten. Zahlreiche Werkzeuge, wie z.B. CASE-Tools, AutoCode-Generatoren oder Rapid Prototyping Systeme, unterstützen den Modellerstellungsprozess, ermöglichen eine oft automatisierte Modelltransformation in ausführbare Software und bieten eine Plattform zur Validierung und Verifikation der erstellten Verhaltensmodelle.

### 4.2.8 Zusammenfassung

Alle behandelten Modellierungskonzepte sind nochmals in Tabelle 4.3 mit ihren zentralen Eigenschaften zusammenfassend dargestellt. Die darin vorgenommene Bewertung stützt sich dabei sowohl auf eigene Untersuchungen als auch auf veröffentlichte Ergebnisse in [Bie03] und [SCJ98]. Im Folgenden dient diese Bewertung zur Überprüfung der Erfüllung der unter Kapitel 3.2 getroffenen Anforderungen.

Wie aus Tabelle 4.3 hervorgeht, erfüllt ausschließlich die hybride Modellierung alle Anforderungen zur Realisierung einer modellbasierten Systemanalyse. Bei der Betrachtung der unterschiedlichen Modellierungskonzepte fällt die relativ klare Trennung zwischen Beschreibungsmitteln für kontinuierliche und diskrete Systeme auf. Insbesondere diese Beschränkung auf jeweils eines dieser Systemkonzepte verhindert die vollständige Erfüllung der getroffenen Anforderungen.



Tabelle 4.3: Bewertungsmatrix der Modellierungskonzepte

Kategorie	Kriterien	Legende: + geeignet 0 vorhanden - nicht vorhanden						
		Graphen	Zustands- automaten	Petri-Netze	CARTRONIC	Qualitative Modelle	Quantitative Modelle	Hybride Modelle
Paradigma	prozessorientiert	-	-	-	-	+	+	+
	produktorientiert	-	-	-	-	-	-	-
	zustandsorientiert	+	+	+	-	-	-	+
	objektorientiert	-	-	-	+	-	-	-
Einsatz- bereich	mathematisch-orientiert	+	+	+	-	-	+	+
	implementierungsorientiert	-	-	-	-	-	-	-
	strukturorientiert	-	-	0	+	0	0	0
	verhaltensorientiert	+	+	+	-	+	+	+
Einsatz- phase	Spezifikation	+	0	+	+	+	-	-
	Modellbildung	+	+	+	+	+	+	+
	Implementierung	-	0	0	-	-	0	0
	Betrieb	-	0	0	-	-	0	0
Eigen- schaften	mathematische Basis	+	+	+	0	0	+	+
	Abbildung der Funktionsstruktur	-	0	+	+	+	+	+
	Abbildung der Komponentenstruktur	-	-	0	-	+	+	+
	Modularität	-	-	+	+	+	+	+
	Dynamik	0	+	+	-	0	+	+
	Umgang mit kontinuierlichen Größen/Vorgängen	-	-	-	-	0	+	+
	Umgang mit diskreten Größen/Vorgängen	-	+	+	-	0	-	+
	Abbildung paralleler Vorgänge	-	-	+	-	+	-	+
	Abbildung sequentieller Vorgänge	+	+	+	-	+	-	+
	Abbildung der Zeit	-	-	0	-	+	+	+
	Simulierbarkeit	+	+	+	0	+	+	+
	Analysierbarkeit	0	+	+	0	0	0	0
	Durchgängigkeit	-	0	0	-	0	0	0
Toolunterstützung	+	+	+	0	0	+	+	
Sicherheits- analyse	Zusammenhang: Ursache und Wirkung	+	+	+	+	+	+	+
	Klassifizieren von Betriebsmodi	0	+	+	0	+	-	+
	Ermittlung des System-Fehlerverhaltens	-	+	+	-	+	+	+

Erst die Kombination aus quantitativen Modellierungskonzepten, wie z.B. Differentialgleichungen oder Übertragungsfunktionen und Zustandsautomaten zu einem Beschreibungsmittel für hybride Systeme wird allen notwendigen Vorgaben gerecht.

Aus Kapitel 3 geht neben den bereits behandelten Anforderungen zur Sicherheitsanalyse und zur dafür notwendigen Modellierungstechnik auch die Kernanforderung einer Automatisierung und der iterativen Anwendbarkeit des Analyseverfahrens hervor. Im folgenden Abschnitt steht deshalb die Frage nach bekannten Methoden zur Automatisierung von Sicherheitsanalysen im Fokus der Betrachtung.

## 4.3 Methoden zur Automatisierung von System- und Sicherheitsanalysen

Mit dem Thema Automatisierung hat man sich bisher hauptsächlich im Bereich der Testmethodik beschäftigt. Die Steigerung der Systemkomplexität geht unweigerlich einher mit einer enormen Steigerung des Testaufwands und der daraus resultierenden Fragestellung der Auswahl geeigneter Teststimuli zur Begrenzung der Aufwendungen für Tests.

In der Literatur [DSSS01, RPGN97, Kai03] existiert eine Vielzahl an Lösungsmöglichkeiten für die Problematik der richtigen Stimulwahl. Der erste nahe liegende Ansatz ist die zufällige Auswahl der Stimuli. Eine zweite Möglichkeit zur Wahl der Eingangswerte besteht in der Selektion durch das Testpersonal. Entweder aufgrund der Kenntnis der Implementierung des Testobjekts (white box test) oder aufgrund der Erfahrung früherer Tests (black box test) bestimmt das Testpersonal intuitiv die Stimuli. Eine weitere Vorgehensweise zur Lösung dieser Problematik liegt im Bestreben, die Eingangswerte so zu beschränken, dass möglichst nur die Stimuli getestet werden, die in der Realität auch tatsächlich auftreten. Dazu ist das Testobjekt in eine virtuelle Systemumgebung aus Fahrzeug oder Fahrer eingebettet und der Kreis zwischen den Ein- und Ausgängen des zu testenden Fahrzeugsystems geschlossen. Je nach Art des Testobjekts handelt es sich um eine andere Ausprägung des Testprozess. Ist das Testobjekt ein Modell oder eine ausführbare Spezifikation, so spricht man von model-in-the-loop (MiL) Tests. Ersetzt man das Modell unter Beibehaltung der Testumgebung durch ein übersetztes und ausführbares Programm in Seriene-Code-Form, wird die Testumgebung software-in-the-Loop (SiL) genannt. Werden Teile des Kraftfahrzeugsystems in realer Hardware realisiert und die Testumgebung samt Testobjekt in Echtzeit ausgeführt, handelt es sich um eine hardware-in-the-loop (HiL) Testplattform. Eine Übersicht zu den vorgestellten Testverfahren zeigt Bild 4.12. Für ausführliche Beschreibungen und Realisierungsbeispiele der einzelnen Verfahren sei auf [DSSS01, LT04, KKS<sup>+</sup>02, Spi01] verwiesen.

Selbst bei einer geeigneten Beschränkung der Teststimuli mit Hilfe der in-the-Loop Testverfahren ist der Umfang an notwendigen Testszenarien in den letzten Jahren aufgrund der gestiegenen Qualitätsanforderungen und der zunehmenden Komplexität der vernetzten Kraftfahrzeugsysteme enorm angewachsen. Um den Testaufwand beherrschen zu können, ist zusätzlich zur geeigneten Stimulwahl die Automatisierung des gesamten Testprozesses von Nöten [GR02]. Dies gelingt mit Hilfe von Testsequenzen, die festlegen, welche Eingangssignale angelegt, in welchen inneren Zustand das Testobjekt versetzt oder welche Umgebungsbedingungen gezielt untersucht werden sollen [Con01]. Die Möglichkeit zur Speicherung entworfener Testsequenzen in einer Testbibliothek erhöht die Wiederverwendbarkeit und reduziert den Testaufwand. Auch

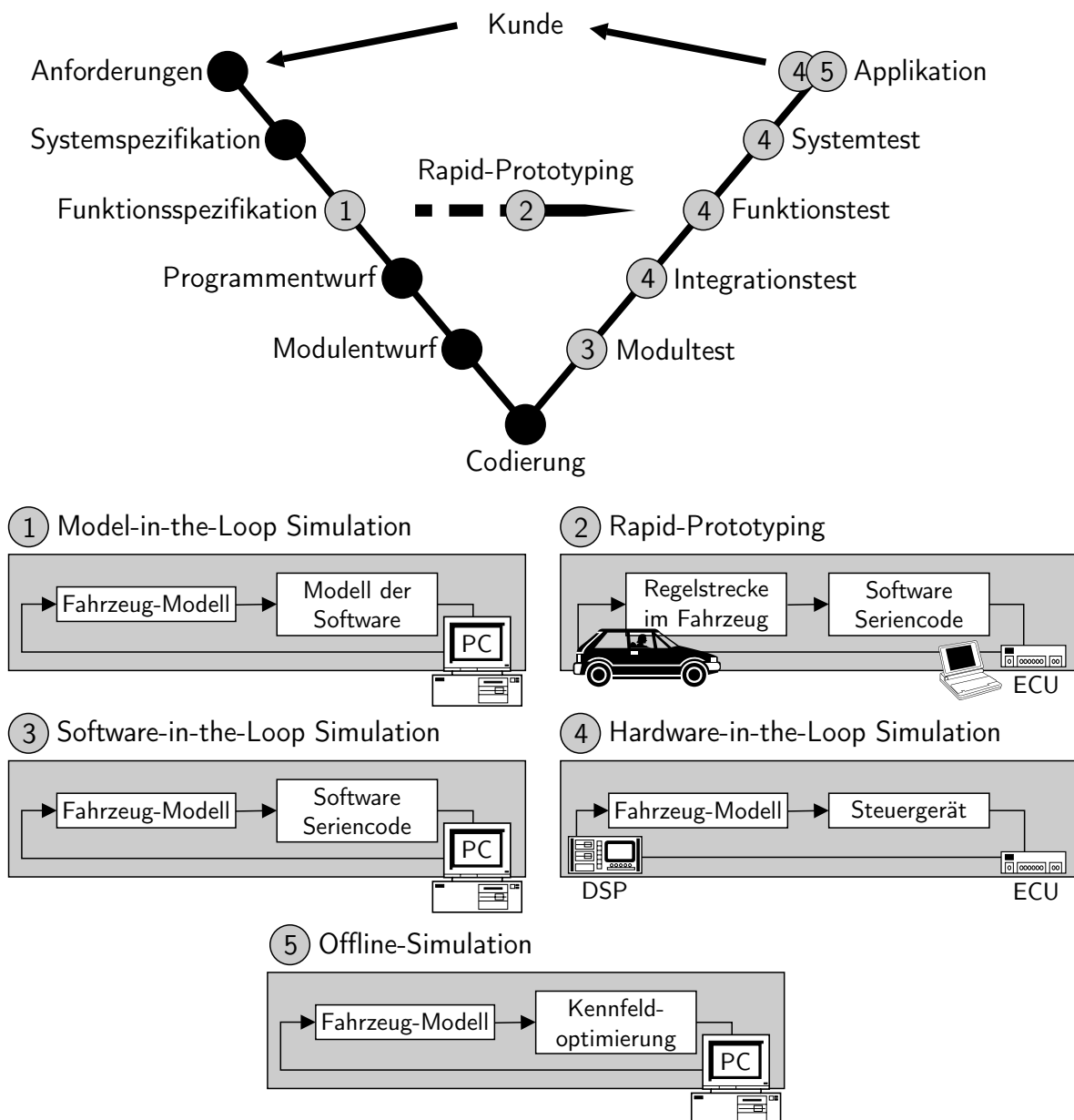


Bild 4.12: Einordnung von in-the-Loop Verfahren im Entwicklungsprozess nach [GR02]

die Testauswertung und deren Dokumentation kann inzwischen durch Signalvergleichsverfahren weitgehend automatisiert werden, so dass heute in der industriellen Praxis funktionale Aspekte schon durch einen weitgehend automatisierten Testprozess überprüfbar sind [CFP03].

Mit der Entwicklung von sicherheitsrelevanten Systemen rückt neben der Funktion immer stärker das Verhalten im Fehlerfall in den Fokus der Testmethodik. Es ist nicht mehr ausreichend, mit Funktionstests die Umsetzung der funktionalen Anforderungen zu überprüfen, sondern mehr und mehr muss die Wirksamkeit der eingesetzten Fehlererkennungsmechanismen und Fehlertoleranzverfahren nachgewiesen werden. Wünschenswert wäre, einen Beweis der Wirksamkeit mit

Hilfe formaler Methoden führen zu können. Tatsächlich stoßen die formalen Methoden jedoch, wie bereits in Kapitel 4.1.6 erörtert, an Aufwandsgrenzen, so dass experimentelle Untersuchungen und Tests herangezogen werden müssen. Zu diesem Zweck sind künstlich erzeugte Fehler in das entwickelte System zu injizieren. Die Zielsetzung und Vorgehensweise weicht damit deutlich von denen der bisher vorgestellten Testverfahren ab.

Seit einigen Jahren hat sich die Fehlerinjektion zu einem eigenständigen Arbeitsgebiet im Bereich der Fehlertoleranzverfahren und der Testmethoden entwickelt. Es existiert inzwischen eine Vielzahl an Fehlerinjektionsmethoden für die unterschiedlichsten Anwendungsgebiete. Eine umfangreiche Zusammenstellung bekannter Methoden und deren Klassifikation in physikalische (Hardware-basierte), Software-implementierte und Simulation-basierte Fehlerinjektion findet sich in [ES98].

Die überwiegende Mehrzahl der zum Thema Fehlerinjektion veröffentlichten Forschungsbeiträge widmet sich bisher der digitalen Schaltungstechnik. Insbesondere die Überprüfung von Fehlertoleranzmechanismen in Hardware-Komponenten, wie der Recheneinheit oder dem Speicher, stehen im Fokus der Untersuchungen. Darüber hinaus existieren einige Ansätze auf Kommunikationsnetz-Ebene zur Validierung der Wirksamkeit von Fehlertoleranzmechanismen in verteilten Rechnernetzen.

Bisher wurden die Verfahren zur Fehlerinjektion jedoch nur sehr selten auf System-Ebene eingesetzt. In [OKC01] wenden die Autoren die Fehlerinjektion auf ein Modell eines brake-by-wire System an und untersuchen damit die Wirksamkeit ihrer Fehlertoleranzkonzepte. Sie beschränken sich dabei auf ein generisches Fehlermodell, das nicht die speziellen Eigenschaften und die damit verbundenen Fehlermöglichkeiten der System-Komponenten berücksichtigt, sondern aus einem Satz allgemein gültiger, signalfluss-orientierter Datenmanipulationsmöglichkeiten besteht. Darüber hinaus setzen auch die Autoren von [CERT04] Fehlerinjektionen auf System-Ebene ein, verfolgen damit jedoch andere Ziele. Sie befassen sich mit dem Einfluss transienter Fehler auf die funktionale Leistungsfähigkeit eines Fahrdynamik-Systems.

# 5 Konzept der ganzheitlichen, dynamischen Systemanalyse

Das vorherige Kapitel hat deutlich gemacht, dass die bekannten Verfahren zur System- und Sicherheitsanalyse Schwachstellen aufweisen. Sie können die aufgestellten Anforderungen nur bedingt erfüllen. Insbesondere mit dem Wandel in der Automobilindustrie hin zu einem modellbasierten Entwicklungsprozess und dem Bestreben sicherheitsrelevante Funktionen im Fahrzeug der Zukunft zu realisieren, ergibt sich sowohl eine Chance als auch ein Bedarf für ein Verfahren einer erweiterten, modellbasierten Analyse unter einer ganzheitlichen Betrachtungsweise von sicherheitsrelevanten Fahrzeugsystemen.

Die folgenden Ausführungen beschreiben ein Verfahren, das, zunächst allgemein, auf den bisher abgeleiteten Erkenntnissen und Anforderungen basiert, bevor in den folgenden Kapiteln detailliert auf die Bestandteile dieser Methodik eingegangen wird.

## 5.1 Methodik der ganzheitlichen, dynamischen Analyse

Das vorliegende Verfahren basiert auf dem Gedanken, das komplexe Verhalten eines sicherheitsrelevanten Fahrzeugsystems im Fehlerfall unter Einbeziehung eines Modells herzuleiten. Die notwendigen Prozessschritte von der Modellerstellung bis zur Modellauswertung und der entsprechenden Ableitung geänderter Anforderungen sind in Abbildung 5.1 dargestellt.

Der erste Ablaufschritt umfasst die Modellierung der Systemkomponenten ausschließlich unter funktionalen Gesichtspunkten. Konkrete Realisierungsfragestellungen, wie z.B. die Frage nach der Zuordnung von Funktionen zu Komponenten, sollen in dieser Phase ausdrücklich unberücksichtigt bleiben. Durch die nachfolgende Integration der so entstandenen funktionalen Komponentenmodelle zu einem funktionalen Systemmodell wird der erste Schritt in Richtung Systemanalyse vollzogen. Das Hinzufügen von Beschreibungen der Systemumgebung, wie z.B. Fahrer- oder Fahrzeugmodell, erlauben die Ausweitung zur Gesamtsystembetrachtung.

Anschließend erfolgt die Erweiterung der Verhaltensbeschreibung des sicherheitsrelevanten Fahrzeugsystems über die funktionalen Aspekte hinaus durch die Berücksichtigung struktureller Aspekte. Hierzu zählt z.B. die Nachbildung einer speziellen Systemarchitektur oder die

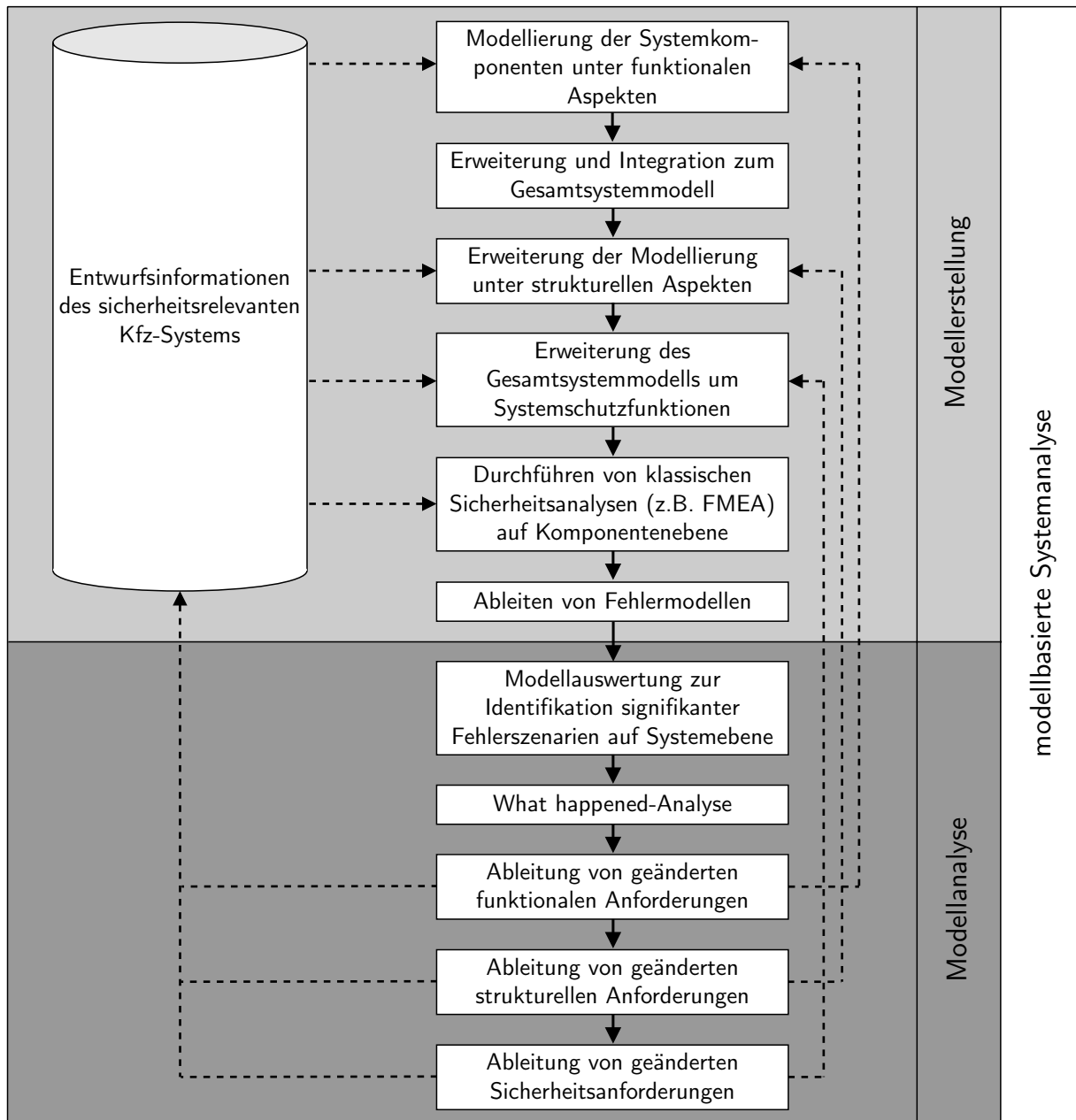


Bild 5.1: Verfahrensablauf der ganzheitlichen, dynamischen Analyse sicherheitsrelevanter Fahrzeugsysteme unter dem Aspekt des Fehlerverhaltens

Modellierung von real existierenden Komponentenschnittstellen. Die zweite Erweiterung des Fahrzeugsystemmodells betrifft die Schutzfunktionen des Systems. In dieser Modellierungsphase müssen im Systemmodell die entsprechenden Rückfallebenen, Fehlererkennungs- und Fehlerbehandlungsmechanismen implementiert werden.

Die nachfolgende Durchführung von klassischen Sicherheitsanalysen auf Komponentenebene ermöglicht die Ermittlung der Fehlermöglichkeiten der Systemkomponenten. Wie bereits in Kapi-

tel 4.1 dargelegt, weisen die klassischen Sicherheitsanalysen bei der Untersuchung großer, komplexer Systeme einige Schwachstellen auf. Sie eignen sich jedoch gut für die Analyse abgegrenzter, überschaubarer Komponenten. Diese Vorteile der klassischen und etablierten Sicherheitsanalysen sollen auch weiterhin genutzt, jedoch auf die Analyse von Komponenten beschränkt werden. Im Anschluss an die klassischen Sicherheitsanalysen werden aus deren Ergebnissen Fehlermodelle für die jeweiligen Komponenten abgeleitet, die das Verhalten der Komponenten im Fehlerfall an ihren Schnittstellen beschreiben.

Die anschließende Modellauswertung stellt den Kern der Analysemethodik dar und dient zur Identifikation signifikanter Fehlerszenarien. Durch Fehlerinjektion und Modellausführung können die Auswirkungen auf Systemebene reproduzierbar bestimmt werden. Beim Auftreten von signifikanten Fehlerauswirkungen stellt sich meist sofort die Frage nach der Ursache. Wurde das Gesamtsystemmodell bisher ausschließlich in vorwärtiger Richtung genutzt, um die Auswirkungen von Fehlern zu ermitteln, ist vor allem die Umkehrung der Analyserichtung für die Ursachenklärung von Bedeutung. Die so genannte „What happened“-Analyse lässt, auf Basis des identischen Gesamtsystemmodells, die Ermittlung der Fehlerausbreitungswege im sicherheitsrelevanten Fahrzeugsystem zu. Aus der Kenntnis dieser systeminternen Abläufe gelingt die Ableitung von geänderten funktionalen, strukturellen oder sicherheitsbezogenen Anforderungen. Eine iterative Anwendung dieser Vorgehensweise ermöglicht eine kontinuierliche Verbesserung der Systemanforderungen und deren Realisierung bei ständiger Überprüfung der vorgenommenen Änderungen.

Um den in Kapitel 3 definierten Anforderungen insbesondere hinsichtlich der Automatisierbarkeit und der iterativen Anwendbarkeit der Methodik gerecht zu werden, ist eine intensive Betrachtung der bisher mit dem Begriff Modellauswertung beschriebenen Vorgehensphase erforderlich. In Bild 5.2 wird der Prozessschritt der Modellauswertung detailliert dargestellt.

Im Zentrum der Modellauswertung steht das Modell des sicherheitsrelevanten Fahrzeugsystems, das, wie bereits beschrieben, nicht nur die Funktion, sondern auch Struktur- und Sicherheitsaspekte berücksichtigt. Durch die Integration des Modells in eine modellbasierte Systemumgebung bestehend aus Fahrzeugmodell, Fahrer, Straße und spezifizierten Fahrmanövern gelingt durch die Ausführung des Gesamtsystemmodells die Analyse des Kraftfahrzeugsystems unter funktionalen Aspekten.

Mit Hilfe der Modellierung von Komponentenfehlern und deren Integration in das Gesamtsystemmodell ist man in der Lage, die Auswirkungen und Ausbreitungen von Fehlern im Fahrzeugsystem zu untersuchen. Bisher ist diese Analyse auf eine ausschließlich manuell durchführbare Modellauswertung beschränkt, da eine Automatisierung mit der bisherigen Vorgehensweise noch nicht realisierbar ist.

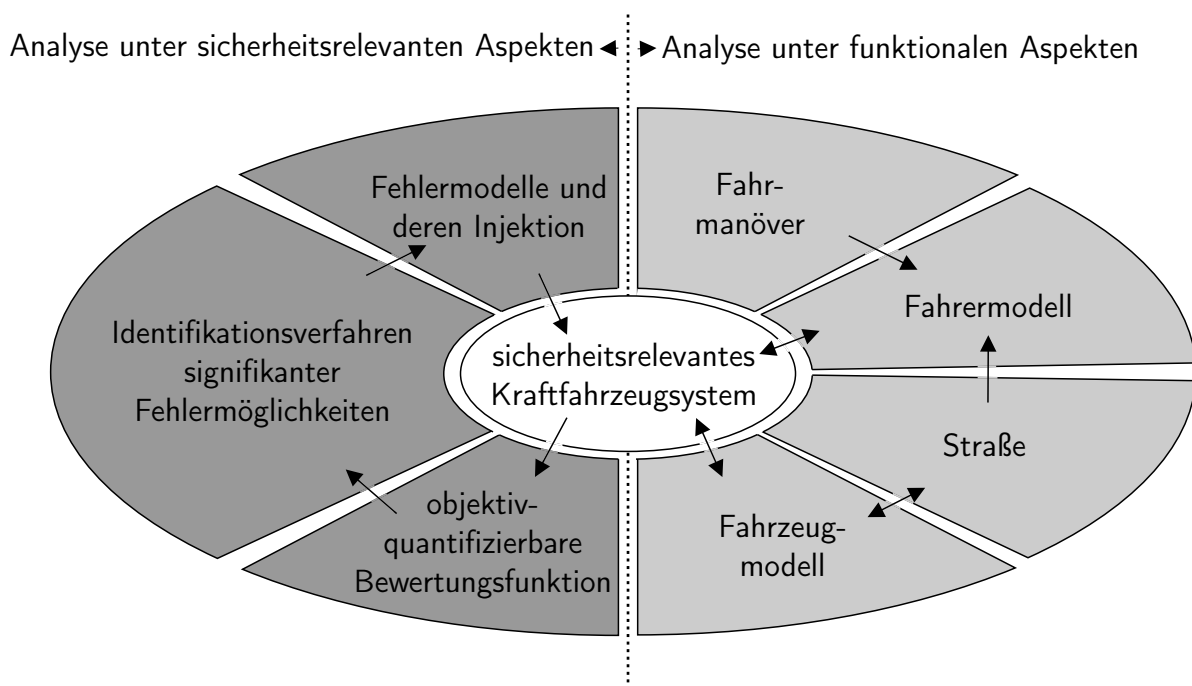


Bild 5.2: Struktur der Modellauswertung zur Identifikation signifikanter Fehlermöglichkeiten

Dazu muss eine weitere Komponente, die objektiv-quantifizierbare Bewertungsfunktion, hinzugefügt werden. Diese Funktion stellt eine mathematische Berechnungsvorschrift für die Schwere eines Fehlers dar. Sie ist deshalb als objektiv-quantifizierbar zu bezeichnen, da für jede mögliche Fehlersituation die selben Kriterien zur Bewertung herangezogen werden. Das Ergebnis der Bewertung liegt damit quantitativ vor und ist reproduzierbar. Selbstverständlich steht „objektiv-quantifizierbar“ zudem für die Tatsache, dass es nicht nur eine gültige Bewertungsfunktion für ein System gibt. Je nach persönlichem Sicherheitsempfinden der Entwickler, Sicherheitsvorgaben von öffentlichen Normen, Bestimmungen firmeninternen Verordnungen sowie der Risikoakzeptanz in der Gesellschaft variieren die Bestandteile und deren Gewichtungen dieser Bewertungsfunktion. Hat man eine Bewertungsfunktion definiert, so wird auf Basis dieser Festlegung eine objektive Bewertung ohne subjektives Empfinden Einzelner vorgenommen.

Erweitert man die Modellauswertung zusätzlich um einen Algorithmus, der Fehler injiziert, das Fahrzeugsystem in einen definierten Systemzustand versetzt und die Auswirkungen mit Hilfe der Bewertungsfunktion erfasst, sind alle Komponenten für eine automatisierte Modellauswertung vorhanden. Wird darüber hinaus ein intelligenter Algorithmus eingesetzt, der selbstständig auf Basis vergangener Bewertungen neue Fehlersituationen bestehend aus modellierten Fehlern, deren Auftretenszeitpunkt, deren Anhaltedauer und des vorherrschenden Systemzustands generiert und untersucht, gelingt die automatische Identifikation von signifikanten Fehlerszenarien.



## 5.2 Einbettung in einen Entwicklungsprozess

Es existiert bereits eine Vielzahl unterschiedlichster Entwicklungsprozesse für sicherheitsrelevante Systeme [DE02, Ben04, SAE96a, SAE96b, Hed01, Stö00, SS04]. Beispielhaft soll an dieser Stelle die Integration des beschriebenen Verfahrens zur ganzheitlichen, dynamischen Analyse unter dem Aspekt des Systemfehlerverhaltens in einem solchen Entwicklungsprozess erläutert werden.

Dazu sei hier exemplarisch das von Stefan Benz entwickelte Vorgehensmodell aus seiner Dissertation „Eine Entwicklungsmethodik für sicherheitsrelevante Elektroniksysteme im Automobil“ [Ben04], das so genannte Doppel-V-Modell herangezogen. Abbildung 5.3 zeigt den Prozessablauf eines erweiterten Doppel-V-Modells mit integrierter Methodik zur ganzheitlichen, dynamischen Systemanalyse.

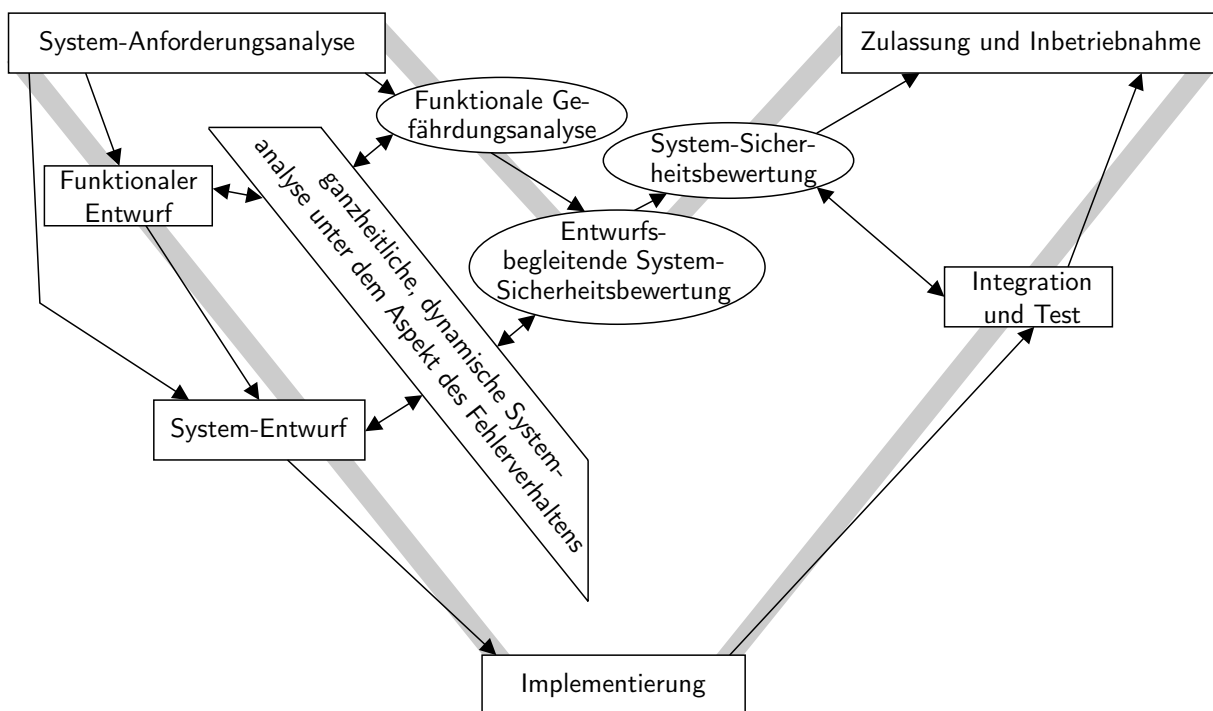


Bild 5.3: Exemplarische Integration der Methodik in den Doppel-V-Entwicklungsprozess

Der von Stefan Benz vorgestellte Entwicklungszyklus ist vor allem durch die zeitliche Parallelität von Funktions- und Sicherheitsentwicklung geprägt. Zwei ineinander angeordnete V-Modelle, jeweils eines für den Entwicklungsprozess der Funktion und der Sicherheit, stellen diese Parallelität dar. Um aus dieser Parallelität nicht ein ungewünschtes Nebeneinander der Prozesse werden zu lassen, sind insbesondere die Schnittstellen zwischen den einzelnen Phasen der parallelen Entwicklungszyklen von großer Bedeutung. An diesen Stellen kann die vorgestellte Metho-

dik eingesetzt und als Mittel zum Informationsaustausch und zur Ableitung von notwendigen Änderungen aus Funktions- bzw. Sicherheitssichtweise herangezogen werden.

Mit der Integration eines funktionalen, modellbasierten Systementwurfs in eine Gesamtsystemumgebung unter funktionalen Aspekten lässt sich bereits sehr früh im Entwicklungsprozess ableiten, welche potentiellen Gefahren, so genannte Hazards, von diesem System ausgehen können. Die Zuordnung eines quantitativen Gefährdungsmaßes während einer funktionalen Gefährdungsanalyse für die Schwere eines Fehlers kann auf Systemebene deutlich leichter ermittelt werden. Mit fortschreitender Entwicklung wird die Erweiterung des Gesamtsystemmodells um Struktur- und Sicherheitsaspekte wichtiger. Die iterative Anwendung der vorgestellten Methodik lässt während dem gesamten Systementwurf eine kontinuierliche Bewertung des aktuellen Entwicklungsstandes sowohl hinsichtlich der Funktion als auch der Sicherheit zu. Die Möglichkeit zur Ableitung kontinuierlicher Verbesserungen, getrieben sowohl durch den Funktions- als auch den Sicherheitsentwicklungsprozess, erlaubt eine abgestimmte, parallele und entwurfsbegleitende Vorgehensweise ohne die Gefahr einer Entfernung der beiden Entwicklungsprozesse vom gemeinsamen Entwicklungsziel.

# 6 Quantitative hybride Gesamtsystemmodellierung

Wie in Kapitel 4.2 dargelegt, erfüllt nur die hybride Modellierung alle an die Modellierung gestellten Anforderungen zur Realisierung einer ganzheitlichen, modellbasierten Systemanalyse. Im Folgenden ist zunächst eine Klassifizierung der zu modellierenden Systeme und Systemkomponenten beschrieben, bevor im Anschluss drei Modellierungsstufen eingeführt werden, die üblicherweise aufeinander folgend durchzuführen sind, um zu einer geeigneten Systembeschreibung für eine ganzheitliche, dynamische Systemanalyse unter dem Aspekt des Fehlerverhaltens zu gelangen. Die Beschreibung jeder Modellierungsstufe beinhaltet deren individuelle Zielsetzung, die daraus abzuleitenden Abstraktionsmöglichkeiten und weitere Erläuterungen zur Umsetzung mit Hilfe der hybriden Modellierung.

## 6.1 Klassifizierung von Systemen und Systemkomponenten

Vor der eigentlichen Gesamtsystemmodellierung bietet sich die Einführung eines Klassenmodells für mögliche zu untersuchende Kraftfahrzeugsysteme an. Die Grundlagen für die allgemeine Klassifizierung bilden spezielle Systemeigenschaften, die das zu modellierende Systemverhalten bereits in besonderer Weise prägen. Für die hier gewählte Klassifizierung sind folgende Eigenschaften von Bedeutung: das Vorhandensein eines Systemgedächtnisses, eines verteilten Systemgedächtnisses oder eines selbst- oder fremdbildenden Systemgedächtnisses. Aus diesen grundlegenden Systemeigenschaften ergibt sich eine Klassifizierung gemäß Bild 6.1.

Aus der Klassifizierung gehen vier Systemgruppen hervor. Die erste Gruppe zeichnet sich durch keinerlei Systemgedächtnis aus. Hierbei handelt es sich um reine Steuerungssysteme, die in Abhängigkeit verschiedener, zu einem Zeitpunkt gültiger Eingangsgrößen eine begrenzte Anzahl an Ausgangsgrößen erzeugen. Das Systemverhalten wird dabei durch keinerlei Systemzustände oder Eingangsgrößen der Vergangenheit beeinflusst, sondern wird nur durch die momentan vorherrschende Umgebungssituation geprägt. Systeme der zweiten Klasse verfügen über ein Gedächtnis. Ihr Verhalten wird ebenfalls durch ihre Vergangenheit bestimmt. Sie weisen dabei eine Systemarchitektur auf, die durch eine zentrale Informationsbeschaffung, -verarbeitung und -verteilung geprägt ist. Die Systemgruppe 3 zeichnet sich durch eine verteilte Systemtopologie

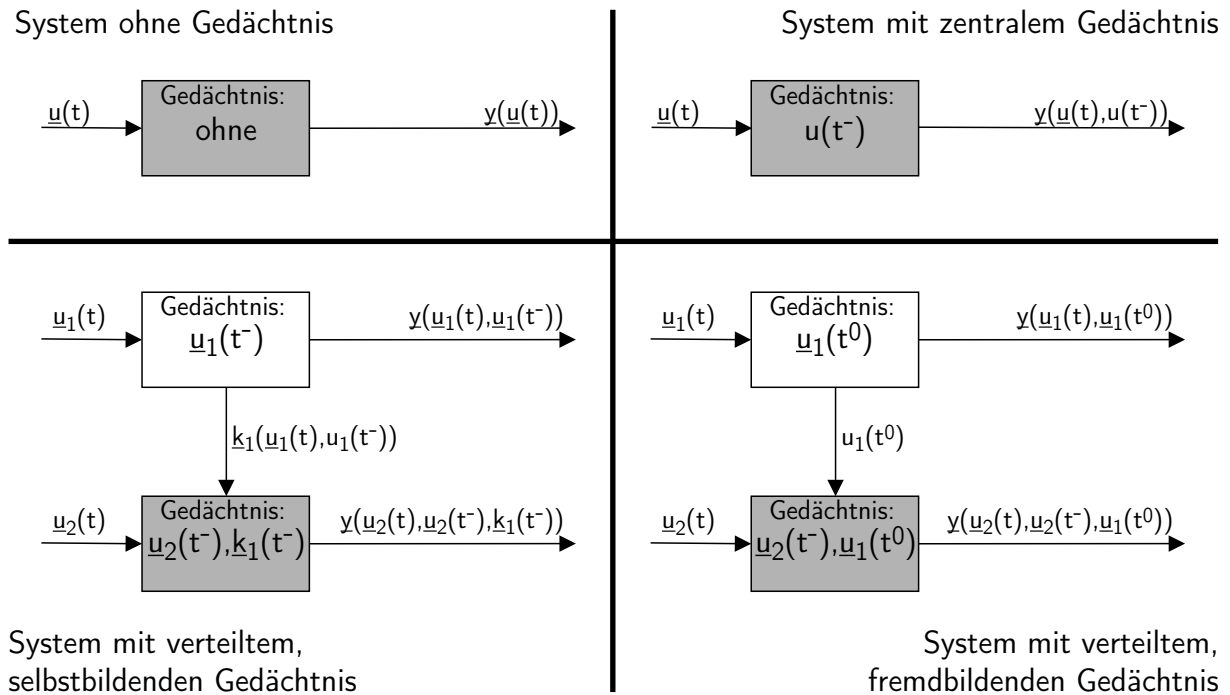


Bild 6.1: Klassifizierung von Systemen

aus. Die Informationshaltung aus der Systemvergangenheit ist dezentral organisiert. Jede Systemeinheit kann die für sie notwendigen Informationen der Systemvergangenheit selbst erfassen und so ihr Gedächtnis selbst bilden. Im Gegensatz zur Systemgruppe 3 sind die Systemeinheiten der Gruppe 4 dazu nicht in der Lage. Sie können die für sie notwendigen Informationen der Vergangenheit nicht selbst erfassen, sondern sind auf eine Kommunikation mit anderen Einheiten zum Bilden ihres eigenen Gedächtnisses angewiesen.

Über die Klassifizierung der Systeme hinaus kann auch für die einzelnen Systemkomponenten eine entsprechende Klassifizierung gefunden werden. Insbesondere in Anbetracht der Forderung einer automatisierten Untersuchung verschiedener Fehlerszenarien ist die Einführung einer Komponentenklassifikation für die Fehlersuche hilfreich. In Abbildung 6.2 ist eine mögliche Zuordnung von Systemkomponenten zu Komponentenklassen dargestellt.

Systemkomponenten der Gruppe 1 sind in der Wirkkette des Systems im Pfad von der Umwelt in Richtung System angesiedelt. Sie verfügen weder über eigene Intelligenz noch über ein Gedächtnis. Ein fehlerhaftes Verhalten dieser Komponente kann seitens des Systems durch Beobachtung der jeweiligen Signal- und Energieflüsse entdeckt werden. Nach transienten Fehlern ist das Verhalten einer Komponente der Klasse 1 nicht vom Nominalverhalten zu unterscheiden. Diese Eigenschaft erlaubt nach der Entdeckung eines transienten Fehlers eine schnelle und einfache Reintegration der Komponente in den Systemablauf.

