Methodische Entwicklung hochintegrierter mechatronischer Systeme unter funktionalen, zuverlässigkeits- und sicherheitstechnischen Aspekten
-
Analyse und Quantifizierung

Von der Fakultät für Ingenieurwissenschaften,
Abteilung Maschinenbau und Verfahrenstechnik der
Universität Duisburg-Essen
zur Erlangung des akademischen Grades
eines
Doktors der Ingenieurwissenschaften
Dr.-Ing.

genehmigte Dissertation

von

Marco Junglas
aus
Cochem

Gutachter: Univ.-Prof. Dr.-Ing. Dirk Söffker
Prof. Dr.-Ing. Bernd Bertsche
Tag der mündlichen Prüfung: 12. Juli 2012

Danksagung

Die hier vorliegende Arbeit entstand im Rahmen eines Forschungsprojektes durch die Zusammenarbeit zwischen der Firma TRW Automotive, Lucas Varity GmbH und dem Lehrstuhl Steuerung, Regelung und Systemdynamik der Universität Duisburg-Essen.

Für die Möglichkeit der Durchführung dieses Forschungsprojektes möchte ich mich bei der Firma TRW, für die Bewilligung dieser Forschungsarbeit und bei meinen Kollegen sowie Herrn Dr.-Ing. Dirk Kesselgruber für die Unterstützung bedanken.

An dieser Stelle möchte ich mich besonders bei Herrn Univ.-Prof. Dr.-Ing. Dirk Söffker, für die Betreuung, Förderung und Anregungen während der Entstehung dieser Arbeit sowie das mir entgegengebrachte Vertrauen bedanken.

Ebenfalls bedanke ich mich bei Herrn Prof. Dr.-Ing. Bertram Bertsche für die Begutachtung der Arbeit.

Herrn Dr.-Ing. Rüdiger Eick gebührt besonderer Dank für die Geduld, die Betreuung, die Unterstützung, die vielfältigen Diskussionen und die Förderung dieser Arbeit und darüber hinaus.

Bei meinen Kollegen am Lehrstuhl Steuerung, Regelung und Systemdynamik möchte ich mich für die gemeinsame Zeit, die kollegiale Zusammenarbeit und die Unterstützung bedanken. Hier insbesondere für die angenehme und freundschaftliche Atmosphäre während meiner Aufenthalte in Duisburg.

Mein größter Dank gilt meinen Eltern und meiner Familie die mich während dieser Zeit stets motiviert und bestärkt sowie mir die notwendigen Freiräume gegeben haben.

Auch möchte ich allen Danken die mir während dieser Zeit das notwendige Verständnis für die Arbeit entgegengebracht und mich stets unterstützt haben.

Koblenz, im Juli 2012

Kurzfassung

Zeitlich dicht aufeinander folgende Produktinnovationen, der hochdynamische Markt und die Hersteller von automotiven Produkten erfordern einen eng an die Kundenbedürfnisse angepassten Entwicklungsprozess. Sicherheitsanforderungen, Zuverlässigkeitsanforderungen und Kundenerwartungen an das endgültige System und dessen Komponenten steigen stetig. Das gleichermaßen gestiegene Bewusstsein für Sicherheit und Zuverlässigkeit verlangt spezielle effektive und effiziente Analyse- und Bewertungsmethoden sicherheitskritischer Systeme.

Die Bewertung von Sicherheit und Zuverlässigkeit basiert auf der Strukturbeschreibung eines Systems. Aufgrund des gestiegenen Anteils an mechatronischen Systemen im Fahrzeug, ist hier aus Entwicklungssicht eine domänenübergreifende Strukturbeschreibung notwendig, die die zuverlässigkeitstechnische Auswertung unterstützt.

Der in dieser Arbeit eingeführte Systembewertungsprozess beschreibt die generelle Vorgehensweise zur Bewertung von Systemen. Des Weiteren wird eine Systemmatrix zur formalen Beschreibung der technischen, funktionalen und zuverlässigkeitstechnischen Systemzusammenhänge definiert. Diese bildet die Grundlage für die Analyse und die Evaluierung von Zuverlässigkeitsnetzen. Die Analyse der Systemmatrix erfolgt unter der Verwendung von standardisierten zuverlässigkeitstechnischen Methoden. Diese Methoden und deren Anwendung werden durch die empirische Ausfallanalyse und die Pfadimportanzanalyse erweitert. Die quantifizierten Informationen der Systemmatrix werden durch die hier eingeführten Zuverlässigkeitsnetze beschrieben und in einen Systembaukasten zusammengefasst. Dieser Baukasten erlaubt bereits quantifizierte Teilsysteme unter Beibehaltung der Bewertung innerhalb des Bewertungsprozesses wiederzuverwenden.

Werden Systemanforderungen bezüglich der Qualitätsmerkmale (z. B. Zuverlässigkeit) nicht erfüllt, besteht die Möglichkeit, das System durch eine Optimierung an diese Anforderungen anzupassen. Diese Vorgehensweise wird anhand eines Entscheidungsprozesses dargestellt. Die Informationen über das System werden dem Optimierungsprozess über das Zuverlässigkeitsnetz bereitgestellt. Ergänzend werden neben der Strukturfunktion Vorschläge für die Optimierung übergeben.

In Kombination mit dem Optimierungsprozess ist es möglich wichtige Konzept-Entscheidungen frühzeitig mit einer hohen Sicherheit zu treffen. Durch die Anwendung des Systembewertungsprozesses wird eine Reproduzierbarkeit der Analyseergebnisse erreicht, was im Garantiefall den Nachweis unterstützt.

Abstract

Methodic Development of Highly Integrated Mechatronic Systems Considering Functional, Reliability and Safety aspects Analysis and Quantification

The automotive industry demands very rapid innovative products and evolutions of those products. Safety and reliability requirements and expectations upon the final system and components are increasing significantly. These in turn are driving the need to evolve more effective and efficient analysis techniques for safety critical systems.

To realize the assessment of the system safety and reliability a structural description of a system is essential. Due to the increased use of mechatronic systems in vehicles, descriptions considering all domains of the structure and supporting technical reliability evaluation are necessary.