Charakteristika der Komponentenklassen				
Klasse 1	Klasse 2	Klasse 3	Klasse 4	Klasse 5
<ul style="list-style-type: none"> <li>♦Wirkrichtung: → System</li> <li>♦Keine Intelligenz</li> <li>♦Kein Gedächtnis</li> <li>♦Fehlererkennung durch Beobachtung der Signal- und Energieflüsse</li> <li>♦Verhalten nach transienten Fehlern entspricht fehlerfreiem Verhalten</li> </ul>	<ul style="list-style-type: none"> <li>♦Wirkrichtung: → Umwelt</li> <li>♦Keine Intelligenz</li> <li>♦Kein Gedächtnis</li> <li>♦Fehlererkennung durch Anregung und Komponenten der Klasse 1</li> <li>♦Reintegration nach transienten Fehlern schwierig</li> </ul>	<ul style="list-style-type: none"> <li>♦Wirkrichtung: → System</li> <li>♦Gedächtnis vorhanden</li> <li>♦Selbstdiagnose</li> <li>♦Fail-silent Eigenschaften</li> <li>♦Verhalten nach transienten Fehlern kann sich vom fehlerfreien Verhalten unterscheiden</li> </ul>	<ul style="list-style-type: none"> <li>♦Wirkrichtung: → Umwelt</li> <li>♦Gedächtnis vorhanden</li> <li>♦Selbstdiagnose</li> <li>♦fail-silent Eigenschaften</li> <li>♦Verhalten nach transienten Fehlern kann sich vom fehlerfreien Verhalten unterscheiden</li> </ul>	<ul style="list-style-type: none"> <li>♦zentrale Einheit</li> <li>♦Interaktion mit Komponenten der Klassen 1 bis 5</li> <li>♦Gedächtnis vorhanden</li> </ul>

Bild 6.2: Klassifizierung von Komponenten

Die zweite Klasse umfasst Komponenten, die sich in der Wirkkette im Pfad vom System in Richtung Umwelt befinden. Sie besitzen ebenfalls keinerlei Intelligenz oder Gedächtnis. Fehler in diesen Komponenten können vom System nur bei entsprechender Anregung und mit Hilfe von Komponenten der Klasse 1 detektiert werden. Die Entdeckung transienter Fehlerszenarien gestaltet sich durch die direkte Kopplung der Komponenten mit der Umwelt schwierig, da mit jeder Anregung zum Zwecke des Komponententests negative Auswirkungen auf die Systemumgebung zu befürchten sind. Dadurch ist auch eine Reintegration der Komponente äußerst kritisch zu betrachten bzw. nur in ganz bestimmten, sicheren Systemzuständen in Erwägung zu ziehen.

Die Komponentenklassen 3 und 4 sind von den Gruppen 1 und 2 abgeleitet. Sie unterscheiden sich jedoch durch das Vorhandensein eigener Intelligenz. Die Komponenten der Gruppe 3 bzw. 4 sind damit in der Lage in unterschiedlichen Betriebszuständen zu arbeiten, eigene Fehler selbst zu diagnostizieren und entsprechende Maßnahmen einzuleiten. Ferner weisen sie dadurch fail-silent Eigenschaften auf. Das Komponentenverhalten während und insbesondere nach der Präsenz von Fehlern kann sich aus diesem Grund signifikant vom Nominalverhalten unterscheiden.

Bei den Komponenten der 5. Klasse handelt es sich um zentrale Systemeinheiten wie z.B. reine Datenverarbeitungseinheiten. Sie bestehen zumeist aus Hardware- und Softwareteilen, die für eine Klassifizierung nicht sinnvoll separiert werden können. Ihre Wirkrichtung ist in den seltensten Fällen eindeutig zu klassifizieren, da sie meist als Verbindungselement fungieren und dabei sowohl mit Komponenten der Klassen 1 oder 3 wie auch der Klassen 2 oder 4 interagieren.

## 6.2 Modellierung unter funktionalen Aspekten

Die erste Stufe der für die Analyse notwendigen Modellierung beschränkt sich auf die Beschreibung eines funktionalen Systemverhaltens. Die wesentlichen Bestandteile des Gesamtsystemmodells sind in dieser Modellierungsstufe nominale Verhaltensbeschreibungen für das zu untersuchende Kraftfahrzeugsystem, das Fahrzeug, den Fahrer und die Fahrzeugumgebung.

### 6.2.1 Modell des zu untersuchenden Kraftfahrzeugsystems

Eine allgemeine Struktur einer Verhaltensbeschreibung für ein Kraftfahrzeugsystem unter dem Aspekt der funktionalen Modellierung zeigt beispielhaft die Abbildung 6.3. Es ist zu erkennen, dass diese Modellierungsstufe sowohl in der Zielsetzung als auch in der Realisierung mit der aus der Regelungstechnik bekannten Vorgehensweise beim Entwurf eines Regelalgorithmus zu vergleichen ist. Bei der funktionalen Modellierung handelt es sich um eine zeit-kontinuierliche Beschreibungsform der systemdynamischen Vorgänge im Kraftfahrzeugsystem. Die strukturellen Aspekte einer konkreten Realisierungsform, wie z.B. die technischen Merkmale der verwendeten Sensorik oder deren tatsächliche Schnittstellen, als auch die Abweichungen vom nominalen Systemverhalten bleiben unberücksichtigt.

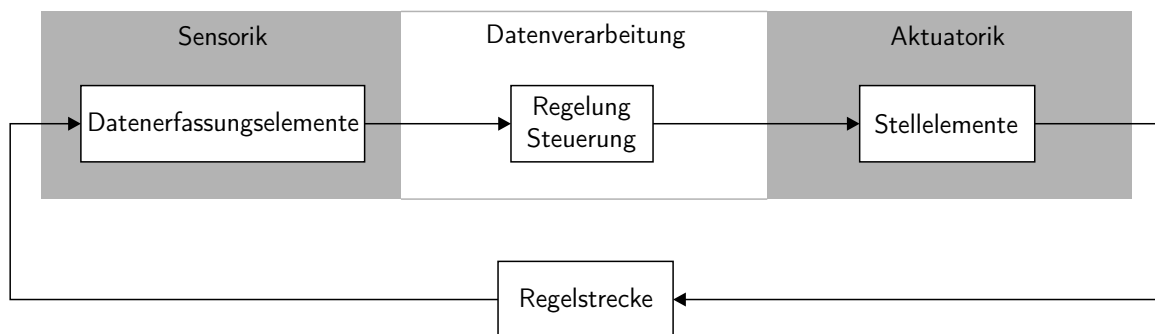


Bild 6.3: Beispielhaftes Modell unter funktionalen Aspekten

### 6.2.2 Fahrzeugmodell

Anhand der theoretischen Modellbildung soll die Kinematik eines Fahrzeugs in ihren Einheiten quantitativ beschrieben werden. Ziel ist es ein Referenzmodell zu erstellen, das eine ausreichende Genauigkeit besitzt, um Größen des Fahrzeugs im modellierten Kraftfahrzeugsystem zu untersuchen. Dabei sind alle wichtigen konstruktiven Merkmale der Kinematik, der

Elastokinematik und der Dynamik zu berücksichtigen. Auf Details von Einzelkomponenten sollte jedoch nur in soweit eingegangen werden, wie diese für die Gesamtsystemmodellierung des zu untersuchenden Kraftfahrzeugsystems relevant sind.

Das Fahrzeugmodell stellt im Zusammenhang mit der modellbasierten Sicherheitsanalyse nur ein Hilfsmittel zur Bestimmung der Auswirkungen auf Gesamtsystemebene dar. Da die zugrunde liegende Theorie sehr umfangreich und bereits in zahlreichen Arbeiten dokumentiert ist [Zom91, Mit95, Mit97, Mit90, Rei92, Rei86a, Rei83, Rei86b, Rei95], wird an dieser Stelle auf eine weitere allgemeine Beschreibung zur Fahrzeugmodellierung verzichtet. Für den konkreten Anwendungsfall, der als Nachweis der Tauglichkeit der in dieser Arbeit präsentierten Methodik zur System- und Sicherheitsanalyse dient, wird in Kapitel 8 das eingesetzte Fahrzeugmodell beschrieben.

### 6.2.3 Fahrermodell

Für die ganzheitliche Systemanalyse ist die Modellbildung des Fahrzeugs alleine nicht ausreichend. Der Fahrer hat Einfluss auf das Gesamtsystemverhalten und muss deshalb im ganzheitlichen Systemmodell berücksichtigt werden. Während das Fahrzeugmodell sowohl in der Literatur als auch in kommerziellen Modellen einen recht hohen Stellenwert erlangt hat, ist dagegen die Modellierung des menschlichen Fahrerverhaltens vergleichsweise untergeordnet behandelt worden.

Die wenigen kommerziell verfügbaren Fahrermodelle, wie z.B. der IPG-Driver [IPG04], beschränken sich meist auf die Nachbildung optimalen Fahrerverhaltens und finden Anwendung z.B. bei der Rundenzeitoptimierung im Motorsport. Die ganzheitlich, modellbasierte Systemanalyse stellt jedoch ganz andere Anforderungen an ein Fahrermodell. Hier besteht das Ziel der Modellierung des Fahrerverhaltens in der Beschreibung der Stärken und Schwächen eines Menschen beim Führen eines Fahrzeugs, um so anhand der Simulation die Auswirkungen möglicher Fehler von Fahrzeugsystemen auf den Fahrer und damit auch auf die Fahrzeugbewegung beurteilen zu können. Vor diesem Hintergrund wird deutlich, dass ein sowohl auf das zu untersuchende Fahrzeugsystem angepasstes als auch auf die daraus resultierenden Einflussmöglichkeiten auf den Fahrer abgestimmtes Fahrermodell notwendig ist.

In der Literatur finden sich einige Ansätze zur Modellierung des Fahrerverhaltens, die sich meist an regelungstechnischen Methoden orientieren. Eine Übersicht verschiedener Ansätze ist Gegenstand in den Arbeiten [Jür97, Rei90, Mac03], die auch auf die Eigenschaften des Menschen beim Führen eines Fahrzeugs, wie z.B. Wahrnehmungsschwellen und Totzeiten, eingehen. Für den konkreten Anwendungsfall steer-by-wire, der die Umsetzung der hier erarbeiteten Methoden

am Praxisbeispiel darstellt, wird in Kapitel 8 ein angepasstes Fahrermodell auf Basis von [Sta06] beschrieben, das den Anforderungen der ganzheitlichen, modellbasierten Systemanalyse für ein elektromechanisches Lenksystem gerecht wird.

### 6.2.4 Umweltmodellierung

Um die Modellierung unter funktionalen Aspekten zu komplettieren, ist die Erstellung einer Nachbildung der Systemumwelt von Nöten. Dazu ist ein Straßenmodell erforderlich mit dessen Hilfe Bahnverläufe unter Einbeziehung äußerer Umwelteinflüsse, beispielsweise Wind oder Fahrbahnoberflächenbeschaffenheiten, für die Gesamtsystem-Simulation festgelegt werden können. Im Einzelnen müssen zur Definition eines Bahnverlaufs folgende Parameter festgelegt werden: Koordinaten in x- und y-Richtung, Bahnkrümmung, Sollgierwinkel, Fahrbahnquer- und Fahrbahnlängsneigung, Fahrbahnoberflächeninformationen und Witterungsbedingungen.

Mit Hilfe der Unterteilung des gesamten Straßenverlaufs in einzelne Segmente unterschiedlicher Ausprägung gelingt es, Bahnverläufe relativ einfach zu definieren. Bild 6.4 zeigt einen exemplarischen Bahnverlauf in neun Segmenten.

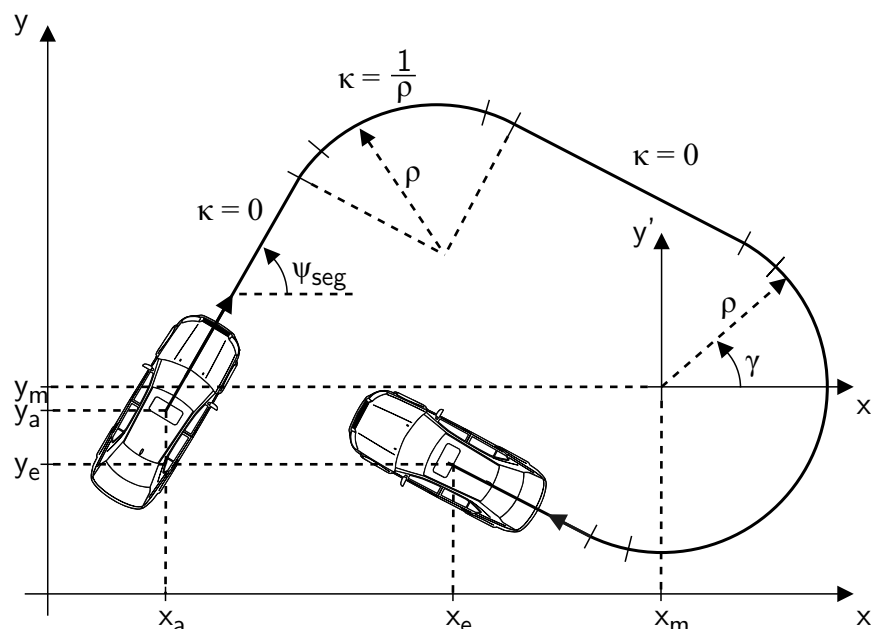


Bild 6.4: Kursdefinition durch Segmentierung

Tatsächlich sind drei Segmentinstanzen ausreichend, um einen Bahnverlauf vollständig zu beschreiben:



**Geradensegmente** stellen die einfachsten Bahnformen eines Straßenmodells dar. Formal lassen sie sich im globalen Koordinatensystem mit Hilfe einer Geradengleichung  $y = m \cdot x + c$  beschreiben. Ihre Bahnkrümmung  $\kappa$  ist gleich Null und der Kurvenradius  $\rho$  entspricht unendlich. Die weiteren zur Beschreibung dieses Segments notwendigen Parameter beschränken sich damit auf die Segmentlänge  $l$  und den Neigungswinkel  $\phi$ .

**Kreisbogensegmente** beschreiben die Kurvenführung einer Bahn. Sie zeichnen sich durch einen konstanten Kurvenradius  $\rho$  und damit durch eine ebenfalls konstante Bahnkrümmung  $\kappa$  aus. Die mathematische Darstellung erfolgt mit Hilfe von Polarkoordinaten zu

$$x = \rho \cdot \cos \gamma(t) \quad (6.1)$$

$$y = \rho \cdot \sin \gamma(t) \quad (6.2)$$

$$\rho = \sqrt{x'^2 + y'^2} \quad (6.3)$$

$$\gamma = \arctan \frac{y'}{x'} = \arctan \left( \frac{y - y_m}{x - x_m} \right) \quad (6.4)$$

Die weiteren zur Beschreibung dieses Segments notwendigen Parameter beschränken sich damit auf die Segmentlänge  $l$  und den Neigungswinkel  $\phi$ .

**Klothoidensegmente** ermöglichen den Übergang zwischen Geraden- und Kreisbogenstücken, da sie die notwendige, stetige Fortsetzung des Straßenverlaufs mit stetiger Bahnkrümmung gewährleisten. Sie zeichnen sich durch die Eigenschaft aus, dass sich ihre Bahnkrümmung  $\kappa$  mit der Segmentlänge  $l$  linear verändert.

$$\rho(s) = \frac{v_{ref}^2}{\pi s} T^2 \quad (6.5)$$

$$x(s) = x_m + \int_0^s \cos \left( \frac{\pi s^2}{2T^2 v_{ref}^2} \right) ds \quad (6.6)$$

$$y(s) = y_m + \int_0^s \sin \left( \frac{\pi s^2}{2T^2 v_{ref}^2} \right) ds \quad (6.7)$$

### 6.3 Modellierung unter strukturellen Aspekten

Als zweite Modellierungsstufe ist die Berücksichtigung struktureller Aspekte anzusehen. Dabei werden die bisher rein funktionalen Modelle um Beschreibungen des Systemzeitverhaltens, der Systemarchitektur, der tatsächlich verwendeten Sensor- und Aktuatorkonzepte sowie der real existierenden Schnittstellen erweitert. Hierbei ist eine Beschränkung auf das zu untersuchende Objekt zulässig, da es sich ausschließlich um Modellerweiterungen handelt, die sich auf die Beschreibung des Kraftfahrzeugmodells beziehen.

### 6.3.1 Erweiterung des Kraftfahrzeugsystemmodells

Die erweiterte Struktur des Gesamtsystemmodells unter strukturellen Aspekten ist anhand des bereits eingeführten Beispielsmodells in der Abbildung 6.5 dargestellt. Es wird deutlich, dass in dieser Modellierungsstufe die Systemarchitektur in der Modellbeschreibung Berücksichtigung findet. Wichtigstes Merkmal ist die Einführung einer Komponentensichtweise. Im Gegensatz zur funktionalen Modellierung, die ausschließlich Funktionen nachbildet und die Frage, wer diese Funktionen erbringt, unbeantwortet lässt, spielt bei der Modellierung unter strukturellen Aspekten die Verteilung der Funktionen auf Komponenten eine Rolle. Damit wird eine Beschreibung der tatsächlich zum Einsatz kommenden Komponenten, deren Funktionen und insbesondere deren Schnittstellen erforderlich.

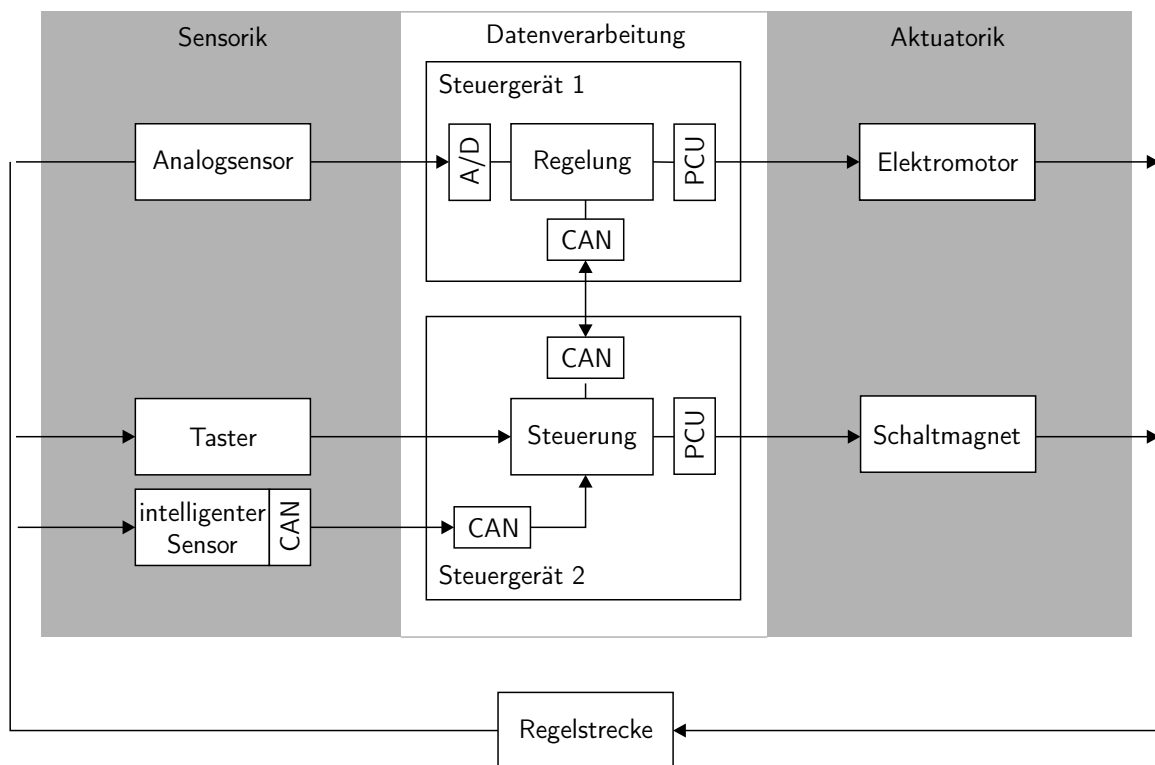


Bild 6.5: Beispielhaftes Modell unter strukturellen Aspekten

Durch die Einführung der Komponentensichtweise werden aber auch redundante und verteilte Funktionen Teil der Modellbeschreibung. Im Zuge der Berücksichtigung von Redundanz und Verteiltheit entstehen jedoch neue Funktionen, die bisher in der funktionalen Modellierung keine Rolle spielten. Hierzu gehören z.B. Voting-Mechanismen für redundante Sensorsignale.

Darüber hinaus erfordert die Modellierung unter strukturellen Aspekten die Betrachtung des Zeitverhaltens des zu modellierenden Kraftfahrzeugsystems. Insbesondere Komponenten-

schnittstellen, die durch ein Kommunikationssystem ausgezeichnet sind, haben erheblichen Einfluss auf das Zeitverhalten. Dieser Einfluss verstärkt sich nochmals, wenn, wie es in vielen sicherheitsrelevanten Systemen der Fall ist, zeitgesteuerte Kommunikationssysteme eingesetzt werden. Diese zeitgesteuerten Konzepte erfordern bei der Modellierung des Zeitverhaltens außerdem die Berücksichtigung eines zeitgesteuerten Taskmanagements.

## 6.4 Modellierung unter Sicherheitsaspekten

Zur Untersuchung von Sicherheitsaspekten ist die bisherige Modellbildung noch nicht ausreichend. Zusätzlich zum nominalen Verhalten muss auch das Verhalten des Systems im Fehlerfall abgebildet werden. Diese Anforderung bringt die Notwendigkeit einer Reihe von zusätzlichen Erweiterungen des Gesamtsystemmodells mit sich. In Abbildung 6.6 sind die Erweiterungen unter Sicherheitsaspekten anhand des eingeführten Beispielsmodells dargestellt.

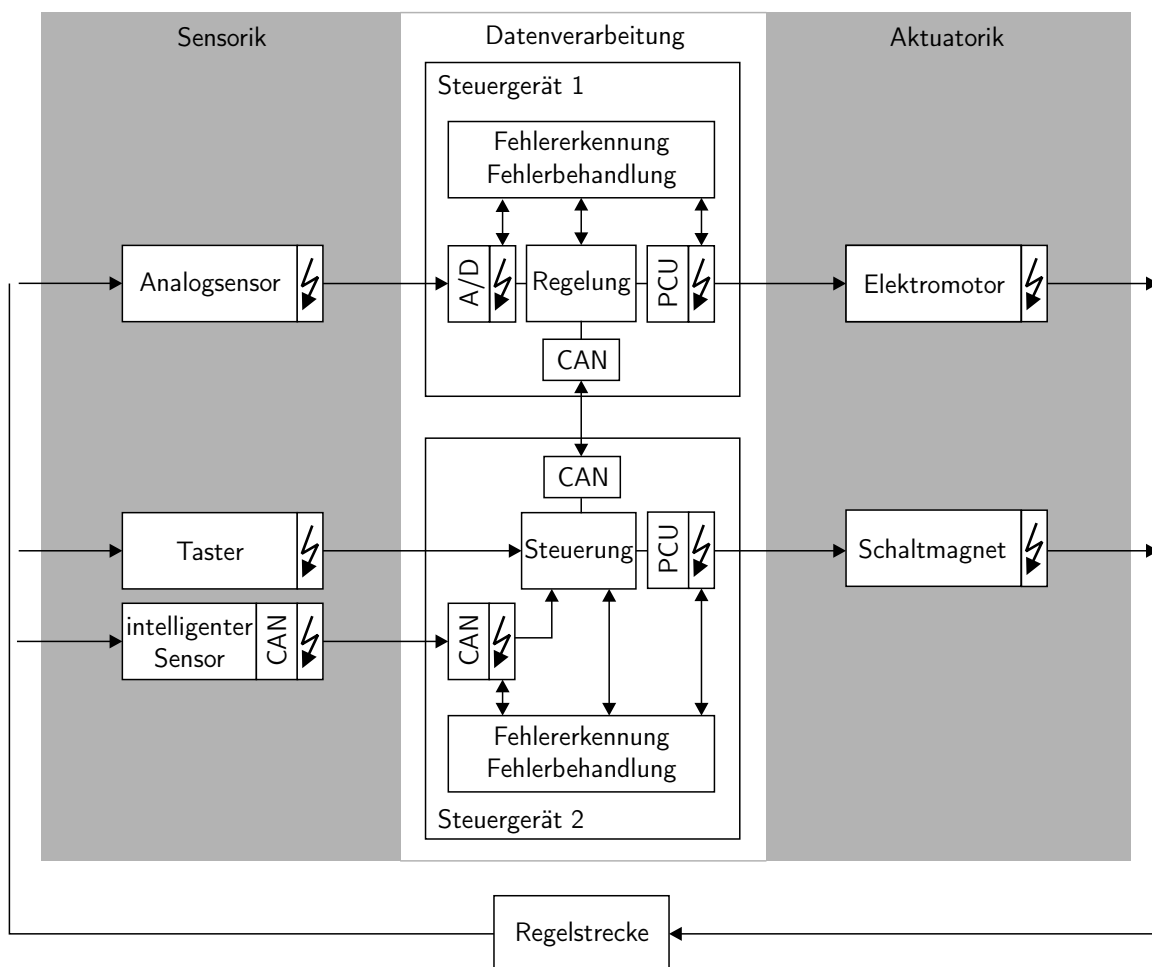


Bild 6.6: Beispielhaftes Modell unter Sicherheitsaspekten

### 6.4.1 Fehlermodelle

In Anlehnung an die mathematische Verhaltensbeschreibung einer Komponente oder Funktion im Nominalfall ist es möglich, das Verhalten auch im Fehlerfall nachzubilden. Diese Beschreibung wird Fehlermodell genannt. Grundsätzlich kann zwischen zwei Arten von Fehlermodellen unterschieden werden: den generischen und den spezifischen Fehlermodellen.

Ein generisches Fehlermodell ist die Verhaltensbeschreibung eines Fehlers, der unabhängig von einer bestimmten Komponente an sehr vielen Stellen im System auftreten kann. Insbesondere bei signalfluss-orientierten Systembeschreibungen sind generische Fehlermodelle geeignet, um Signale bei der Übertragung von einem Punkt des Systems zu einem anderen zu manipulieren. Dieser Signalfluss kann entweder zwischen oder innerhalb von Systemkomponenten stattfinden und darüber hinaus sowohl eine rein logische oder auch tatsächlich physikalische Realisierung aufweisen. An jeder dieser Stellen ist jedenfalls eine Veränderung des Signalflusses in seinem Werte- oder Zeitbereich durch einen Fehler möglich. Dabei spielt die eigentliche Funktion, bzw. die zugehörige Komponente, keine Rolle. Das der Signalveränderung zu Grunde liegende Fehlermodell ist allgemein gültig und an vielen Stellen im System einsetzbar. Ein mögliches generisches Fehlermodell zeigt die Abbildung 6.7.

Im Unterschied dazu berücksichtigen spezifische Fehlermodelle spezielle Eigenschaften einzelner Systemkomponenten und sind damit in ihrem Einsatz auf ausgewählte Fehlerorte begrenzt. Mittels einer FMEA können die Fehlermöglichkeiten einer speziellen Komponente systematisch ermittelt werden. Dabei geht sowohl das Wirkprinzip als auch die elektrische und mechanische Realisierung der Komponente in die Analyse der Fehlermöglichkeiten mit ein. Die Bestimmung der Fehlerauswirkungen ist lediglich auf Komponentenebene durchzuführen und gelingt damit im Vergleich zu der in Kapitel 4.1 erwähnten System-FMEA deutlich einfacher, objektiver und zuverlässiger.

Sehr wichtig für die spezifische Modellierung von Fehlern ist die Frage nach deren notwendigem Detaillierungsgrad. Generell sollte der Detaillierungsgrad einer Modellierung dem konkreten Anwendungsfall angepasst ausgewählt werden. Deshalb kann die Frage nach dem notwendigen Detaillierungsgrad eines Fehlermodells am geeignetsten mit der Zielsetzung der ganzheitlichen, modellbasierten Systemanalyse beantwortet werden. Wie in Kapitel 5 erläutert liegt ein Hauptuntersuchungsziel auf der Analyse der Fehlerpropagation im Gesamtsystem. Damit ist nicht der tatsächlich ablaufende physikalische Vorgang innerhalb einer Komponente im Fehlerfall von vordergründiger Bedeutung, sondern vielmehr die aus diesem Fehler resultierende Veränderung des Komponentenverhaltens. Für ein Fehlermodell in diesem Anwendungskontext ist es somit ausreichend eben dieses Verhalten an den Schnittstellen der Komponente abzubilden.

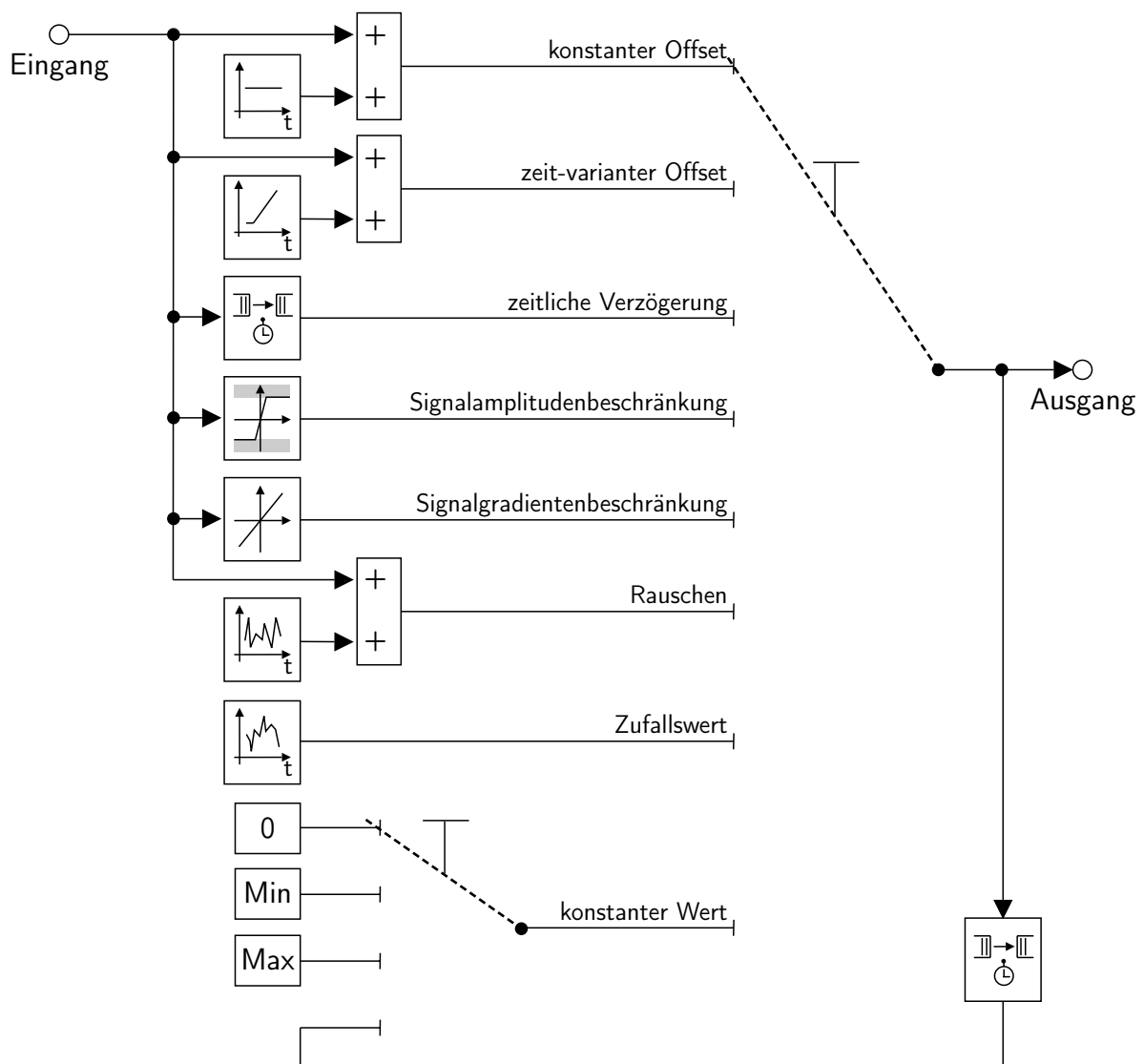


Bild 6.7: Generisches Fehlermodell

Diese Nachbildung gelingt meist erheblich leichter und exakter als der Versuch die tatsächlichen physikalischen Abläufe innerhalb einer Komponente im Fehlerfall zu modellieren.

Darüber hinaus spielt für den Detaillierungsgrad auch die Eingruppierung der Komponente in eine der vorgestellten Komponentenklassen eine große Bedeutung. Insbesondere bei Komponenten, die durch eigene Intelligenz, Fehlererkennungs- und -behandlungsmechanismen fail-silent Eigenschaften besitzen, kann die Frage nach dem notwendigen Detaillierungsgrad der Fehlermodellierung erneut mit dem konkreten Untersuchungsziel beantwortet werden. Komponenten, die, wie in der Klasse 3 und 4 definiert, ihre eigenen Fehlererkennungs- und -behandlungsmechanismen mit sich bringen, weisen im Fehlerfall fail-silent Eigenschaften auf. Aus globaler Sicht spielt deshalb ein Fehler innerhalb einer Komponente und der damit verbun-

dene Fehlererkennungsmechanismus keine Rolle. Für die Analyse ist hingegen von Bedeutung, in welcher Art und Weise das Verhalten der Komponente durch die Fehlerbehandlungsmechanismen geändert wird. Im Hinblick auf eine Komponente der Klasse 3 oder 4 bedeutet dies, dass die Zielsetzung nicht die Frage nach der Überprüfung der fail-silent Eigenschaft der Komponente selbst ist, sondern vielmehr deren Integration in einen Verbund aus anderen Komponenten und die daraus hervorgehenden Fehlerszenarien im Gesamtsystem selbst. Daraus lässt sich ableiten, dass auf eine Modellierung der Fehlermöglichkeiten und den dazugehörigen Fehlererkennungsmechanismen auf Komponenten-Ebene verzichtet werden kann. Wichtig ist hingegen eine Berücksichtigung des durch die Fehlerbehandlungsmechanismen hervorgerufenen äußeren Verhaltens der Komponente. Dies ist überwiegend bestimmt durch die Zustände, die die Komponente im Nominal- und Fehlerfall einnehmen kann sowie durch die jeweiligen Bedingungen, die zu einem Übergang zwischen den Zuständen führen. Diese Bedingungen sind mit den Fehlermöglichkeiten der Komponente direkt verknüpft und können im Laufe des Identifikationsprozesses von einer Fehlerinjektion eingespeist werden.

Die Komponenten der Klassen 1 und 2 verfügen nicht über diese fail-silent Eigenschaft. Fehler innerhalb dieser Komponenten zeigen sich deshalb an deren Schnittstellen und breiten sich darüber im Gesamtsystem aus. Die zugehörigen Fehlermodelle sind so zu gestalten, dass das durch einen Fehler hervorgerufene Verhalten an den Komponentenschnittstellen abgebildet wird. In der Regel kommen in diesen Fällen auf System-Ebene Fehlererkennungs- und -behandlungsmechanismen zur Vermeidung der Fehlerausbreitung zum Einsatz. Sie sind Teil des Gesamtsystems und müssen deshalb bei der ganzheitlichen, modellbasierten Systemanalyse berücksichtigt und damit ebenfalls modelliert werden.

### 6.4.2 Erweiterung des Kraftfahrzeugsystemmodells

Sowohl die Fehlererkennungs- als auch die Fehlerbehandlungsmechanismen eines Systems spielen für die Analyse des Systemverhaltens hinsichtlich dessen Sicherheitsaspekten die zentrale Rolle. Fehler im System müssen erkannt und vom System angemessen behandelt werden, um die Betriebssicherheit eines Kraftfahrzeugsystem gewährleisten zu können. Daraus resultiert die Un-erlässlichkeit einer Beschreibung der Fehlererkennungsmechanismen für ein Modellierung unter Sicherheitsaspekten. Darüber hinaus ist auf einen erkannten Fehler entsprechend zu reagieren und damit eine Nachbildung der Fehlerbehandlungsmechanismen und Systemrückfallebenen erforderlich. Diese möglichen Systemzustände können mittels Zustandsautomaten mit ihren jeweiligen Zustandübergängen in der hybriden Modellierungssprache realisiert werden.

### 6.4.3 Fehlerinjektion

Zur Aktivierung der beschriebenen Fehlermodelle wird auf die Mechanismen der Fehlerinjektion zurückgegriffen. Sie stellt die Mittel zur Verfügung, um die beschriebenen Fehlermodelle im Laufe der Modellausführung zu aktivieren bzw. zu deaktivieren, um so das Auftreten permanenter und transienter Fehler auf System-Ebene nachzubilden. Gemäß den erarbeiteten Anforderungen muss diese Injektion zu jedem beliebigen Zeitpunkt möglich sein. Dazu werden alle Fehlerorte auf System-Ebene mit einer Identifikationsnummer versehen. Mit jedem Fehlerort ist eine Vielzahl zur Injektion zugelassener Fehlermodelle verknüpft. Dabei kann es sich um generische oder spezifische Fehlermodelle handeln. So kann verhindert werden, dass unzulässige Kombinationen aus Fehlerort und Fehlermodell untersucht werden. Aus diesen beiden Identifikationsnummern für Fehlerort und Fehlermodell wird die so genannte *Fehlernummer* bestimmt, die genau ein Fehlermodell an genau einem Fehlerort im Gesamtsystem beschreibt.

Zur Injektion des Fehlers mit der zugehörigen Fehlernummer wird vor Ausführung des Gesamtsystemmodells die Verhaltensbeschreibung für den Nominalfall durch die Verhaltensbeschreibung für den Fehlerfall ersetzt. Dieser Austausch erfolgt vollständig automatisiert und unter Beachtung der zugehörigen Parameter für die Fehleraktivierungs- bzw. -deaktivierungszeitpunkte.

### 6.4.4 Objektiv-quantifizierbare Bewertungsfunktion

Um der Forderung nach einer automatisierten Systemanalyse gerecht zu werden, muss die zum einen zeitintensive und zum anderen fehlerträchtige Bewertung der Funktion und Sicherheit des Kraftfahrzeugsystems durch menschliche Experten ersetzt werden. Insbesondere auch im Hinblick auf die Anforderung der Reproduzierbarkeit ist eine Abkehr vom herkömmlichen Bewertungsprozess durch Experten-Teams unumgänglich.

Wünschenswert ist hingegen ein mathematischer Algorithmus, der das Systemverhalten sowohl unter funktionalen Aspekten als auch im Falle injizierter Fehler selbstständig bewertet. Nahe liegend für eine automatisierte Bewertung ist die Verwertung der durch die Gesamtsystemsimulation ohnehin verfügbaren Fahrzeugdaten. Hierzu zählt z.B. die laterale Fahrzeugbewegung bei der Untersuchung von Lenksystemen sowie die Fahrzeugverzögerung bei der Analyse von Bremssystemen. Eine Bewertung aufgrund von Daten der Fahrzeugebene fällt meist leichter als eine auf spezieller Systemparameter basierende Bewertung. Aus funktionaler Sicht ist vor allem der Vergleich von Ist- und Solldaten als ein geeignetes Bewertungskriterium anzusehen. Unter dem Aspekt der Sicherheitsrelevanz kann vor allem das unterschiedliche Systemverhalten im fehlerfreien und fehlerbehafteten Fall als Kriterium dienen.

Darüber hinaus sind weitere problemspezifische Kriterien zur Bewertung des Systemverhaltens denkbar. Wie im Abschnitt 6.2.3 beschrieben, besitzt der menschliche Fahrzeugführer basierend auf seinen Erfahrungen eine gewisse Erwartung vom Verhalten seines Fahrzeugs. Diese Erfahrung ist auch im Modell des Fahrers abgebildet. Weicht das tatsächliche vom erwarteten Fahrzeugverhalten ab, so wird dies dem Fahrer auffallen und je nach Ausprägung eine Empfindung von Störung bis Gefahr in ihm hervorrufen. Wird dieser Unterschied zwischen Ist- und Erwartungsverhalten durch einen numerischen Wert ausgedrückt, so kann er als weiteres Kriterium zur Bewertung herangezogen werden.

Auch interne Größen und Zustände des zu untersuchenden Systems selbst können als Indikatoren für eine mathematisch beschreibbare Bewertungsfunktion herangezogen werden. Insbesondere Fehlererkennungsmechanismen oder Zustandswechsel in den Betriebsmodi eines Fahrzeugsystems können ihren Anteil zur Bewertungsfunktion beitragen. Müssen z.B. aufgrund eines injizierten Fehlers Komponenten abgeschaltet und damit die Funktion eingeschränkt bzw. auf Fehlertoleranzmöglichkeiten verzichtet werden, so kann dies in einen Bewertungsalgorithmus einfließen.

Die hier erwähnten möglichen Bestandteile einer Bewertungsfunktion stellen nur eine kleine Auswahl denkbarer Einflussgrößen dar. Je nach spezifischem Anwendungsgebiet variieren die Bewertungskriterien und deren Gewichtung für das Gesamtergebnis. Leider ist es in den wenigsten Fällen möglich, tatsächlich eine allgemein gültige, objektive Bewertungsfunktion zu finden. Hierfür sind die Empfindungen für Gefahr bei jedem Menschen aufgrund seiner individuellen Erfahrungen zu unterschiedlich ausgeprägt. Die Auswahl und Gewichtung der Bewertungskriterien muss deshalb weiterhin in einem Abstimmungsprozess durch ein Experten-Team erfolgen. Hierbei führt die Konsensbildung zwischen den persönlichen Einschätzungen der Experten zu einem gemeinschaftlich akzeptierten Regelwerk, das als Grundlage für die Erstellung des Bewertungsalgorithmus dient. Endergebnis ist eine Bewertungsfunktion, die ein objektiv-quantifizierbares Maß für das gezeigte Systemverhalten bietet und deren Kriterien klar im Algorithmus verankert und dokumentiert sind. Auf diese Weise kann sowohl den Anforderungen nach der Automatisierbarkeit als auch der Reproduzierbarkeit nachgekommen werden.



# 7 Identifikation signifikanter Fehler

Bisher stand vor allem das Modell des Gesamtsystems und dessen Beschreibung im Vordergrund der Betrachtungen. Die Durchführung einer Sicherheitsanalyse auf Basis dieses Modells bedingt aufgrund der enormen Menge an analysierbaren Fehlermöglichkeiten zusätzliche Maßnahmen. Das folgende Kapitel befasst sich deshalb zum einen mit der Automatisierung der modellbasierten Analyse und zum anderen mit der Beschränkung auf die relevanten Fehlermöglichkeiten. Dazu ist ein selbständig arbeitender, intelligenter Suchalgorithmus von Nöten, der auf Basis des Gesamtsystemmodells die signifikanten, sicherheitsrelevanten Fehler identifiziert.

## 7.1 Algorithmen zur globalen Suche

Zur Lösung dieser Aufgabe sind mehrere durchaus sehr unterschiedliche Ansätze denkbar. Die charakteristischen Eigenschaften der bekanntesten Vertreter der Algorithmen zur globalen Suche und Optimierung werden im Folgenden kurz vorgestellt.

### 7.1.1 Monte Carlo Methode

Eine Monte Carlo Simulation [Gen03] ist ein stochastisches Verfahren zur Lösung komplexer, mathematischer Probleme, die nicht oder nur sehr schwierig deterministisch gelöst werden können. Die Bezeichnung ist eine Anspielung auf den für Glücksspiele bekannten monegasischen Ort, da die Grundlage des Verfahrens Zufallszahlen sind, wie man sie auch mit einem Roulette-Rad erzeugen könnte. Die meisten Such- und Optimierungsverfahren, die auf Monte Carlo Methoden basieren, vollziehen zufällige Wege durch den zumeist mehrdimensionalen Lösungsraum. Dabei wird eine Vielzahl an unterschiedlichen Lösungsvektoren ausprobiert, um eine hinsichtlich eines Optimierungskriteriums verbesserte Problemlösung zu finden.

### 7.1.2 Simulated Annealing

Einen deutlichen Vorteil gegenüber der Monte Carlo Methode haben die so genannten gerichteten Such- und Optimierungsverfahren. Dabei handelt es sich um iterative Algorithmen, die

die Informationen aus Lösungsvorschlägen vorhergehender Iterationen zum Auffinden besserer Lösungsvorschläge verwenden. Zu diesen Verfahren zählt auch die Simulated Annealing Methode [KGV83, Vid93]. Vorbild für dieses Verfahren ist der langsame und kontrollierte Abkühlungsprozess beim Härten von Metallen. Während dieses Vorgangs finden die Metallatome eine hinsichtlich ihrer Stabilität bzw. Energie verbesserte Anordnung. Um die Simulated Annealing Methode auf ein Problem anwenden zu können, ist die Definition von einigen Parametern wie z.B. der Abkühlungsfunktion oder der Akzeptanzwahrscheinlichkeit notwendig. Die Wahl dieser Parameter ist entscheidend für die Rechenzeit und das Ergebnis des Algorithmus. Leider gestaltet sich das Auffinden geeigneter Parameter oft sehr schwierig und gelingt meist nur mit viel Erfahrung und Kenntnis über die Problemfunktion.

### 7.1.3 Genetische Algorithmen

Genetische Algorithmen [Gol89, Rec94] sind ebenfalls globale und gerichtete Such- und Optimierungsverfahren, die als Vorbild die biologische Evolution haben. Sie zählen zu den metaheuristischen Verfahren, die vor allem bei Problemen eingesetzt werden, für die keine analytische Lösung angegeben werden kann. Genetische Algorithmen gelangen zu ihrer Lösung, indem sie eine Vielzahl möglicher Lösungsvorschläge kontinuierlich verändern und miteinander kombinieren, bis einer dieser Vorschläge den Abbruchbedingungen gerecht wird. Um den Anforderungen von genetischen Algorithmen gerecht zu werden, müssen alle Veränderlichen des tatsächlichen Problems einer binären Repräsentation zugeordnet werden. Die Art und Weise dieser Zuordnung hat jedoch erheblichen Einfluss auf den Suchprozessverlauf und das Suchergebnis.

## 7.2 Evolutionäre Programme

Evolutionäre Programme basieren auf den Grundideen der genetischen Algorithmen. Sie haben jedoch eine Erweiterung in Bezug auf die verwendeten Datenstrukturen erfahren. Können die genetischen Algorithmen nur mit binären Repräsentationen umgehen, so ist es bei den evolutionären Programmen möglich, globale Such- und Optimierungsprobleme zu lösen, die auch ganzzahlige oder realwertige Variablen aufweisen. Damit entfällt die komplexe Zuordnung der Variablen zu einer binären Darstellung.

Das hier eingesetzte Verfahren ist dem Anwendungsgebiet angepasst und unterscheidet sich von den klassischen Verfahren, die in [Poh00, Mic99, GKK04] vorgestellt werden, teils deutlich. Aus diesem Grund wird im Folgenden detailliert auf die einzelnen Operatoren und Methoden eingegangen.

### 7.2.1 Aufbau und Funktionsweise

Betrachtet man die Natur, so ist augenscheinlich, dass die äußerst unterschiedlichen Lebewesen gut an ihren jeweiligen Lebensraum angepasst sind. Jedes Individuum hat sich im Laufe der Evolution auf seine Umgebungsbedingungen eingestellt und optimierte Eigenschaften entwickelt, die ein Leben unter den jeweiligen Bedingungen ermöglichen oder erleichtern. Diese Eigenschaften sind in den Genen der Lebewesen gespeichert und können so auch an ihre Nachkommen weitergegeben werden. Überträgt man diese Beobachtung in eine technische Betrachtung, so lässt sich schließen, dass jedes Individuum eine bestimmte Kombination von Variablen darstellt. Mathematisch gesehen repräsentiert ein Vektor bestehend aus einer Menge Veränderlicher jedes Individuum  $i_n^t = (v_1, v_2, \dots, v_k)$ . Diese Kombination wird im Lebensraum, der so genannten Zielfunktion oder Fitnessfunktion, immer wieder neu im Sinne der Überlebenswahrscheinlichkeit, dem so genannten Zielfunktionswert oder Fitnesswert, für jedes Individuum bewertet. Je besser ein Individuum an seine Lebensbedingungen angepasst ist, desto größer ist auch dessen Überlebenswahrscheinlichkeit. Das Prinzip „Der Stärkere überlebt“ wird dabei sowohl gleichzeitig auf  $n$  Individuen, die so genannte Population  $P(t)$ , als auch über einen Zeitraum, den so genannten Generationen  $t$ , angewendet.

Überträgt und vereinfacht man die komplexen Vorgänge in der Natur auf einen technischen Prozess, so entsteht ein Ablauf gemäß Bild 7.1 - das evolutionäre Programm.

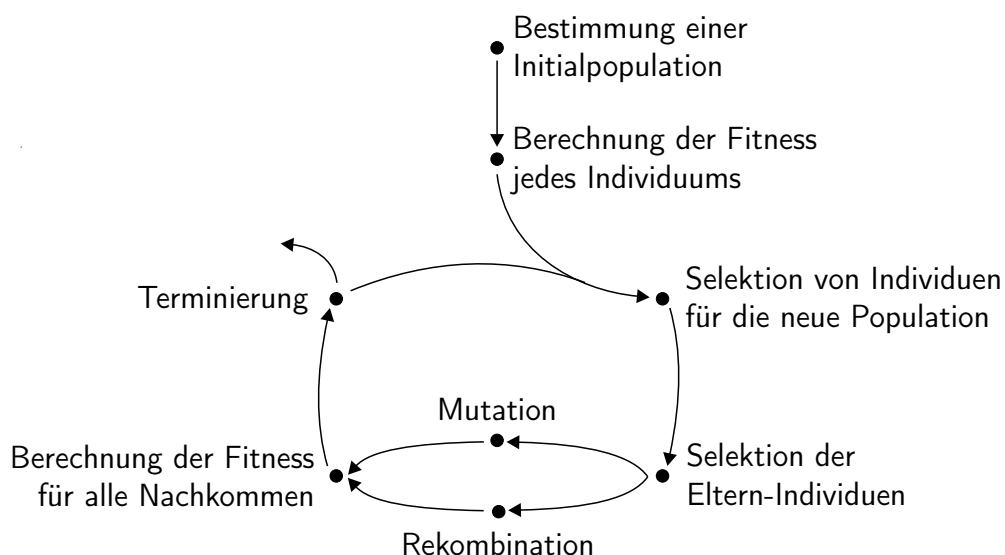


Bild 7.1: Ablaufdiagramm eines evolutionären Programms

Zu Beginn eines evolutionären Programms erfolgt die Initialisierung. Diese beinhaltet neben der Parameterierung vor allem die Erstellung der Anfangspopulation. Normalerweise werden

die Individuen der Anfangspopulation zufällig im Definitionsbereich der Variablen initialisiert. Die erstellte Anfangspopulation wird mit Hilfe der Fitnessfunktion bewertet und stellt die Population der ersten Generation dar.

Damit kann der eigentliche evolutionäre Kreislauf und die Erstellung einer neuen Generation beginnen. Gemäß eines so genannten Selektionsverfahrens werden zunächst die Individuen aus der ersten Generation ausgewählt, die überleben und in der neuen Generation fortbestehen sollen. Ein gewisser vorher festgelegter Anteil der Population stirbt aus. Um die freien Plätze in der Population wieder auszufüllen, selektiert ein zweites Auswahlverfahren aus der vorangegangenen Generation die Individuen, die als Eltern für die Produktion von Nachkommen für die neue Generation verantwortlich sind. Diese Nachkommen entstehen durch Anwendung so genannter evolutionärer Operatoren und füllen die Population der neuen Generation zu konstanter Größe. Damit ist gewährleistet, dass besonders geeignete Individuen mit hoher Wahrscheinlichkeit sowohl in einer neuen Generation überleben als auch zur Produktion von Nachkommen herangezogen werden.

Die Bewertung der neu entstandenen Generation erfolgt erneut mit der Fitnessfunktion. Wenn ein definiertes Terminierungskriterium noch nicht erfüllt ist, beginnt der Kreislauf von vorne. Als Terminierungsfunktion sind unter anderem z.B. das Erreichen eines festgelegten Fitnessfunktionswertes, einer vorher definierten maximalen Anzahl entwickelter Generationen oder das Verstreichen einer wählbaren Anzahl an Generationen ohne Fortschritt, denkbar.

Die im vorliegenden Anwendungsfall eingesetzten Selektionsverfahren und evolutionären Operatoren werden in den folgenden Abschnitten näher erläutert.

### 7.2.2 Selektionsverfahren

Das im vorliegenden Anwendungsfall eingesetzte Selektionsverfahren sieht folgende Arbeitsschritte vor:

**Schritt 1** Auswahl von  $(popsize - d)$  Individuen aus Population  $P(t)$  als Basis für die neue Population  $P(t+1)$  der nächsten Generation. Dabei handelt es sich bei der Größe *popsize* um die konstante Populationsgröße und bei  $d$  um die Anzahl der Individuen, die von einer Generation zur nächsten aussterben.

**Schritt 2** Selektion von  $d$  Eltern-Individuen aus der Population  $P(t)$ . Jedes ausgewählte Individuum wird genau einem evolutionären Operator fest zugeordnet.

**Schritt 3** Aus den  $d$  Eltern-Individuen werden durch die direkt zugeordneten evolutionären Operatoren genau  $d$  Nachkommen erzeugt.

Im Gegensatz zur klassischen Vorgehensweise, die zur Produktion von mehrfachen, exakten Kopien eines Individuums in einer Population neigt, verbessert diese Vorgehensweise die Ausnutzung der Ressource Population. Insbesondere die Wahrung der Diversität durch Vermeidung exakter Kopien kann erreicht und damit die Effizienz des Suchalgorithmus gesteigert werden.

Beim eigentlichen Selektionsalgorithmus kommt das *stochastic universal sampling* zum Einsatz, das die Individuen entsprechend ihres absoluten Fitnessfunktionswertes aus dem Selektionspool auswählt. Damit gehört das *stochastic universal sampling* zur Gruppe der fitnessproportionalen Selektionsverfahren, das nach folgendem Prinzip arbeitet:

Allen Individuen eines Selektionspools werden jeweils einzelne Abschnitte einer Linie zugeordnet. Die Größe eines Abschnittes entspricht der Fitness des jeweiligen Individuums. Über der Linie sind, wie in Bild 7.2 dargestellt, Zeiger äquidistant im Abstand  $\frac{1}{\text{Anzahl Zeiger}}$  angeordnet, deren Anzahl über die Anzahl der ausgewählten Individuen bestimmt. Die Zeiger werden mit Hilfe einer Zufallszahl im Bereich  $\left[0, \frac{1}{\text{Anzahl Zeiger}}\right]$  vom Nullpunkt der Linie aus zusammenhängend verschoben. Im letzten Schritt des Auswahlverfahrens werden die Individuen selektiert, deren Abschnitte die Zeiger markieren.

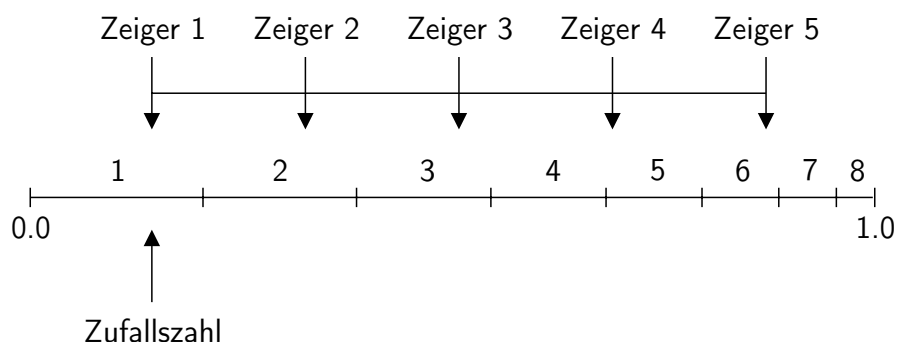


Bild 7.2: Selektion gemäß dem *stochastic universal sampling* Verfahren [Bak87]

### 7.2.3 Evolutionäre Operatoren

Evolutionäre Operatoren lassen sich in zwei Gruppen unterteilen: die Mutations- und die Rekombinationsoperatoren. Während die Mutationsoperatoren ausschließlich Veränderungen an den Variablen eines Eltern-Individuums vornehmen, benutzen die Rekombinationsoperatoren hingegen grundsätzlich zwei Eltern-Individuen, um durch gegenseitigen Austausch von Erbinformationen zwei Nachkommen zu erzeugen. Zur nachfolgenden Erläuterung der einzelnen im Identifikationsprozess für signifikante Systemfehler verwendeten Operatoren seien die Eltern-Individuen als  $i_n^t = (v_1, v_2, \dots, v_k)$  und  $i_m^t = (w_1, w_2, \dots, w_k)$  definiert:

**Uniform Mutation** Ist  $i_n^t = (v_1, v_2, \dots, v_k)$  ein Eltern-Individuum und eine zufällig ausgewählte Variable von  $i_n^t$  sei  $v_j$ , dann ergibt sich durch den Operator *Uniform Mutation* der neu erzeugte Nachkomme zu

$$i_n^{t+1} = (v_1, v_2, \dots, v'_j, \dots, v_k). \quad (7.1)$$

Dabei nimmt  $v'_j$  einen zufälligen Wert aus dem für diese Variable gültigen Wertebereich  $[l_j, u_j]$  an. Der Einsatz dieses Operators bewirkt während einer gerichteten globalen Suche das Vordringen in neue, bisher unbekannte Suchunterräume.

**Boundary Mutation** Die Wahrscheinlichkeit mit dem Operator *Uniform Mutation* auch die oberen bzw. unteren Grenzbereiche des Wertebereichs einer Variablen zu durchsuchen ist sehr gering. Aus diesem Grund stellt die *Boundary Mutation* eine abgewandelte Version der *Uniform Mutation* dar. Die Festlegung von  $v'_j$  gehorcht nun folgender Berechnungsvorschrift:

$$v'_j = \begin{cases} l_j & \text{falls } \text{Zufallszahl} = 0 \\ u_j & \text{falls } \text{Zufallszahl} = 1 \end{cases} \quad (7.2)$$

**Non-Uniform Mutation** Waren die bisherigen Operatoren unabhängig vom Alter der Population, so verändern sich die Eigenschaften des *Non-Uniform Mutation* Operators mit zunehmendem Suchfortschritt. Zu Beginn ist der zulässige Veränderungsspielraum an einem Eltern-Individuum durch diesen Operator noch sehr groß. Mit fortschreitendem Suchverlauf engen sich diese Veränderungsmöglichkeiten mehr und mehr ein. Am Ende des Suchprozesses sind schließlich nur noch geringe Manipulationen an den Eltern-Individuen erlaubt. Von einer anfänglichen globalen Suche hat sich die Zielsetzung des Operators im Laufe der Zeit zu einer lokalen Suche gewandelt. Damit ist dieser Operator für die Feinabstimmung während der gerichteten Suche verantwortlich. Seine mathematische Beschreibung ergibt sich folgendermaßen: Ist  $i_n^t = (v_1, v_2, \dots, v_k)$  ein Eltern-Individuum und eine zufällig ausgewählte Variable von  $i_n^t$  sei  $v_j$ , dann ergibt sich der neu erzeugte Nachkomme zu

$$i_n^{t+1} = (v_1, v_2, \dots, v'_j, \dots, v_k) \quad (7.3)$$

mit

$$v'_j = \begin{cases} v_j + \Delta(t, u_j - v_j) & \text{falls } \text{Zufallszahl} = 0 \\ v_j - \Delta(t, v_j - l_j) & \text{falls } \text{Zufallszahl} = 1 \end{cases} \quad (7.4)$$

wobei die Funktion  $\Delta(t, y)$  einen Wert im Bereich  $[0, y]$  liefert, der mit zunehmendem  $t$  immer näher bei 0 liegt. Die Wahl der hier eingesetzten Funktion

$$\Delta(t, y) = y \cdot \left(1 - r^{\left(1 - \frac{t}{T}\right)^b}\right) \quad (7.5)$$

ist auf eine Empfehlung in [Mic99] zurückzuführen. Dabei ist  $r$  eine Zufallszahl zwischen 0 und 1,  $T$  die maximale Anzahl an Generationen und  $b$  ein Parameter zur Festlegung der zeitlichen Veränderung des zulässigen Suchraums.

**Simple Crossover** Werden die Nachkommen  $i_n^{t+1}$  und  $i_m^{t+1}$  der Eltern-Individuen  $i_n^t$  und  $i_m^t$  auf folgende Art und Weise erzeugt

$$i_n^{t+1} = (v_1, v_2, \dots, v_j, w_{j+1}, \dots, w_k) \quad (7.6)$$

$$i_m^{t+1} = (w_1, w_2, \dots, w_j, v_{j+1}, \dots, v_k) \quad (7.7)$$

so spricht man vom Rekombinationsoperator *Simple Crossover*. Er dient zum Austausch von Informationen zwischen zwei Individuen, um neue Suchunterräume zu erschließen.

**Arithmetical Crossover** Ausgehend von den Eltern-Individuen  $i_n^t$  und  $i_m^t$  ist das Ergebnis des evolutionären Operators *Arithmetical Crossover* definiert als Linearkombination von  $i_n^t$  und  $i_m^t$  der Form:

$$i_n^{t+1} = b \cdot i_n^t + (1 - b) \cdot i_m^t \quad (7.8)$$

$$i_m^{t+1} = b \cdot i_m^t + (1 - b) \cdot i_n^t \quad (7.9)$$

wobei  $b$  entweder ein konstanter Parameter für einen zeitunabhängigen Operator oder eine Variable für einen vom Alter der Population beeinflussten Operator sein kann.

## 7.2.4 Erweiterungen zum Umgang mit Nebenbedingungen

Ein zentrales Problem der evolutionären Algorithmen stellt deren mangelhafte Unterstützung im Umgang mit Variablen-Nebenbedingungen dar. Darunter versteht man kombinatorische, lineare oder auch nicht-lineare Abhängigkeiten unter den Variablen eines Individuums. Die jeweiligen Definitionsbereiche einer Variablen sind damit veränderlich und insbesondere abhängig von der Wahl der anderen Variablenwerte eines Individuums. Diese Abhängigkeiten führen bei der Produktion von neuen Individuen mit evolutionären Operatoren im herkömmlichen, oben beschriebenen Sinne, immer wieder zu Nachkommen, die die Nebenbedingungen verletzen und aus diesem Grund keine gültigen Individuen darstellen.

Diese Problematik stellt sich auch im vorliegenden Anwendungsfall. So existieren bei der Suche nach signifikanten Fehlerauswirkungen diverse Abhängigkeiten unter den einzelnen Variablen eines Individuums, die zusammen ein Fehlerszenario definieren. Die Beschreibung eines solchen Fehlerszenarios kann beispielsweise die Variablen *Fehleraktivierungszeitpunkt* und *Fehlerdeaktivierungszeitpunkt* enthalten. Sie müssen in einer zeitlich richtigen Reihenfolge liegen, um eine sinnvolle Definition eines Fehlerszenarios zu repräsentieren. Ebenso spielen bei der Untersuchung von Fehlerkombinationen bzw. Fehlersequenzen die Abhängigkeit zwischen Erst- und Zweitfehler eine wichtige Rolle.

Eine oftmals angewandte Lösungsmöglichkeit ist die so genannte Bestrafungsmethode. Dabei werden ungültige Individuen in ihrer Bewertung durch die Fitnessfunktion deutlich herabgestuft. Die folgende Selektion ist dann in der Lage, diese ungültigen Individuen mit hoher Wahrscheinlichkeit aus der Population aus zu sortieren. Eine zweite Möglichkeit zur Behebung dieser Problematik repräsentieren die Reparaturfunktionen. Sie beheben eventuelle Verletzungen der Variablen-Beschränkungen, die bei der Produktion eines Individuums aufgetreten sind, im Nachhinein durch Anpassung der Variablenwerte.

Alle bisher erwähnten Verfahren zum Umgang mit Nebenbedingungen bringen jedoch deutliche Nachteile mit sich. Sie reichen vom Performance-Verlust über die Beschränkung der Allgemeingültigkeit durch speziell an das Problem angepasste Lösungsmethoden bis hin zur Gefahr von minderwertigen Suchergebnissen aufgrund zu weniger gültiger Individuen innerhalb einer Population. Deshalb basiert der hier verfolgte Ansatz auf der Verwendung angepasster evolutionärer Operatoren, die in der Lage sind, schon bei der Produktion der Nachkommen die Nebenbedingungen zu berücksichtigen. Ziel ist damit nicht die nachgelagerte Erkennung und Korrektur von ungültigen Individuen, sondern die Produktion ausschließlich gültiger Nachkommen. Die aus dieser Zielsetzung heraus erweiterten evolutionären Operatoren sind:

**Uniform Mutation** Ist  $i_n^t = (v_1, v_2, \dots, v_k)$  ein Eltern-Individuum und eine zufällig ausgewählte Variable von  $i_n^t$  sei  $v_j$ , dann ergibt sich durch den erweiterten Operator *Uniform Mutation* der neu erzeugte Nachkomme zu

$$i_n^{t+1} = (v_1, v_2, \dots, v'_j, \dots, v_k). \quad (7.10)$$

Dabei nimmt  $v'_j$  einen zufälligen Wert aus dem für diese Variable gültigen, aber dynamischen Wertebereich  $[l_j^{i_n^{t+1}}, u_j^{i_n^{t+1}}]$  an. Die untere Grenze  $l_j^{i_n^{t+1}}$  bzw. die obere Grenze  $u_j^{i_n^{t+1}}$  sind veränderliche Werte, die von den jeweiligen Nebenbedingungen für die Variable  $j$  und den Werten der anderen Variablen des Individuums  $i_n^{t+1}$  abhängen. Damit ist garantiert, dass beim Vordringen in neue Suchunterräume, hervorgerufen durch den evolutionären Operator *Uniform Mutation*, keine ungültigen Individuen entstehen können.



**Boundary Mutation** ist ein evolutionärer Operator, der eine Variante der *Uniform Mutation* darstellt. Er ist insbesondere für die Expansion der Suche am Rand des dynamischen Wertebereichs der Variablen eines Individuums verantwortlich. Die Festlegung von  $v'_j$  gehorcht folgender Berechnungsvorschrift:

$$v'_j = \begin{cases} l_j^{i_n^{t+1}} & \text{falls } \text{Zufallszahl} = 0 \\ u_j^{i_n^{t+1}} & \text{falls } \text{Zufallszahl} = 1 \end{cases} \quad (7.11)$$

Auch dieser erweiterte Operator ist durch die Einführung der dynamischen Bereichsgrenzen in der Lage, nur gültige Individuen zu erzeugen.

**Non-Uniform Mutation** Die Erweiterung zur Produktion ausschließlich gültiger Individuen kann an diesem evolutionären Operator folgendermaßen vorgenommen werden: Ist  $i_n^t = (v_1, v_2, \dots, v_k)$  ein Eltern-Individuum und eine zufällig ausgewählte Variable von  $i_n^t$  sei  $v_j$ , dann ergibt sich der neu erzeugte Nachkomme zu

$$i_n^{t+1} = (v_1, v_2, \dots, v'_j, \dots, v_k) \quad (7.12)$$

mit

$$v'_j = \begin{cases} v_j + \Delta \left( t, u_j^{i_n^{t+1}} - v_j \right) & \text{falls } \text{Zufallszahl} = 0 \\ v_j - \Delta \left( t, v_j - l_j^{i_n^{t+1}} \right) & \text{falls } \text{Zufallszahl} = 1 \end{cases} \quad (7.13)$$

wobei  $\Delta(t, y)$  der Funktion aus dem vorangegangenen Abschnitt entspricht und  $l_j^{i_n^{t+1}}$  bzw.  $u_j^{i_n^{t+1}}$  erneut die dynamischen Bereichsgrenzen der Variablen  $v_j$  in Abhängigkeit der Nebenbedingungen und der anderen Variablenwerte darstellen.

**Simple Crossover** Gültige Nachkommen  $i_n^{t+1}$  und  $i_m^{t+1}$  können unter Berücksichtigung der Nebenbedingungen mit dem erweiterten *Simple Crossover* aus den Eltern-Individuen  $i_n^t$  und  $i_m^t$  auf folgende Art und Weise erzeugt werden:

$$i_n^{t+1} = (v_1, v_2, \dots, v_j, w_{j+1} \cdot a + v_{j+1} \cdot (1 - a), \dots, w_k \cdot a + v_k \cdot (1 - a)) \quad (7.14)$$

$$i_m^{t+1} = (w_1, w_2, \dots, w_j, v_{j+1} \cdot a + w_{j+1} \cdot (1 - a), \dots, v_k \cdot a + w_k \cdot (1 - a)) \quad (7.15)$$

dabei ist  $a$  ein Parameter im Intervall  $[0..1]$ . Er ist mit Hilfe einer lokalen Optimierung im Sinne eines größtmöglichen Informationsaustausches so groß zu wählen, wie unter den Nebenbedingungen nur möglich. Leider muss bei diesem evolutionären Operator zunächst die Erzeugung des Nachkommen erfolgen, bevor über die Korrektheit und die Wahl des Parameters  $a$  entschieden werden kann.

**Arithmetical Crossover** Erfüllen die Eltern  $i_n^t$  und  $i_m^t$  die Nebenbedingungen so erfüllen auch die durch Linearkombinationen

$$i_n^{t+1} = b \cdot i_n^t + (1 - b) \cdot i_m^t \quad (7.16)$$

$$i_m^{t+1} = b \cdot i_m^t + (1 - b) \cdot i_n^t \quad (7.17)$$

produzierten Nachkommen ohne Erweiterung des Operators die Nebenbedingungen. *Arithmetical Crossover* produziert damit ausschließlich gültige Nachkommen.

**Extended Arithmetical Crossover** Der evolutionäre Operator *Extended Arithmetical Crossover* ist eine Abwandlung des normalen *Arithmetical Crossover*, wobei der Parameter  $b$  einen Vektor der Größe korrespondierend zur Anzahl an Variablen pro Individuum aus Zufallszahlen zwischen 0 und 1 darstellt. Da nun die Variablen-Nebenbedingungen nicht mehr automatisch erfüllt sind, ist es notwendig jeden produzierten Nachkommen auf dessen Einhaltung der Nebenbedingungen zu kontrollieren. Gegebenenfalls muss das erste Ergebnis verworfen und der Operator mit einem neuen Parameter  $b$  nochmals angewendet werden.

## 7.3 Identifikation von Einzelfehlern

Der erste Schritt hin zu einer modellbasierten und automatisierten Analyse von Fehlermöglichkeiten in Bezug auf deren Auswirkungen ist mit der Entwicklung des angepassten, evolutionären Programms erfolgt. Es ist in der Lage auf Basis eines Systemmodells selbständig signifikante Fehlerszenarien zu identifizieren. Der zweite Schritt besteht in der Definition der Fehlerszenarien. Dabei werden zunächst die notwendigen Festlegungen für eine Identifikation von signifikanten Einzelfehlern diskutiert.

Bei der Identifikation signifikanter Einzelfehler kommen Fehlerszenarien in Betracht, die nur ein singuläres Fehlerereignis aufweisen. Dieses Fehlerereignis kann permanenter oder transienter Natur sein und enthält zu seiner detaillierteren Beschreibung Angaben zum Fehlerort, der Fehlerart und der entsprechenden Ausprägung. Um die Beschreibung zu komplettieren, umfassen die Festlegungen der Fehlerszenarien von Einzelfehlern auch den aktuellen Systemzustand zum Zeitpunkt des Fehlereintritts. Dazu zählen z.B. der aktuelle Betriebsmodus des Fahrzeugsystems, die im Moment vorherrschenden Umgebungsbedingungen oder das gerade durchzuführende Fahrmanöver.

Aus dieser Definition eines Fehlerszenarios für Einzelfehler lässt sich eine mathematische Beschreibung für die Individuen des evolutionären Programms zur Identifikation von Einzelfehlern

ableiten. Jedes Individuum  $i_n^t$  beschreibt genau ein Fehlerszenario und beinhaltet dafür folgende Variablen:

$$i_n^t = \begin{pmatrix} \textit{Systemzustand} \\ \textit{Fehlernummer} \\ \textit{Fehlerschwere} \\ \textit{Fehleraktivierungszeitpunkt} \\ \textit{Fehlerdeaktivierungszeitpunkt} \end{pmatrix}' \quad (7.18)$$

Der Inhalt der Variable Systemzustand beschreibt in kodierter Form das eingeleitete Fahrmanöver, den internen Systemzustand und die vorherrschenden Umgebungsbedingungen. Mit Hilfe der Variablen Fehlernummer werden sowohl der Fehlerort als auch die Fehlerart kodiert spezifiziert. Der Inhalt aller anderen Variablen korrespondiert direkt mit deren Variablennamen.

Mit dieser Definition eines Individuums kann ein Einzelfehlerszenario nach den obig erwähnten Anforderungen eindeutig spezifiziert werden. Die Notwendigkeit von Nebenbedingungen zur Definition von gültigen Individuen beschränkt sich auf den zeitlichen Aspekt der Fehlerereignisse. Es muss für alle gültigen Individuen gelten:

$$\textit{Fehleraktivierungszeitpunkt} < \textit{Fehlerdeaktivierungszeitpunkt} \quad (7.19)$$

## 7.4 Erweiterungen

### 7.4.1 Identifikation von Fehlerkombinationen

Wie sich in zahlreichen Untersuchungen insbesondere im Bereich der Luftfahrt gezeigt hat, sind es jedoch meist nicht die Einzelfehler, die zu katastrophalen Auswirkungen während des Betriebs eines sicherheitsrelevanten Systems führen. Vielmehr zeigt die Vergangenheit, dass die Ursache für das Versagen sicherheitsrelevanter Systeme in der Regel in einer Verkettung von mindestens zwei Fehlerereignissen begründet ist.

Die bereits vorgestellte Methodik zur Identifikation signifikanter Einzelfehler kann mit Hilfe einer Erweiterung dieser Tatsache Rechnung tragen. Es ist möglich, durch eine veränderte Definition der Individuen, ebenfalls Fehlerkombinationen mit dieser Vorgehensweise zu analysieren. Unter Fehlerkombinationen wird im Folgenden das gleichzeitige Auftreten mindestens

zweier permanenter oder transienter Fehlerereignisse verstanden. Die erweiterte Individuen-Definition  $i_n^t$  zur Identifikation von Fehlerkombinationen lautet:

$$i_n^t = \left( \begin{array}{c} \textit{Systemzustand} \\ \textit{Fehlernummer 1} \\ \textit{Fehlerschwere 1} \\ \vdots \\ \textit{Fehlernummer n} \\ \textit{Fehlerschwere n} \\ \textit{Fehleraktivierungszeitpunkt} \\ \textit{Fehlerdeaktivierungszeitpunkt} \end{array} \right)' \quad (7.20)$$

Der eigentliche Identifikationsalgorithmus ist von dieser Erweiterung nicht betroffen und kann unverändert angewendet werden. Sinnvoll ist jedoch eine Einteilung der Fehlerereignisse in Fehlerklassen. Hiermit kann unter Berücksichtigung der eingeführten Nebenbedingungen gewährleistet werden, dass Fehlerkombinationen, die ohnehin als nicht möglich oder unzulässig bekannt sind, im Laufe des automatisierten Injektionsvorgangs vermieden werden. Dies führt während des Identifikationsverfahrens zu einer deutlichen Aufwandsreduktion.

### 7.4.2 Identifikation von Fehlersequenzen

Eine noch allgemein gültigere Analyse kann mit der Identifikation von Fehlersequenzen durchgeführt werden. Der Begriff Fehlersequenzen umschreibt das Auftreten mindestens zweier Fehlerereignisse, die sowohl kausal als auch zeitlich in Bezug auf ihren Fehlereintritt und ihre Anhaltedauer völlig unabhängig voneinander sind. Dazu lautet die erweiterte Individuen-Definition  $i_n^t$  folgendermaßen:

$$i_n^t = \left( \begin{array}{c} \textit{Systemzustand} \\ \textit{Fehlernummer 1} \\ \textit{Fehlerschwere 1} \\ \textit{Fehleraktivierungszeitpunkt 1} \\ \textit{Fehlerdeaktivierungszeitpunkt 1} \\ \vdots \\ \textit{Fehlernummer n} \\ \textit{Fehlerschwere n} \\ \textit{Fehleraktivierungszeitpunkt n} \\ \textit{Fehlerdeaktivierungszeitpunkt n} \end{array} \right)' \quad (7.21)$$

Der Identifikationsalgorithmus ist auch in diesem Fall nicht von der Erweiterung betroffen und kann unverändert eingesetzt werden. Jedoch bietet sich auch in diesem Fall die erwähnte Fehlerklassenbildung an, um den benötigten Identifikationsaufwand durch Vermeidung sinnloser Fehlersequenzen zu reduzieren.

### 7.4.3 Verknüpfung mit Fehlerauftrittswahrscheinlichkeiten

Dem bisherigen Identifikationsablauf liegt die Annahme zu Grunde, dass alle Fehlerereignisse mit der gleichen Wahrscheinlichkeit eintreten. Der Fokus der Untersuchung liegt dabei ausschließlich auf der Identifikation der Fehler mit signifikanten Auswirkungen. Es ist jedoch durchaus möglich, dass diese identifizierten Fehler extrem selten auftreten. Deshalb besteht über die Verkettung mehrerer Fehlerereignisse hinaus die Möglichkeit, die Auftretenswahrscheinlichkeit der Fehlerereignisse im Identifikationsprozess zu berücksichtigen. Hierdurch kann der Fokus der Untersuchung auf die Fehler gerichtet werden, die sowohl mit einer höheren Wahrscheinlichkeit auftreten als auch signifikantere Auswirkungen zeigen als andere Fehlerszenarien.

Zur Anwendung dieser Erweiterung sind zwei Dinge erforderlich. Zum einen muss die Auftretenswahrscheinlichkeit in der Bewertungsfunktion berücksichtigt werden. Dies kann z.B. durch Multiplikation der Auftretenswahrscheinlichkeit mit dem bisherigen Fitnesswert erfolgen. Die so erzeugte neue Fitnessfunktion bewertet das Individuum sowohl nach deren Auswirkung auf das Systemverhalten als auch nach dessen Auftretenshäufigkeit. Zum anderen ist eine Quantifizierung der Auftretenswahrscheinlichkeiten der Fehler vorzunehmen. Die dazu erforderliche Kenntnis ist jedoch, wie auch schon bei den einführenden Untersuchungen zu den bekannten Sicherheitsanalysen erwähnt, äußerst schwierig zu erlangen. Diese Problematik wird hier noch verschärft, da für jeden speziellen und modellierten Fehler mit seiner spezifischen Fehlerschwere eine Auftretenswahrscheinlichkeit angegeben werden muss.

Es hat sich während der praktischen Anwendung dieser Erweiterung gezeigt, dass es zwar prinzipiell möglich ist, den Identifikationsalgorithmus mit probabilistischen Methoden zu verknüpfen, die Auftretenswahrscheinlichkeiten in dieser frühen Entwicklungsphase jedoch nur in seltenen Fällen in der notwendigen Genauigkeit ermittelbar sind.

Um im Folgenden die Übersichtlichkeit und Verständlichkeit zu wahren, wird nicht mehr auf die Erweiterungen eingegangen, sondern die Identifikation von Einzelfehlern als Basis zu Grunde gelegt. Die Umsetzung und die Anwendung aller im Folgenden beschriebenen Methoden sowie die Erzeugung von Ergebnissen am Beispiel einer steer-by-wire Anwendung können für die erweiterten Identifikationsverfahren analog erfolgen.

## 7.5 Visualisierung

Evolutionäre Programme erreichen ihr komplexes Verhalten und ihre Eigenschaft zur globalen Suche durch die Verknüpfung mehrerer, aber in ihren algorithmischen Grundlagen einfach strukturierten Verfahren. Dabei werden in den einzelnen Iterationen beim Ablauf eines evolutionären Programms eine große Menge an Daten produziert. Dem Anwender eröffnen diese Daten in den meisten Fällen nur einen geringen Einblick in das Verhalten des evolutionären Programms. Ein Verständnis für die Funktion, die Möglichkeit zur Überwachung der Abläufe und die Fähigkeit zur Identifikation von Optimierungspotentialen sind auf dieser Datenbasis nur sehr schwer bzw. nicht zu erlangen.

Erst durch eine entsprechende Datenaufbereitung und deren grafische Darstellung ist es möglich, dem Anwender ein Hilfsmittel an die Hand zu geben, das eine leichte Interpretation der Abläufe und belastbare Aussagen zur Funktion des evolutionären Programms zulässt. Aus einer Vielzahl möglicher Visualisierungsverfahren [Poh00] konnten diejenigen ausgewählt und weiterentwickelt werden, die bei der Verwendung im vorliegenden Einsatzgebiet als besonders geeignet eingestuft wurden. Im Folgenden werden diese Datenaufbereitungsverfahren und die dazugehörigen Darstellungsvarianten vorgestellt.

Die wichtigste Visualisierung ist die Darstellung der Konvergenz der Population. Dazu wird der Fitnessfunktionswert des besten Individuums über den vergangenen Generationen aufgetragen. Bild 7.3 zeigt ein Beispiel eines so genannten Konvergenz-Diagramms, das in dieser Darstellung um den Mittelwert aller Fitnessfunktionswerte ergänzt ist. Das Diagramm gibt einen guten Überblick, wie schnell bessere Lösungen durch den verwendeten Algorithmus gefunden werden und wie groß der Fortschritt zwischen den einzelnen Generationen ist. Es lässt sich deutlich erkennen, dass zu Beginn einer Suche durch die zufällige Initialisierung der Anfangspopulation sehr schlechte Fitnessfunktionswerte erzielt werden. Innerhalb weniger Generationen werden diese jedoch deutlich besser und ändern sich im folgenden Verlauf der Suche prozentual gesehen nur noch geringfügig. Ist der Fitnessfunktionswert des besten Individuums über eine große Anzahl von Generationen konstant, kann dies als ein Indiz für eine geeignete Terminierung des Suchalgorithmus gewertet werden.

Da beim Konvergenz-Diagramm pro Generation immer nur der Fitnessfunktionswert abgebildet wird, ist das Aussehen der einzelnen Individuen aus dieser Darstellungsform nicht ersichtlich. Mittels einer weiteren Visualisierungsvariante eröffnet sich jedoch die Möglichkeit eines tieferen Einblicks in die Abläufe der globalen Suche. Dazu werden die Variablenwerte des besten Individuums normiert, über den Generationen aufgetragen und in einer dreidimensionalen Liniengrafik gemäß Bild 7.4 dargestellt.