The system evaluation process introduced in this paper describes the general procedure for the evaluation of systems. Furthermore a system matrix for the formal description of the technical, functional and reliability interrelations is introduced. This forms the basis for the analysis of the matrix and the development and evaluation of reliability networks. The analysis of the system matrix is made using standardized methods of reliability engineering. These methods and their applications are extended by the empirical failure analysis and path importance analysis. Reliability networks are used to describe the system structure and additional quantified information. They are summarized into a system construction set. This construction set allows the reuse of already quantified subsystems within the evaluation process.

If system requirements are not met (e.g. reliability), it is possible to adapt and optimize the system. This procedure is illustrated using a decision-making process. All information about the system is provided to the optimization process through the reliability network. In addition to structural information proposals for optimization are provided to the process.

In combination with the optimization process, important decisions at an early concept stage with a high security become possible. Through the application of the system evaluation process, reproducibility of the analytical results is achieved, which provides additional support in case of warranty verification.

Inhaltsverzeichnis

Verzeichnis der Abkürzungen und Formelzeichen

AADL ..... 'Architecture Analysis and Description Language'; Architektur Analyse- und Beschreibungssprache
ADL ..... Architecture Description Language
ASIC ..... Application Specific Integrated Circuit
ASIL ..... Automotive Safety Integrity Level
C ..... 'Controllabilitiy'; Beherrschbarkeit
CV ..... 'Concept Verification'; Konzeptverifikation
DV ..... 'Design Verification'; Designverifikation
E ..... 'Exposure'; Auftretenswahrscheinlichkeit
ECU ..... Electronic Control Unit (Steuergerät)
EOL ..... 'End of Life'; Außerbetriebnahme
EOP ..... 'End of Production'; Produktionsende
FMEA ..... Fehler Möglichkeits- und Einfluss- Analyse
FTA ..... 'Fault Tree Analysis'; Fehlerbaumanalyse
IEC ..... International Electrotechnical Commission
ISO ..... International Organization for Standardization
Kfz ..... Kraftfahrzeug
OEM ..... Original Equipment Manufacturer; Automobilhersteller
PV ..... 'Product and process Verification'; Produkt und Prozessverifikation
QM ..... 'Quality Management'; Qualitätsmanagement
RAMS ..... 'Reliability, Availability, Maintainability, Safety'; Zuverlässigkeit, Verfügbarkeit, Wartbarkeit und Sicherheit
RBD ..... 'Reliability Block Diagram'; Zuverlässigkeitsblockdiagramm
RN ..... Zuverlässigkeitsnetze
S ..... 'Serverity'; Schwere der Gefährdung
SOP ..... 'Start of Production'; Produktionsbeginn
SysML ..... System Modeling Language
UML ..... Unified Modeling Language
VDA ..... Verband der Automobilindustrie
$\phi(\underline{x})$ ..... Strukturfunktion
$\underline{x}$ ..... Zustandsvektor der Systemkomponenten
A ..... Auftretenswahrscheinlichkeit
ADN ..... Ausgezeichnete oder kanonische disjunktive Normalform
AKN ..... Ausgezeichnete oder kanonische konjunktive Normalform
B ..... Bedeutung
C_{Pf} ..... Menge der Minimalpfade - Erfolgspfade
C_{Sch} ..... Menge der Minimalschnitte - Ausfallschnitte
DN ..... Disjunktive Normalform
E ..... Entdeckungswahrscheinlichkeit
F ..... Ausfallwahrscheinlichkeit

I_{strukt} ..... strukturelle Importanz
KN ..... Konjunktive Normalform
p_i ..... Überlebenswahrscheinlichkeit der Komponente i
q_i ..... Ausfallwahrscheinlichkeit der Komponente i
R ..... Überlebenswahrscheinlichkeit
RPZ ..... Risikopriorität szahl
$y(x_1, \ldots, x_n)$. Boolesche Funktion

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einleitung

Gegenwärtige Trends in der Automobilindustrie, steigender internationaler Wettbewerb, variierende Umfeldbedingungen, höhere Sicherheitsansprüche, steigender Innovationsdruck sowie verkürzte Entwicklungszeiten, erfordern einen optimalen und auf den Produktbereich angepassten Produktentwicklungsprozess. Steigende Kundenerwartungen und zeitlich dicht aufeinanderfolgende Modellvariation erhöhen den Druck auf die Hersteller weiter. Mit steigendem Wettbewerb unter den Zulieferern der Automobilindustrie steigen die Anforderungen an die Rahmenbedingungen der Produkte (u. a. Kosten, Zeit, Qualität). Die wichtigsten Rahmenbedingungen für die Produktentwicklung werden durch

- höhere Komplexität,
- rechtliche Anforderungen,
- Verringerung von Fehlerkosten,
- geringere Entwicklungskosten,
- höhere Funktionalität,
- höhere Integrationsdichte,
- stärkere Vernetzung,
- kürzere Entwicklungszeiten und
- steigende Kundenerwartungen

dargestellt (in Anlehnung an [Pic09]). Die steigende Komplexität wird unter anderem durch die stärkere Vernetzung zwischen mechanischer und elektronischer Hardware (inkl. Software) beeinflusst. Diese Rahmenbedingungen wirken sich ebenfalls indirekt auf die Zuverlässigkeit von Systemen aus.

1.1 Motivation

Im Jahresbericht 2010 des Kraftfahrzeugbundesamtes [Kra10] wird auf die stetig wachsende Anzahl von Rückrufen aufgrund sicherheitskritischer Mängel[1] hingewiesen (Abbildung 1.1). Die meisten Mängel (60%) sind auf mechanische Mängel zurückzuführen. Elektrische/elektronische Mängel sind mit 27% die zweithäufigste Ursache für einen Rückruf. Diesen Mängeln werden auch abweichende Umsetzungen im Bereich der funktionalen Anforderungen zugeordnet. Aufgrund der kurzen Zeitspanne zwischen Projektstart und Markteinführung, aber dennoch in weiten Teilen seriell

[1] Mangel: In diesem Kontext beschreibt ein Mangel unter anderem die Abweichungen von einem erwartungskonformen Systemverhalten. Fehler, die aufgrund einer fehlerhaften Anforderungsumsetzung entstehen und somit zu einem nicht erwünschten Zustand eines Systems führen, werden auch Mängeln zugeordnet. Entsteht durch den Ausfall eines Systems eine sicherheitskritische Situation, ist dieses ebenfalls ein Mangel.