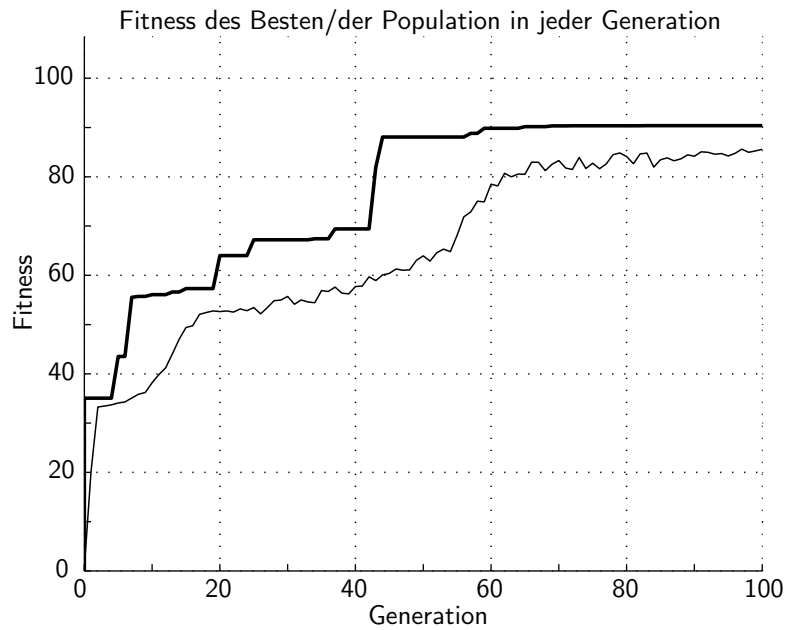


Bild 7.3: Fitnessfunktionswert des besten Individuums aufgetragen über den Generation. Zusätzlich mit durchschnittlichem Fitnessfunktionswert aller Individuen

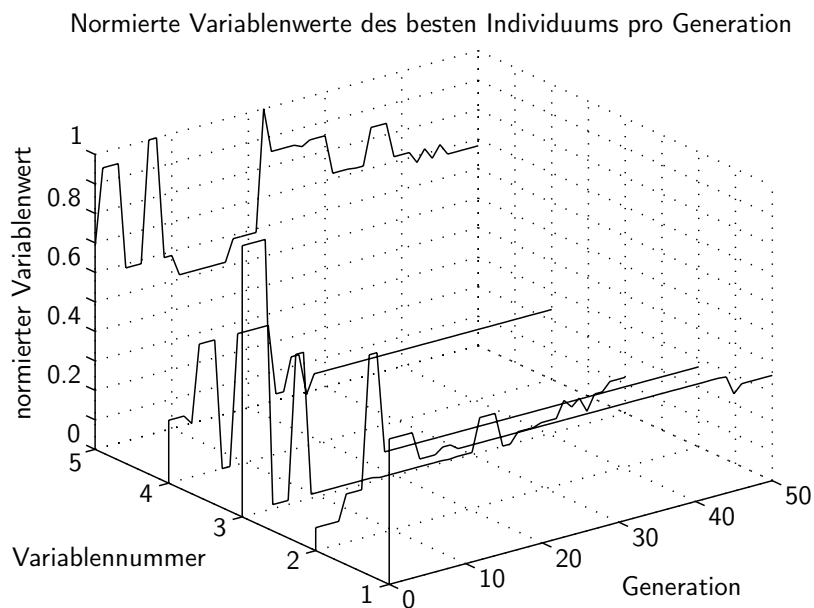


Bild 7.4: Werte der Variablen des besten Individuums aufgetragen über mehreren Generationen

Mit dieser Visualisierung lässt sich auf einen Blick erkennen, ob im Verlauf der Suche große Sprünge in den Variablenwerten auftreten. Insbesondere gegen Ende der Suche, wenn das Suchergebnis bereits konvergiert, kann diese Darstellungsform zusammen mit dem Konvergenzdiagramm einen guten Einblick in den Zusammenhang zwischen der Veränderung der Fitnessfunktions-

tionswerte und der korrespondierenden Veränderung der Variablenwerte geben. Damit ist in der Regel eine schnelle Einschätzung möglich, ob die Suche frühzeitig als fehlgeschlagen, noch nicht beendet oder bereits als erfolgreich abgeschlossen anzusehen ist.

Waren bisher immer nur einzelne Individuen Gegenstand der Betrachtung, so soll im Folgenden das Verhalten der gesamten Population mit Hilfe eines Diagramms verdeutlicht werden. Ein zweidimensionaler Farbteppich, in dem die Individuen über den Variablen aufgetragen werden und in dem die jedem Individuum pro Variable zugeordnete Fläche die Farbe enthält, die mit dem entsprechenden Variablenwert auf einer Farbskala korrespondiert, eignet sich für einen Einblick in die Vorgänge innerhalb der gesamten Population. In der linken Hälfte von Bild 7.5 sieht man deutlich die aus der zufälligen Initialisierung herrührende, unstrukturierte Verteilung der Variablenwerte innerhalb der Anfangspopulation. Zum Ende des Suchprozesses, dargestellt in der rechten Hälfte von Bild 7.5, hat sich innerhalb der Population eine erkennbare Struktur herausgebildet. Aus ihr ist abzulesen, dass die gesamte Population in Richtung einer Lösung konvergiert und trotzdem nicht aus einer einzigen, alles dominierenden Lösung besteht. Sollten im Laufe der Suche mehrere, nahezu gleichwertige, aber diversitäre Lösungen gefunden werden, so wird dieser Sachverhalt erst durch diese Darstellungsform ersichtlich.

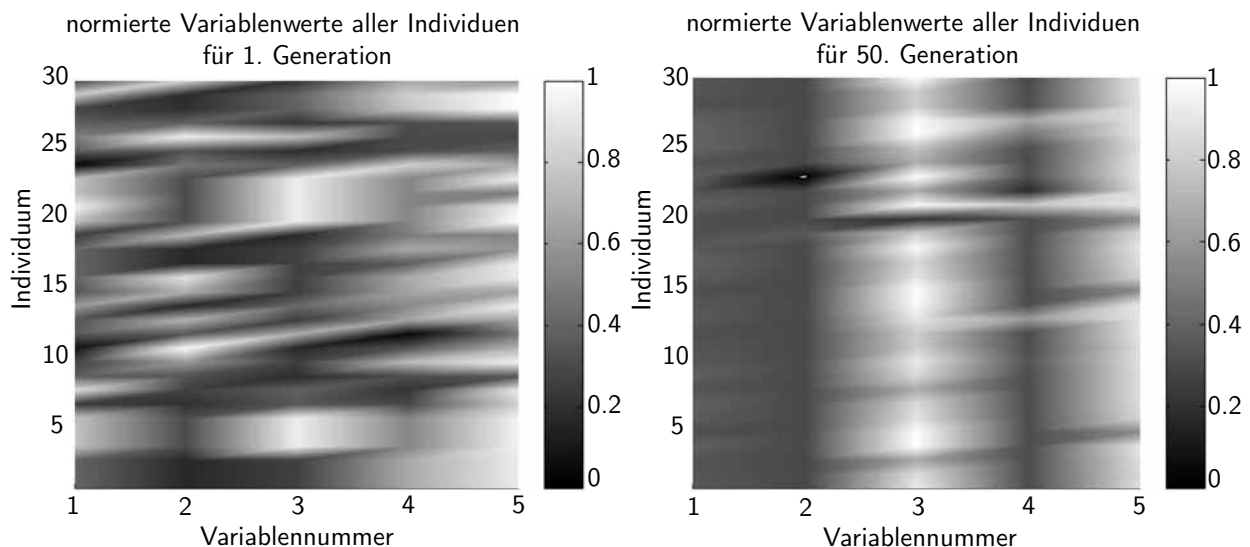


Bild 7.5: Variablenwerte aller Individuen einer Population in zweidimensionaler Farbteppich-Darstellung; links: zu Beginn eines Suchlaufs; rechts: am Ende der Suche

Oftmals sind einzelne Variablen und deren Wertigkeit für die Gesamtlösung von signifikanter Bedeutung. Zur Identifizierung jener Variablen werden so genannte Spinnennetz-Diagramme verwendet. Dazu wird jeder Variablen eine Zahlenachse zugeteilt und die Variablenwerte eines Individuums an den entsprechenden Achsen abgetragen. Verbindet man jeweils benachbarte



Achsenabschnitte miteinander, so entsteht eine geschlossene geometrische Form für jedes Individuum der Population. Wie in der linken Hälfte von Bild 7.6 zu erkennen ist, sind die möglichen Lösungen in der ersten Generation noch weit gestreut. Nach 100 Generationen ha-

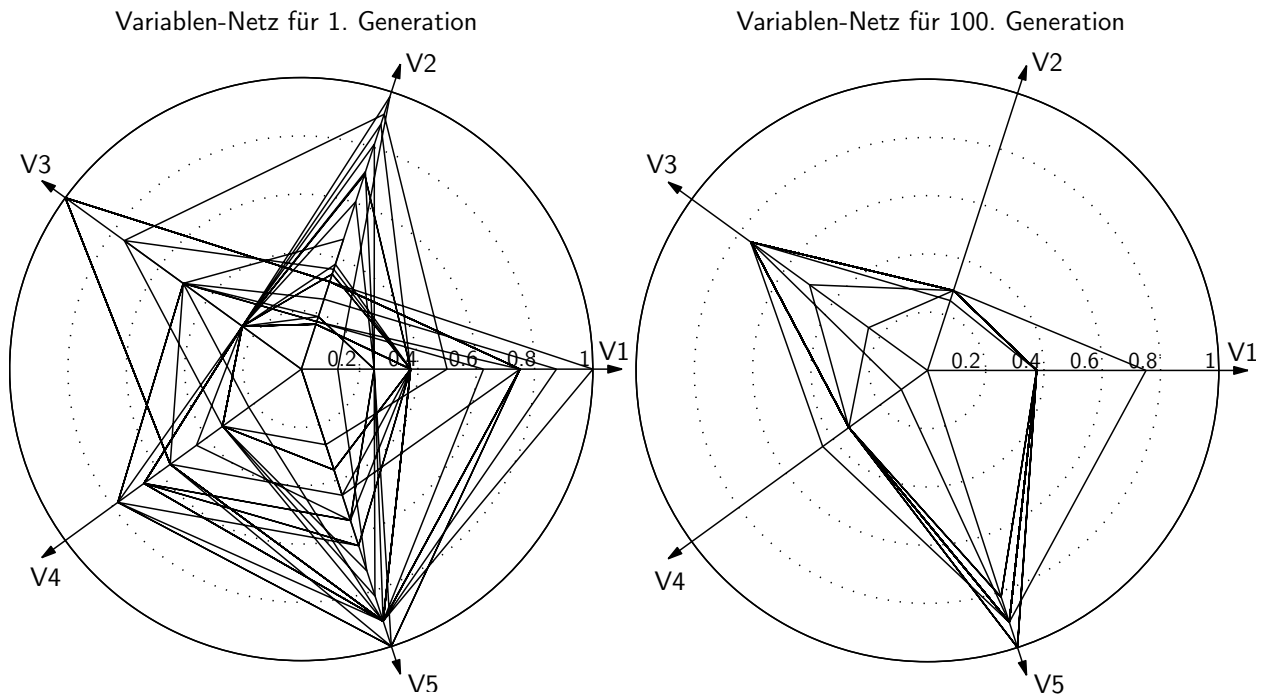


Bild 7.6: Darstellung aller Individuen einer Population und deren normierte Variablenwerte in Spinnennetz-Form; links: zu Beginn eines Suchlaufs; rechts: am Ende der Suche

ben sich jedoch schon deutliche Häufungen bei einzelnen Variablenwerten herausgebildet. In der rechten Hälfte von Bild 7.6 wird dieses Verhalten besonders bei den Variablennummern 2 und 5 deutlich. Diese Darstellungsform ermöglicht darüber hinaus auch die Überwachung der Abläufe innerhalb der Population. Zeigt die Population keine Konvergenz im Laufe der Suche oder beschränkt sie sich auf ein dominierendes Individuum, so ist dieses Fehlverhalten mit Hilfe des Spinnennetz-Diagramms schnell zu erkennen.

## 7.6 Parallele und skalierbare Anwendung

Der größte Nachteil der evolutionären Programme und deren Verknüpfung mit modellbasierten Analysemethoden ist die zur Identifikation signifikanter Systemfehler notwendige Rechenzeit. Zur Analyse eines Fehlerszenarios muss das Gesamtsystem-Modell ausgeführt und die zu Tage tretenden Reaktionen entsprechend bewertet werden. Je nach Detaillierungsgrad der Systemmodellierung nehmen diese Berechnungen mehr oder weniger Zeit in Anspruch. Die Identifikation

der signifikanten Fehler erfordert jedoch die Untersuchung einer Vielzahl unterschiedlichster Fehlerannahmen, so dass sich die notwendige Rechenzeit durchaus zu einigen Stunden aufaddieren kann.

Aus diesem Grund liegt der Wunsch nahe, den gesamten Identifikationsprozess zu parallelisieren und so deutlich zu beschleunigen. Unter der Parallelisierung einer Aufgabe versteht man die geeignete Aufspaltung des Gesamtproblems in kleinere Subberechnungen. Dabei kommt es vor allem auf die Unabhängigkeit der einzelnen Teilprobleme hinsichtlich der Einhaltung einer zeitlichen Reihenfolge oder eines notwendigen Informationsaustausches an. Denn nur unter Berücksichtigung dieser Randbedingung können sie auch tatsächlich auf unterschiedlichen Rechenknoten zeitgleich bearbeitet werden.

Gerade für die Parallelisierung des Identifikationsprozesses für signifikante Fehler ist die Verknüpfung von evolutionärem Programm mit modellbasierter Analysemethode jedoch hervorragend geeignet. Da in den meisten Anwendungsfällen des Identifikationsprozesses davon auszugehen ist, dass die Ausführung des Gesamtsystemmodells zur Analyse des Systemverhaltens den weitaus größten Anteil zur benötigten Gesamtrechenleistung beiträgt, muss eine Parallelisierung genau an dieser Stelle erfolgen. Bild 7.7 zeigt eine Struktur eines verteilten Identifikationsprozesses, die die gewünschte Parallelisierung bietet.

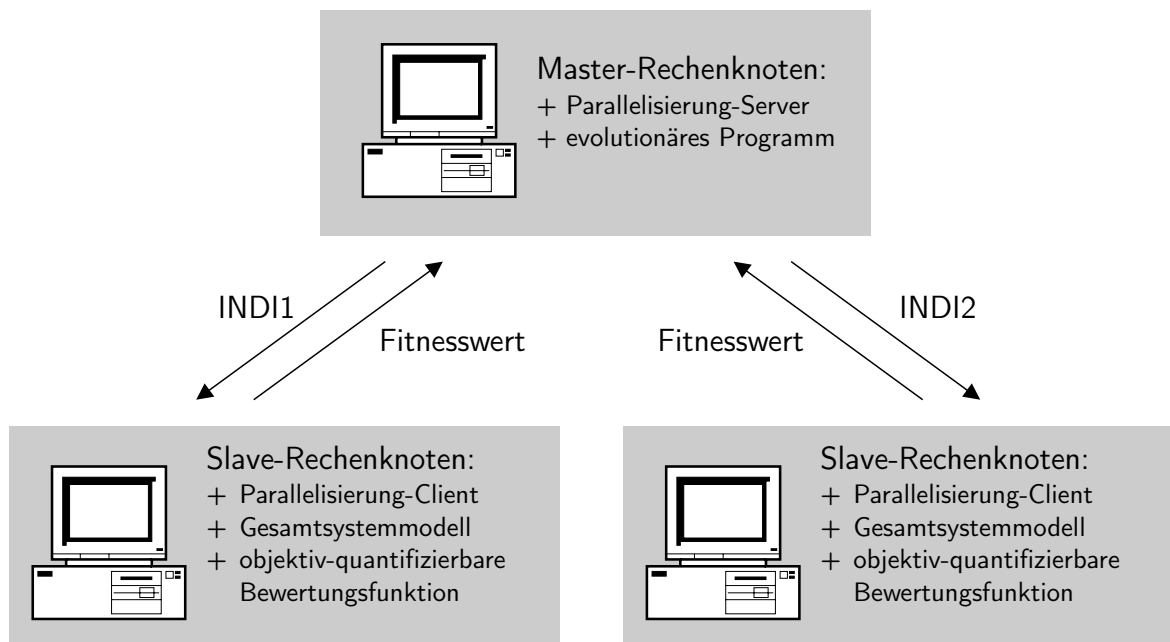


Bild 7.7: Prinzip des verteilten und skalierbaren Identifikationsprozesses für signifikante Fehler

Ein zentraler Berechnungsknoten, der so genannte Master, ist mit der Durchführung des evolutionären Programms und der damit verbundenen Produktion geeigneter Fehlerszenarien be-

auftragt. Die modellbasierte Analyse der Reaktion des Systems während diesen Fehlerszenarien übernehmen eine variable Anzahl an verteilten, parallel vorhandenen Slave-Berechnungsknoten. Ihnen wird lediglich über ein Kommunikationsnetz der Auftrag zur Analyse jeweils eines Fehlerszenarios übermittelt. Deren Ausführung und Bewertung erfolgt selbständig und unabhängig durch die einzelnen Slave-Berechnungsknoten. Sie stellen die so erlangten Fitnessfunktionswerte dem zentralen Master zur Verfügung, der diese Ergebnisse in die folgende Suche nach noch signifikanteren Fehlerszenarien einfließen lässt.

Die Realisierung dieser Struktur erfolgt ebenfalls in Matlab/Simulink mit Hilfe einer entwickelten Parallelisierung-Toolbox, so dass eine enge Verknüpfung und eine einfache Schnittstelle zur Entwicklungsumgebung des Kraftfahrzeugsystems und dem Gesamtsystemmodells gewährleistet ist. Die Komplexität und die zur Verfügung stehende Entwicklungs- und Analysezeit bestimmen die notwendige Anzahl der Slave-Berechnungsknoten. Experimente, dargestellt in Bild 7.8, haben gezeigt, dass die Steigerung der Leistung durch den Einsatz eines verteilten Identifikationsprozesses nahezu proportional zur Anzahl der verwendeten Slave-Rechenknoten ist. Der notwendige Aufwand für die Verteilung und Parallelisierung kann somit sehr gering gehalten werden.

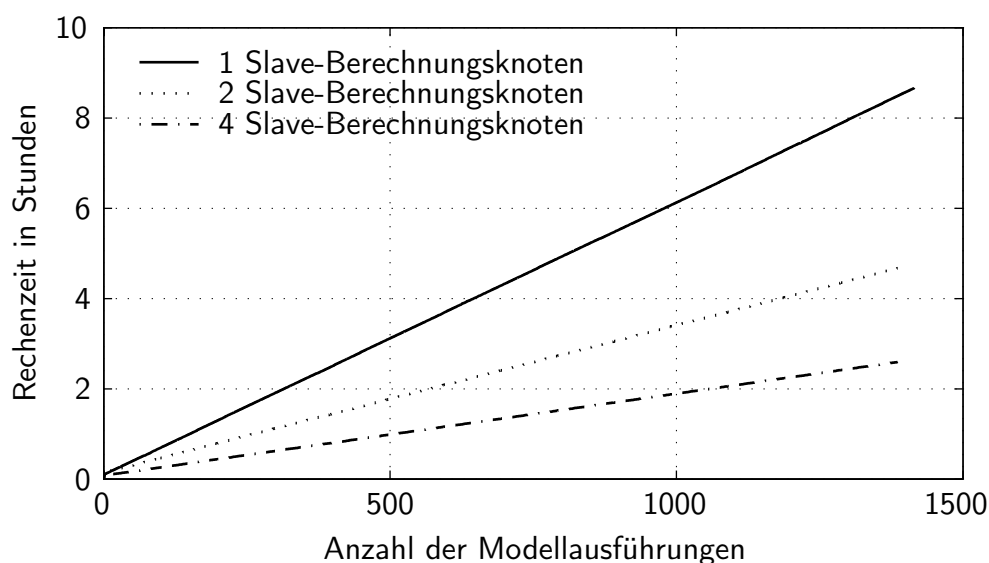


Bild 7.8: Vergleich der Rechenzeiten in Abhängigkeit der Anzahl an Berechnungsknoten

## 7.7 „What happened“-Analyse

Standen bisher vor allem die Fragen: „*Welche Fehler führen zu welchen Auswirkungen?*“ und „*Wie sind diese Auswirkungen in ihrer Schwere zu bewerten?*“ im Mittelpunkt der Untersuchung, so ändert sich mit Abschluss der Identifizierung von Fehlern mit signifikanten Auswirkungen der Fokus der weiteren Betrachtung. Denn mit der Kenntnis von Fehlerszenarien mit besonders hervorstechenden Konsequenzen für das Systemverhalten stellt sich unweigerlich die Frage nach dem Grund für diese Systemreaktion und nach möglichen Abhilfemaßnahmen. Diese sich ergebende Verlagerung des Untersuchungsschwerpunkts stellt eine Umkehr der bisher vorherrschenden Analyse-Richtung dar. Wurde die Wirkkette bislang in vorwärtsgerichteter Orientierung vom Fehler zur Auswirkung betrachtet, so impliziert die jetzige Fragestellung eine rückwärtsgerichtete Analyse von der Auswirkung zu zwingend notwendigen Ereignissen um das gezeigte Systemverhalten zu erreichen. Dieser Sachverhalt ist nochmals in Abbildung 7.9 verdeutlicht.

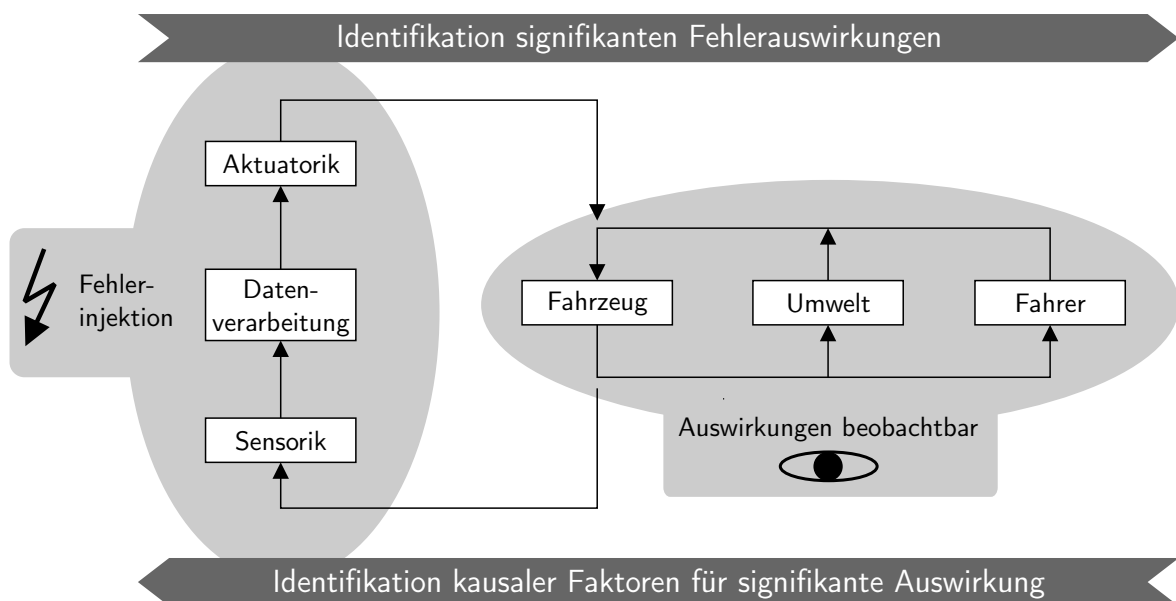


Bild 7.9: Inversion der Untersuchungsrichtung mit der „What happened“-Analyse

In der Unfallforschung der Luft- und Raumfahrttechnik werden bereits ähnliche Fragestellungen insbesondere nach schweren Luftfahrtunfällen behandelt. Eine der dabei zum Einsatz kommenden Methoden ist die Why-Because-Analyse [Lad01, Pau05]. Sie geht auf Peter Ladkin zurück, der, angeregt durch Unzulänglichkeiten in den Unfallberichten vergangener Luftfahrtunfälle, in den neunziger Jahren ein erstes Konzept der Why-Because-Analyse entwickelte. Ausgehend von einem Unglück werden Informationen zu allen involvierten Systemen, Umgebungseinflüssen und Bedieneingaben gesammelt und rückwärts Schritt für Schritt miteinander

verknüpft, um so Ursachen des Unfalls identifizieren zu können. Bei jeder potentiellen Ursache muss sich die Frage gestellt werden, ob der Unfall auch eingetreten wäre, wenn diese Ursache nicht existiert hätte. Im Falle einer positiven Beantwortung dieser Frage wird diese Ursache als kausaler Faktor des Unfalls bezeichnet.

Im Gegensatz zur Why-Because-Analyse, deren Ablauf meist durch das Sammeln unterschiedlichster Informationen und deren schrittweiser, theoretischer Verknüpfung und Auswertung geprägt ist, kann auf Basis des bereits erstellten Systemmodells ebenfalls eine rückwärtsgerichtete Identifikation der kausalen Faktoren durchgeführt werden. Im Gesamtsystemmodell sind bereits implizit alle Informationen und deren Verknüpfungen enthalten und in einer quantitativen und ausführbaren Art und Weise beschrieben. Die relevanten Stimuli, die potentielle, äußere Ursachen für unerwünschtes Systemverhalten darstellen, sind aus der Identifikation von Fehlern mit signifikanter Auswirkung bekannt. Die Analyse der internen Systemabläufe zur Identifikation von kausalen Faktoren, die in der Systemimplementierung begründet sind, erfolgt entlang der internen Signal- und Energieflüsse des Modells. Schritt für Schritt wird das Verhalten von den Systemausgaben in Richtung der Eingaben nachvollzogen und eventuelle Abweichungen und deren Ursachen aufgedeckt. Dabei kommt der Analyse vor allem die Eigenschaft der Reproduzierbarkeit der Modellausführung zu Gute, so dass jeder rückwärtsgerichtete Iterationsschritt unter den gleichen, definierten Bedingungen durchgeführt werden kann. Damit bietet die Verknüpfung des Systemmodells, der Identifikation signifikanter Fehlerauswirkungen und der rückwärtsgerichteten Analyse einige Vorteile bei der Bestimmung der kausalen Faktoren eines gezeigten Systemverhaltens im Vergleich zur Why-Because-Analyse.

Die Inversion der bisherigen Analyse-Richtung auf Basis des bereits erstellten Systemmodells unter Einbeziehung der identifizierten, signifikanten Fehlerauswirkungen wird im Folgenden „What happened“-Analyse genannt.

## 7.8 Iterative Durchführung

Nach der Identifikation der kausalen Faktoren für signifikante Fehlerauswirkungen kann der kreative Prozess der Ideenfindung zur Behebung der Ursachen eingeleitet werden. Ob bei der Beseitigung der Schwachstellen tatsächlich geeignete Maßnahmen ergriffen wurden, kann eine wiederholte Ausführung des Gesamtsystemmodells unter den Randbedingungen des Fehlerszenarios aufzeigen. Hat sich die Schwere der beobachtbaren Auswirkungen des ermittelten Fehlerszenarios reduziert, kann zunächst von einer erfolgreichen Verbesserung des Systemverhaltens in dieser speziellen Situation ausgegangen werden.

Es ist jedoch nicht selten der Fall, dass im Zuge solcher Verbesserungsmaßnahmen andere, neue Fehlermöglichkeiten ins System eingebaut werden, die eventuell ganz andere Systembereiche betreffen und zudem ganz andere Auswirkungen mit sich bringen. Um derartige vermeintliche Verbesserungen zu entdecken und zu vermeiden, sollte der Identifikationsprozess im Anschluss an die vorgenommenen Änderungen wiederholt werden. Eine iterative Anwendung des Identifikationsprozesses und die wiederholte Beseitigung der Ursachen führen somit zu einer kontinuierlichen Absenkung des Gefährdungspotentials des Kraftfahrzeugsystems.

## 7.9 Anmerkungen

Das vorgestellte Verfahren darf nicht mit dem Führen eines Sicherheitsnachweises missverstanden werden. Es bietet zwar die Möglichkeit nach sicherheitsrelevanten Fehlerszenarien zu suchen, stellt dabei jedoch keinen Beweis für die Fehlerfreiheit oder Fehlertoleranzeigenschaft eines entwickelten Systems dar. Ein solches Beweisverfahren ist aufgrund der enormen Menge an unterschiedlichen Kombinationen aus Fehlermöglichkeiten und Systemzuständen für solche umfangreiche und komplexe Systeme heute nicht bekannt und auch in näherer Zukunft nicht zu erwarten.

Ferner ist es aus eben diesen Gründen für die Entwicklung sicherheitsrelevanter Kraftfahrzeugsysteme von grundlegender Bedeutung, dass zum einen die Einflüsse der Teilsysteme eines mechatronischen Systems für das Gesamtsystemverhalten transparent gemacht werden. Der modellbasierte Ansatz der vorgestellten Methodik trägt entscheidend dazu bei, dass ein interdisziplinäres Gesamtverständnis für das Systemverhalten im nominal wie auch im fehlerbehafteten Betriebsfall entstehen kann. Zum anderen wird jedoch auch eine Konzentration auf sinnvolle und zielgerichtete Testszenarien immer wichtiger. Aus der enormen Menge an Kombinationen aus Fehlermöglichkeiten und Systemzuständen müssen Testfälle ausgewählt werden, die zur Überprüfung der Sicherheitseigenschaften eines Systems genutzt werden können. Für eben diese schwierige Aufgabe der Auswahl geeigneter Testszenarien bietet der Identifikationsprozess signifikanter Fehlerauswirkungen eine Strategie.

Der Nachweis (*und nicht der Beweis*) der Sicherheit ist auch weiterhin nur durch eine Kombination unterschiedlichster Verfahren zu erbringen. Die modellbasierte, ganzheitliche Analyse und Identifikation kann als eines dieser Verfahren zur Verbesserung des Sicherheitsniveaus von neu entwickelten Kraftfahrzeugsystemen beitragen. Ihr Nutzen wird im folgenden Kapitel anhand eines Beispiels aus der Praxis verdeutlicht.

## 8 Anwendung am Beispiel eines steer-by-wire Systems

Der Nachweis der Anwendbarkeit der entwickelten Methodik wird am Beispiel eines verteilten fehlertoleranten Kraftfahrzeugsystems erbracht und in diesem Kapitel beschrieben. Als exemplarisches Kraftfahrzeugsystem dient ein im Rahmen eines Forschungsprojekts entwickeltes steer-by-wire System. Um auch tatsächlich die Anwendbarkeit der modellbasierten Systemanalyse und deren Vor- und Nachteile bewerten zu können, wurde der komplette Entwicklungsstand zu einem festgelegten Zeitpunkt während der laufenden Entwicklung eingefroren. Dieser Entwicklungsstand dient als Basissystem für alle folgenden Untersuchungen, wobei damit gewährleistet ist, dass nicht alle Systembestandteile vollständig entwickelt sind, bisher nur partielle Optimierungen durchgeführt wurden, vielfältige Verbesserungen am vorliegenden System möglich sind und eine Vielzahl an Fehlern noch im System verborgen sind.

Im folgenden Kapitel wird zunächst die zugrunde gelegte Systemarchitektur des steer-by-wire Systems erläutert, ehe auf die Erstellung des korrespondierenden Systemmodells eingegangen wird. Anschließend erfolgt eine Beschreibung des Ausbaus zum Gesamtsystemmodell und der Verknüpfung mit den modellbasierten Analysemethoden. Abschließend wird die konkrete Vorgehensweise erläutert und die erlangten Ergebnisse der modellbasierten Systemanalyse bei der Identifikation von Fehlern vorgestellt.

### 8.1 Systemarchitektur der Beispielanwendung

Das Grundkonzept eines steer-by-wire Systems basiert auf dem Gedanken, die mechanische Verbindung zwischen Lenkrad und gelenkten Vorderrädern vollständig durch elektromechanische Komponenten zu ersetzen. Sowohl die Energie- als auch die Informationsübertragung vom Fahrer zum Fahrzeug erfolgt damit rein elektrisch bzw. elektronisch.

Die mechanische Entkopplung von Lenkrad und gelenkten Rädern ermöglicht die Implementierung von Funktionen, die mit konventionellen Lenksystemen nicht oder nur sehr schwer und mit großem Aufwand zu realisieren sind. So sind z.B. Lenkeigenschaften individuell und rein über Softwareparameter zu beeinflussen, Lenkübersetzungen in Abhängigkeit der Geschwindigkeit





TTCAN, kommunizieren die beiden Lenkrad-Steuergeräte mit den entsprechenden Steuergeräten des Radaktuators. Der Aufbau des Systembereichs Radaktuator entspricht im Wesentlichen dem des Lenkradaktuators. Auf Grund der komplett redundanten Auslegung der Systemstruktur spricht man auch von einem zwei-kanaligen Systemaufbau, der in der Lage ist wenigstens einen Fehler im System zu tolerieren.

Zum Zeitpunkt des Einfrierens des Entwicklungsstandes für die spätere Anwendung der modellbasierten Systemanalyse war das steer-by-wire System bereits so weit entwickelt, dass in einem Prototypen-Fahrzeug mit Hilfe einer Rapid Prototyping Umgebung Funktionsentwicklung und Fahrversuche durchgeführt werden konnten. Dies bedeutet damit auch das Vorhandensein der mechanischen Komponenten für Lenkrad- und Radaktuator. Die Systemstruktur, bestehend aus den vier Steuergeräten, war zwar im Software-Modell des Rapid Prototyping Systems verwirklicht, jedoch in der Realität nicht in Hardware umgesetzt. Die Software-Funktionen aller vier Steuergeräte wurden zusammen auf einem Rapid Prototyping System im Fahrzeug implementiert. Sowohl das zeitgesteuerte, redundante Kommunikationssystem als auch das zeitgesteuerte Betriebssystemkonzept blieben damit noch unberücksichtigt.

## 8.2 Gesamtsystemmodellierung

Gemäß den Ergebnissen aus den untersuchten Modellierungsmethoden wurde ein Gesamtsystemmodell auf Basis der hybriden Modellierung mit dem Werkzeug Matlab/Simulink/Stateflow für das steer-by-wire System erstellt. Die folgenden Abschnitte geben zunächst einen Überblick zur entwickelten Simulationsumgebung und anschließend einen Einblick in die Realisierung ausgewählter Komponentenmodelle hinsichtlich deren funktionalen, strukturellen und sicherheitsgerichteten Modellierungsschwerpunkten.

### 8.2.1 Überblick zur Simulationsumgebung

Die Abbildung 8.2 zeigt die Struktur der entwickelten Simulationsumgebung. Kernstück der Simulationsumgebung bildet das in Simulink/Stateflow modellierte steer-by-wire System. Es verfügt über eine Nachbildung der Lenkradmechanik, der Elektromotoren für die Rückstellmomentgenerierung und die entsprechende Lenkradsensorik. All jene Komponenten sind im Strukturblock *Modell der Lenkradaktuatorik / -sensorik* zusammengefasst. Daran angeschlossen sind die beiden Lenkradsteuergeräte-Modelle, die die Funktions- und Schutzfunktionssoftware zur Regelung und Steuerung der Lenkradeinheit beinhalten. Diese beiden Steuergeräteeinheiten

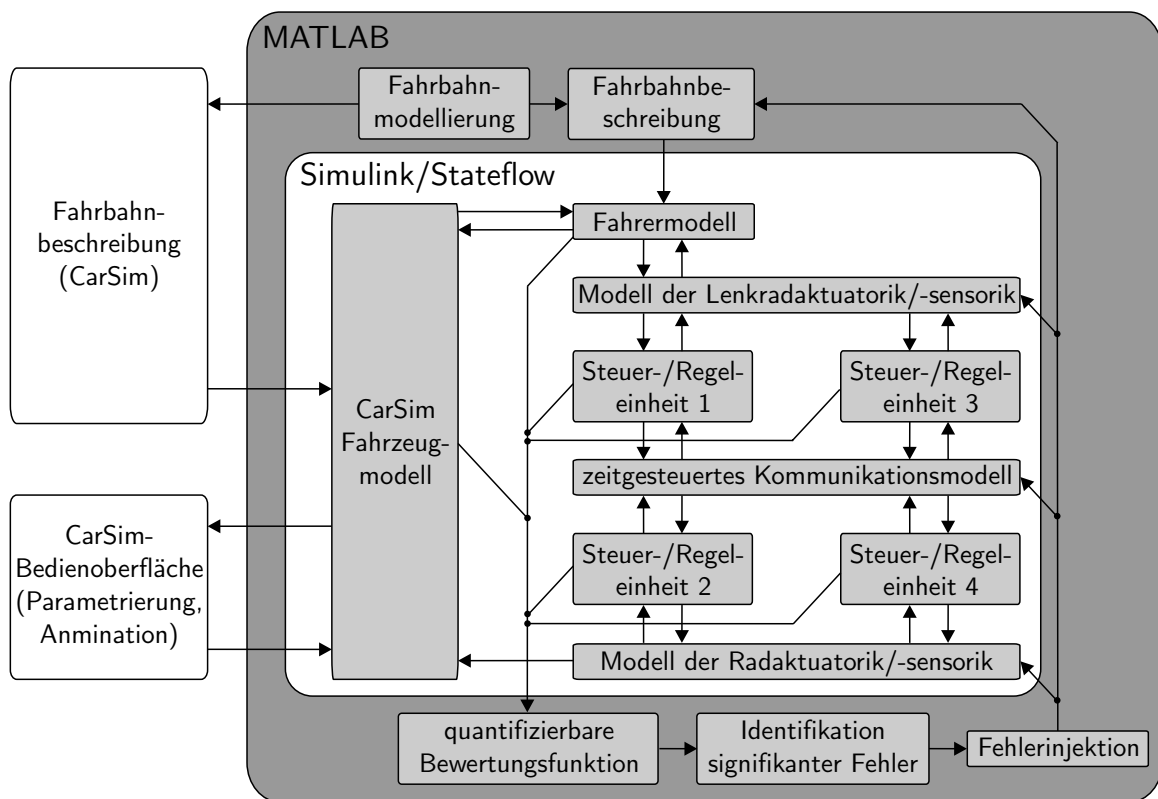


Bild 8.2: Struktur und Signalfluss der Simulationsumgebung unter MATLAB/Simulink

kommunizieren über ein Modell eines zeitgesteuerten Kommunikationssystems, das das Zeitverhalten bei der Übertragung der Daten zur Radaktuatoreinheit berücksichtigt. Über die beiden Radsteuergeräte-Modelle wird mithilfe des *Modells der Radaktuatorik/-sensorik* das *CarSim Fahrzeugmodell* beeinflusst.

Die für die Beschreibung der Systemumgebung notwendige Fahrbahnmodellierung, sowohl für das Fahrer- als auch für das Fahrzeugmodell, wird vor der eigentlichen Modellausführung in Matlab durchgeführt. Nach dem Simulationslauf wird mithilfe der objektiv-quantifizierbaren Bewertungsfunktion aus internen Größen des Fahrzeugmodells, der Steuer- und Regeleinheiten und dem Fahrermodell eine Kenngröße für die Sicherheitsbewertung berechnet. Sie ist Grundlage für den folgenden Identifikationsprozess, der neue, zu untersuchende Fehlerszenarien erzeugt und mit Hilfe der Fehlerinjektion ins steer-by-wire Modell einbringt.

Im Folgenden wird auf die Modellbildung des steer-by-wire Gesamtsystemmodells unter den drei Aspekten der Funktion, Struktur und der Sicherheit eingegangen. Ausgehend vom vorhandenen Rapid Prototyping Modell ist der schrittweise Ausbau der jeweiligen Modellkomponenten und die dabei zugrunde gelegte Vorgehensweise auf dem Weg zu einem Gesamtsystemmodell Gegenstand der Ausführungen.

## 8.2.2 Modellierung unter funktionalen Aspekten

### 8.2.2.1 Vom Rapid Prototyping zum funktionalen steer-by-wire Gesamtmodell

Das Rapid Prototyping Modell, das in einem Versuchsfahrzeug eingesetzt wurde, umfasst hauptsächlich die Systembestandteile: Signalerfassung, Regelung, Stellgrößengenerierung, Fehlererkennung und Systemzustandsmanagement. Sie sind alle in unveränderter Form Bestandteil des funktionalen Gesamtsystemmodells. Alle real vorhandenen Komponenten, wie z.B. die Leistungselektronik oder mechanischen Aktuatorkomponenten, die im Fahrzeug den Regelkreis des steer-by-wire Systems schlossen, wurden durch entsprechende Verhaltensbeschreibungen nachgebildet. Mit dem in den weiteren Abschnitten näher beschriebenen Fahrer-, Fahrzeug- und Systemumgebungsmodell ergänzt sich das funktionale steer-by-wire Modell zu einem Gesamtsystemmodell.

### 8.2.2.2 Fahrzeugmodell

Das im Rahmen dieser Arbeit eingesetzte, kommerzielle Simulationswerkzeug CarSim [Mec06] modelliert das Fahrzeug als Mehrkörpersystem mit neun Massen, deren Freiheitsgrade in Abbildung 8.3 dargestellt sind. Die gefederte Aufbaumasse hat drei rotatorische und drei translatorische Freiheitsgrade. Darüberhinaus verfügen sowohl die vier ungefederten Massen der Einzelradaufhängung als auch die vier Radmassen über je einen rotatorischen Freiheitsgrad.

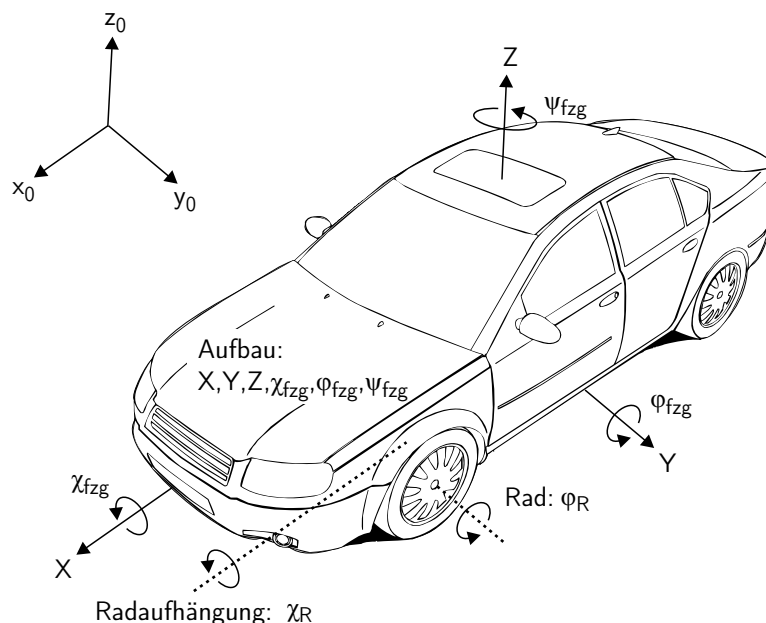


Bild 8.3: Körper des Fahrzeugmodells CarSim und deren Freiheitsgrade

Die Parametrierung des Fahrzeugmodells erfolgte anhand gemessener Daten eines realen Fahrzeugs. Die Modellgüte kann daher anhand eines direkten Vergleichs der Messdaten mit den korrespondierenden Simulationsdaten beurteilt werden. Dabei zeigt sich, dass das Modell das Verhalten des Fahrzeugs bis in den fahrdynamischen Grenzbereich nachbilden kann [Meh04, Tra05].

### 8.2.2.3 Fahrermodell

Für die Identifikation signifikanter Fehlerauswirkungen ist neben einem bis in den fahrdynamischen Grenzbereich aussagekräftigen Fahrzeugmodell auch eine geeignete Nachbildung des Fahrerhaltens unumgänglich. Insbesondere die Stärken und Schwächen des Menschen im Umgang mit den Reaktionen eines fehlerbehafteten sicherheitsrelevanten Kraftfahrzeugsystems sind für die Bewertung der Fehlerschwere von Bedeutung. Wie bereits in Kapitel 6.2.3 dargelegt, macht diese Anforderung den Einsatz eines eigenen, angepassten Fahrermodells erforderlich.

Ein Versuch, die im Menschen ablaufenden psychologischen und physiologischen Vorgänge beim Führen eines Fahrzeugs detailliert zu modellieren, erscheint im Rahmen dieser Arbeit nicht zielgerichtet, angemessen und Erfolg versprechend. Viel wichtiger, als die Modellierung eines individuellen Fahrerhaltens, ist für die Sicherheitsanalyse die Nachbildung der Verhaltensweise einer möglichst großen Fahrerpopulation. Aus [Sta06] geht hervor, dass für das Verhalten eines Durchschnittsfahrers folgende Annahmen getroffen werden können:

1. Durchschnittsfahrer erwarten aufgrund ihrer Erfahrung ein lineares Fahrzeugverhalten.
2. Durchschnittsfahrer reagieren in kritischen Fahrsituationen über.
3. Durchschnittsfahrer reagieren in einer Fehlersituation verzögert, da sie ihr Verhalten der ungewohnten Fahrzeugreaktion anpassen müssen.
4. Durchschnittsfahrer reduzieren die Anforderung am Fahrpedal, wenn sie feststellen, dass sie sich in einer kritischen Fahrsituation befinden.

Für die Modellierung des Fahrers wird zunächst dessen Verhalten in fehlerfreien und problemlos beherrschbaren Fahrsituationen durch geeignete Regler für die Quer- und Längsbewegung beschrieben. Dieses Verhalten wird als nominales Fahrerhalten bezeichnet. Die in dieser Beispielanwendung zum Einsatz kommenden Ansätze für Quer- und Längsregelung sind in [Sta06] beschrieben.

Die in fehlerbehafteten Fahrsituationen vom Fahrer hervorgerufenen Reaktionen werden, wie in Abbildung 8.4 dargestellt, durch nachgeschaltete Überlagerung des Nominalverhaltens mit si-

tuationsabhängigen Verhaltensmodellen berücksichtigt. Als Indikator für eine unerwartete Fahrzeugbewegung dient dabei das Gefahrenmaß

$$\sigma_{\dot{\Psi},\delta} = \frac{1}{2} \left( \tanh \left( s_{\dot{\Psi},\delta} \left( \Delta\dot{\Psi}_\delta - \Delta\dot{\Psi}_{\delta,max} \right) \right) + 1 \right) \quad (8.1)$$

im Intervall  $[0, 1]$  mit

$$\Delta\dot{\Psi}_\delta = \left| \dot{\Psi} - \dot{\Psi}_\delta \right|$$

$s_{\dot{\Psi},\delta}$  Anstiegskoeffizient  
 $\dot{\Psi}$  Gierratenabweichung  
 $\dot{\Psi}$  Istgierrate des Fahrzeugs  
 $\dot{\Psi}_\delta$  aufgrund des Fahrerlenkwinkels zu erwartende Gierrate

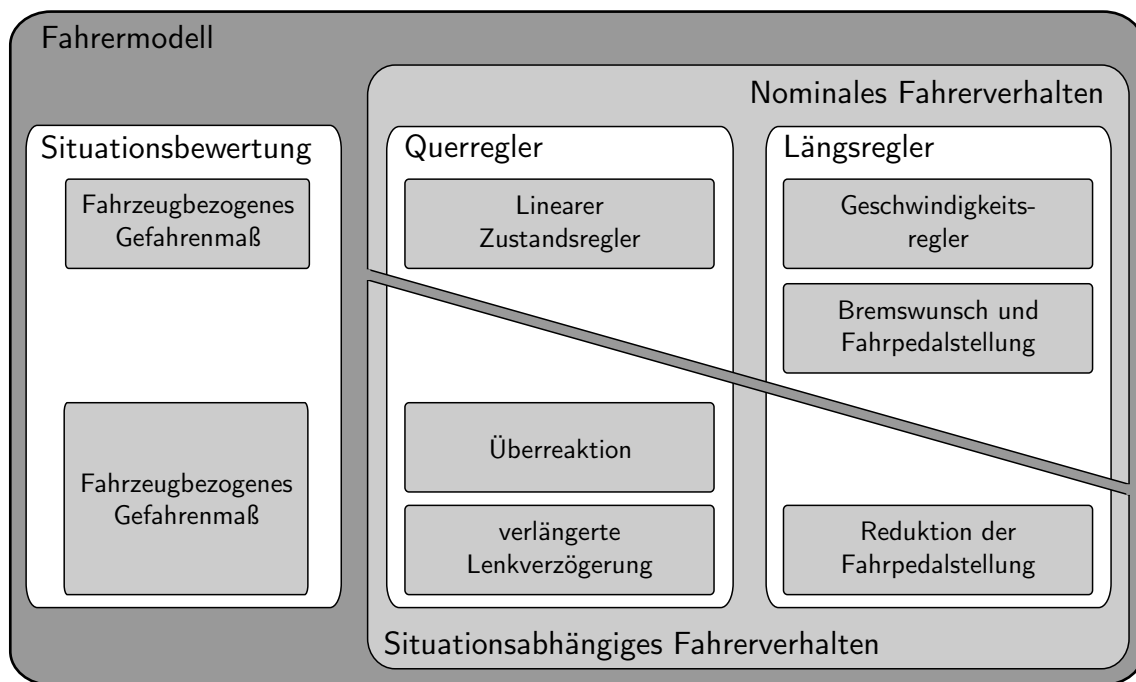


Bild 8.4: Struktur des Fahrermodells aus Quer- und Längsregler mit Unterteilung in nominales und situationsabhängiges Fahrerverhalten nach [Sta06]

Übersteigt das Gefahrenmaß  $\sigma_{\dot{\Psi},\delta}$  eine Schwelle, wird ein Intensitätsfaktor  $e_p \in [0, 1]$  sprunghaft auf 1 gesetzt und die Überlagerung des nominalen Fahrerverhaltens durch dessen Fehlverhalten ausgelöst. Die Adaption des Fahrerverhaltens an die neue Fahrsituation erfolgt durch eine Abnahme von  $e_p$  mit einer vom Bewegungszustand des Fahrzeugs unabhängigen Mindestgeschwindigkeit und wird beschleunigt, wenn  $\sigma_{\dot{\Psi},\delta}$  klein wird.

Die Parametrierung der Regler hinsichtlich der Vorausschauweite des Fahrers auf den vor ihm liegenden Sollkurs, der situationsabhängigen Überreaktion, der Totzeit in ungewohnten Fehlerzenarien und der maximalen Lenkgeschwindigkeit wurde anhand von Daten eigener Fahrversuche und veröffentlichter Untersuchungsergebnisse [NK03] exemplarisch durchgeführt. Die detaillierte Beschreibung der Fahrerreaktion bei der Interaktion mit fehlerbehafteten Fahrzeugsystemen war nicht Kernziel der vorliegenden Arbeit, sondern ist weitergehend in zukünftigen Forschungsaktivitäten zu untersuchen.

### 8.2.3 Modellierung der Systemumwelt

Um die Untersuchungsgrundlage variantenreich zu gestalten, kommt im Falle des steer-by-wire Systems ein Fahrmanöver-Katalog aus verschiedensten Fahrsituationen zum Einsatz. Insgesamt sind zehn verschiedene Fahrmanöver definiert, die die Fahrzeug-Fahrer-Einheit zusammen mit dem steer-by-wire System in völlig unterschiedliche Fahrszenarien versetzt. Damit ist gewährleistet, dass sowohl das Fahrzeug als auch das steer-by-wire System in unterschiedlichsten Betriebszuständen untersucht werden können. Die festgelegten Umweltszenarien reichen von langsamen Parkiervorgängen bis hin zu hochdynamischen Fahrsituationen und sind im einzelnen wie in Tabelle 8.1 definiert. Sie besitzen jeweils eine Zeitdauer von 15 Sekunden. Auf die Modellierung unterschiedlicher Straßenreibwerte oder Windbedingungen wurde verzichtet.

Tabelle 8.1: Fahrmanöver-Katalog

Fahrmanövernummer	Beschreibung des Fahrmanövers	Fahrzeugsollgeschwindigkeit
1	Parkiervorgang	
2	Abbiegevorgang rechts	30 km/h
3	Slalomkurs mit Pylonenabstand 18 Meter	40 km/h
4	Doppelter Spurwechsel	120 km/h
5	Doppelter Spurwechsel	80 km/h
6	Hochgeschwindigkeitsoval	100 km/h
7	Spurgasse geradeaus mit Breite 2,5 Meter	200 km/h
8	Spurgasse geradeaus mit Breite 2,5 Meter	150 km/h
9	Spurgasse geradeaus mit Breite 2,5 Meter	80 km/h
10	Spurgasse geradeaus mit Breite 2,5 Meter	50 km/h

## 8.2.4 Modellierung unter strukturellen Aspekten

### 8.2.4.1 Komponentenstruktur

Wie sich bereits aus der Beschreibung der Simulationsumgebung erkennen lässt, ist auch die Systemtopologie des steer-by-wire Systems im Gesamtsystemmodell umgesetzt. Dies bezieht sich insbesondere auf die Abbildung der tatsächlich vorhandenen Schnittstellen. Spielt dieser Modellierungsgrad aus rein funktionaler Sicht noch keine Rolle und ist auch im Rapid Prototyping Modell nicht realisiert, ist dieses Detail jedoch in Bezug auf die spätere Fehlerinjektion nicht mehr zu vernachlässigen. Wird beispielsweise ein unter funktionalen Aspekten modellierter Winkelsensor direkt einen Winkelwert ausgeben, ist unter strukturellen Aspekten dessen tatsächliche Schnittstelle z.B. durch einen Ausgang mit analogem Spannungswert abzubilden. Aus struktureller Sicht und insbesondere hinsichtlich der späteren Fehlerinjektion hat die Art und Weise der Informationswandlung und -übertragung einen erheblichen Einfluss auf die möglichen Fehler und deren Auswirkungen auf den gemessenen Winkelwert.

Auch redundante Funktionen und Komponenten, die aus einer rein funktional orientierten Betrachtungsweise keine zusätzlichen Erkenntnisse bringen, müssen modelliert werden. Die Zweikanaligkeit oder auch das redundante Kommunikationssystem des steer-by-wire Systems sind dafür Beispiele.

### 8.2.4.2 Zeitstruktur

Neben der Systemtopologie spielt jedoch bei der Berücksichtigung struktureller Aspekte der zeitliche Systemablauf eine wichtige Rolle. Insbesondere in einem zeitgesteuerten System, wie es bei dem hier betrachteten steer-by-wire System der Fall ist, muss zusätzlich die Zeitstruktur für eine Betrachtung der Fehlerauswirkungen berücksichtigt werden.

Da weder eine zeitgesteuerte Tasksteuerung noch das zeitgesteuerte Kommunikationssystem im Rapid Prototype Berücksichtigung fand, mussten diese Systembestandteile im Gesamtsystemmodell zusätzlich abgebildet werden. Kern der systemweiten Zeitstruktur bildete der TTCAN-Fahrplan für die zeitgesteuerte Kommunikation. Er ist in Abbildung 8.5 dargestellt. Sowohl die Kommunikation als auch die Datenverarbeitung richteten sich im Gesamtsystemmodell nach diesem Zeitplan. Bei der Modellierung des Kommunikationssystems wurde auf eine Nachbildung des Nachrichtenprotokolls auf Bitebene verzichtet und stattdessen eine Abstraktion auf Botschaftsebene gewählt. Diese Abstraktion wird als zulässig angesehen, da erstens die Implementierung des Nachrichtenprotokolls selbst nicht im Fokus der Untersuchung steht, sondern der Untersuchungsschwerpunkt auf der Integration eines solchen Kommunikationssystems in

ein übergeordnetes Kraftfahrzeugsystem liegt. Zum zweiten sind die relevanten Fehlermöglichkeiten, wie z.B. eine fehlerhafte Synchronisation zwischen TTCAN und Betriebssystem oder inkonsistente Datenübertragung innerhalb des redundanten Kommunikationssystem, auch auf dieser Abstraktionsstufe abbildbar (vgl. Kap 8.2.5.2). Diese Abstraktion ließ die Reduktion der Simulationsschrittweite auf  $62,5\mu s$  zu.

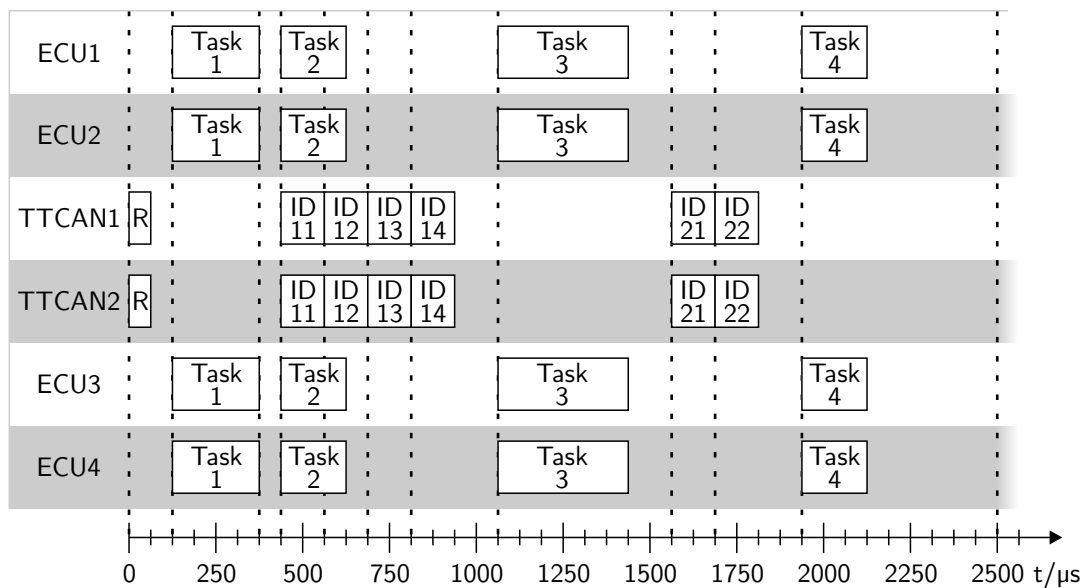


Bild 8.5: Zeitstruktur des Gesamtsystemmodells

## 8.2.5 Modellierung unter Sicherheitsaspekten

### 8.2.5.1 Fehlererkennung und -behandlung

Die bis zum Zeitpunkt des Einfrierens implementierten Fehlererkennungen und die zugehörigen -behandlungen waren schon Bestandteil des Rapid Prototyping Modells. Sie wurden unverändert in das Gesamtsystemmodell übernommen, denn sie zählen unter anderem zu den zu analysierenden Objekten in der modellbasierten ganzheitlichen Sicherheitsanalyse.

### 8.2.5.2 Fehlermodelle

Anhand der bereits vorgestellten Klassifizierung von Komponenten werden im weiteren exemplarisch einige ausgewählte Systemkomponenten bezüglich ihres Fehlerverhaltens und dessen Nachbildung im Modell diskutiert. Dabei liegen die Schwerpunkte zum ersten auf der Frage nach der Vorgehensweise zur Bestimmung der Fehlermöglichkeiten, zum zweiten auf der Frage



nach dem notwendigen Modelldetaillierungsgrad und zum dritten auf den tatsächlich in diesem Anwendungsbeispiel modellierten Komponentenfehlern.

Um die Vorgehensweise anhand möglichst unterschiedlicher Komponenten aufzeigen zu können, sind im folgenden neben einem generischen Fehlermodell Komponenten der Klassen 1,3 und 4 für die Fehlermodellierung ausgewählt.

### Generisches Fehlermodell

Das in diesem Anwendungsfall eingesetzte generische Fehlermodell zeigt Abbildung 8.6. Es beinhaltet die Modellierungen von Fehlern im Wertebereich, wie z.B. konstante und zeit-variante Offsets, additives Rauschen oder Signalamplitudenbeschränkungen, und im Zeitbereich, wie z.B. Verzögerungen oder Gradientenbeschränkungen.

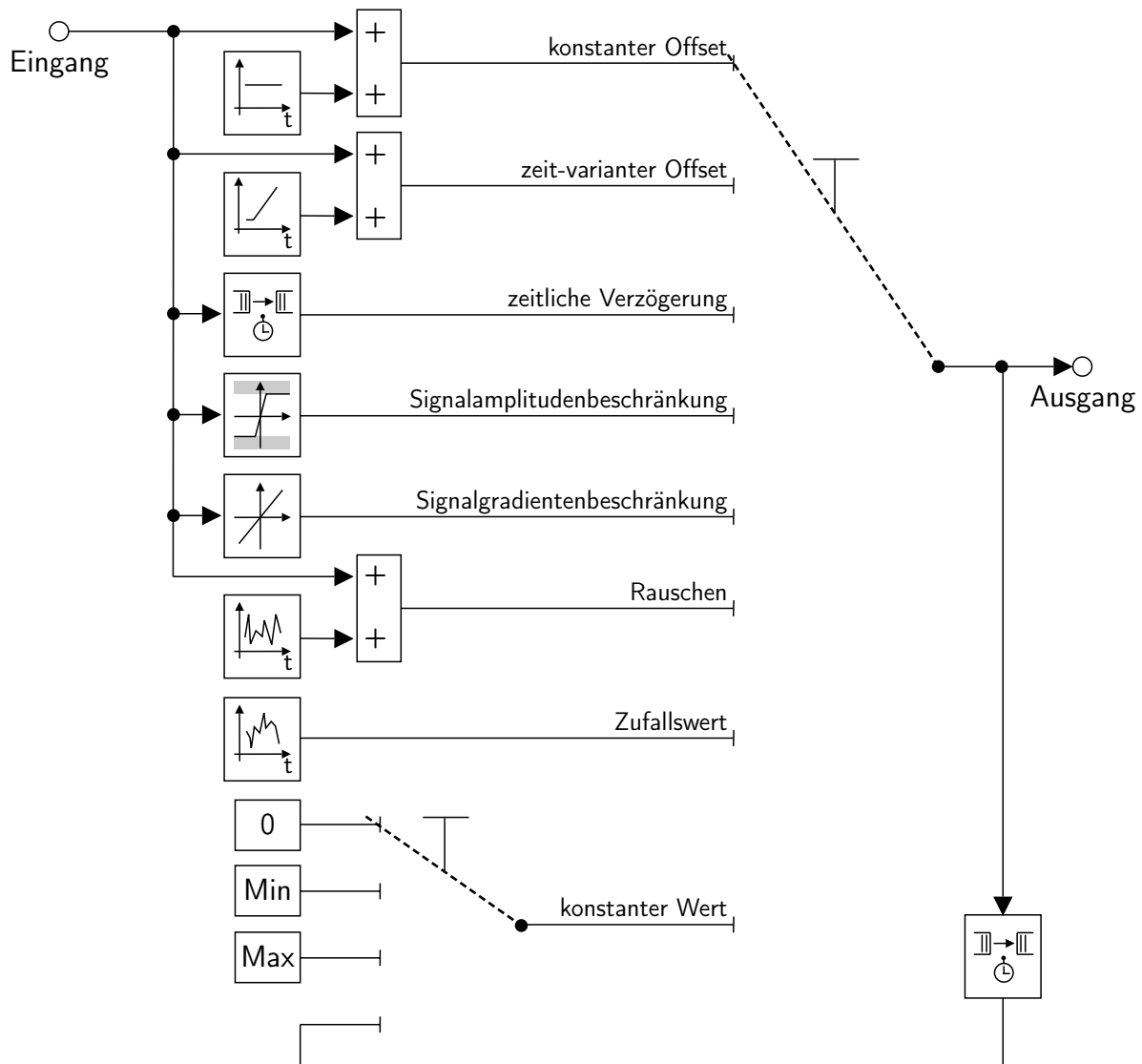


Bild 8.6: Generisches Fehlermodell

Die Schwere dieser Fehler, definiert als die Größe der Veränderung des Ausgangssignals im Vergleich zum Eingangssignal, ist unter Berücksichtigung der spezifizierten Eingangssignalwerte für Amplitude und Gradient in fünf Abstufungen wählbar. Außerdem finden weitere Fehler ohne Schwereabstufung im generischen Fehlermodell Berücksichtigung. Hierzu zählt die Erzeugung von Zufalls- oder Konstantwerten am Ausgang des Fehlermodells.

### Sensorfehler des Rotorpositionssensors (RPS)

Die zur Regelung der eingesetzten brüstenlosen permanentmagneterregten Synchronmaschinen verwendeten Rotorpositionssensoren (RPS) erfassen den Winkel der aktuellen Rotorstellung. Sie gehören zur Komponentenklasse 1 und verfügen damit weder über eigene Intelligenz noch über Fehlererkennungs- und -behandlungsmechanismen. Das Messprinzip der Sensorik basiert auf dem anisotropen Magnetwiderstandseffekt (AMR). Dieser Effekt gründet auf dem Prinzip, dass ein stromdurchflossener Leiter in einem Magnetfeld eine Widerstandsänderung erfährt, die eine Funktion des Winkels zwischen Magnetisierungs- und Stromrichtung ist. Das differentielle Ausgangssignal ( $+V_0$ ,  $-V_0$ ) einer Wheatstone-Brücke repräsentiert diese Widerstandsänderung und ist proportional zu  $\sin(2\alpha)$  (vgl. Bild 8.7).

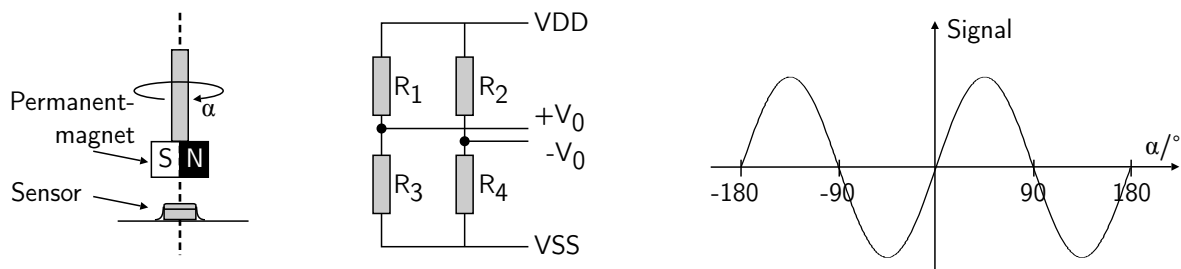


Bild 8.7: Grundlegendes Funktionsprinzip eines AMR-Sensors

Integriert man zwei dieser Sensoren in einem Gehäuse und positioniert sie in einer Ebene in einem Winkel von  $45^\circ$  zueinander, so weisen die beiden elektrischen Ausgangssignale eine Phasendifferenz von  $90^\circ$  auf. Mathematisch lassen sie sich somit durch

$$X(\alpha, T) = X_0(T) \sin(2\alpha) \quad (8.2)$$

$$Y(\alpha, T) = Y_0(T) \cos(2\alpha) \quad (8.3)$$

beschreiben. Nimmt man an, dass die integrierten Sensoren der gleichen Umgebungstemperatur  $T$  ausgesetzt sind und damit die gleiche Amplitude  $X_0 = Y_0$  aufweisen, so kann der Winkel  $\alpha$  mit folgender Berechnungsvorschrift aus den beiden Ausgangssignalen des RPS-Sensors ermittelt werden:

$$\alpha = \frac{1}{2} \arctan \frac{X}{Y} \quad (8.4)$$

Aufgrund dieses Messprinzips ist das Verfahren unabhängig von Feldstärkenänderungen während der Betriebszeit, von Änderungen des Magnetfelds durch Temperatureinflüsse, von mechanischen Fertigungstoleranzen und von mechanischen Veränderung hervorgerufen durch thermische Belastungen.

Führt man für diesen Sensortyp eine FMEA-Analyse durch, so erhält man folgende Fehlermöglichkeiten:

**Schwaches magnetisches Feld** Bei einer nicht ausreichenden magnetischen Feldstärke lassen sich zwei Effekte beobachten. Zum einen kommt es zu einer Überlagerung des Ausgangssignals mit einer sinusförmigen Störung, die proportional zu  $\sin^2(4\alpha)$  ist (vgl. Bild 8.8). Die Amplitude dieser Störung ist abhängig vom tatsächlichen Schwächungsgrad des magnetischen Felds. Der Fehler wirkt auf beide Ausgangssignale gleichermaßen und hat damit einen Fehler im gemessenen Winkel zur Auswirkung. Zum anderen tritt beim Messen über größere Winkelbereiche eine Hysterese auf. Sie führt zu einer Messwinkeldifferenz zwischen identischen Messpositionen.

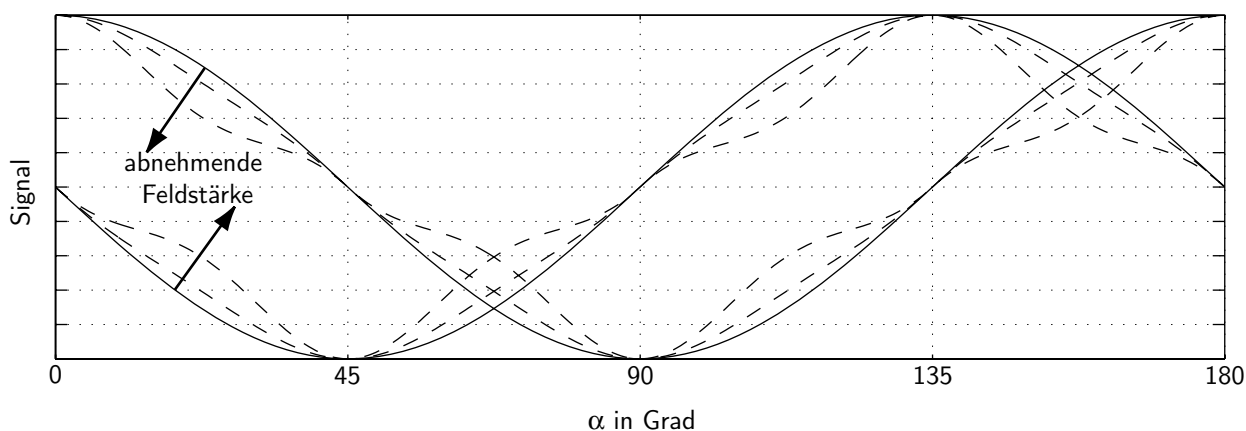


Bild 8.8: Signalverlauf bei zu schwachem Magnetfeld

**Inhomogenes magnetisches Feld** Befindet sich aufgrund einer mechanisch, nicht präzisen Anordnung die Rotationsachse des Magneten nicht exakt in der Mitte des Sensors, so wird die Messung in einem inhomogenen Magnetfeld durchgeführt. Der Grad des daraus resultierenden Messfehlers hängt stark von der jeweiligen Anordnung ab. Er lässt sich jedoch verallgemeinert abschätzen zu:

$$E_{inhomogen} = \frac{320^\circ}{(w + l)^2} \cdot R^2 \quad (8.5)$$

mit

$R$  Abstand der Rotationsachse des Magneten zum Sensormittelpunkt

$w$  Breite des Magneten

$l$  Länge des Magneten

**Konstanter und zeitvarianter Offset** Ein Signaloffset kann sowohl auf nur einen als auch auf beide Ausgangskanäle Auswirkungen haben. Wird zur Beschreibung dieser Fehlereffekte ein Offset in das mathematische Modell des Sensors eingeführt, so ergibt sich:

$$X(\alpha, T) = X_0(T) \sin(2\alpha) + \Delta x(t) \quad (8.6)$$

$$Y(\alpha, T) = Y_0(T) \cos(2\alpha) + \Delta y(t) \quad (8.7)$$

Der daraus resultierende Winkelfehler wird berechnet zu:

$$E_{offset}(\alpha, \Delta x(t), \Delta y(t)) = \left| \alpha - \frac{1}{2} \arctan \left( \frac{X_0 \sin 2\alpha + \Delta x(t)}{Y_0 \cos 2\alpha + \Delta y(t)} \right) \right| \quad (8.8)$$

**Unterschiedliche Signalamplituden der beiden Kanäle** Sind die Signalamplituden der beiden integrierten Sensorelemente unterschiedlich, so führt dies zu einem Winkelfehler folgender Größe:

$$E_{amplitude}(\alpha, A) = \left| \alpha - \frac{1}{2} \arctan \left( A \frac{\sin 2\alpha}{\cos 2\alpha} \right) \right| \quad \text{mit} \quad A = \frac{X_0}{Y_0} \quad (8.9)$$

**Phasendifferenz der beiden Kanäle** Beträgt die Phasendifferenz zwischen den beiden Signalen nicht exakt  $90^\circ$ , zeigt sich ein Messfehler, der folgendermaßen mathematisch beschrieben werden kann:

$$X = X_0 \sin(2\alpha + \Delta\beta(\alpha)) \quad (8.10)$$

$$Y = Y_0 \cos 2\alpha \quad (8.11)$$

Daraus ergibt sich für den absoluten Winkelfehler folgende Funktion:

$$E_{phase}(\alpha, \Delta\beta) = \left| \alpha - \frac{1}{2} \arctan \left( \frac{\sin(2\alpha + \Delta\beta(\alpha))}{\cos 2\alpha} \right) \right| \quad (8.12)$$

**Schnittstellenfehler** Zu den bisher erörterten Fehlermöglichkeiten, die ausschließlich aus dem Messprinzip des Sensors abgeleitet werden können, sind an der Schnittstelle des Sensors weitere Fehler möglich. Hierzu zählen alle Fehlermöglichkeiten, die auch schon im generischen Fehlermodell betrachtet wurden und deshalb nicht erneut erwähnt werden.

Alle durch die FMEA erfassten Fehlermöglichkeiten und die daraus resultierenden Verhaltensmuster des Sensors sind ebenfalls in einem Matlab/Simulink-Modell nachgebildet. Insgesamt besitzt das komplettierte Sensormodell 28 Fehlermöglichkeiten. Darüber hinaus sind all jene Fehlermöglichkeiten, die in Abhängigkeit weiterer Parameter veränderliche Fehlerauswirkungen zeigen, mit einem so genannten Schwereparameter versehen. Er ermöglicht die Variation der Schwere der Fehlerauswirkungen in fünf festgelegten Abstufungen von minimaler bis schwerwiegender Abweichung vom Nominalverhalten.

### **Fehlermöglichkeiten im TTCAN-Kommunikationssystem**

Im Vergleich zum Rotorpositionssensor handelt es sich beim Kommunikationssystem um eine deutlich komplexere Systemkomponente. Sie verfügt gemäß der Einteilung in die Komponentenklasse 3 bzw. 4 über eigene Intelligenz und Fehlererkennungs- und -behandlungsmechanismen. Ihre Fehlermöglichkeiten können deshalb gemäß den Erläuterungen aus Abschnitt 6.4.1 mit Hilfe des Zustandautomaten des TTCAN-Fehlermanagements modelliert werden. Das vollständige und in Matlab/Simulink modellierte Zustandübergangsdiagramm für den einkanaligen TTCAN-Kommunikationsknoten ist in Abbildung 8.9 dargestellt.

Die Erweiterung des einkanaligen TTCAN-Kommunikationsnetzes zu einem redundanten, zeitlich synchronisierten Kommunikationssystem ist in Anhang A beschrieben. Durch den dabei zum Einsatz kommenden Synchronisationsalgorithmus entstehen auf Systemebene neue Fehlermöglichkeiten, die in einer Fehleranalyse bestimmt und untersucht werden konnten. Das Ergebnis dieser Analyse findet sich im Anhang A.3. Alle dort beschriebenen Fehler und deren Auswirkungen wurden ebenfalls in einem Matlab/Simulink-Modell nachgebildet. Insgesamt erhält man damit 50 Fehlerszenarien in wiederum fünf Schwereabstufungen für das redundante, synchronisierte TTCAN-Kommunikationssystem mit vier Kommunikationsteilnehmern.

## **8.3 Identifikation signifikanter Fehlerszenarien**

Mit Abschluss der Modellierung unter funktionalen, strukturellen und sicherheitsgerichteten Aspekten ist die Grundlage geschaffen, um die in Kapitel 7 beschriebene Vorgehensweise zur Identifikation signifikanter Fehlerszenarien anzuwenden. Die dazu notwendigen Funktionen und Parametrierungen sowie die Ergebnisse einer Identifikation beispielhaft ausgewählter Fehler sind Bestandteil der nachfolgenden Ausführungen.

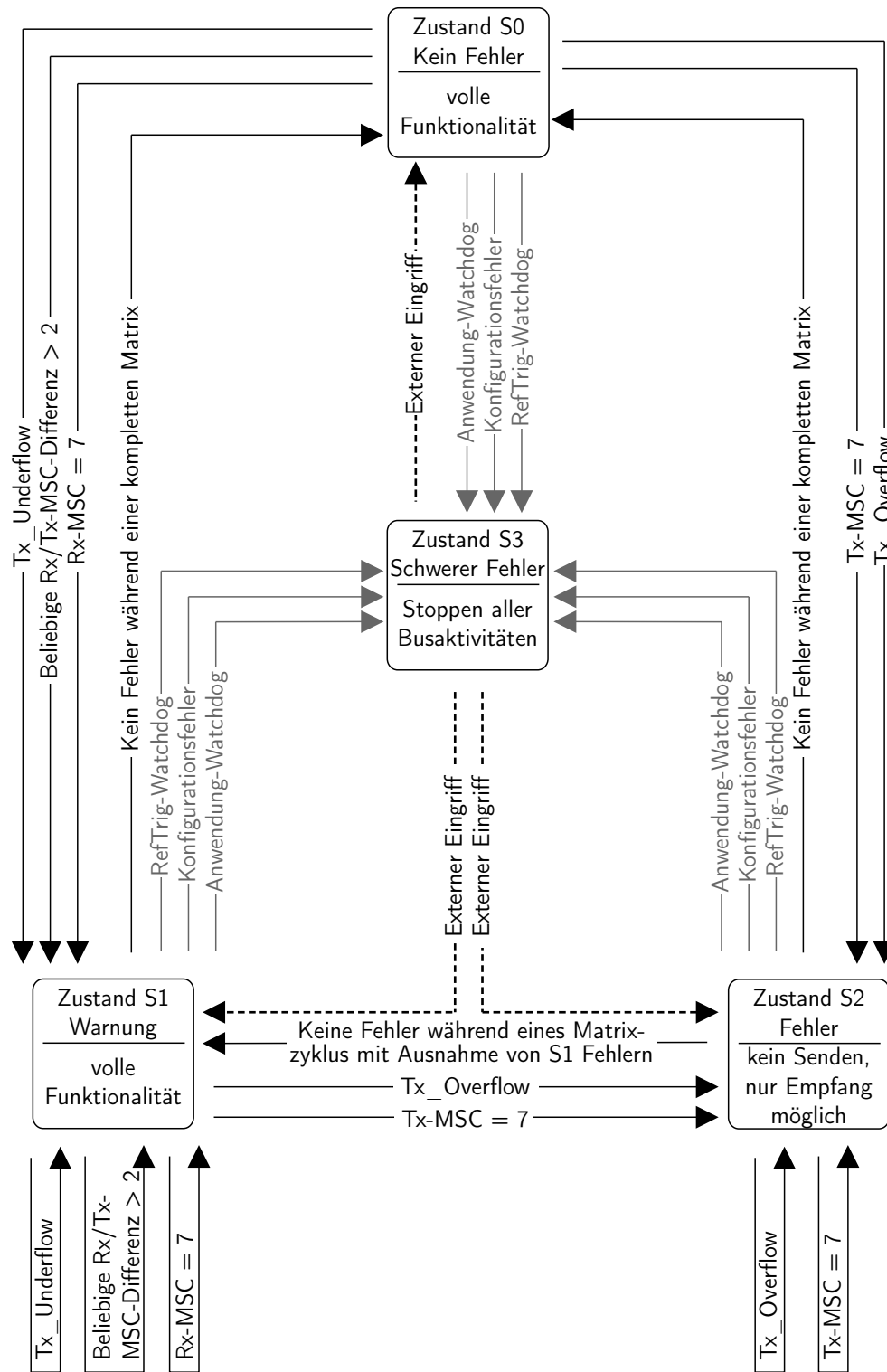


Bild 8.9: Zustandsautomat im TTCAN-Protokoll

### 8.3.1 Objektiv-quantifizierbare Bewertungsfunktion

Die objektiv-quantifizierbare Bewertungsfunktion für das steer-by-wire System setzt sich im vorliegenden Anwendungsfall aus drei Teilen zusammen. Zunächst spielt für die Bewertung der

Schwere eines eingespeisten Fehlers das Gefühl des Fahrers während dieses Fehlers eine wichtige Rolle. Er hat, wie bereits erläutert, als Führer seines Kraftfahrzeugs eine Erwartungshaltung vom Verhalten seines Fahrzeugs. Sobald während eines Fehlers das gezeigte vom erwarteten Fahrzeugverhalten abweicht, ist es dem Fahrer möglich einen eventuellen Fehler zu erkennen und zu reagieren. Diese Reaktion kann entweder richtig erfolgen und die Fehlerauswirkungen mildern oder im deutlich schlimmeren Szenario werden die Auswirkungen des Fehlers sogar verstärkt. In jedem Fall reicht das Gefühl eines Fahrers aber in einer derartigen Situation von unwohl bis unsicher oder gefährdet. Als Gradmesser für dieses Gefühl dient die in Formel 8.1 hergeleitete Größe  $\sigma_{\dot{\psi},\delta}$ .

Das zweite Bewertungskriterium zur Ermittlung der Schwere des eingespeisten Fehlers stellt die tatsächliche Abweichung des Fahrzeugs von der vorgegebenen Soll-Trajektorie dar. Der Fahrer soll in der Lage sein, das Fahrzeug grundsätzlich in der Mitte seiner Fahrbahn führen zu können. Ist dies aufgrund des eingespeisten Fehlers nicht möglich, kommt es zur Abweichung zwischen Soll- und Ist-Fahrzeugbewegung. Die Fläche zwischen diesen Trajektorien wird als Indikator für die Schwere des eingespeisten Fehlers verwendet. Um Abweichungen, die zu einem Verlassen der eigenen Fahrspur in Richtung Seitenstreifen oder gar dem Gegenverkehr führen, als besonders schwerwiegend einzustufen, wird zusätzlich für diese Fälle ein Bestrafungsfaktor  $\rho_{spur} > 1$  eingeführt. Zusammen sieht die für dieses Bewertungskriterium entwickelte Berechnungsvorschrift folgendermaßen aus:

$$\sigma_{fahrzeug} = \int_0^s |Abstand\_Soll-Ist-Trajektorie(s)| ds \cdot \begin{cases} 1 & \text{falls } |Abstand\_Soll-Ist-Trajektorie(s)| \leq 0,5\text{m} \\ \rho_{spur} & \text{falls } |Abstand\_Soll-Ist-Trajektorie(s)| > 0,5\text{m} \end{cases} \quad (8.13)$$

Nimmt das Kraftfahrzeugsystem aufgrund des eingespeisten Fehlers darüber hinaus einen Zustand ein, in dem es die eigentliche Funktion zwar weiterhin erbringen kann, das System aber seine Fehlertoleranzeigenschaften einbüßt, so kann dies mit den bisherigen Kriterien bei der Bewertung der Fehlerschwere noch nicht berücksichtigt werden. Deshalb bildet eine Zuordnungstabelle aus Rückfallebene und Bewertungsmaß  $\sigma_{system}$  den dritten Bestandteil der objektiv-quantifizierbaren Bewertungsfunktion. Je nach eingennommener Rückfallebene und der damit noch verfügbaren Funktionalität und Fehlertoleranz geht eine konstante Bewertung der Situation in die Gesamtbewertung ein.

Die objektiv-quantifizierbare Bewertungsfunktion für den vorliegenden Anwendungsfall ergibt sich damit zu

$$\sigma_{gesamt} = wp_1 \cdot \sigma_{\dot{\psi},\delta} + wp_2 \cdot \sigma_{fahrzeug} + wp_3 \cdot \sigma_{system} \quad (8.14)$$

Dabei stellen die Parameter  $wp_1$ ,  $wp_2$  und  $wp_3$  frei wählbare Gewichtungsfaktoren für die einzelnen Bestandteile der Bewertungsfunktion dar.

### 8.3.2 Parametrierung, Fehlerinjektion und Ergebnisse

Das Untersuchungsziel für den hier ausgewählten Beispielfall beschränkt sich auf Einzelfehler der RPS-Sensorik. In Anlehnung an das zugehörige Fehlermodell und die Systemarchitektur beläuft sich die Anzahl auf 112 injizierbare Fehler mit jeweils bis zu fünf Schwereabstufungen. Das steer-by-wire System befindet sich vor der Fehlerinjektion immer in einem fehlerfreien Zustand. Die Systemumgebungsbedingungen können vom Identifikationsalgorithmus gemäß dem vorgestellten Fahrmanöverkatalog frei ausgewählt werden. Der Aktivierungszeitpunkt des injizierten Fehlers kann maximal zwischen der ersten und der zehnten Sekunde, der Deaktivierungszeitpunkt zwischen der ersten und der zwölften Sekunde des Fahrmanövers variieren. Die Parameter des Identifikationsprozesses sind folgendermaßen gewählt: 30 Individuen bilden die Population. Die maximale Anzahl der durch den evolutionären Algorithmus erzeugten Generationen beträgt 50. Dabei werden zur Erzeugung von Nachkommen folgende evolutionäre Operatoren in jedem Generationsschritt eingesetzt: viermal Uniform Mutation, einmal Boundary Mutation, dreimal Non-Uniform Mutation, einmal Simple Crossover, einmal Arithmetical Crossover und einmal Extended Arithmetical Crossover. Daraus ergibt sich, dass in jeder neuen Generation 16 Individuen der Vorgängergeneration überleben und die verbleibenden 14 Plätze in der Population durch neu kreierte Nachkommen aufgefüllt werden. Der Einfluss dieser Parameter auf das Untersuchungsergebnis wird im nachfolgenden Abschnitt 8.3.6 noch detailliert diskutiert.