ablaufenden Entwicklungs- und Produktumsetzung, können Mängel häufig erst nach der Markteinführung entdeckt werden. Mängel die nach der Markteinführung oder sehr spät im Lebenszyklus entdeckt werden, sind aufgrund Rückrufaktionen oder hiermit verbundenen Entwicklungsverzögerungen mit hohen Kosten verbunden und generell zu vermeiden. Durch die Wahl geeigneter Methoden der Zuverlässigkeitstechnik ist die Vermeidung schon in der Entwicklungsphase möglich. Wurden früher geringe Mängel erst beim nächsten Service behoben, werden heute Mängel nicht mehr akzeptiert. Hierdurch wächst der öffentliche Druck auf die Hersteller und Lieferanten weiter und kann sich sogar im Absatz bemerkbar machen. Rückrufaktionen haben somit einen negativen Einfluss auf das Image des Herstellers und minimieren die Kundenzufriedenheit.

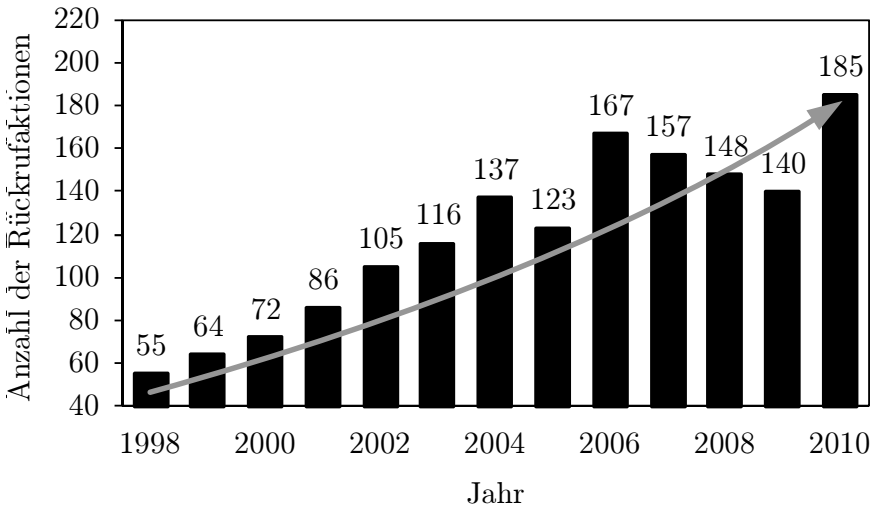

Abbildung 1.1: Rückrufaktionen für Kraftfahrzeuge nach [Kra10]

Aus Herstellersicht ist die Kundenzufriedenheit von entscheidender Bedeutung und bestimmt den Markterfolg. Zudem wird die Qualität des Produktes indirekt durch den Wettbewerb, den Zeit- und Kostendruck auf den Hersteller beeinflusst (verkürzte Entwicklungszeiten und die Forderung nach günstigen Produkten). Weil die Kosten stetig steigen, wird eine Kostenoptimierung von den Herstellern gefordert. Die Faktoren Kundenzufriedenheit und Kostendruck beeinflussen den Preis und die Qualität des Produktes zwar nur indirekt, werden jedoch direkt vom Markt wahrgenommen. Eine Einstufung der wichtigsten Qualitätsmerkmale wird in Abbildung 1.2 dargestellt. Es ist deutlich zu erkennen, dass die Qualitätsmerkmale sehr unterschiedlich gewichtet werden. Hat sich der Markt ursprünglich an Kosten und Funktionalität orientiert, so hat sich die Bedeutung von Zuverlässigkeit und Sicherheit in der Vergangenheit gewandelt (Abbildung 1.2). Bei der Entwicklung von sicherheitskritischen Produkten kann die Sicherheit als Alleinstellungsmerkmal für den Hersteller nützlich sein und für die Vermarktung ein wichtiges Erfolgskriterium darstellen. Dabei werden

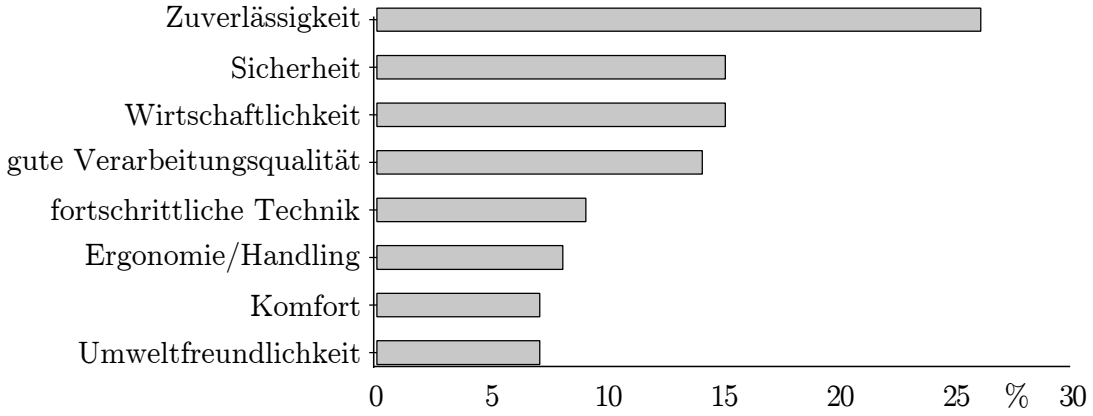

Abbildung 1.2: Einstufung von Qualitätsmerkmalen nach [VDA00]

irreführender Weise die Begriffe Sicherheit und Zuverlässigkeit häufig mit gleicher Bedeutung verwendet.

Für einen Automobilhersteller (OEM - „Original Equipment Manufacturer“) ist der Produkterfolg sehr stark von der Auswahl der Zulieferer abhängig. In [Voe99] werden die Qualität, der Preis, die Termintreue und die Lieferzeit als wichtige Kriterien für die Lieferantenauswahl genannt. Für den Zulieferer entsteht hierdurch ein enormer Druck, denn er wird dazu veranlasst, ein hochwertiges, günstiges Produkt zeitnah zu realisieren. Auch so haben sich im Laufe der Jahre die Entwicklungszeiten für ein Produkt wesentlich verkürzt.

Um Marktanforderungen gerecht zu werden, ist ein planvolles ökonomisches Handeln notwendig. Es werden entwicklungsbegleitende Methoden benötigt, die die Zieleinhaltung überwachen können. Für die Produktentwicklung können die bereits genannten Rahmenbedingungen

- hohe Funktionalität,
- geringe Kosten,
- geringe Entwicklungszeit,
- hohe Sicherheit,
- hohe Zuverlässigkeit und
- hohe Verfügbarkeit

als Ziele für ein wettbewerbsfähiges und letztlich markttaugliches Produkt festgelegt werden. Dieser Zusammenhang führt zu einem Konflikt, indem sich Kosten, Qualität und Zeit (Entwicklungszeit: Zeitspanne von der Idee bis zur Markteinführung) in ihrer Zielstellung scheinbar widersprechen. In diesem Konflikt gibt es keinen eindeutigen Favoriten, es muss methodisch nach einer Kompromisslösung gesucht werden.

Um den Anforderungen des Produktes gerecht zu werden und wettbewerbsfähig zu sein, ist es besonders wichtig, Qualitäts- und Zuverlässigkeitsmethoden entwicklungsbegleitend in den Produktentstehungsprozess zu integrieren. Eine kosten- und

zeitneutrale Integration bei stetiger Überwachung der Sicherheitsziele ist erforderlich. Hierdurch kann die Einhaltung der Ziele bereits in frühen und in den weiterführenden Phasen, bis hin zum Serienprodukt, bewertet und überwacht sowie der oben genannte Konflikt gelöst werden.

Technische Systeme im Bereich der Kraftfahrzeugtechnik zeichnen sich aktuell durch eine steigende Komplexität und eine zunehmende Vernetzung zwischen Elektronik, Informatik und Mechanik aus. Durch Innovationen und die gestiegenen Anforderungen an ein System werden Elektronik und Informatik in diesem Umfeld immer bedeutender. Systeme, die früher hauptsächlich aus mechanischen Komponenten bestanden, werden immer häufiger durch integrierte Systeme verdrängt. Begünstigt durch den Preisverfall im Bereich der Elektronik und einem hohen Entwicklungsgrad in der Informationstechnik zeichnet sich ein Trend in Richtung by-wire Systeme (u. a. steer-by-wire, brake-by-wire) ab. Bei diesen Systemen wird auf die durchgehende konventionelle mechanische Verbindung/Übertragung/Komponenten verzichtet und partiell durch elektrische Verbindungen/Komponenten ersetzt. Sensoren erfassen den Fahrerwunsch, den Systemzustand und die Systemänderungen und leiten diesen über weitere Komponenten zu den Aktoren des Systems. Hierdurch gewinnen Elektronik, Sensorik und Software von Systemen immer mehr an Bedeutung.

Bestand ein Kraftfahrzeug früher hauptsächlich aus mechanischen Komponenten, ist durch die Einführung von Sicherheits-, Komfort- und Optimierungsfunktionen der Anteil an Elektronik im Kraftfahrzeug angestiegen. In diesem Zusammenhang wurden zunehmend Funktionen zur Unterstützung und Überwachung des Fahrers, sowie zur Steigerung der Sicherheit integriert. Die Erweiterung der Funktionalität inklusive der Sicherheitsanforderung steht jedoch in einem engen Zusammenhang mit der Komplexität des Systems, denn mit wachsender Funktionalität (automatische Systeme) erhöht sich deren Komplexität. Beispielsweise werden in Kraftfahrzeugen zunehmend Assistenzsysteme oder autonom agierende Systeme zur Steigerung der Sicherheit integriert. Neue Funktionen unterstützen den Fahrer und erhöhen die Sicherheit des Fahrzeugs in der Umgebung, in der es eingesetzt wird. Dieser Zusammenhang kann anhand von brake-by-wire Systemen verdeutlicht werden. Konventionelle Bremssysteme werden rein mechanisch betätigt und sind zuverlässigkeitstechnisch bedingt meist überdimensioniert und sind somit robust. Durch die Integration von Elektronik wird im ersten Schritt auf die mechanische Kopplung zwischen Bedienelement und Bremsmechanik verzichtet. Im zweiten Schritt werden hydraulische und pneumatische Verbindungen durch elektrische ersetzt und somit ein reines by-wire-System realisiert. Obwohl die rein elektrische Kopplung zwischen den Komponenten und der Bremsmechanik mehr Anforderungen an Sicherheit und Zuverlässigkeit stellt, ist eine Steigerung von Funktionalität und Sicherheit möglich.

Die Anforderungskriterien für die Markteinführung eines Produktes stehen in engem Zusammenhang mit der Komplexität und Funktionalität des Systems. Diese lassen sich in notwendige und hinreichende Kriterien aufteilen. Notwendige Kriterien

werden vom Gesetzgeber, dem Kunden[2] oder dem Unternehmen selbst vorgegeben. Beispielsweise definiert der Gesetzgeber die erwartete Lebensdauer des Systems und die Sicherheitsanforderungen (SIL - Safety Integrity Level), die an das System gestellt werden. Hinreichende Kriterien sind nicht zwingend für den Systembetrieb und die Sicherheitserfüllung notwendig oder lassen sich über andere Anforderungen abdecken; zu diesen lassen sich die kundenspezifischen Erweiterungen zuordnen. Die Systemanforderungen setzen sich aus Mindestanforderungen, funktionalen Anforderungen und Sicherheitsanforderungen zusammen. Sicherheitsanforderungen zählen zu den Mindestanforderungen und sind bindend. Die funktionalen Anforderungen werden häufig den Sicherheitsanforderungen untergeordnet. Mit der Komplexität des Systems und den steigenden Anforderungen, nimmt der Aufwand bei der Umsetzung und beim Nachweis der Erfüllung der Anforderungen, insbesondere der Sicherheitsanforderungen, zu. Sicherheitskritische Systeme können bei Ausfall eine potentielle Gefahrenquelle für den Nutzer und die Umwelt darstellen. Der Gesetzgeber erlaubt keine Ausfälle, die zu einer Sicherheitsgefährdung führen können. Diese notwendigen Anforderungen sind ein Grund dafür, dass bei sicherheitskritischen mechatronischen Systemen die Sicherheit, insbesondere die Ausfallsicherheit, durch eine mechanische Rückfallebene gesteigert wird. Bei der elektrischen Lenkung wurde aufgrund der Verfügbarkeitsanforderungen bisher stets eine mechanische Kopplung als Rückfallebene implementiert.