Der Verlauf des Identifikationsprozesses ist in den Abbildungen 8.10 - 8.13 dokumentiert. Es ist deutlich zu erkennen, dass das objektiv-quantifizierbare Bewertungsmaß der Fehlerauswirkungen im Laufe des Identifikationsprozesses gegen ein Maximum konvergiert (vgl. Bild 8.10). Das lässt sich auch aus der Darstellung der Variablenwerte des signifikantesten Fehlers in Abbildung 8.11 erkennen. Hier kann beobachtet werden, dass am Anfang des Identifikationsprozesses zunächst noch deutliche Schwankungen in den Variablenwerten auszumachen sind. Gegen Ende der Identifikation hingegen haben sich die Variablenwerte stabilisiert. Dieser fortwährende Verbesserungsprozess während der Suche nach signifikanten Fehlerszenarien wird in den Abbildungen 8.12 und 8.13 deutlicher. Diese Darstellungen zeigen im Vergleich die Strukturen der Population zum Zeitpunkt der ersten und letzten Generation. Es lässt sich sehr gut erkennen, dass der Identifikationsalgorithmus innerhalb des aufgespannten Fehlerraums eine klare Struktur von Variablenkombinationen gefunden hat, die signifikante Fehlerauswirkungen zufolge haben.



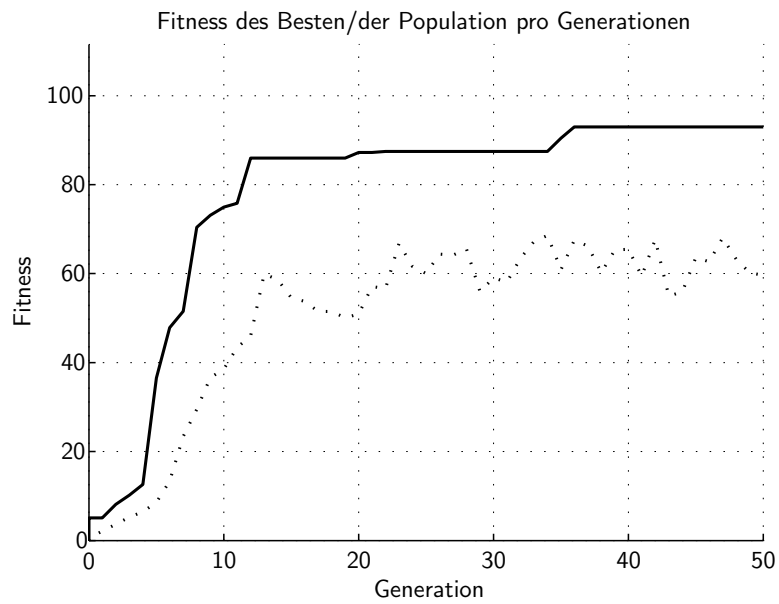


Bild 8.10: Fitnesswert des signifikantesten Fehlers und durchschnittlicher Fitnesswert aller injizierten Fehler über den Generationen

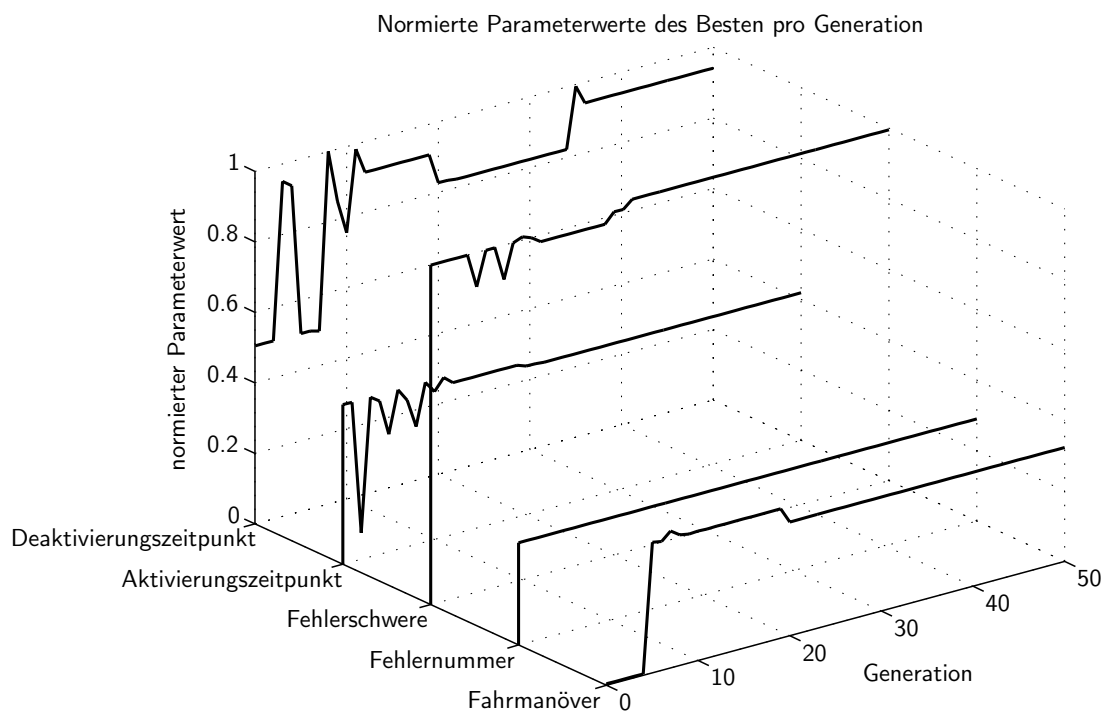


Bild 8.11: Variablenwerte des signifikantesten Fehlers über den Generationen

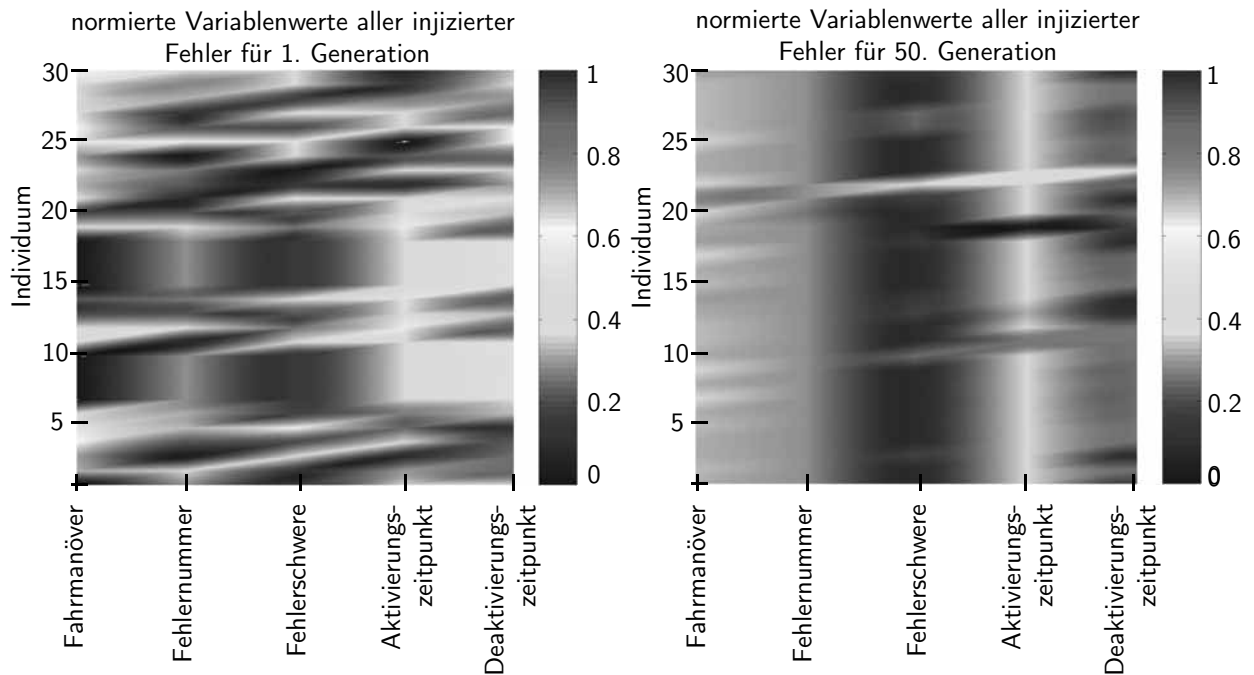


Bild 8.12: Normierte Variablenwerte aller injizierten Fehler in Farbteppich-Darstellung; links: zu Beginn der Identifikation; rechts: am Ende der Identifikation

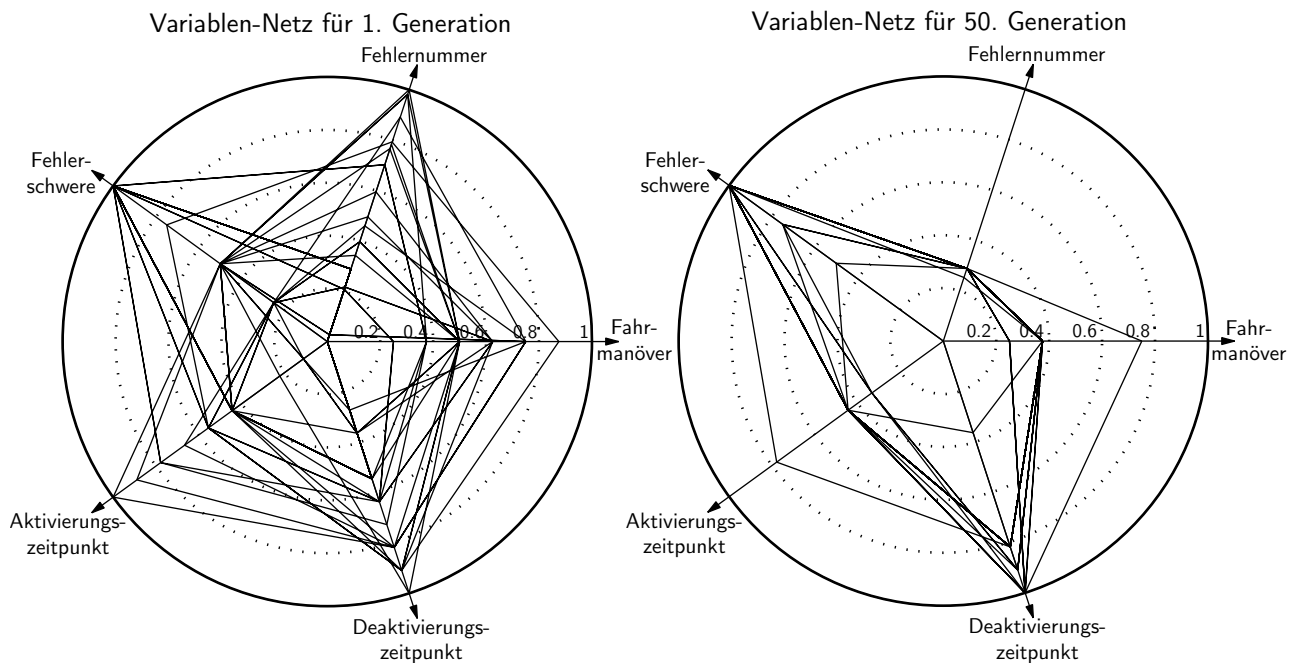


Bild 8.13: Normierte Variablenwerte aller injizierten Fehler in Spinnennetz-Form; links: zu Beginn der Identifikation; rechts: am Ende der Identifikation

Am Ende des Identifikationsprozesses steht eine Population, die eine Menge unterschiedlichster Fehler mit signifikanten Auswirkungen enthält. Die schwerwiegendsten Fehlerauswirkungen, die durch das größte gefundene Bewertungsmaß ausgedrückt werden, zeigt dabei folgender injizierter Fehler:

Fahrmanöver	=	4	...	Doppelter Spurwechsel 120 km/h
Fehlernummer	=	33	...	Rauschen im Sinus-Kanal des RPS-Sensors
Fehlerschwere	=	5	...	bis zu 10% Abweichung durch Rauschen
Aktivierungszeitpunkt	=	1,2	Sekunden	
Deaktivierungszeitpunkt	=	8,8	Sekunden	

Das bei der Injektion dieses Fehlers auftretende Systemverhalten auf Fahrzeugebene visualisiert die Abbildung 8.14. Beim dunklen Fahrzeug handelt es sich um den Versuchsträger mit steer-by-wire-System und injiziertem Fehler, wohingegen das weiße Fahrzeug als Vergleichsobjekt ohne injizierten Fehler dient. Es ist der erste Teil des Fahrmanövers „Doppelter Spurwechsel bei 120 km/h“ dargestellt. Deutlich zu erkennen ist, dass der Fahrer des weißen Fahrzeugs in der Lage ist das Fahrmanöver durchzuführen und dem vorgegebenen Kurs des Spurwechsels folgen kann. Der Fahrer des dunklen Fahrzeug ist jedoch nach der Injektion des identifizierten Fehlers nicht mehr in der Lage der Vorgabe des Fahrmanövers zu folgen. Er kann das Fahrzeug nicht auf dem gewünschten Kurs halten und verlässt die Fahrbahn.

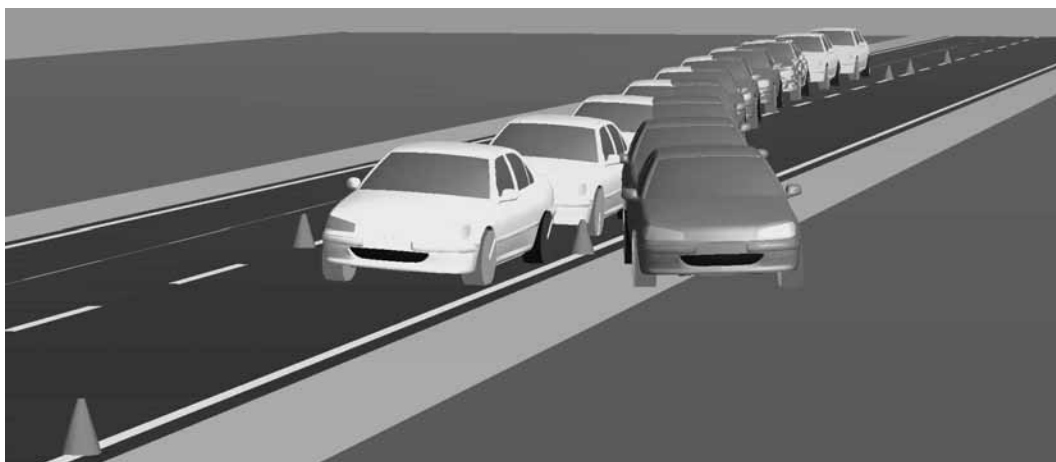


Bild 8.14: Systemverhalten auf Fahrzeugebene bei Injektion des identifizierten Fehlers mit den signifikantesten Auswirkungen

### 8.3.3 What happened - Analyse

Unmittelbar verknüpft mit der Identifikation dieses signifikanten Fehlers und der daraus resultierenden Fahrzeugbewegung, ist die Frage nach dem Grund für dieses Systemverhalten. Insbesondere vor dem Hintergrund, dass - wie bereits erwähnt - das steer-by-wire System über ein redundantes und gegenüber Einzelfehlern tolerantes Systemkonzept verfügt. Es lässt sich folgern, dass der Identifikationsprozess ein Fehlerszenario gefunden haben muss, das in der momentanen Systemrealisierung das Fehlertoleranzkonzept außer Kraft setzt und das Fahrzeug bereits bei der Injektion eines Einzelfehlers in einen unkontrollierbaren Zustand versetzt.

Zur Klärung der Frage nach der Ursache des gezeigten Systemverhaltens kommt die beschriebene „What happened“-Analyse zum Einsatz. Mithilfe der rückwärtsgerichteten Untersuchung des Systemmodells ergibt sich die Ursache-Wirkung-Kette, die in Abbildung 8.15 dargestellt ist.

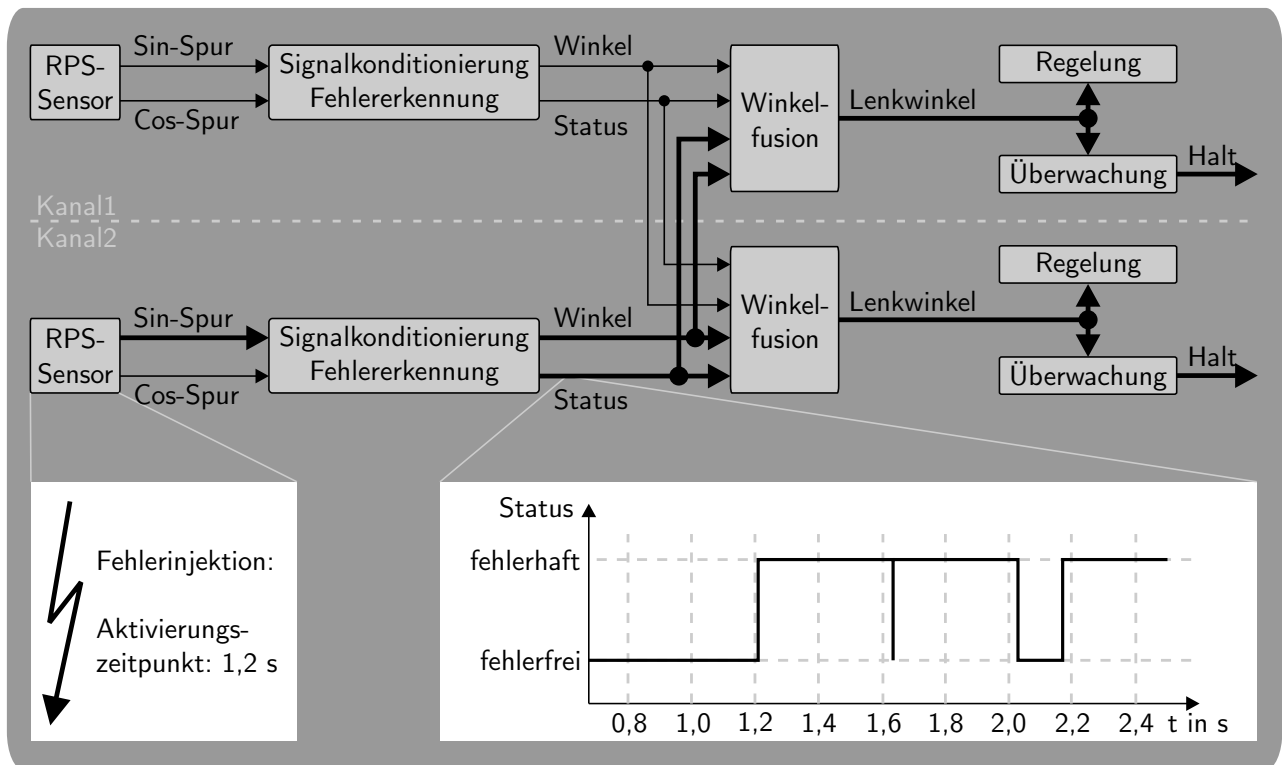


Bild 8.15: Ursache-Wirkung-Kette für den signifikantesten, identifizierten Fehler

Der signifikanteste Fehler wird zum Zeitpunkt 1,2 Sekunden im RPS-Sensor des zweiten Kanals injiziert. Dieser Sensor verfügt zur Ausgabe des aktuellen Rotorlagewinkels über zwei Ausgänge - ein Sinus- und ein Cosinus-Signal. Der injizierte Fehler verfälscht die Signale der Sinus-Spur. Die dem Sensor nachgelagerte Signalkonditionierung und Fehlererkennung berechnet aus den

beiden Sensorsignalen den momentan vorherrschenden Rotorlagewinkel und führt gleichzeitig eine Fehlererkennung durch Plausibilisierung der beiden Signale zueinander durch. Der Winkelwert und eine zugehörige Statusinformation werden zwischen den Kanälen ausgetauscht und einer Winkelfusion zugeführt. Wie aus dem dargestellten Diagramm für den Winkelstatus zu erkennen ist, wird der injizierte Fehler kurz nach seiner Aktivierung bei 1,2 Sekunden erkannt und die Statusinformation zum Winkelwert des RPS2 auf fehlerhaft gesetzt. Damit wird der Winkelfusion, die im vorliegenden Prototypenstadium hauptsächlich aus einer Mittelwertbildung besteht, signalisiert, den als fehlerhaft erkannten Signalwert bei der Berechnung des Lenkwinkels auszuschließen.

Nach etwa 1,6 Sekunden des Fahrmanövers versagt allerdings die Fehlererkennung und gibt den verfälschten Winkelwert kurzzeitig als gültig aus. Dies führt zu einer fehlerhaften Winkelfusion und einem Sprung im resultierenden Lenkwinkel. Eine auf diesem Lenkwinkel basierende Stromüberwachung des Aktuatormotors erkennt aufgrund des Lenkwinkelsprungs eine Unstimmigkeit zwischen den vorherrschenden Motorströmen und der aktuellen Rotorlage. Als Reaktion auf diese Abweichung deaktiviert das kanalspezifische Fehlermanagement den Aktuator. Fatalerweise geschieht diese Abschaltung aufgrund des kommunizierten, fehlerhaften Winkelwertes in beiden Kanälen des steer-by-wire Systems. Als Resultat dieser Vorgänge ergibt sich das zu beobachtende Fahrzeugverhalten.

### 8.3.4 Iterative Anwendung

Aus den Erkenntnissen der „What happened“-Analyse gelingt die systematische Ableitung von Verbesserungspotentialen zur Beseitigung der signifikanten Fehlerszenarien. Im dargestellten Anwendungsfall ergeben sich beispielhaft drei Möglichkeiten zur Erhöhung der Fehlertoleranz des steer-by-wire Systems:

1. Verbesserungen zur Steigerung der Zuverlässigkeit der Sensorfehlererkennung
2. Implementierung einer modellbasierten, fehlertoleranten Winkelfusion, die aufgrund von echter oder analytischer Redundanz einen Fehler in einem Winkelwert selbst erkennt und behandelt (Voting)
3. Überprüfung der Abschaltsschwellen der Motorstromüberwachung hinsichtlich zu sensibler Parametrierung

Unabhängig von der Entscheidung eines Entwicklungsingenieurs für die eine oder andere Verbesserungsvariante ist eine erneute Durchführung einer Identifikation nach erfolgter Realisierung

hilfreich. Zum einen kann sehr schnell analysiert werden, ob die realisierten Verbesserungen die Auswirkungen der signifikanten Fehler tatsächlich in ihrer Schwere reduziert haben. Zum anderen bringen jedoch gerade Veränderungen im System die Gefahr von neuen Fehlermöglichkeiten mit sich. Mit einer iterativen Anwendung des Identifikationsverfahrens werden alle vermeintlichen Verbesserungen einer Analyse unterzogen und hinsichtlich ihrer Fehlermöglichkeiten und -auswirkungen auf das Gesamtsystem untersucht. So kann die Gefahr die ursprünglichen Fehlerszenarien zwar beseitigt, aber neue deutlich schwerwiegendere Fehlermöglichkeiten ins System implementiert zu haben, reduziert werden.

### 8.3.5 Reproduzierbarkeit des Identifikationsergebnisses

Da es sich beim Identifikationsprozess für signifikante Fehlerszenarien um evolutionäre Algorithmen handelt, die auf metaheuristischen Ansätzen basiert, steht der Nachweis der Reproduzierbarkeit der Identifikationsergebnisse noch aus. Zur Klärung dieser Fragestellung wurde der Identifikationsprozess zehn Mal unter den gleichen Randbedingungen wiederholt und sowohl die jeweiligen Verläufe als auch die Ergebnisse der Identifikationen einander gegenübergestellt (vgl. Abbildung 8.16).

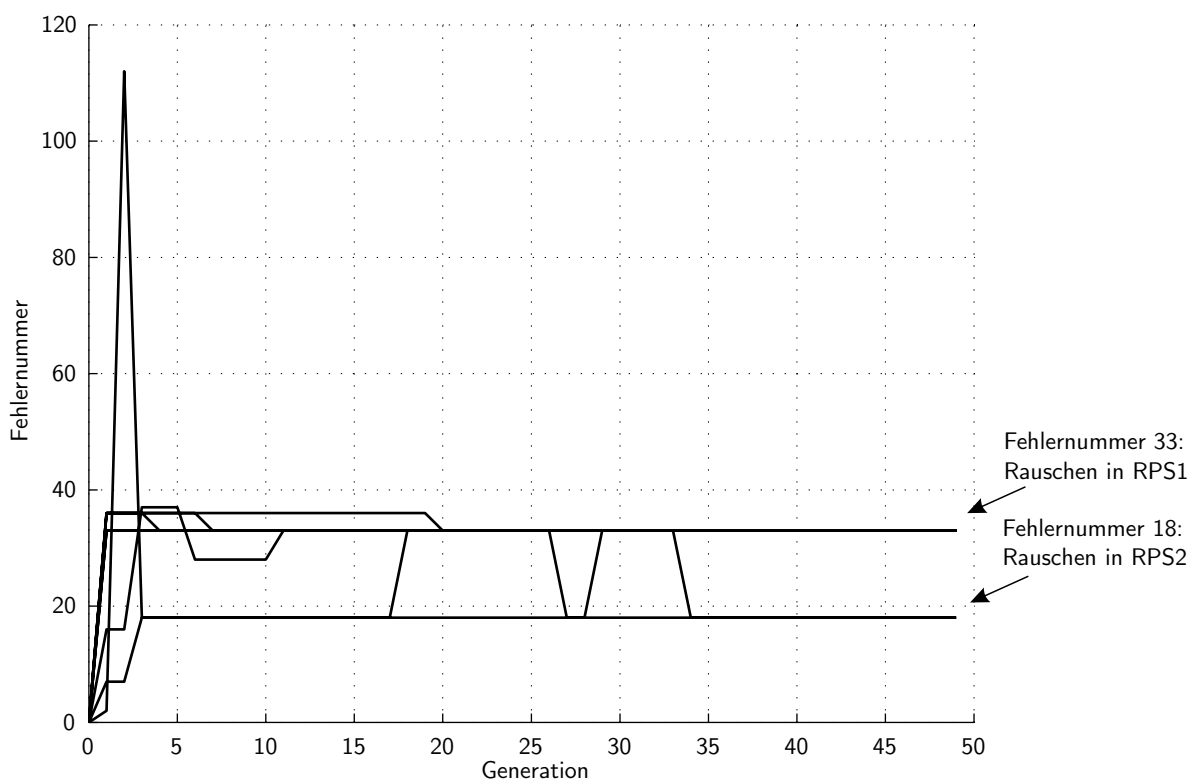


Bild 8.16: Identifikationsverlauf für 10 Wiederholungen

In Abbildung 8.16 ist für die zehn Identifikationen jeweils der signifikanteste Fehler über den jeweiligen Generationen aufgetragen. Es ist deutlich zu erkennen, dass die 10 Identifikationen völlig unterschiedliche Verläufe aufweisen und an ihrem Ende als Ergebnisse zwei von einander abweichende, signifikante Fehler, Fehlernummer 18 und 33, liefern. Bei näherer Betrachtung wird jedoch augenscheinlich, dass es sich bei diesen beiden Identifikationsergebnissen um ein und den selben Fehler handelt. Aufgrund der komplett redundanten Systemarchitektur des steer-by-wire Systems existiert jeder Fehlerort und damit auch jedes Fehlerszenario zweimal - einmal im ersten und einmal im zweiten Kanal. Fehlernummer 18 repräsentiert einen Fehler im Rotorpositionssensor der Lenkradeinheit des ersten Kanals. Fehlernummer 33 steht entsprechend für den selben Fehler im RPS des zweiten Kanals. Die signifikanten Fehlerauswirkungen sind in beiden Fällen identisch.

Die zehn Wiederholungen stellen unter statistischen Gesichtspunkten eine zu kleine Menge für eine belastbare Aussage bzw. den Nachweis der Reproduzierbarkeit dar, sie geben jedoch einen Eindruck von der Robustheit und Ergebnisgüte der beschriebenen Vorgehensweise zur Identifikation signifikanter Fehlerszenarien.

### 8.3.6 Einfluss der Parameter des evolutionären Algorithmus auf das Identifikationsergebnis

Neben der Frage der Reproduzierbarkeit des Identifikationsergebnisses stellt sich auch die Frage des Einflusses der Parameterwahl des Identifikationsverfahrens auf das zu erzielende Identifikationsergebnis. Die Beantwortung dieser Fragestellung und die Ableitung einiger Handlungsvorschläge für die Auswahl geeigneter Identifikationsparameter erfolgt anhand der Ergebnisse zweier Identifikationen mit unterschiedlichen Parametersätzen. Zur Wahrung der Vergleichbarkeit sind die Parametersätze so ausgewählt worden, dass jeweils eine annähernd gleiche Anzahl an Modellauswertungen für die Identifikation erforderlich ist. Die Ergebnisse der Identifikationen mit unterschiedlichen Parametersätzen sind in Abbildung 8.17 zusammengetragen.

Das linke Diagramm in Abbildung 8.17 bezieht sich auf einen Parametersatz 1, der gekennzeichnet ist, durch eine große Populationsgröße, viele Generationen und eine geringe Anzahl evolutionärer Operatoren. Das Diagramm zeigt die Verläufe und Ergebnisse von insgesamt 10 Identifikationen, die mit diesem Parametersatz und unter gleichen Randbedingungen nacheinander durchgeführt wurden. Das rechte Diagramm in Abbildung 8.17 stellt ebenfalls die Verläufe und Ergebnisse von 10 Identifikationen dar. Bei diesen Identifikationen kam jedoch ein anderer Parametersatz 2 zum Einsatz, der sich vor allem durch eine kleinere Populationsgröße, weniger Generationen und wesentlich mehr evolutionäre Operatoren vom erst genannten unterscheidet.

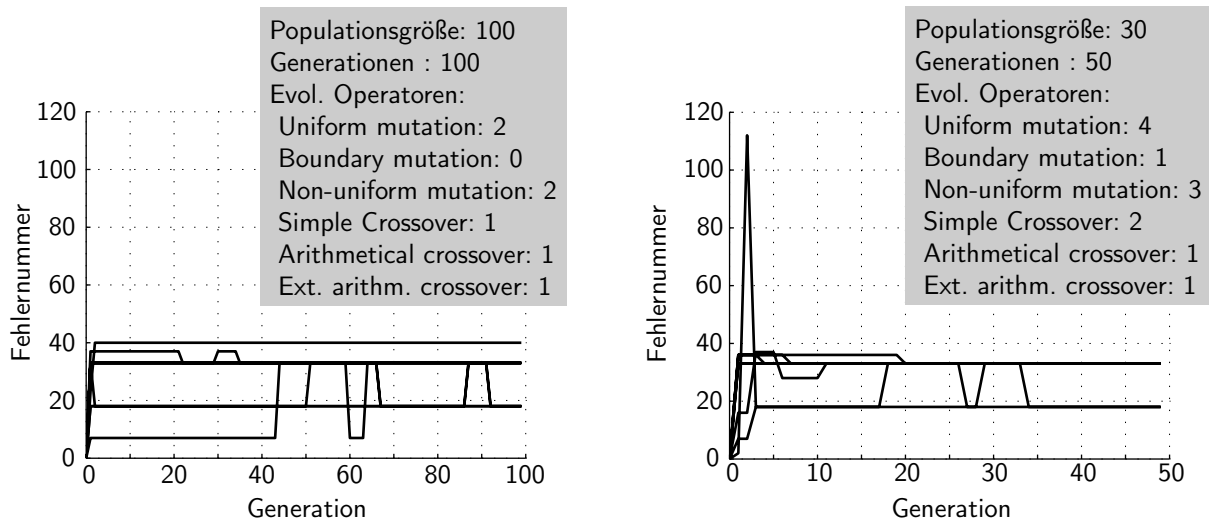


Bild 8.17: Identifikationsergebnisse mit unterschiedlichen Parametersätzen

Anhand der beiden Diagramme lässt sich beispielhaft das unterschiedliche Verhalten des Identifikationsverfahrens, hervorgerufen durch die unterschiedlichen Parametersätze, verdeutlichen. Im Falle des Parametersatzes 1 wird am Anfang des Identifikationsverfahrens zunächst eine große Population initialisiert. Die Wahrscheinlichkeit bereits in dieser Population ein signifikantes Fehlerszenario gefunden zu haben, ist im Gegensatz zum Verfahren mit Parametersatz 2 deutlich höher. Die Anfangspopulation kann sich bei Parametersatz 1, aufgrund der geringen Anzahl evolutionärer Operatoren, nur erheblich träger verändern. Dieses Verhalten zeigt sich auch in den Verläufen der jeweiligen Identifikationen. Das rechte Diagramm weist zu Beginn der Identifikationen eine größere Dynamik auf als es im linken Diagramm der Fall ist. Mit fehlender Dynamik wächst jedoch die Gefahr, bei der Identifikation mit Parametersatz 2 nicht das tatsächlich signifikanteste Fehlerszenario zu identifizieren, da zu wenig Bereiche des mehrdimensionalen, komplexen Fehlerraums untersucht werden. Dieser Fall ist bei den 10 Identifikationen mit Parametersatz 1 einmal aufgetreten. Anstatt der beiden, bereits erwähnten, signifikantesten Fehler mit den Nummern 18 und 33 liefert der Identifikationsalgorithmus einmal die Fehlernummer 40 als Ergebnis der Identifikation.

Aus dem Beispiel ist ersichtlich, dass bei der Wahl der Parameter insbesondere die Dynamik der Identifikation zu berücksichtigen ist. Eine große Anzahl an evolutionären Operatoren stellt sicher, dass das Identifikationsverfahren eine hohe Dynamik aufweist und damit den komplexen, mehrdimensionalen Fehlerraum zielgerichtet untersuchen kann. Dies wird insbesondere dann wichtig, wenn der Fehlerraum durch die Injektion von Mehrfachfehlern oder Fehlersequenzen deutlich an Komplexität gewinnt.

Andererseits darf die Reduktion der Populationsgröße zu Gunsten der Anzahl der evolutionären



Operatoren nicht zu groß ausfallen. In Anbetracht der Tatsache, dass die Population während des Identifikationsablaufs den Speicherplatz für gute Individuen darstellt, schränkt eine geringe Größe die Möglichkeiten zur Suche im Fehlerraum stark ein und erhöht die Gefahr, in einem lokalen Optimum zu enden.

# 9 Zusammenfassung und Ausblick

## 9.1 Zusammenfassung

In zukünftigen Generationen von Kraftfahrzeugen werden in zunehmendem Maße mechatronische Systeme eingesetzt, die auch sicherheitsrelevante Funktionen erbringen. Insbesondere die Entwicklung so genannter x-by-wire Systeme bringt in diesem Zusammenhang neue Herausforderungen mit sich.

In der vorliegenden Arbeit wurden zunächst die neuen Anforderungen an eine modellbasierte System- und Sicherheitsanalyse für zukünftige, sicherheitsrelevante Fahrzeugsysteme ermittelt. Sie bildeten die Basis für die sich anschließende vergleichende Bewertung einer Auswahl klassischer Analyseverfahren.

Es konnte gezeigt werden, dass keines der klassischen Verfahren allein alle an eine modellbasierte System- und Sicherheitsanalyse für sicherheitsrelevante Kraftfahrzeugsysteme gestellte Anforderungen erfüllen kann. Meist war die Modellbildung, die dem jeweiligen klassischen Analyseverfahren zu Grunde liegt, Hauptursache für Einschränkungen. Das Modellierungskonzept und der Umfang der Beschreibungssprache war nicht vollständig mit den Anforderungen in Einklang zu bringen. Aus diesem Grund erfolgte eine zusätzliche Bewertung unterschiedlichster Modellierungskonzepte.

Aus den Ergebnissen der verschiedenen Bewertungen ist in der vorliegenden Arbeit ein Verfahren entstanden, das das klassische Analyseverfahren FMEA mit neuen Konzepten der hybriden Modellierung kombiniert und erweitert. Damit gelingt die Ausweitung der Methodik zur modellbasierten, dynamischen Systemanalyse auf eine ganzheitliche Betrachtung von Kraftfahrzeugsystemen, bei der das kausale Zusammenspiel der Systembestandteile Fahrzeugsystem, Fahrzeug, Fahrzeugumgebung und Fahrer rechnergestützt analysiert wird.

Das hierzu angewandte Vorgehen beruht zunächst auf einer funktionalen, hybriden Modellierung des nominalen Verhaltens der Systemkomponenten. Durch Modellaggregation gemäß einer bekannten Systemarchitektur gelingt die Integration zu einem Fahrzeugsystem und die Einbettung in die Fahrzeugsystemumgebung aus Fahrzeug, Fahrer und Straße. Damit finden neben den Modellen zum nominalen Komponentenverhalten auch strukturelle Systemaspekte Berücksichtigung im Gesamtsystemmodell.

Die über das Nominalverhalten hinausgehende Modellierung des Fehlerverhaltens von Systemkomponenten gliedert sich in einzelne weitere Vorgehensschritte. Das auf die Systemkomponente beschränkte Fehlverhalten wird im ersten Prozessschritt mit Hilfe der FMEA ermittelt. Im Folgenden ist das entsprechende Fehlverhalten an den Komponentenschnittstellen mit Hilfe der hybriden Modellierung zu beschreiben und in die funktionalen Modelle zu integrieren.

Die Idee des ganzheitlichen Ansatzes beruht auf der Annahme, dass eine sicherheitsrelevante Gefahr einer Komponente eindeutig zuzuordnen ist. Die Ursachen, die zur Auslösung einer Gefahr führen, können jedoch im gesamten technischen System - d.h. im Kraftfahrzeugsystem selbst, aber auch im Fahrereingriff oder der vorherrschenden Umgebungssituation - begründet sein. Bestehen Sicherheitsanforderungen für das zu entwickelnde Kraftfahrzeugsystem, so können diese mit den Aussagen des ganzheitlichen, hybriden Modells verglichen werden und dadurch die Einhaltung der Sicherheitsanforderungen manuell überprüft werden.

Durch den Einsatz evolutionärer Algorithmen und einer objektiv-quantifizierbaren Bewertungsfunktion ist das in dieser Arbeit entwickelte Verfahren, über die manuelle Überprüfung von Sicherheitszielen hinaus, in der Lage, selbstständig signifikante Szenarien aus Einzelfehlern, Mehrfachfehlern oder Fehlersequenzen zu identifizieren, die die Sicherheitsanforderungen verletzen.

Der Grundgedanke basiert dabei auf dem Vorbild der biologischen Evolution. Es werden automatisiert Fehlerszenarien, bestehend aus einem oder mehrerer Fehlermöglichkeiten, einem oder mehrerer Fehlerauftretenszeitpunkten und einer Definition der Systemumgebung, generiert. Sie werden mit Hilfe von Fehlerinjektion in das hybride Gesamtsystemmodell eingebracht und anhand einer objektiv-quantifizierbaren Bewertungsfunktion in ihren Auswirkungen auf das Kraftfahrzeugsystem selbst, aber auch auf den Fahrer, das Fahrzeug und die Systemumgebung klassifiziert. Aus den Fehlerszenarien mit signifikanten Auswirkungen werden mit Hilfe evolutionärer Operatoren neue Fehlerszenarien erzeugt. Eine iterative Anwendung dieses Vorgehens identifiziert kontinuierlich Schwachstellen im System, die zum Verletzen der Sicherheitsanforderungen führen und ermöglicht darüber hinaus deren schnelle und zielgerichtete Beseitigung.

## 9.2 Bewertung und Erkenntnisse

Durch die Anwendung der entwickelten Vorgehensweise am Anwendungsbeispiel eines in der Entwicklung befindlichen steer-by-wire Systems konnte das Leistungspotential des Konzepts aufgezeigt, die Stärken und Schwächen ermittelt und das Aufwand-Nutzen-Verhältnis abgeschätzt werden.

Die Stärken des Konzepts für die modellbasierte, ganzheitliche System- und Sicherheitsanalyse liegt in der Kombination des klassischen Analyseverfahrens FMEA, der hybriden Gesamtsystemmodellierung unter funktionalen, strukturellen und sicherheitsgerichteten Aspekten sowie der rechnergestützten Identifikation und Bewertung von Fehlerszenarien mit Hilfe der evolutionären Algorithmen und einer objektiv-quantifizierbaren Bewertungsfunktion:

- Mit Hilfe der ganzheitlich modellbasierten System- und Sicherheitsanalyse ist es möglich, das komplexe Zusammenspiel der Fahrzeugsystemkomponenten und darüber hinaus die Interaktion des Fahrzeugsystems mit dessen Systemumgebung umfassend zu analysieren und zu bewerten. Dadurch lassen sich insbesondere im Fehlerfall eine Ursache-Folge-Analyse auf Systemebene realisieren und wichtige Erkenntnisse für die Ableitung von Änderungsmaßnahmen gewinnen.
- Die System- und Sicherheitsanalyse beschränkt sich aufgrund der zugrunde gelegten hybriden Modellierung nicht mehr auf eine statische Betrachtung, wie es bei vielen klassischen Analyseverfahren der Fall ist, sondern kann das dynamische Systemverhalten nachbilden. Der Einfluss des Zeitpunktes eines Fehlereintritts, die Zeit zur Fehlererkennung, das zeitliche Verhalten bei der Fehlerausbreitung im System und die Dynamik der fehlerhaften Reaktion des Systems bleibt nicht länger unberücksichtigt.
- Darüber hinaus erfolgt die Bewertung im Gegensatz zu vielen klassischen Sicherheitsanalyseverfahren nicht subjektiv durch ein Bewertungsteam aus Systemexperten, sondern objektiv-quantifizierbar durch die Definition von mathematischen Bewertungsfunktionen. Dies ermöglicht zum einen die rechnergestützte Systembewertung, garantiert darüber hinaus die Reproduzierbarkeit der Analyseergebnisse.
- Die iterative Anwendbarkeit des verfolgten Ansatzes erlaubt eine kontinuierliche Überprüfung der Sicherheitsanforderungen und stetige Verbesserung des Fahrzeugsystems hinsichtlich Funktion und Sicherheit. Insbesondere die vielfach durchgeführten Änderungen an Systembestandteilen zur vermeintlichen Fehlerbeseitigung oder Funktionsverbesserung können fortlaufend auf ihre tatsächliche Reaktion im komplexen Gesamtsystem untersucht werden.
- Um den dazu notwendigen Zeitaufwand gering zu halten, ist die entwickelte Methodik voll automatisierbar und identifiziert selbstständig signifikante Fehlerszenarien, die die gestellten Sicherheitsanforderungen verletzen.