Bei industriellen Entwicklungsprozessen sind die Gesamtkosten ein weiteres wichtiges Entscheidungskriterium für die Systemauswahl. Eine Optimierung bezüglich der Kosten kann jedoch leicht zu einem Konflikt mit der Komplexität und der Zuverlässigkeit des Systems führen. Daher dürfen die Kosten nur unter Berücksichtigung der sicherheitsrelevanten Anforderungen optimiert werden. Die Zuverlässigkeit kann durch die Variation der Komponenten oder der Systemarchitektur[3] - unter Berücksichtigung der Komplexität und der Kosten - optimiert werden. Im Rahmen der zuverlässigkeitstechnischen Strukturoptimierung kann es beispielsweise günstiger sein, redundante Komponenten statt einer teuren Einzelkomponente in ein System zu integrieren.

[2]Kunde: Der Kunde kann im Allgemeinen als Auftraggeber oder Endverbraucher für ein Produkt gesehen werden. Für Automobilhersteller sind dies die Käufer der Fahrzeuge. In der Zulieferindustrie sind unter Kunde der Automobilhersteller sowie die Endverbraucher zu sehen.

[3]Der Begriff Architektur wird synonym mit dem Begriff Struktur verwendet. Aufgrund der Begriffserweiterung aus der Informatik, bezieht sich die Architektur auf die Struktur informationstechnischer Systeme und beschreibt die Zusammensetzung und Anordnung der Komponenten und das Zusammenwirken dieser. Der Begriff wurde weiter verallgemeinert und auf geplante komplexe Strukturen und deren Konzeption (Entwurf) übertragen.

1.2 Ziele und Aufbau der Arbeit

Der Nachweis von Sicherheit, Zuverlässigkeit und Verfügbarkeit ist für die Etablierung im Markt, die Zulassung durch den Gesetzgeber und die Akzeptanz der Versicherer wichtig. Zur Bewertung dieser Kriterien, unter der Berücksichtigung von Kosten[4], muss das zu entwickelnde System analysiert werden. Eine Beschreibung der Systemstruktur ist notwendig und darauf aufbauend können die geforderten Kriterien bewertet werden.

Ziel der vorliegenden Arbeit ist die Unterstützung des Entwicklungsprozesses von der frühen Entwicklungsphase (Innovation/Idee) bis zum Beginn der Produktion. Dies soll durch einen transparenten Ansatz realisiert werden. Zu Beginn wird eine abstrakte Architektur definiert und im weiteren Verlauf des Prozesses detailliert. Die automatische zuverlässigkeitsorientierte Bewertung während der Entwicklungsphase ist sehr stark von der vorhandenen Information über das System abhängig. Für die Beschreibung des Gesamtsystems wird das System solange in Teilsysteme aufgeteilt, bis sogenannte „atomare Einheiten“[5] vorhanden sind. Diese Einheiten werden dann durch die Anwendung des Quantifizierungsprozesses bewertet. Durch die kontinuierliche Durchführung des Prozesses werden immer mehr Daten über die verwendeten Standardkomponenten gewonnen und es entsteht ein Baukasten (Kapitel 5). Darin sind alle Merkmale enthalten, die für die Qualitäts- und Eigenschaftssicherung des Systems notwendig sind. Die Wiederverwendung von Informationen über bereits quantifizierte Teilsysteme oder Gesamtsysteme wird durch den Baukasten ermöglicht. Die Gesamtsysteme lassen sich aus bekannten, bewerteten Teilsystemen zusammensetzen.

Die hier vorgestellte Methode unterstützt den Entwickler durch die Analyse und Bewertung von Zuverlässigkeitskenngrößen während der Entwicklungsphase und darüber hinaus. Durch Anwendung dieser Methode können

- Zuverlässigkeitskenngrößen,
- Vorlagen für Fehlerbäume,
- Zuverlässigkeitsblockdiagramme,
- Systemstrukturinformationen,
- Optimierungspotential im System sowie
- Gegenüberstellung der Entscheidungsmerkmale

für weitere Analysen zur Verfügung gestellt werden.

[4] Die Kosten werden in diesem Fall durch die Anforderungen an das System beeinflusst. Zuverlässigkeits-, Verfügbarkeits-, Kunden- und Sicherheitsanforderungen sind unter anderem Faktoren, die bei der Optimierung zu berücksichtigen sind und die Kosten beeinflussen.

[5] „atomare Einheiten“: Untersysteme (Module - Funktionale Einheiten) die nicht weiter in ihrer Funktionalität aufgeteilt werden können. Die Aufteilung erfolgt unter zuverlässigkeitstechnischen Gesichtspunkten und ist unabhängig von physikalischen oder informatorischen Merkmalen.

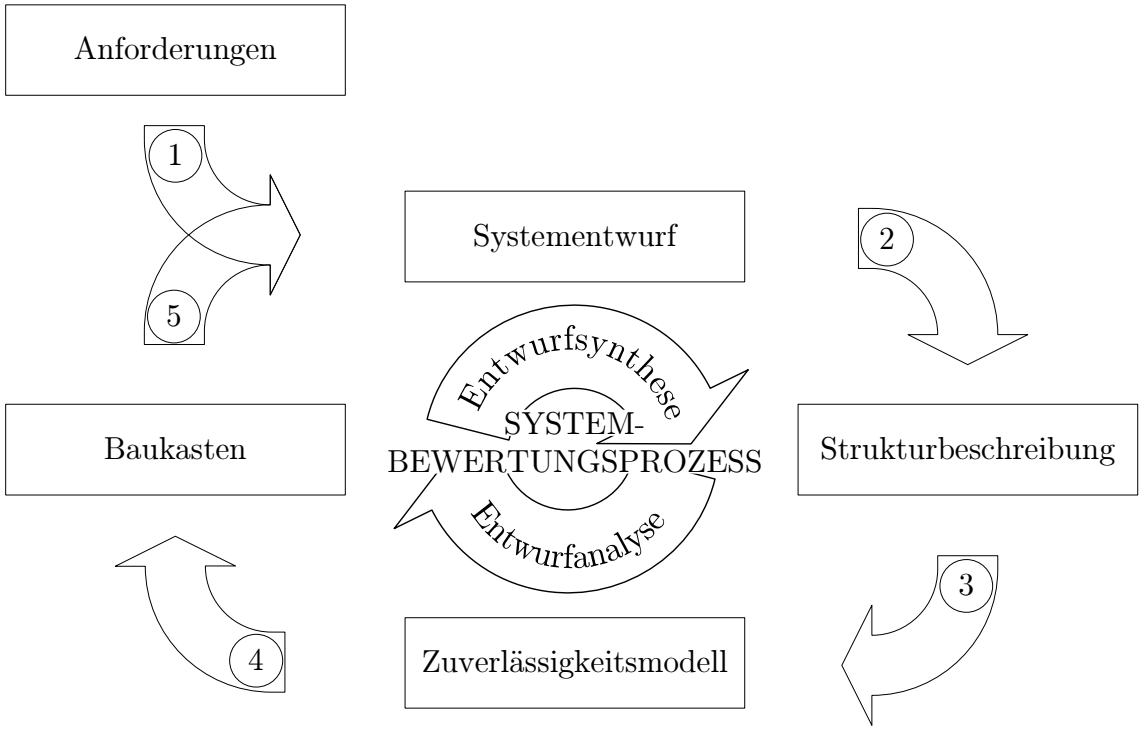

Abbildung 1.3: Systembewertungsprozess (Struktur Hauptteil)
(1) Anforderungsanalyse, (2) Strukturanalyse, (3) Modellbildung, (4) Quantifizierung, (5) Systemkomposition

Im folgenden Kapitel wird der aktuelle Stand der Wissenschaft und die Grundlagen für die Beschreibung von Systemstrukturen und die Bewertung von Systemzuverlässigkeit erläutert (Kapitel 2). Der Produktlebenszyklus innerhalb der Automobilindustrie und der Begriff Mechatronik sowie seine Bedeutung in der Automobilindustrie werden näher betrachtet. Die Vorgehensweise zur Bewertung von Zuverlässigkeit und Sicherheit in Bezug auf mechatronische Systeme wird eingeführt. Abschließend wird die Norm ISO 26262 betrachtet und ihre Anwendung im Bereich der Automobilindustrie beschrieben.

Den Hauptteil dieser Arbeit bildet die Analyse und Quantifizierung von mechatronischen Systemen. Dabei handelt es sich um einen neuen Ansatz zur formalen Beschreibung von Systemstrukturen. Hierzu wird ein Systembewertungsprozess, der die generelle Vorgehensweise zur Analyse und Quantifizierung von Systemen beschreibt, eingeführt (Kapitel 3). Zur Beschreibung der Systemstruktur wird eine Systemmatrix (Kapitel 4) verwendet. Diese Matrix bildet die Grundlage für die Quantifizierung des Systems. Zur Bewertung von Qualitätsmerkmalen ist es notwendig, die Matrix in geeignete Modelle zu überführen. Zuverlässigkeitsmodelle erlauben unter Anwendung von standardisierten Methoden der Zuverlässigkeitstechnik die Quantifizierung der Zuverlässigkeitsmerkmale. Die Auswertung der Systemmatrix

liefert wichtige Informationen über die Merkmale des Systems und seiner Komponenten. In diesem Zusammenhang werden neue Methoden zur Strukturbewertung des Systems eingeführt.

Um eine hohe Verfügbarkeit der Informationen zu erhalten, werden die quantifizierten Merkmale in einem Zuverlässigkeitsnetz abgelegt und in einer Wissensdatenbank (Systembaukasten) zusammengefasst (Kapitel 5). Mit Hilfe dieser Wissensdatenbank besteht die Möglichkeit, während der Produktspezifikation auf bereits bekannte, bewährte und bewertete Systeme sowie Teilsysteme zurückzugreifen. Der Systembaukasten und seine Anwendung zur Gegenüberstellung von verschiedenen Systemen bildet die Grundlage für die Systemvariation und den Optimierungsprozess. Ein durchgängiger Systemquantifizierungs- und Optimierungsprozess erläutert die Schnittstellen zwischen der Optimierung und der Systemanalyse. Die Systemanalyse und -bewertung liefern die Grundlage und bilden die Schnittstelle für den Optimierungsprozess. Während der Systemanalyse können die für die Optimierung geeigneten Komponenten identifiziert und charakterisiert werden. Die Optimierung kann für die Struktur des Gesamtsystems, die Struktur der Teilsysteme und einzelne Komponenten durchgeführt werden. Dem Optimierungsprozess folgt stets der Systemquantifizierungsprozess, hierbei wird die Systemstruktur erneut analysiert und in der Wissensdatenbank abgelegt.

Das hieran anschließende Kapitel 6 erläutert die Erstellung und Auswertung von Systemmatrizen anhand von einigen ausgewählten Beispielen aus der Automobilindustrie. Hier wird die grundlegende Vorgehensweise der Systemstrukturanalyse, der Quantifizierung, der Modularisierung und Darstellung von Zuverlässigkeitsnetzen verdeutlicht.

Abschließend werden die wesentlichen Bausteine dieser Arbeit dargestellt und mögliche Erweiterungen des Ansatzes vorgeschlagen.

2 Stand der Wissenschaft

Qualität und Kosten eines Produktes entscheiden über die Akzeptanz am Markt und den Erfolg des Produktes. Der Wettbewerb und die gestiegenen Anforderungen an technische Systeme erfordern Methoden, die eine Bewertung der Qualität schon in frühen Phasen des Lebenszyklus ermöglichen. Die Qualität setzt sich aus unterschiedlichen Merkmalen zusammen und wird zur Bewertung des Produktes genutzt. Zuverlässigkeit und Sicherheit sind Qualitätsmerkmale, die in den letzten Jahren immer mehr in den Vordergrund getreten sind.