Trotz der Möglichkeit zur Automatisierung ist die entwickelte Vorgehensweise mit einem erhöhten Aufwand verbunden. Die zeitaufwändigen Schritte sind zum einen die Modellerstellung und

zum anderen dessen Auswertung. Um die Problematik der langen Rechenzeit der Modellauswertung zumindest abschwächen zu können, konnte im Rahmen dieser Arbeit ein Ansatz zur parallelen Modellauswertung vorgestellt werden. Die Identifikation von signifikanten Fehlerszenarien mit Hilfe von evolutionären Algorithmen eignet sich für die gleichzeitige Berechnung auf verteilten Rechnerknoten. Damit kann der zur Modellauswertung notwendige Zeitbedarf deutlich reduziert werden. Der erhöhte Aufwand für die Modellerstellung wird heute meist schon zu einem Großteil betrieben, denn die für die modellbasierte ganzheitliche System- und Sicherheitsanalyse eingesetzten Modelle unterscheiden sich nicht von den Modellen aus dem Rapid-Prototyping. Allerdings sind die rein funktional orientierten Modelle des Rapid-Prototyping um die erwähnten strukturellen und sicherheitsgerichteten Aspekte in der Modellierung zu erweitern. Dies bringt einen Mehraufwand mit sich.

Darüber hinaus darf das vorgestellte Verfahren nicht als alleiniger Sicherheitsnachweis missverstanden werden. Es kann nicht den Beweis liefern, dass alle Sicherheitsanforderungen eingehalten werden. Dazu ist die Menge an möglichen Kombinationen von Fehlern, Eintrittszeitpunkten und Umgebungsbedingungen zu groß. Aus heutiger Sicht ist noch kein Verfahren bekannt, das heute oder in naher Zukunft diesem Anspruch an Vollständigkeit bei wachsender Systemkomplexität gerecht werden kann. Ein Sicherheitsnachweis für ein sicherheitsrelevantes Kraftfahrzeugsystem ist auch weiterhin durch die Kombination von verschiedensten Sicherheitsanalysen zu erbringen. Die vorgestellte Methodik bietet sich jedoch als ein Bestandteil des Sicherheitsnachweises an, um die Identifizierung und Fokussierung auf Fehlerszenarien mit signifikanten Fehlerauswirkungen aus der Vielzahl der Kombinationen zu ermöglichen.

Der Nutzen der modellbasierten, ganzheitlichen System- und Sicherheitsanalyse konnte, gerade hinsichtlich der Identifizierung und Fokussierung, anschaulich am Anwendungsbeispiel *steer-by-wire* gezeigt werden. Das Identifikationsverfahren ist in der Lage, Fehler zu finden, die im komplexen Zusammenspiel unterschiedlichster Funktionen im System begründet sind. Der kausale Zusammenhang zwischen Winkelfehlerinjektion, misslungener Fehlererkennung, falscher Winkelfusion und Abschaltung aufgrund der Motorstromüberwachung ist nicht offensichtlich.

### 9.3 Ausblick

Es liegt nahe, die modellbasierte ganzheitliche Analyse mit quantitativen Wahrscheinlichkeiten zum Fehlereintritt zu verknüpfen. Leider sind zuverlässige Ausfallraten in der Automobilindustrie bisher schwer zu ermitteln. Sollte jedoch im Zuge der Einführung einer branchenweit gültigen Anpassung der Norm IEC61508[DE02] eine systematische Erfassung und Ermittlung

von Felddaten notwendig werden, könnten diese Daten bei der Identifikation von signifikanten Fehlermöglichkeiten berücksichtigt werden.

Darüber hinaus wird sich in Zukunft, ebenfalls mit der Einführung der automobil-spezifischen Ableitung der Norm IEC61508[DE02], sehr viel häufiger die Frage stellen, welche Fehlerauswirkungen für den Fahrer spürbar, störend oder gar gefährlich sind. Dabei kann das in dieser Arbeit verwendete Fahrermodell als Teil der Bewertungsfunktion nicht den Ansprüchen genügen. Auf diesem Gebiet sind systematische Untersuchungen erforderlich, die in einem ersten Schritt die Grundfunktionen eines Fahrzeugs und insbesondere deren Fehlfunktionen bewertet. Im zweiten Schritt sind anschließend entweder physikalische Messgrößen und deren Schwellwerte zu bestimmen, die der Durchschnittsfahrer bei einer Fehlfunktion als gefährlich empfindet, oder es sind Fahrermodelle von Nöten die das menschliche Verhalten insbesondere in Situationen mit fehlerhaften Fahrzeugreaktionen nachbilden können.



# A Ein redundantes, synchronisiertes TTCAN-Kommunikationsnetz

Mit dem verbreiteten Einsatz elektronischer Steuergeräte, intelligenter Sensoren und Aktoren in modernen Kraftfahrzeugen, haben auch fahrzeuginterne Netzwerke zum fahrzeugweiten Datenaustausch Einzug erhalten. Ein Kommunikationsprotokoll, das sich im Kraftfahrzeug und in der Automatisierungstechnik etabliert hat, ist das Controller Area Network (CAN). Es handelt sich dabei um ein ereignisgesteuertes Kommunikationssystem, das in ISO 11898 standardisiert ist und auf Grund seiner Robustheit und Flexibilität in vielen Fahrzeugklassen und -generationen eingesetzt wird.

Zukünftige - insbesondere sicherheitsrelevante - Fahrzeugsysteme stellen neue Anforderungen an die Kommunikationssysteme. Vor allem hinsichtlich der Beherrschbarkeit der enorm gestiegenen Komplexität und dem Wunsch eines deterministischen Systemverhaltens scheint ein Übergang von ereignis- zu zeitgesteuerten Systemen vorteilhaft. Aus diesem Grund wurden sowohl Neuentwicklungen, wie TTP [TTT05] oder FlexRay [Fle05], als auch eine Erweiterung des bestehenden CAN-Protokolls um eine zeitgesteuerte Variante, dem sogenannten Time-Triggered Controller Area Network (TTCAN) [ISO04], durchgeführt.

## A.1 Einführung in das TTCAN-Kommunikationsprotokoll

Die Grundvoraussetzung für ein zeitgesteuertes Kommunikationssystem, in dem allein das Fortschreiten der Zeit das Kommunikationsverhalten bestimmt, ist eine systemweite synchronisierte Zeitbasis. Nur wenn alle Kommunikationsteilnehmer das gleiche Zeitverständnis aufweisen, ist ein sinnvoller Datenaustausch möglich. Zu diesem Zweck wird im TTCAN-Protokoll die Synchronisation mittels sogenannter Referenz-Nachrichten eingeführt. Dabei handelt es sich um mit einem bestimmten Identifier gekennzeichnete Nachrichten, die den Beginn einer neuen Kommunikationsrunde anzeigen und eine Synchronisation des lokalen Zeitverständnisses jedes einzelnen Kommunikationsteilnehmers ermöglicht. Die Referenz-Nachricht wird von einem sogenannten Zeitmaster gesendet. Im TTCAN-Netz besteht die Möglichkeit, mehrere potentielle Zeitmaster zu installieren, so dass bei einem Ausfall des aktuellen Zeitmasters diese Funktion von einem potentiellen Zeitmaster übernommen werden kann.



Der Zeitraum zwischen zwei Referenznachrichten wird Basiszyklus genannt. Er enthält einen Fahrplan, also eine a priori festgelegte Reihe von Zeitfenstern, in denen der Datenaustausch erfolgen kann. Sobald die synchronisierte lokale Zeit eines Kommunikationsteilnehmers eine ihm zugeordnete Zeitmarke erreicht, kann er seine Nachricht auf dem Kommunikationsmedium platzieren, bzw. eine für ihn bestimmte Nachricht empfangen.

Für die Nachrichtenübertragung stehen prinzipiell zwei verschiedene Zeitfenstertypen zur Verfügung. Zum einen existiert das Exklusiv-Zeitfenster, das ausschließlich einer speziellen Nachricht zugeordnet ist, zum anderen das Arbitrating-Zeitfenster, in dem mehrere Kommunikationsteilnehmer um die Berechtigung zum Senden ihrer Nachricht konkurrieren.

Die Gesamtheit mehrerer Zeitfenster, eingebettet in mehrere Basiszyklen, ergibt den Fahrplan für die zeitgesteuerte Kommunikation, die sogenannte System Matrix. Sie ist beispielhaft in Abbildung A.1 dargestellt.

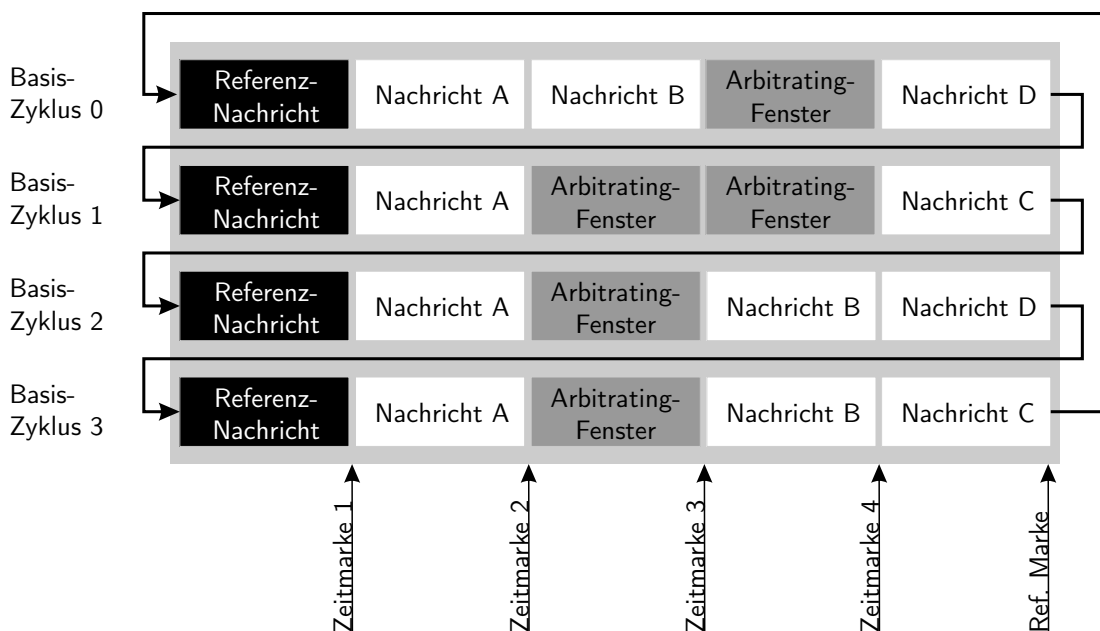


Bild A.1: Beispielhafte Systemmatrix eines TTCAN-Kommunikationssystems

Die weiterführenden Mechanismen des TTCAN-Protokolls sollen an dieser Stelle nicht näher detailliert werden. Stattdessen wird hierzu auf den ISO Standard 11898-4 [ISO04] und die Literatur [LD02, FDH<sup>+</sup>01] verwiesen.

## A.2 Synchronisationsalgorithmus

Da der neue TTCAN-Standard ISO 11898-4 eine Erweiterung des bisherigen CAN-Standard ISO 11898 darstellt, ist eine Nachrichtenübertragung über redundante Kommunikationsmedien bisher nicht berücksichtigt. Sicherheitsrelevante Systeme erfordern jedoch ein fehlertolerantes Kommunikationssystem, das eben diese Redundanz ermöglicht. Zur Beseitigung dieses Defizit ist der Einsatz mehrerer, zunächst unabhängiger TTCAN-Kommunikationskanäle denkbar. Um jedoch als Ganzes von einem integrierten, zeitgesteuerten Kommunikationssystem sprechen zu können, müssen die Nachrichtenübertragungen auf den jeweiligen Kommunikationskanälen sowohl im Werte- als auch im Zeitbereich untereinander in Zusammenhang stehen. Dies kann durch einen systemweit konsistenten TTCAN-Fahrplan und eine Synchronisation zwischen den redundanten TTCAN-Kommunikationskanälen erreicht werden.

Zur Synchronisation von mindestens zwei TTCAN-Kommunikationskanälen wird eine zusätzliche Kommunikationsschicht - der sogenannte Sync-Layer - entwickelt. Er basiert auf dem im TTCAN-Protokoll vorgesehenen GAP-Mechanismus. Dazu wird nach jedem Basiszyklus eine Lücke im TTCAN-Fahrplan eingefügt. Die zeitliche Länge dieser Lücke kann innerhalb vordefinierter Grenzen vom TTCAN-Controller beeinflusst werden. Abbildung A.2 zeigt schematisch den Aufbau eines Basiszyklus bei aktiviertem GAP-Mechanismus.

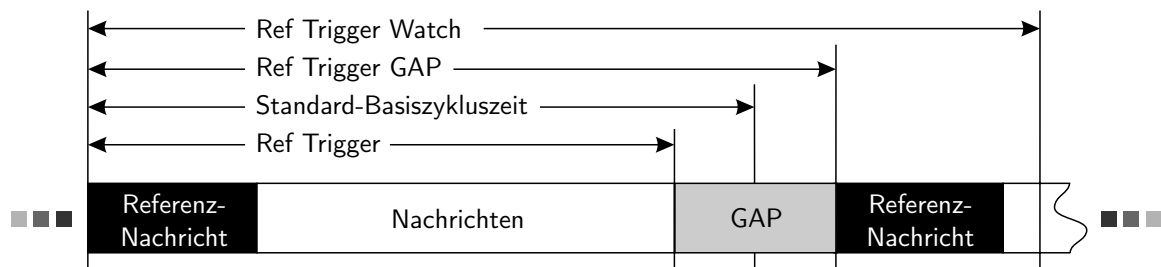


Bild A.2: Variable Basiszykluszeit durch GAP-Mechanismus

Die Lücke wird immer an der vorher festgelegten Zeitmarke *Ref Trigger* in den TTCAN-Fahrplan eingefügt. Ihre Länge ist bis zur Zeitmarke *Ref Trigger GAP* limitiert. In der Regel wird das GAP durch den Zeitmaster durch Aussendung einer Referenz-Nachricht mittig abgebrochen, so dass die gewünschte Standard-Basiszykluszeit entsteht. Sollte auf Grund eines Fehlers der aktuelle Zeitmaster nicht in der Lage sein, die Lücke abubrechen, so verstreicht die Zeit der maximalen GAP-Länge. Anschließend erkennen alle anderen potentiellen Zeitmaster das Ausbleiben der Referenz-Nachricht und konkurrieren gemäß ihren Prioritäten um die Erlaubnis, ihre Referenz-Nachricht auf dem Kommunikationsmedium zu platzieren. Damit entsteht im Fehlerfall eine maximale Abweichung von der Standard-Zykluszeit von einer halb-

en GAP-Länge. Sollte bis zur Zeitmarke *Ref Trigger Watch* ein potentieller Zeitmaster nicht in der Lage gewesen sein, eine Referenz-Nachricht auszusenden oder hat ein Zeitslave bis zu diesem Zeitpunkt keine neue Referenz-Nachricht empfangen, so stellt der jeweilige TTCAN-Kommunikationscontroller die Teilnahme an der Kommunikation ein.

Um den GAP-Mechanismus des TTCAN-Protokolls zu Synchronisationszwecken von zwei oder mehr Kommunikationskanälen zu verwenden, wird zu einem bestimmten Zeitpunkt während des Basiszyklus eine Differenzmessung zwischen den lokalen Zeiten der TTCAN-Kommunikationsbausteine durchgeführt. Überschreitet diese Differenz einen vorher festgelegten Toleranzbereich, so wird im vorausgehenden TTCAN-Kanal das GAP verlängert, bzw. im nachfolgenden verkürzt. Dieses Synchronisationsprinzip ist in Abbildung A.3 nochmals anschaulich dargestellt.

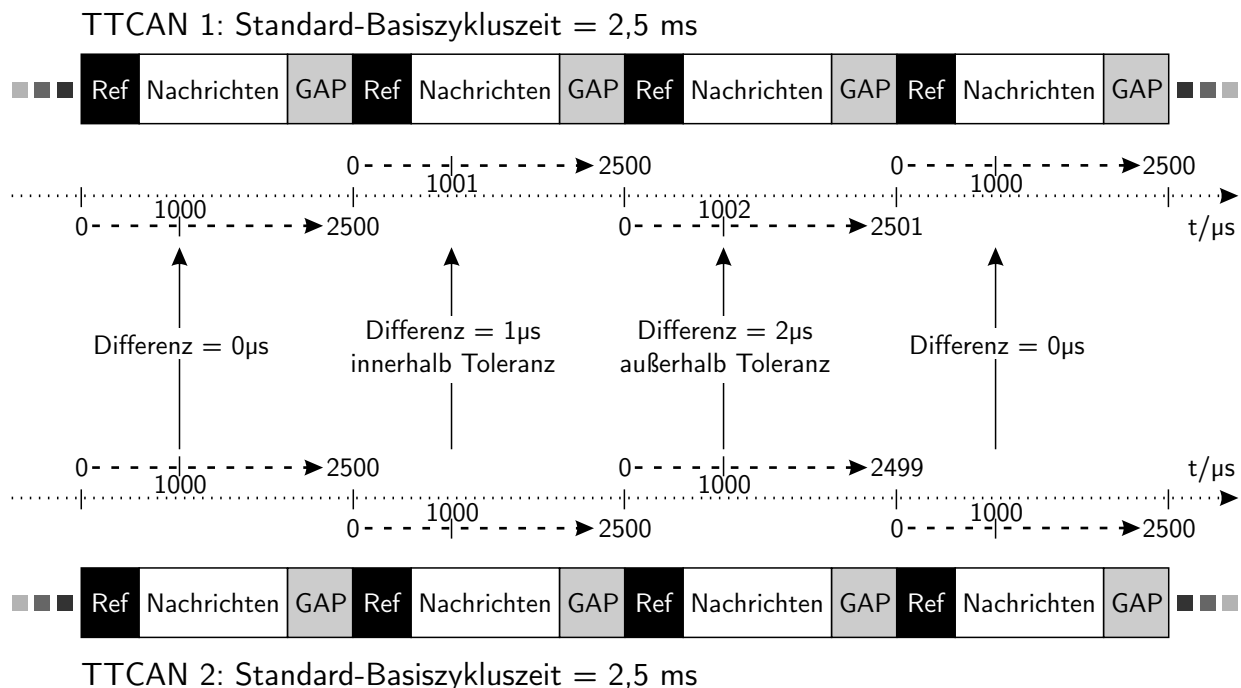


Bild A.3: Synchronisation von zwei TTCAN-Kommunikationskanälen

Während des ersten dargestellten Basiszyklus sind die Kommunikationskanäle TTCAN1 und TTCAN2 noch synchron. Eine Zeitmessung an der Zeitmarke  $1000\mu\text{s}$  im ersten Basiszyklus ergibt eine Differenz zwischen den Kanälen von  $0\mu\text{s}$ . Auf Grund von Fertigungstoleranzen und unterschiedlichen Signallaufzeiten kommt es während der Übertragung des zweiten Basiszyklus zu zeitlichen Abweichungen. Der TTCAN1-Kanal eilt dem TTCAN2-Kanal voraus. Die initiierte Zeitmessung an der Zeitmarke  $1000\mu\text{s}$  im zweiten Basiszyklus ermittelt eine zeitliche Differenz von  $1\mu\text{s}$ . Im dargestellten Beispiel wird der Synchronisationsalgorithmus noch nicht aktiv, da sich die Abweichung innerhalb des tolerierbaren Wertebereichs befindet. Während des

dritten Basiszyklus ist die Differenz der Zeitverständnisse von TTCAN1 und TTCAN2 auf  $2\mu\text{s}$  angestiegen und hat damit die Toleranzgrenze überschritten. Der Synchronisationsalgorithmus veranlasst in diesem Fall den Ausgleich der entstandenen Differenz durch eine Verlängerung des GAP im TTCAN1-Kanal und eine Verkürzung der Lücke im TTCAN2-Kanal. Damit kommt im vierten Basiszyklus erneut eine exakt synchrone Kommunikation zu Stande.

Die Länge der eingefügten Lücke ist ausschlaggebend für die Wiederherstellungsgeschwindigkeit synchroner Kommunikation. Je größer das GAP, desto schneller können auch größere zeitliche Differenzen zwischen den Nachrichtenkanälen abgebaut werden. Im Hinblick auf das übertragbare Datenvolumen sollte das GAP jedoch möglichst klein sein, da in der Lücke keine Nachrichtenübertragen stattfindet. Außerdem kann ein Ausfall eines Zeitmasters durch eine kurze Lücke schneller detektiert und damit die zeitliche Abweichung von der Standard-Basiszykluszeit kleiner gehalten werden. Wie groß die Lücke gewählt werden sollte, muss deshalb für jede Anwendung individuell geprüft werden, um einen optimalen Kompromiss zwischen Wiederherstellungsgeschwindigkeit von synchronen Nachrichtenkanälen und Übertragungsvolumen zu erreichen.

Ein Prototyp eines redundanten TTCAN-Kommunikationsnetzes mit integriertem Synchronisationsalgorithmus folgender Struktur konnte zu Funktions- und Fehleranalysen aufgebaut werden:

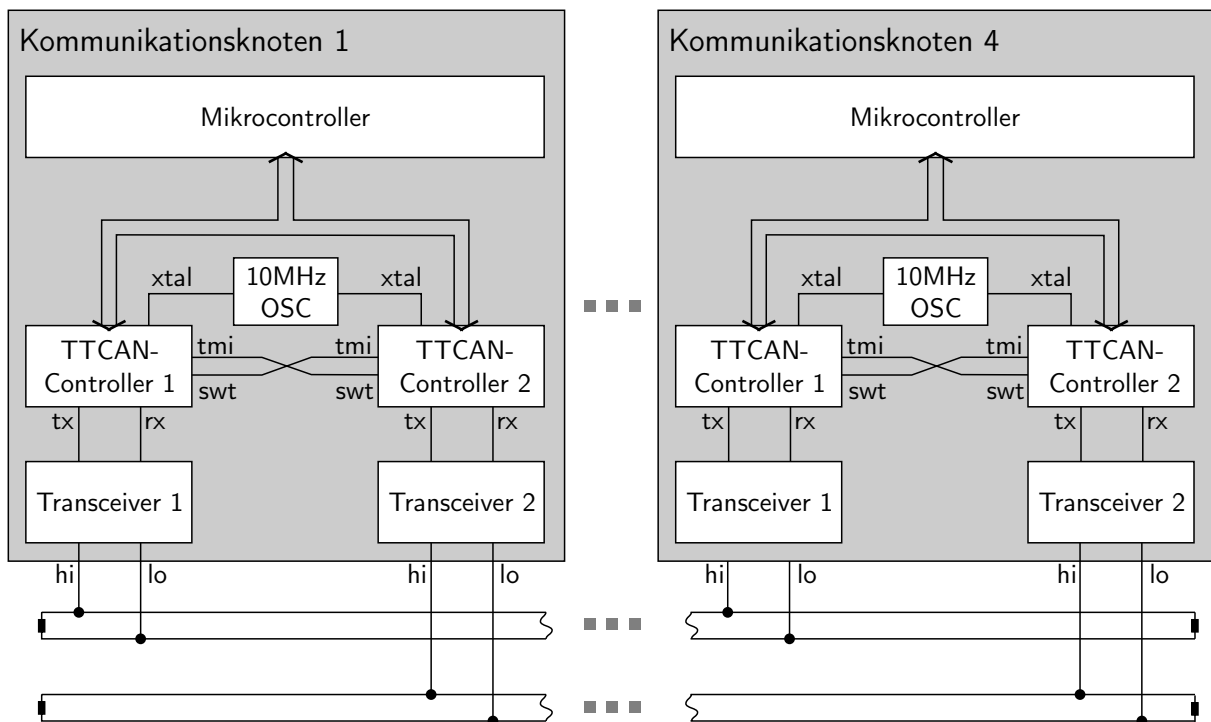


Bild A.4: Aufbau des prototypischen TTCAN-Netzwerk

Das Netz besteht aus vier Kommunikationsknoten, die jeweils über zwei TTCAN-Controller verfügen. Alle Kommunikationsteilnehmer sind somit an die redundanten Kommunikationsmedien angeschlossen. Der hier vorgestellte Synchronisationsalgorithmus ist in einem Mikrocontroller implementiert. Zur Durchführung der zeitlichen Differenzmessung müssen die Anschlüsse *Time Mark Interrupt (TMI)* und *Stop Watch Trigger (SWT)* der TTCAN-Bausteine wechselseitig über Kreuz miteinander verbunden sein. Der implementierte TTCAN-Fahrplan weist eine Standard-Basiszykluszeit von  $2.5ms$  auf. Die maximale GAP-Länge ist auf  $250\mu s$  limitiert, wird jedoch in der Regel nach  $125\mu s$  abgebrochen. Die zulässige Abweichung der lokalen Zeitverständnisse zwischen TTCAN1 und TTCAN2 wird auf  $\pm 1\mu s$  beschränkt. Die Einhaltung der Synchronität des Prototyp-Kommunikationsnetzes zeigt Abbildung A.5.

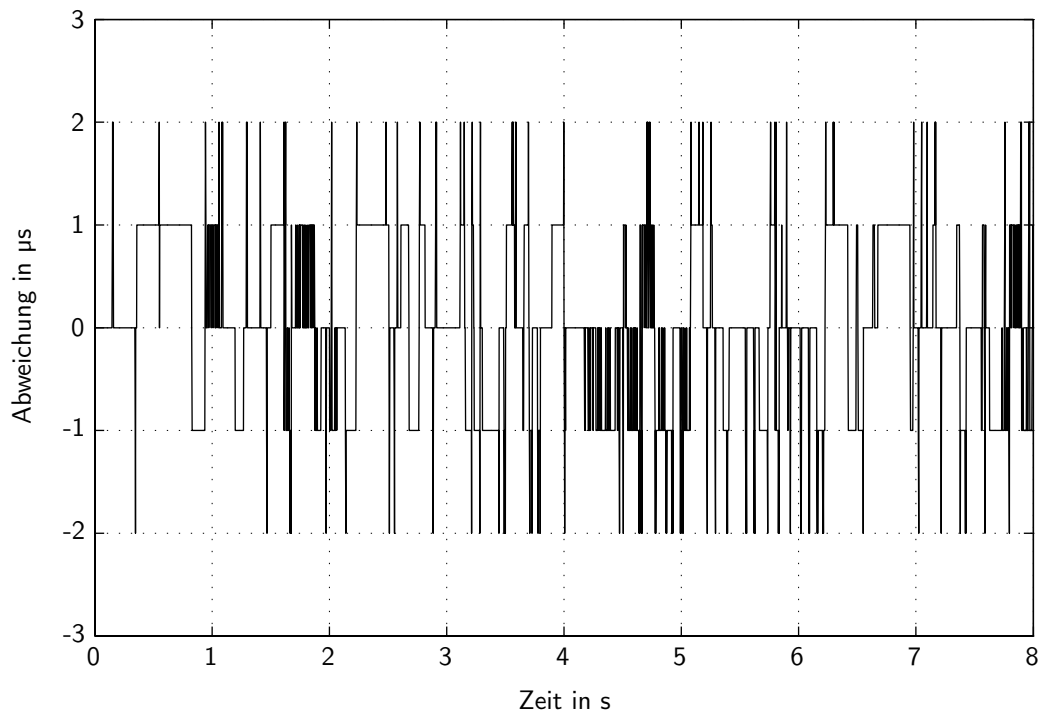


Bild A.5: Zeitliche Abweichung zwischen TTCAN1 und TTCAN2 bei einer Toleranz von  $\pm 1\mu s$

### A.3 Analyse des Fehlerverhaltens

Ein zweiter wichtiger Aspekt des Prototyp-Aufbaus ist neben der Funktionsentwicklung und -untersuchung vor allem die Analyse des Fehlerverhaltens des redundanten, synchronisierten TTCAN-Netzes. Dabei stehen nicht die Fehler der einzelnen TTCAN-Kanäle und deren Behandlungsmöglichkeiten im TTCAN-Protokoll im Fokus der Betrachtung, sondern vielmehr die

Fehlermöglichkeiten und deren Auswirkungen, die durch die Integration zu einem redundanten Kommunikationsnetz zusätzlich entstehen. Analysiert man diese Fehlermöglichkeiten, so lassen sich zwei Fehlerklassen identifizieren. Die erste Fehlerklasse beinhaltet alle Fehlermöglichkeiten, die das zeitliche Verhalten des zeitgesteuerten Kommunikationssystems beeinflussen, wohingegen in der zweiten Fehlerklasse alle Fehlermöglichkeiten zusammengefasst sind, die Auswirkungen auf die übertragenen Nachrichteninhalte haben.

Im Folgenden werden die definierten Fehlerklassen detailliert, die erfassten Fehlermöglichkeiten im Prototyp-Netzwerk injiziert und das zu beobachtende Verhalten analysiert.

### **A.3.1 Fehler, die das zeitliche Verhalten beeinflussen**

Das Verhalten zeitgesteuerter Kommunikationssysteme ist hauptsächlich durch das Fortschreiten der globalen Zeit geprägt. Wird dieses Zeitverständnis durch Fehler gestört, so ist davon auch die Art und Weise der Nachrichtenübertragung unter den Kommunikationsteilnehmern betroffen. Im Nachfolgenden werden die Fehlermöglichkeiten dieser Ausprägung näher betrachtet.

#### **A.3.1.1 Fehler hervorgerufen durch Konfigurationswechsel**

Wie bereits erläutert ist im TTCAN-Protokoll der Zeitmaster für die Synchronisation aller Kommunikationsknoten durch Aussendung der Referenz-Nachricht verantwortlich. Welcher Knoten im Netzwerk aktuell der Zeitmaster ist, welche Teilnehmer potentielle Zeitmaster sind und welche Knoten lediglich Zeitslave-Funktionalität besitzen, wird als die momentan vorherrschende Konfiguration bezeichnet. Insbesondere dann, wenn sich die aktuelle Konfiguration auf Grund eines Fehlers ändert, ist eine Beeinflussung des zeitlichen Verhaltens des redundanten, synchronisierten TTCAN zu erwarten. Es lassen sich Konfigurationswechsel hervorgerufen durch folgende Fehlermöglichkeiten unterscheiden:

##### **durch Ausfall des aktuellen Zeitmasters**

Ist der aktuelle Zeitmaster nicht mehr in der Lage seine Referenz-Nachricht auf dem Kommunikationsmedium zu platzieren, so wird diese Funktionalität in einem konkurrierenden Verfahren einem anderen potentiellen Zeitmaster übertragen. Der Ausfall des Zeitmasters kann jedoch durch die anderen Kommunikationsteilnehmer frühestens nach Ablauf der maximalen GAP-Länge detektiert werden. Das zeitliche Verhalten des Kommunikationssystems ändert sich dahingehend, dass mindestens für einen Basiszyklus eine

längere Zykluszeit zu Stande kommt. Das beschriebene Fehlerbild wurde ins Prototypen-Netzwerk injiziert und die resultierenden Basiszykluszeiten gemessen. Das Ergebnis ist in Abbildung A.6 gezeigt.

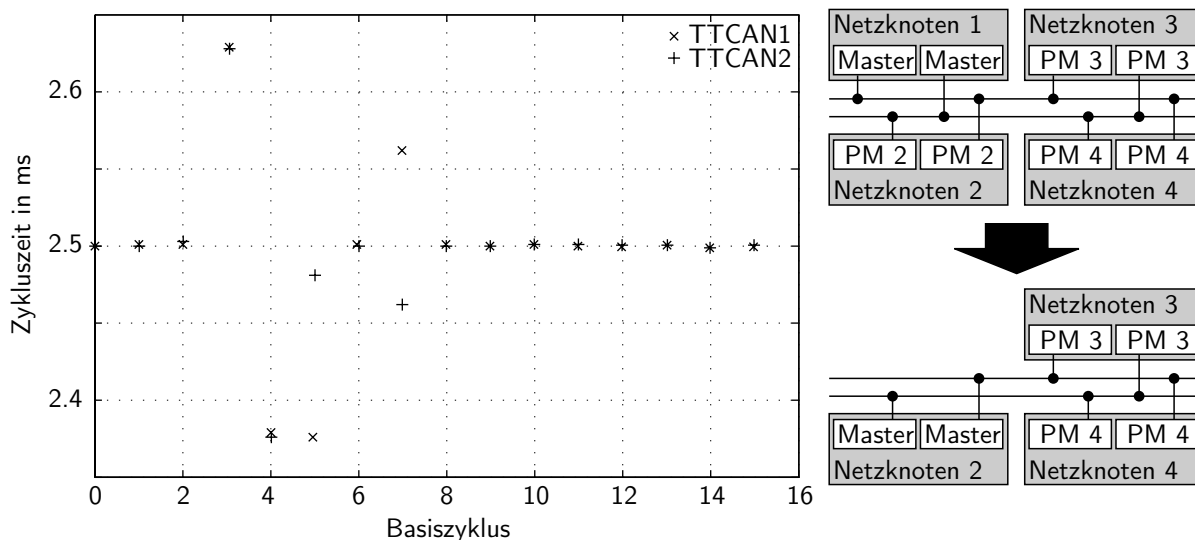


Bild A.6: Konfigurationswechsel durch Ausfall des Zeitmaster-Knotens

Im dritten Basiszyklus wird ein Fehler eingespeist, der zur Abschaltung des aktuellen Zeitmasters führt. Auf Grund des ausbleibenden GAP-Abbruchs verlängert sich die Basiszykluszeit im dritten Basiszyklus in beiden TTCAN-Kanälen auf die Maximallänge von  $2625\mu\text{s}$ . Die anschließende Übernahme der Zeitmaster-Funktionalität hat zur Folge, dass auf Grund von Abläufen im TTCAN-Protokoll im vierten Basiszyklus noch keine Lücke eingefügt werden kann. Deshalb ergibt sich im fünften Basiszyklus eine auf  $2375\mu\text{s}$  verkürzte Basiszykluszeit. Es ist gut zu erkennen, dass die anschließend auftretende Asynchronität im siebten Basiszyklus beseitigt wird und von nun an eine synchrone Kommunikation wieder hergestellt ist.

#### durch Ausfall eines TTCAN-Controllers des aktuellen Zeitmasters

Deutlich komplexer ist das Verhalten des redundanten, synchronisierten TTCAN-Netzwerks, wenn nicht der komplette Zeitmaster-Knoten ausfällt, sondern nur einer seiner beiden TTCAN-Controller defekt ist. In diesem Fall besitzen zwei unterschiedliche Kommunikationsknoten jeweils einen Zeitmaster-Controller. Da auf Grund des fehlenden Bezugssystems auf dem bisherigen Zeitmaster-Knoten keine Differenzmessung mehr möglich ist, kann der Grad der Asynchronität durch diesen Zeitmaster nicht bestimmt und deshalb auch nicht ausgeglichen werden. Im Gegenzug ist aber auch der andere Teilnehmer mit Zeitmaster-Funktionalität nicht in der Lage, die Synchronisierung zu übernehmen.

Ihm ist es zwar möglich, die exakte Zeitdifferenz zwischen TTCAN-Kanal 1 und 2 zu ermitteln, da er aber nur über einen Zeitmaster-Controller verfügt, kann er nur einen der beiden Kommunikationskanäle beeinflussen und damit keine Synchronisation im bisherigen Sinne durchführen. Darüber hinaus ist im TTCAN-Protokoll das „freiwillige“ Abgeben einer Zeitmaster-Funktionalität an einen nieder prioren Zeitmaster nicht vorgesehen, so dass nur die folgende Synchronisationsvariante als möglich erscheint: Der Knoten mit nur einem funktionsfähigen TTCAN-Kontroller gibt die globale Zeit vor und der zweite TTCAN-Kanal wird durch den anderen Teilnehmer mit Zeitmaster-Funktionalität auf diese Vorgabe synchronisiert. Dieses Fehlerverhalten ist in Abbildung A.7 dargestellt.

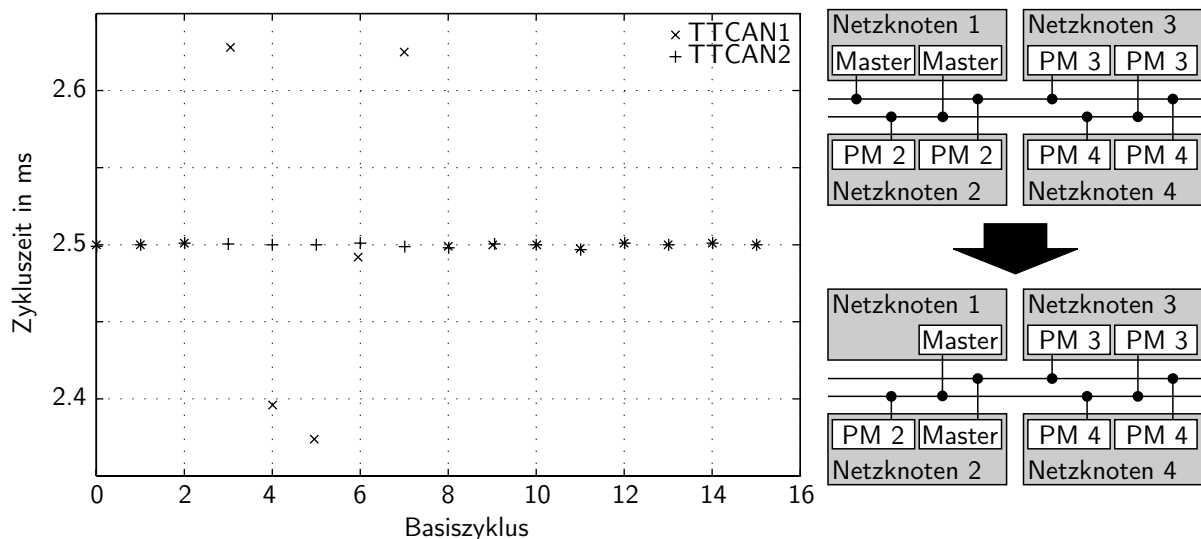


Bild A.7: Konfigurationswechsel durch Ausfall eines TTCAN-Controllers auf dem Zeitmaster-Knoten

Während der zweite TTCAN-Kanal die vorgegebene Zykluszeit von  $2,5\text{ms}$  einhält, verlängert sich der Basiszyklus des ersten Kommunikationskanals durch den Ausfall des Zeitmaster-TTCAN-Controllers im dritten Basiszyklus. Auch hier verstreicht die maximale GAP-Länge bis der nächste potentielle Zeitmaster als Ersatz für den ausgefallenen TTCAN-Controller einspringen kann. Da der erste Kommunikationskanal nicht mehr beeinflusst werden kann, erfolgt die anschließend stattfindende Synchronisation nur durch Anpassung des zweiten TTCAN-Kanals.

#### durch Hinzufügen eines höher prioren Zeitmasters

Wird durch zu späte Integration oder bei Reintegration nach einem Fehler eines Kommunikationsknotens die Konfiguration gewechselt, führt dies ebenfalls zu einem geänderten Kommunikationsverhalten. Durch Hinzufügen eines höher prioren potentiellen Zeitmaster-



Knotens muss die Zeitmaster-Funktionalität an diesen Knoten übergehen. Das dabei zutage tretende zeitliche Verhalten ist in Abbildung A.8 dargestellt.

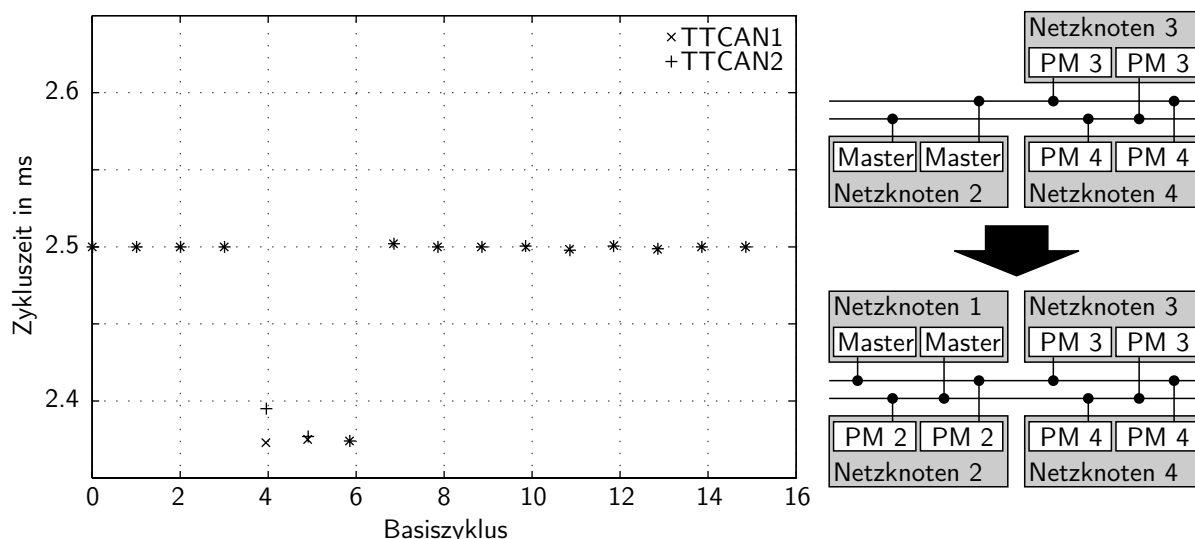


Bild A.8: Konfigurationswechsel durch Hinzufügen eines höher prioreren Zeitmaster-Knotens

Im Normalfall informiert der aktuelle Zeitmaster alle anderen Kommunikationsteilnehmer über das bevorstehende Einfügen einer Lücke mit einer Notiz in seiner Referenz-Nachricht. Wird, wie hier im vierten Basiszyklus, ein neuer, höher priorer, potentieller Zeitmaster-Knoten hinzugefügt, so fehlt ihm diese aktuelle GAP-Information. Er geht nach  $2375\mu s$  Basiszykluszeit von einer ausbleibenden Referenz-Nachricht aus, platziert seine eigene Referenz-Nachricht mit höchster Priorität und bricht sofort die eingefügte Lücke ab. Bis der Synchronisationsalgorithmus die erste Zeitmessung initiiert hat und der neue Zeitmaster in seiner Referenz-Nachricht über das bevorstehende Einfügen einer Lücke informiert hat, vergehen weitere zwei Basiszyklen ohne GAP. Ab dem siebten Basiszyklus sind die beiden Kommunikationskanäle wieder synchron und weisen die Standard-Basiszykluszeit auf.

### A.3.1.2 Fehler hervorgerufen durch verzögerte Referenz-Nachricht

Neben den Konfigurationswechseln können vor allem Fehler, die die Referenz-Nachrichten betreffen, das zeitliche Verhalten eines TTCAN-Kanals beeinflussen. Der folgende Abschnitt betrachtet Fehler, die zu einer verzögerten Aussendung der Referenz-Nachricht führen. Dazu gehören z.B. Fehler im Kommunikationsmedium selbst, wie Kurzschluss oder Übertragungsfehler durch EMV, oder auch das zeitlich fehlerhafte Senden eines anderen Kommunikationsteilneh-

mers einhergehend mit dem daraus resultierenden temporären Blockieren des Nachrichtenkanals. Durch Injektion eben dieser Fehler im Prototyp kann das Verhalten des redundanten, synchronisierten TTCAN-Netzwerks analysiert werden. Die Abbildung A.9 zeigt die Messergebnisse für die Einspeisung eines Fehlers, der zu einer Verzögerung der Referenz-Nachricht um etwa  $200\mu s$  führt.

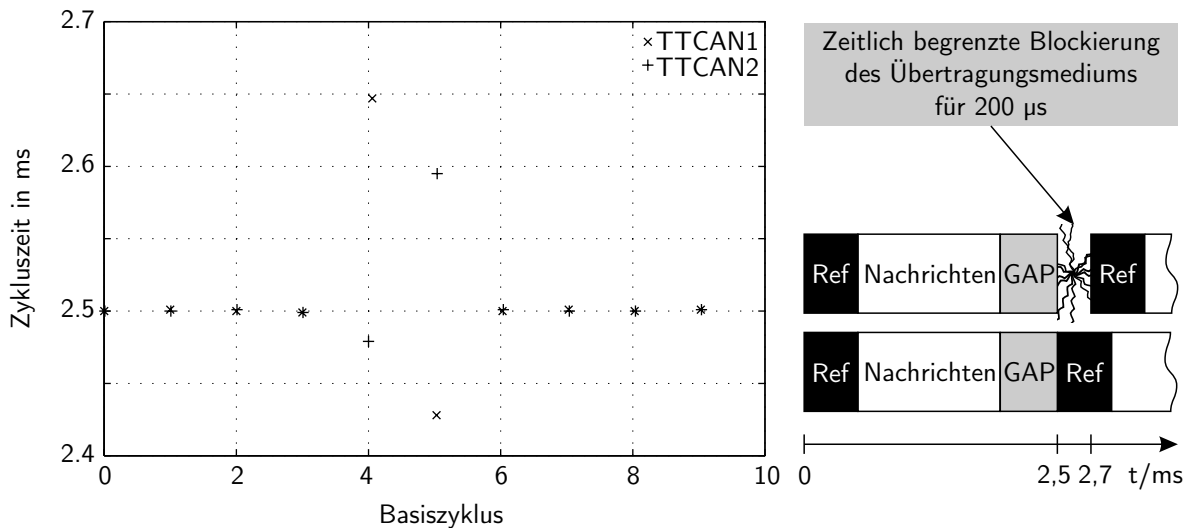


Bild A.9: Asynchronität durch verzögerte Referenz-Nachricht

Zu Beginn des vierten Basiszyklus wird ein Fehler in den ersten TTCAN-Kanal injiziert, der zu einer temporären Blockierung des Kommunikationsmediums von  $200\mu s$  führt. Aufgrund der vorliegenden Störung ist es keinem Zeitmaster dieses Kommunikationskanals möglich, eine Referenz-Nachricht zu senden. Bis zur bereits beschriebenen Zeitmarke *Ref Trigger Watch* versuchen alle Zeitmaster wiederholt eine Referenz-Nachricht auszusenden. Ist das transiente Störereignis vor *Ref Trigger Watch* beendet, so wird eine verspätete Referenz-Nachricht gesendet. Die dadurch entstehende Asynchronität kann anschließend auf Grund der GAP-Größe in nur einem Basiszyklus ausgeglichen werden.

### A.3.1.3 Hervorgerufen durch fehlerhafte Referenz-Nachrichten

Neben verzögerten Referenz-Nachrichten sorgen natürlich auch fehlerhafte Referenz-Nachrichten für ein geändertes zeitliches Verhalten des Kommunikationssystems. Die Klasse der fehlerhaften Referenz-Nachrichten umfasst sowohl zeitlich als auch inhaltlich fehlerhafte Nachrichten.

Zeitlich fehlerhafte Referenz-Nachrichten werden durch Fehler ausgelöst, die einen Kommunikationsteilnehmer veranlassen, eine Referenz-Nachricht völlig unabhängig vom aktuellen Status

des Basiszyklus auszusenden. Diese Fehlermöglichkeit könnte z.B. durch einen Hardware-Fehler oder durch eine falsche Konfiguration (zusätzliche Vergabe der Nachrichten-ID einer Referenz-Nachricht an eine andere Nachricht) auftreten. Die dabei entstehenden zeitlichen Differenzen zwischen den beiden Kommunikationskanälen können recht groß werden, wie Abbildung A.10 zeigt. Die plötzlich während der Ausführung des 15. Basiszyklus im ersten TTCAN-Kanal auftretende fehlerhafte Referenz-Nachricht verkürzt den Basiszyklus zu einer Zykluszeit von etwa  $1,2\text{ms}$ . Trotz verhältnismäßig großem GAP sind die Kommunikationskanäle für fünf Basiszyklen asynchron. Nach dieser Zeitspanne hat der Synchronisationsalgorithmus jedoch die aufgetretene Differenz vollständig kompensiert.

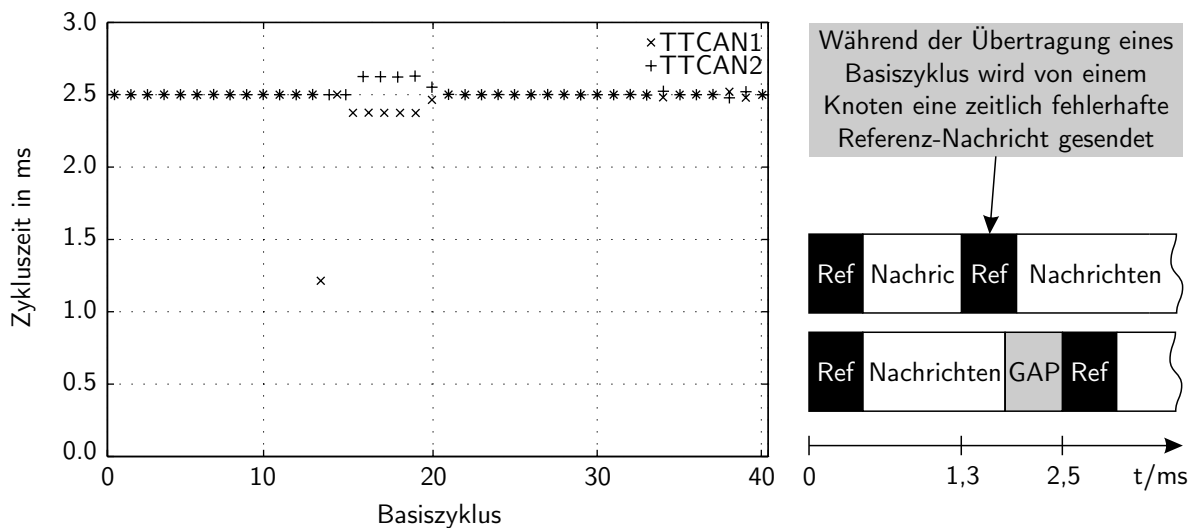


Bild A.10: Asynchronität durch zeitlich fehlerhafte Referenz-Nachricht

Die Referenz-Nachrichten werden neben ihrer Synchronisierungsaufgabe auch zum Datenaustausch verwendet. Im verwendeten TTCAN-Modus beinhalten sie Informationen zur Aktivierung des GAP-Modus und eine Indexierung zur Identifikation des aktuellen Basiszyklus innerhalb der Systemmatrix. Damit kann zum einen der zeitliche als auch der inhaltliche Aspekt einer Referenz-Nachricht durch einen Fehler beeinflusst werden. Das Kommunikationsverhalten bei inhaltlichen Fehlern in Bezug auf die GAP-Information wurde bereits im Zusammenhang mit dem Konfigurationswechsel behandelt. Bei der Verfälschung des Basiszyklus-Index hingegen handelt es sich um eine neue, noch nicht untersuchte Fehlermöglichkeit.

Abbildung A.11 zeigt die Messung des TTCAN-Verhaltens bei der Injektion eines Fehlers, der zu einer Referenz-Nachricht mit einem fehlerhaften Basiszyklus-Index führt. Der in diesem Fall gesendete Basiszyklus-Index ist in der Systemmatrix nicht definiert. Nach der Fehlerinjektion im vierten Basiszyklus verlängert sich dessen Zykluszeit auf etwa  $2750\mu\text{s}$ . Es fällt sowohl auf,

dass die anschließend notwendige Synchronisation nur durch den zweiten TTCAN-Kanal erbracht wird, als auch, dass ab dem 12. Basiszyklus trotzdem die synchrone Kommunikation wieder hergestellt ist. Der Grund für das gezeigte Verhalten ist bisher unklar und muss weiter untersucht werden. Im Hinblick auf die Zielsetzung der Funktionsanalyse des Synchronisationsalgorithmus und insbesondere der Ableitung eines Fehlerverhaltensmodells ist dies jedoch von untergeordneter Bedeutung.

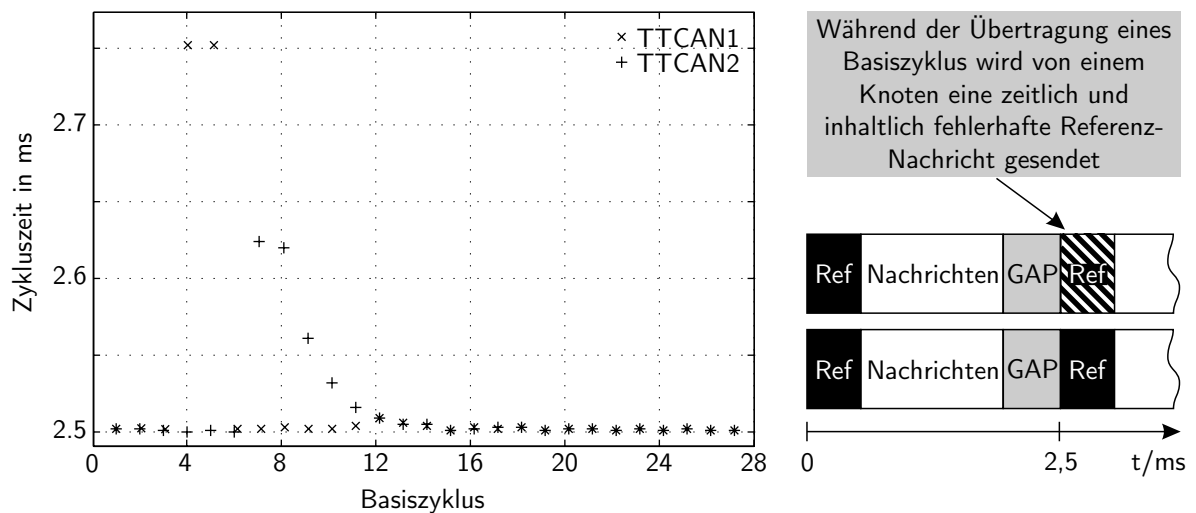


Bild A.11: Asynchronität durch inhaltlich fehlerhafte Referenz-Nachricht

### A.3.2 Fehler, die die Nachrichteninhalte beeinflussen

Neben der Einhaltung der zeitlichen Rahmenbedingungen besteht die Hauptaufgabe eines Kommunikationssystems im korrekten Datenaustausch zwischen Kommunikationsteilnehmern. Aus diesem Grund bilden Fehler, die den Informationsaustausch hinsichtlich des Wertebereichs beeinflussen, die zweite große zu untersuchende Fehlerklasse. Dieser Fehlertyp wiegt insbesondere in einem redundanten Kommunikationssystem schwer, denn für den Fall, dass empfängerseitig von einer Quelle zwei unterschiedliche Daten zur Verfügung stehen, kann dieser Konflikt zwar erkannt, aber nicht ohne weiteres gelöst werden. Das zu analysierende Kommunikationsverhalten bei empfängerseitigen Dateninkonsistenzen ist in folgenden Fehlermöglichkeiten begründet:

#### A.3.2.1 Fehler hervorgerufen durch Störungen im Übertragungsmedium

Empfängerseitig können unterschiedliche Daten von derselben Quelle durch Veränderungen auf dem Transportweg entstehen. Fehlermöglichkeiten, die solche Auswirkungen zeigen, können von

Übertragungsfehlern durch EMV-Strahlung bis hin zum kompletten Nachrichtenverlust reichen.

Zur Analyse des Kommunikationsnetzes im Falle der Präsenz von Fehlern dieser Klasse muss das Prototypen-Netzwerk erweitert werden. Auf jedem Kommunikationsknoten ist zusätzlich ein Echtzeitbetriebssystem implementiert, das zeitlich synchronisiert zum redundanten TTCAN-Netz arbeitet. Es verwaltet vier Software-Tasks: Datengenerierung (Zähler), Senden, Empfangen und Datenauswertung. Der Ablauf ist im oberen Teil von Abbildung A.12 dargestellt.

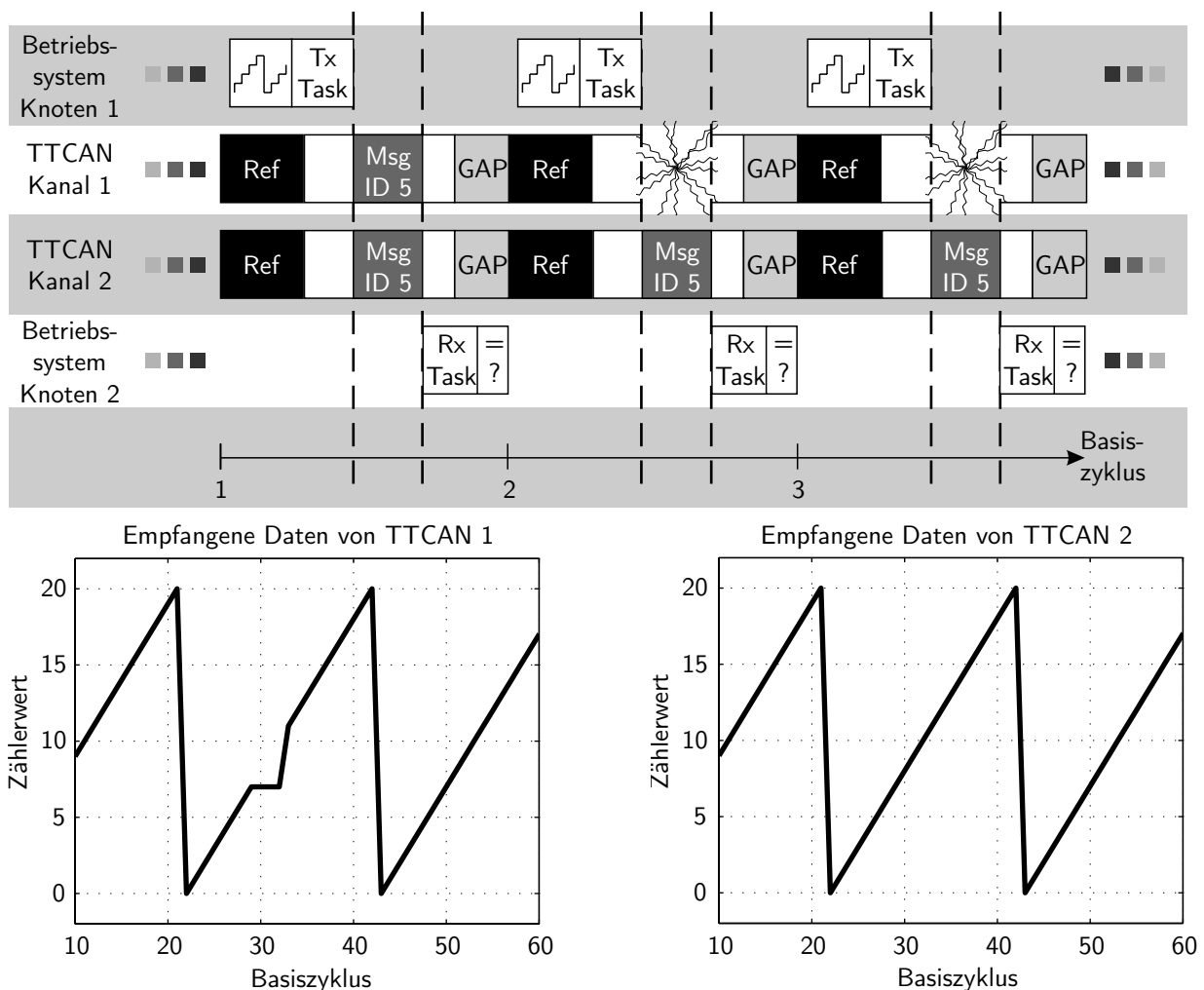


Bild A.12: Inkonsistente Daten durch Störungen in einem Kommunikationskanal

Das zum redundanten TTCAN-Netz synchrone Echtzeitbetriebssystem aktiviert zunächst die Datengenerierung auf dem Kommunikationsknoten 1. Dabei handelt es sich um einen Zähler im Intervall von  $[0\ 20]$ . Im Anschluss wird die Task ausgeführt, die den aktuellen Zählerwert an die beiden TTCAN-Controller weiterleitet. Sie erzeugen daraus gültige Kommunikationsnachrichten und übertragen diese gemäß dem definierten Zeitplan als Nachricht mit der ID 5. Beim Empfänger, dem Kommunikationsknoten 2, werden zum Zeitpunkt des erwarteten Übertra-

gungsende die empfangenen Daten aus den TTCAN-Controllern ausgelesen und anschließend verarbeitet. Treten während der Übertragung Störungen im Kanal auf, wie im TTCAN-Kanal 1 dargestellt, werden keine Daten empfangen oder die empfangenen Daten als verfälscht erkannt und verworfen. Da in diesem Fall die letzten gültigen Daten im Empfangsspeicher der TTCAN-Controller verbleiben, werden diese durch die Empfangen-Task erneut eingelesen. Dies führt, wie im unteren Teil von Abbildung A.12 zu erkennen ist, zu Dateninkonsistenzen beim Empfänger.

### A.3.2.2 Fehler hervorgerufen durch Asynchronität der TTCAN-Kanäle

Arbeiten die TTCAN-Kanäle asynchron, beeinträchtigt das nicht nur den zeitlichen Ablauf im Kommunikationsnetzwerk, sondern auch den Datenaustausch. Im oberen Teil von Abbildung A.13 wird dieser Fehlerfall wieder am erweiterten Prototypen-Netzwerk verdeutlicht.

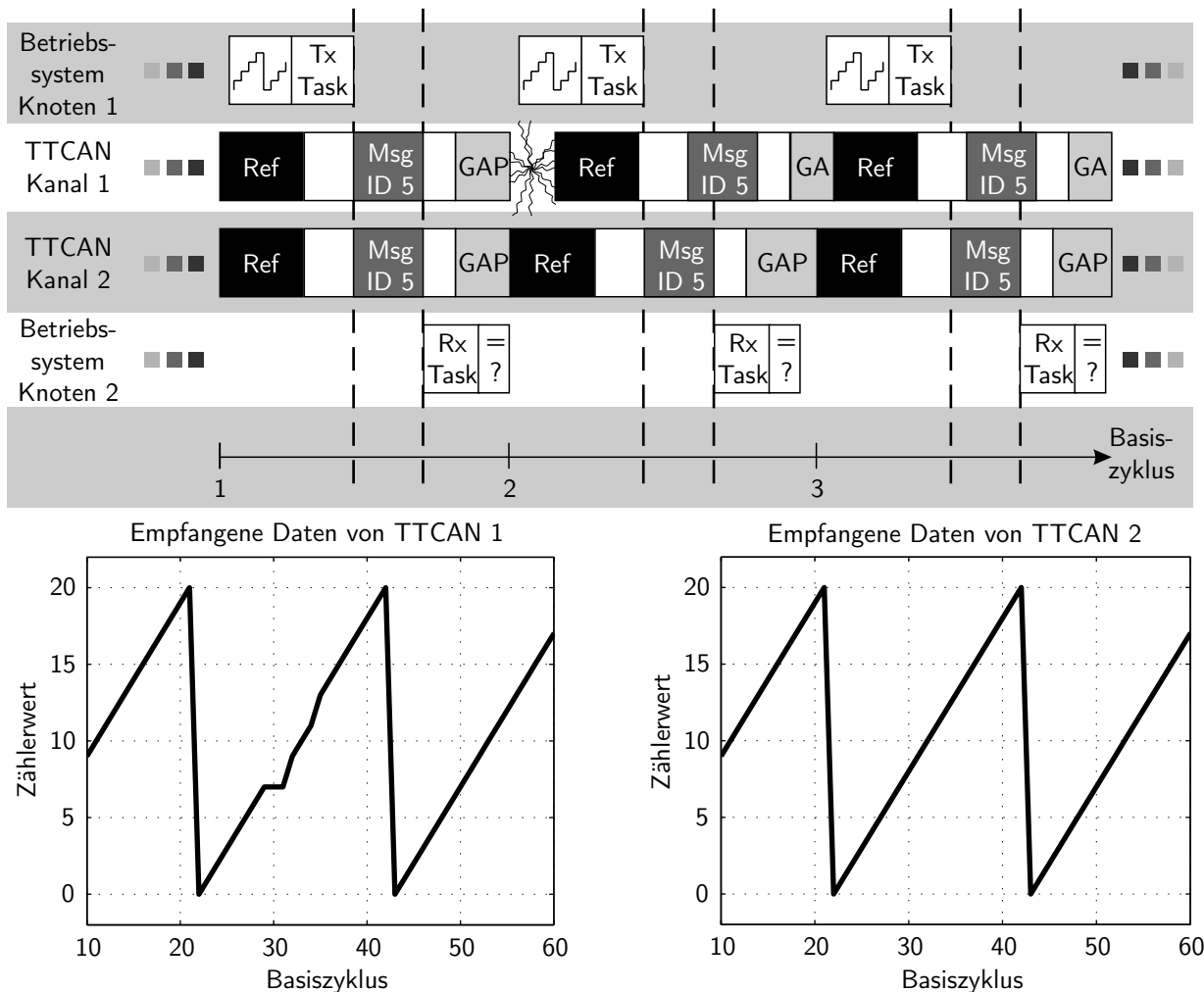


Bild A.13: Inkonsistente Daten durch asynchrone Kommunikationskanäle

Der prinzipielle Ablauf entspricht der Betrachtung aus dem vorangegangenen Fehlerszenario. Der Unterschied besteht jedoch in der Tatsache, dass in diesem Fall nicht einzelne Nachrichten bei ihrer Übertragung in einem TTCAN-Kanal gestört, sondern die Aussendungen der Referenz-Nachrichten verzögert werden. Damit entstehen vorübergehend asynchrone TTCAN-Kanäle. In diesem Fall stimmt das vom Empfänger erwartete Übertragungsende der Nachricht zumindest für einen TTCAN-Kanal nicht mehr mit dem tatsächlichen überein. Zum Zeitpunkt der Aktivierung der Empfangen-Task ist die Übertragung in nur einem der beiden Kommunikationskanäle abgeschlossen. Das Auslesen der übertragenen Daten liefert deshalb zum einen den aktuellsten Wert und zum anderen den Wert aus dem vorangegangenen Basiszyklus. Dieser Vorgang ist im unteren Teil von Abbildung A.13. Die Ausprägung und der zeitliche Verlauf dieses Fehlerverhaltens hängen stark von der tatsächlichen Verzögerung, der GAP-Größe und dem zeitlichen Spielraum zwischen Abschluss der Nachrichtenübertragung und Aktivierung der Empfangen-Task ab.

---

# Literaturverzeichnis

- [Bak87] BAKER, J.: Reducing Bias and Inefficiency in the selection algorithm. In: *Proceedings of the Second International Conference on Genetic Algorithm and their Application, Hillsdale, New Jersey, USA: Lawrence Erlbaum Associates, 1987*
- [Ben04] BENZ, S.: *Eine Entwicklungsmethodik für sicherheitsrelevante Elektroniksysteme im Automobil*, Universität Karlsruhe, Dissertation, 2004
- [BFS<sup>+</sup>02] BERTRAM, T. ; FLORES, P. T. ; SCHIRMER, J. ; PETERSEN, J. ; LAPP, A. ; KRAFT, D. ; HERMSEN, W.: Software-Entwicklung für vernetzte Steuergeräte - Von der Cartronic-Domänenstruktur zum Steuergerätesoftwarecode. In: *ATZ / MTZ / Automotive Engineering Partners-Sonderausgabe Automotive Electronics (2002)*
- [Bie03] BIEGERT, U.: *Ganzheitliche modellbasierte Sicherheitsanalyse von Prozessautomatisierungssystemen*, Universität Stuttgart, Dissertation, 2003
- [Bis90] BISHOP, P.: *Dependability of Critical Computer Systems 3*. Techniques Directory, EWICS TC7, Elsevier Applied Science, London, New York, 1990
- [BSDV97] BERTRAM, T. ; SCHRÖDER, W. ; DOMINKE, P. ; VOLKART, A.: CARTRONIC - ein Ordnungskonzept für Steuerungs- und Regelungssysteme in Kraftfahrzeugen. In: *VDI-Berichte 1374: Systemengineering in der Kfz-Entwicklung, VDI-Verlag, 1997, S. 369–397*
- [CERT04] CORNO, F. ; ESPOSITO, F. ; REORDA, M. S. ; TOSATO, S.: Evaluating the Effects of Transient Faults on Vehicle Dynamic Performance in Automotive Systems. In: *IEEE International Test Conference (ITC), Charlotte (NC), USA, 2004, S. 1332–1339*
- [CFP03] CONRAD, M. ; FEY, I. ; POHLHEIM, H.: Automatisierung der Testauswertung für Steuergerätesoftware. In: *VDI-Berichte Band 1789, 2003, S. 299–315*
- [Con01] CONRAD, M.: Beschreibung von Testszenerarien für Steuergeräte-Software - Vergleichskriterien und deren Anwendung. In: *VDI-Berichte Band 1646, 2001, S. 381–399*
- [Dai02] DAIS, S.: Elektronik und Sensorik: Basis der Sicherheit. In: *Technischer Kongress „Sicherheit durch Elektronik“, VDA - Verband der Automobilindustrie, 2002*



- [DE02] DIN EN 61508, Teil 3.: *Funktionale Sicherheit sicherheitsbezogener elektrischer / elektronischer / programmierbarer elektronischer Systeme*. VDE-Verlag, 2002
- [Deu01] DEUTSCHE GESELLSCHAFT FÜR QUALITÄT E.V.: *FMEA - Fehlermöglichkeits- und Einflussanalyse*. Beuth-Verlag, 2001
- [DFM<sup>+</sup>97] DILGER, E. ; FÜHRER, T. ; MÜLLER, B. ; POLEDNA, S. ; THURNER, T.: X-by-Wire: Design von verteilten, fehlertoleranten und sicherheitskritischen Anwendungen in modernen Kraftfahrzeugen. In: *VDI-Berichte 1374*, 1997, S. 427–442
- [Dil99] DILGER, E.: Fehlertolerante Rechnerarchitekturen für Kraftfahrzeuganwendungen. In: *5. Esslinger Forum für Kfz-Mechatronik, Esslingen*, 1999
- [DIN81] DIN 25424, TEIL 1: *Fehlerbaumanalyse; Methode und Bildzeichen*. Beuth-Verlag, 1981
- [DIN85a] DIN 25419: *Ereignisablaufanalyse; Verfahren, graphische Symbole und Auswertung*. Beuth-Verlag, 1985
- [DIN85b] DIN 25448: *Ausfalleffektanalyse (Fehler-Möglichkeits- und -Einfluß-Analyse)*. Beuth-Verlag, 1985
- [DSSS01] DORNSEIFF, M. ; STAHL, M. ; SIEGERVAND, M. ; SAX, E.: Durchgängige Testmethoden für komplexe Steuerungssysteme - Optimierung der Prüftiefe durch effiziente Testprozesse. In: *10. Internationaler Kongress, Elektronik im Kraftfahrzeug, Baden-Baden*, 2001
- [Ech90] ECHTLE, K.: *Fehlertoleranzverfahren*. Springer-Verlag, 1990
- [EPK<sup>+</sup>02] ECKRICH, M. ; PISCHINGER, M. ; KRENN, M. ; BARTZ, R. ; MUNNIX, P.: Aktivlenkung - Anforderungen an Sicherheitstechnik und Entwicklungsprozess. In: *11. Aachener Kolloquium: Fahrzeug- und Motorentechnik*, 2002
- [ES98] ECHTLE, K. ; SILVA, J.G.: Fehlerinjektion - ein Mittel zur Bewertung der Maßnahmen gegen Fehler in komplexen Rechensystemen. In: *Informatik-Spektrum 21* (1998), S. 328–336
- [FDH<sup>+</sup>01] FÜHRER, T. ; DIETERLE, W. ; HARTWICH, F. ; HUGEL, R. ; KRAFT, D. ; MÜLLER, B.: TTCAN: Zeitgesteuerter Nachrichtenverkehr in CAN-Netzwerken. In: *VDI-Berichte Band 1646*, 2001, S. 43–52

- 
- [Fle05] FLEXRAY CONSORTIUM: *Flexray Specification*. <http://www.flexray.com>, letzter Abruf: März 2005, 2005
- [FMH<sup>+</sup>01] FREITAG, R. ; MOSER, M. ; HARTL, M. ; KOEPERNIK, J. ; ECKSTEIN, L.: Anforderungen an das Sicherheitskonzept von Lenksystemen mit Steer-by-Wire Funktionalität. In: *VDI-Berichte*, 2001
- [GBW01] GOTTWICK, U. ; BOOZ, O. ; WILLMANN, K.-H.: Sicherheitskonzept der elektrohydraulischen Bremse. In: *4. Stuttgart Symposium Kraftfahrwesen und Verbrennungsmotoren*, Expert Verlag, 2001, S. 668–682
- [Gen03] GENTLE, J.: *Random number generation and Monte Carlo methods*. Springer-Verlag, 2003
- [GKK04] GERDES, I. ; KLAWONN, F. ; KRUSE, R.: *Evolutionäre Algorithmen*. 1. Auflage, Wiesbaden, Vieweg-Verlag, 2004
- [Gol89] GOLDBERG, D.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, 1989
- [GR02] GÜHMANN, C. ; RIESE, J.: Testautomatisierung in der Hardware-in-the-Loop Simulation. In: *VDI-Berichte 1672*, 2002, S. 511–527
- [Hed01] HEDENETZ, B.: *Entwurf von verteilten fehlertoleranten Elektronikarchitekturen in Kraftfahrzeugen*, Universität Tübingen, Dissertation, 2001
- [Höf96] HÖFLING, T.: *Methoden zur Fehlererkennung mit Parameterschätzung und Paritätsgleichungen*, Universität Darmstadt, Dissertation, 1996
- [IPG04] IPG AUTOMOTIVE GMBH: *IPG-Driver*. <http://www.ipg.de>, letzter Abruf: März 2004, 2004
- [ISO04] ISO/PRF 11898 PART 4: *TTCAN*. ISO Standards, 2004
- [ISS02] ISERMANN, R. ; SCHWARZ, R. ; STÖLZL, S.: Systemsicherheit des Active Front Steering. In: *IEEE Control Systems Magazine* 22 (2002), Nr. 5, S. 64–81
- [Jür97] JÜRGENSOHN, T.: *Hybride Fahrermodelle*, Technische Universität Berlin, Dissertation, 1997
- [Kai03] KAISER, R.: Das Einmaleins des Testens. In: *embedded world conference*, 2003, S. 243–251

- [KGV83] KIRPATRICK, S. ; GELATT, C. ; VECCHI, M.: Optimization by Simulated Annealing. In: *Science* 220 (1983), Nr. 4598, S. 671–680
- [KKN95] KIENCKE, U. ; KYTÖLA, T. ; NEUMANN, K.: Architectural Trends in Automotive Electronics. In: *1. IFAC-Workshop on Advances in Automotive Control, Ascona, Schweiz*, 1995
- [KKS<sup>+</sup>02] KLUGE, J. ; KLAGES, B. ; SPICHALSKY, C. ; HEINRICH, A. ; LEOHOLD, J. ; HENKE, T.: Hardware-in-the-Loop-Simulation und Testautomatisierung - Einsatz von Simulationstools zur Funktionserprobung. In: *Sonderausgabe ATZ/MTZ VW Phaeton* (2002), S. 138–144
- [Kop97] KOPETZ, H.: *Real-time systems: design principles for distributed embedded applications*. Kluwer-Verlag, 1997
- [Lad01] LADKIN, P.: *Causal system analysis*. RVS Group, Faculty of Technology, University of Bielefeld, Germany, 2001
- [Lau96] LAUFENBERG, X.: *Ein modellbasiertes qualitatives Verfahren für die Gefahrenanalyse*, Universität Stuttgart, Dissertation, 1996
- [LD02] LEEN, G. ; D.HEFFERNAN: TTCAN: a new time-triggered controller area network. In: *Elsevier Microprocessors and Microsystem* (2002), Nr. 26, S. 77–94
- [Lev95] LEVESON, N.: *SAFWARE: system safety and computers : a guide to preventing accidents and losses caused by technology*. Reading, Mass., USA, Addison-Wesley, 1995
- [LFS<sup>+</sup>01] LAPP, A. ; FLORES, P. T. ; SCHIRMER, J. ; KRAFT, D. ; HERMSEN, W. ; BERTRAM, T. ; PETERSEN, J.: Softwareentwicklung für Steuergeräte im Systemverbund - Von der CARTRONIC-Domänenstruktur zum Steuergerätecode. In: *10. Internationaler VDI Kongress Elektronik im Kraftfahrzeug*, 2001
- [LG99] LAUBER, R. ; GÖHNER, P.: *Prozessautomatisierung Band 2*. Springer-Verlag, 1999
- [LLF<sup>+</sup>02] LÄNGST, W. ; LAPP, A. ; FLORES, P. T. ; SCHIRMER, J. ; KRAFT, D. ; KIENCKE, U.: CARTRONIC based Safety Analysis: Introducing Safety Aspects In Early Development Phases. In: *SAE International Congress and Exposition, Detroit*, 2002
- [Län03] LÄNGST, W.: *Formale Anwendung von Sicherheitsmethoden bei der Entwicklung verteilter Systeme*, Universität Karlsruhe, Dissertation, 2003

- 
- [LT04] LEOHOLD, J. ; THEUERKAUF, H.J.: Hil-Methoden zur Entwicklung und Applikation mechatronischer Kfz-Systeme und sicherheitskritischer Systemarchitekturen. In: *Virtual Product Creation, Stuttgart, 2004*
- [Mac03] MACADAM, C.: Understanding and Modeling the Human Driver. In: *Vehicle System Dynamics 40* (2003)
- [Mec06] MECHANICAL SIMULATION: *CarSim*. <http://www.carsim.com>, letzter Abruf: März 2006, 2006
- [Meh04] MEHL, V.: Entwicklung mechatronischer Fahrzeugregelsysteme mit kombiniertem Einsatz von Simulation und Fahrversuch. In: *VDI-Berichte 1842: Mechatronischer Systementwurf, Darmstadt* (2004)
- [Mic99] MICHALEWICZ, Z.: *Genetic Algorithms + Data Structure = Evolution Programs*. 3. Auflage, Berlin, Heidelberg, Springer-Verlag, 1999
- [Mit90] MITSCHKE, M.: *Dynamik der Kraftfahrzeuge: Band C: Fahrverhalten*. Springer-Verlag, 1990
- [Mit95] MITSCHKE, M.: *Dynamik der Kraftfahrzeuge: Band A: Antrieb und Bremsung*. Springer-Verlag, 1995
- [Mit97] MITSCHKE, M.: *Dynamik der Kraftfahrzeuge: Band B: Schwingungen*. Springer-Verlag, 1997
- [Moi01] MOIK, A.: *Ingenieurgerechte formale Methoden für die Entwicklung von sicheren Automatisierungssystemen*, Universität Stuttgart, Dissertation, 2001
- [MP03] MEYNA, A. ; PAULI, B.: *Taschenbuch der Zuverlässigkeits- und Sicherheitstechnik*. Carl Hanser Verlag, 2003
- [MT00] MÜLLER, D. ; TIETJEN, T.: *FMEA-Praxis : das Komplettpaket für Training und Anwendung*. München, Hanser-Verlag, 2000
- [Nen01] NENNINGER, G.: *Modellbildung und Analyse hybrider dynamischer Systeme als Grundlage für den Entwurf hybrider Steuerungen*, Universität Karlsruhe, Dissertation, 2001
- [NK03] NEUKUM, A. ; KRÜGER, H.-P.: Fahrerreaktionen bei Lenksystemstörungen - Untersuchungsmethodik und Bewertungskriterien. In: *VDI-Berichte 1791*, 2003, S. 297–318

- [OKC01] OLSSON, P. ; KELLING, N. ; CHAUMETTE, P.: EU-projekt BRAKE, development of a distributed brake by wire system. In: *10. Internationaler Kongress, Elektronik im Kraftfahrzeug, Baden-Baden*, 2001
- [Pau05] PAUL-STÜVE, T.: *Performing a Why-Because Analysis - A Practical Guide to the Why-Because Analysis Method*. RVS Group, Faculty of Technology, University of Bielefeld, Germany, 2005
- [Poh00] POHLHEIM, H.: *Evolutionäre Algorithmen: Verfahren, Operatoren und Hinweise für die Praxis*. Springer-Verlag, 2000
- [PPG04] PAPADOPOULOS, Y. ; PARKER, D. ; GRANTE, C.: Automating the Failure Modes and Effects Analysis of Safety Critical Systems. In: *8. IEEE International Symposium in High Assurance Systems Engineering*, 2004
- [Rec94] RECHENBERG, I.: *Evolutionstrategie*. Frommann Holzboog, 1994
- [Rei83] REIMPELL, J.: *Fahrwerktechnik: Federung und Fahrwerkmechanik*. Springer-Verlag, 1983
- [Rei86a] REIMPELL, J.: *Fahrwerktechnik: Lenkung*. Springer-Verlag, 1986
- [Rei86b] REIMPELL, J.: *Fahrwerktechnik: Reifen und Räder*. Springer-Verlag, 1986
- [Rei90] REICHELT, W.: *Ein adaptives Fahrermodell zur Bewertung der Fahrdynamik von Pkw in kritischen Situationen*, Technische Universität Braunschweig, Dissertation, 1990
- [Rei92] REIMPELL, J.: *Fahrwerktechnik: Grundlagen*. Springer-Verlag, 1992
- [Rei95] REIMPELL, J.: *Fahrwerktechnik: Radaufhängungen*. Springer-Verlag, 1995
- [RKR05] REINELT, W. ; KLIER, W. ; REIMANN, G.: Systemsicherheit des Active Front Steering. In: *at - Automatisierungstechnik* 53 (2005), Nr. 1, S. 36–43
- [RPGN97] RUDNICK, E. ; PATEL, J. ; GREENSTEIN, G. ; NIERMANN, T.: A Genetic Algorithm Framework for Test Generation. In: *IEEE Transaction on computer-aided design of integrated circuits and systems* 16 (1997), S. 1034–1044
- [SAE96a] SAE AEROSPACE RECOMMENDED PRACTICE 4754: *Certification considerations for highly-integrated or complex aircraft systems*. Standard, SAE - The Engineering Society for Advancing Mobility Land Sea Air and Space, 1996

- 
- [SAE96b] SAE AEROSPACE RECOMMENDED PRACTICE 4761: *Guidelines and methods for conducting the safety assessment process on civil airborne systems and equipment*. Standard, SAE - The Engineering Society for Advancing Mobility Land Sea Air and Space, 1996
- [Sch99] SCHNEEWEISS, W.: *Die Fehlerbaum-Methode*. Hagen, LiLoLe-Verlag, 1999
- [Sch03] SCHOOF, J.: OSEKtime - Betriebssystem-Standard für X-by-Wire. In: *VDI Fortschritt-Berichte, Reihe 12: Verkehrstechnik/Fahrzeugtechnik, Band 525*, 2003, S. 122–133
- [Sch05] SCHWOON, S.: *Networks and Processes: Lecture Slides*. Institute of Formal Methods in Computer Science, Universität Stuttgart, Germany, <http://www.fmi.uni-stuttgart.de/szs/teaching/ws0405/nets/>, letzter Abruf: März 2005, 2005
- [SCJ98] SCHNIEDER, E. ; CHOUIKHA, M. ; JANHSEN, A.: Klassifikation und Bewertung von Beschreibungsmitteln für die Automatisierungstechnik. In: *at - Automatisierungstechnik* 46 (1998), S. 582ff.
- [Sie03] SIEDERSLEBEN, J.: *Softwaretechnik : Praxiswissen für Software-Ingenieure*. München, Hanser-Verlag, 2003
- [Spi01] SPITZER, B.: *Modellbasierter Hardware-in-the-Loop Test von eingebetteten elektronischen Systemen*, Universität Karlsruhe, Dissertation, 2001
- [SS04] SMITH, D. ; SIMPSON, K.: *Functional safety: a straightforward guide to applying IEC 61508 and related standards*. Oxford, Butterwoth-Heinemann, 2004
- [Stö00] STÖLZL, S.: *Fehlertolerante Pedaleinheit für ein elektromechanisches Bremssystem (Brake-by-Wire)*, Technische Universität Darmstadt, Dissertation, 2000
- [Sta06] STABREY, S.: *Adaptive Fahrdynamikregelung unter Nutzung von Fahrspurinformationen*, Technische Universität Ilmenau, Dissertation, 2006
- [Tra05] TRAUTNER, P.: *Fahrdynamische Validierung eines Fahrzeugmodells*, Fachhochschule Karlsruhe, Fachbereich Mechatronik und Naturwissenschaften, Diplomarbeit, 2005
- [TTT05] TTTTECH: *Time Triggered Protocol Specification*. <http://www.tttech.com>, letzter Abruf: März 2005, 2005
- [Ver96] VERBAND DER AUTOMOBILINDUSTRIE E.V.: *Sicherung der Qualität vor Serieneinsatz: System-FMEA*. VDA-Band 4, Teil 2, 1996

[Vid93] VIDAL, R.: *Applied Simulated Annealing*. Springer-Verlag, 1993

[XBWT98] X-BY-WIRE-TEAM: X-by-Wire: Safety related fault tolerant systems in vehicles -  
Final report / Europäische Union. 1998. – Forschungsbericht

[Zom91] ZOMOTOR, A.: *Fahrzeugtechnik: Fahrverhalten*. Vogel-Verlag, 1991