Die Entwicklung und Integration mechatronischer Systeme im Automobil begann im Jahr 1979 [VDA00] mit der ersten Generation elektronischer Antiblockiersysteme. Aufgrund des großen Potentials mechatronischer Systeme werden seitdem immer mehr Teilsysteme durch mechatronische Systeme ersetzt. Dies führte zu einer stetigen Verbesserung der Leistungsfähigkeit, Funktionalität und Sicherheit. Durch das Zusammenwirken der Domänen Mechanik, Elektronik und Informationstechnik wurde es möglich, den immer strenger werdenden Anforderungen an sicherheitskritische Systeme gerecht zu werden.

Mit der Einführung der Mechatronik ergeben sich für die Realisierung von Systemen neue Möglichkeiten, die innerhalb der einzelnen Domänen nicht umsetzbar sind. Allerdings entstehen hierdurch auch besondere Anforderungen an den Entwicklungsprozess [VDI04]. Mechatronische Systeme besitzen eine hohe Komplexität, welche durch die große Anzahl von gekoppelten Elementen, die Vernetzung der unterschiedlichen Domänen und die erhöhte Funktionalität entsteht. Neben der Komplexität ist die Entwicklung von mechatronischen Systemen durch verringerte Produkt- und Entwicklungskosten bestimmt.

Durch das Zusammenwirken von Informatik und Elektronik ist es möglich, programmierbare Komponenten zu realisieren. Diese werden häufig in der Mechatronik eingesetzt und vereinfachen die Anpassung an variierende Anforderungen. Die einfache Integration und Änderung von Funktionen, die sich durch Software abbilden lassen, wird möglich. Das System wird hierdurch skalierbar und Kundenwünsche lassen sich teilweise ohne Hardwareänderungen umsetzen. Auch für die Realisierung von fehlertoleranten Systemen sind programmierbare Komponenten vorteilhaft, da sich Notlaufkonzepte einfacher integrieren lassen. Jedoch wird die Struktur der Systeme komplexer und innerhalb der Komponenten lässt diese sich nicht mehr eindeutig erkennen. Aus diesem Grund ist die Analyse, insbesondere die Zuverlässigkeitsanalyse, sehr schwierig.

Im Folgenden werden die oben beschriebenen Zusammenhänge in Bezug auf die Automobilindustrie näher dargestellt und die Schwachpunkte der standardisiert verwendeten Methoden aufgezeigt. Hierzu wird der allgemeingültige Produktlebenszyklus automobiler Produkte betrachtet und die Bedeutung der Quantifizierung von Qualitätsmerkmalen dargestellt. Anschließend wird der Begriff Mechatronik eingeführt

und die Methoden zur Bewertung der Zuverlässigkeit erläutert. Die Schwierigkeiten bei der Bewertung und Analyse von Systemen werden aufgezeigt.

2.1 Produktlebenszyklus kraftfahrzeugtechnischer Systeme

Die Idee für eine Produktveränderung, ein neues Produkt oder eine Funktionserweiterung ist in der Regel der Beginn für einen häufig erfolgreichen Produktlebenszyklus. Ebenso können Marktbedürfnisse, zu lösende Probleme, Systemvariationen oder Ähnliches einen solchen Zyklus initiieren. Der gesamte Lebenszyklus beinhaltet den Produktentstehungsprozess, den Produktionsprozess, den Betrieb und die Produktrückführung. Der Produktentstehungsprozess beginnt häufig mit einer Innovationsidee (Abbildung 2.1) und beinhaltet die Produktentwicklung. Für die weitere Betrachtung steht der Produktlebenszyklus der Automobilindustrie im Vordergrund (Abbildung 2.2). Hier sind die Abhängigkeiten der Reifegrade des Produktes und die äußeren Einflüsse auf die Produktentwicklung zu erkennen [PBFG07]. Während der Produktentwicklung ist stets der gesamte Lebenszyklus zu berücksichtigen. Selbst die Anforderungen die an die Entsorgung gestellt werden müssen bei der Planung mit einbezogen werden. Eine kontinuierliche Produktüberwachung ist empfehlenswert und wird aufgrund der Qualitätssicherung verlangt. Während der letzten Jahre der Produktentwicklung wurde hier der Produktlebenszyklus insgesamt [GW91] und daher die Entwicklungszeit immer weiter verkürzt. Die Entwicklung eines neuen Fahrzeugs dauert zurzeit etwa 3-4 Jahre. Diese Dauer hat sich in den letzten Jahrzehnten über die Hälfte verkürzt (z. B. 1970: 8 Jahre). Zudem wird zusätzlich alle 1,5 bis 2 Jahre eine Modellpflege durchgeführt, die eine zusätzliche Herausforderung für Zulieferer und Hersteller gleichermaßen darstellt.

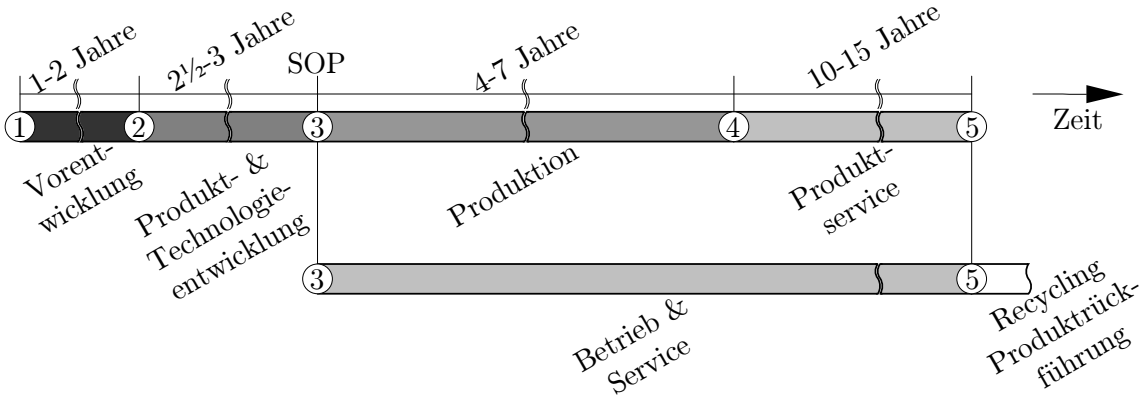

Abbildung 2.1: Vereinfachte Darstellung des Produktlebenszyklus
(1) Idee, (2) Projektstart, (3) Produktionsstart, (4) Produktionsende, (5) Außerbetriebnahme

Das Bedürfnis der Hersteller, mit neuen Modellen schneller auf dem Markt zu sein als die Konkurrenz, könnte in der Zukunft die Entwicklungszeiten weiter verkürzen

(vgl. [Bor10]). Hierdurch wird die Amortisierungsdauer verkürzt, was indirekt einen negativen Einfluss auf die Kosten zur Folge hat.

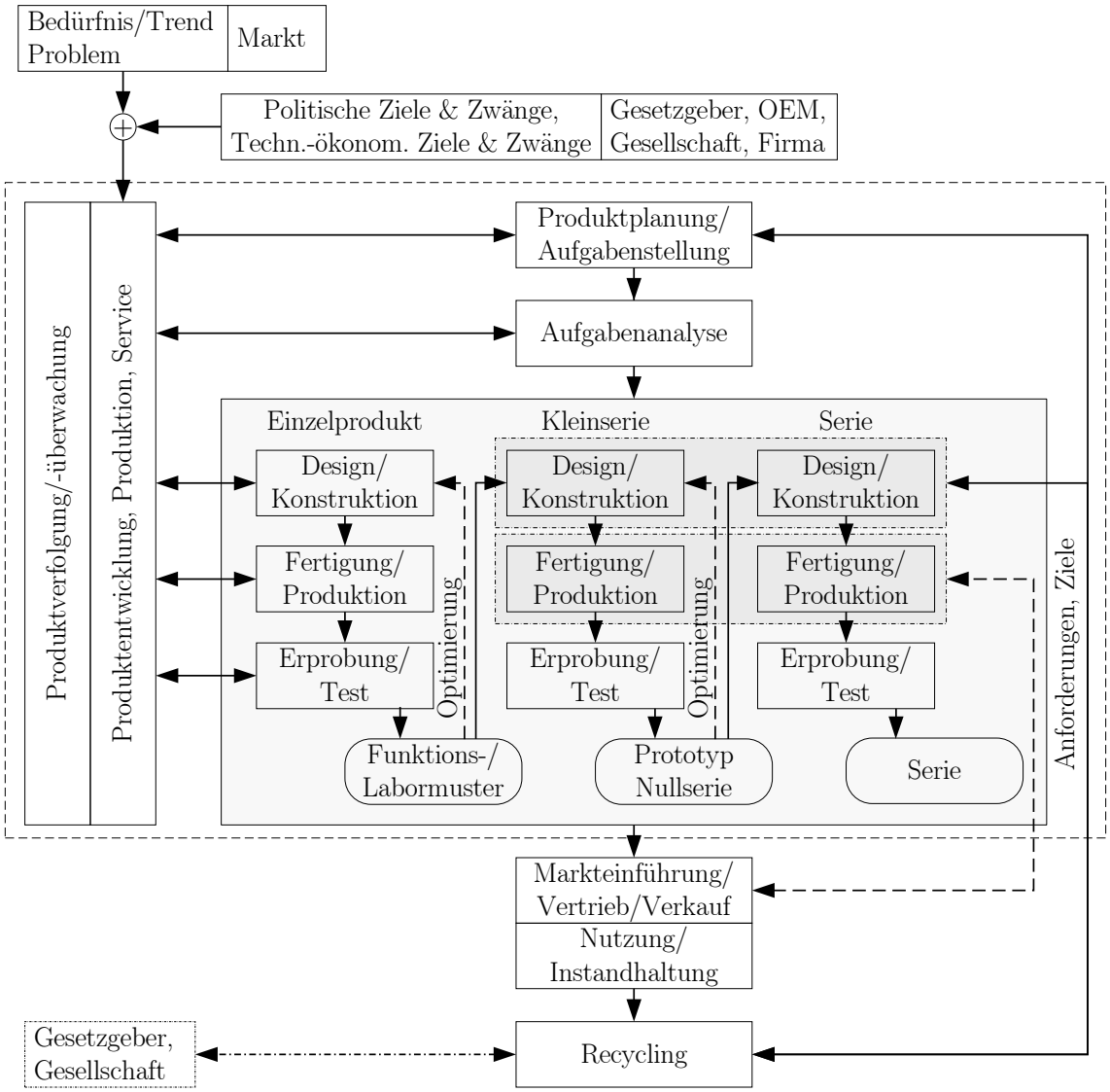

Abbildung 2.2: Produktlebenszyklus in der Automobilindustrie

Der globale Wettbewerb ist geprägt von hoher Innovationsgeschwindigkeit, verkürzten Entwicklungs- und Produktionszyklen, sowie steigenden Anforderungen und Erwartungen an Qualität, Preis und Leistung (Magisches Dreieck[1]). Produktinnovationen

[1]Magisches Dreieck: Ein magisches Vieleck veranschaulicht qualitativ alle betrachteten konkurrierenden Ziele. Für das hier betrachtete magische Dreieck wurden die Kosten, Zeit und Qualität als Ziele festgelegt. Diese Ziele bedingen sich gegenseitig und sind zumeist gegenläufig. Es ist schwierig ein Optimum über alle Einflussfaktoren zu bestimmen. Um dieses Problem zu umgehen, kann eine Priorisierung vorgenommen werden. Ein Netzdiagramm bietet die Möglichkeit, die Ziele des magischen Vielecks quantitativ darzustellen.

tragen in entscheidender Weise dazu bei, um sich in diesem globalen Wettbewerb zu behaupten.

Der enorme Kostendruck führt in der Automobilindustrie in Verbindung mit hohen Stückzahlen dazu, dass die proportionalen Herstellungskosten häufig den Fahrzeugpreis dominieren. Ein weiteres Problem in der Fahrzeugindustrie ist der lange Produktlebenszyklus. Die Entwicklungszeit von maximal drei Jahren, ein Produktionszeitraum von etwa sieben Jahren und eine abschließende Betriebs- und Servicephase von 10 bis 15 Jahren ergibt einen Produktlebenszyklus von ca. 25 Jahren (Abbildung 2.1). Für Zulieferer innerhalb der Automobilindustrie bedeutet dies eine Servicebereitstellung und evtl. Bevorratung von Bauelementen/Steuergeräten von 15 Jahren nach Einstellung der Produktion bzw. Rückwärtskompatibilität.

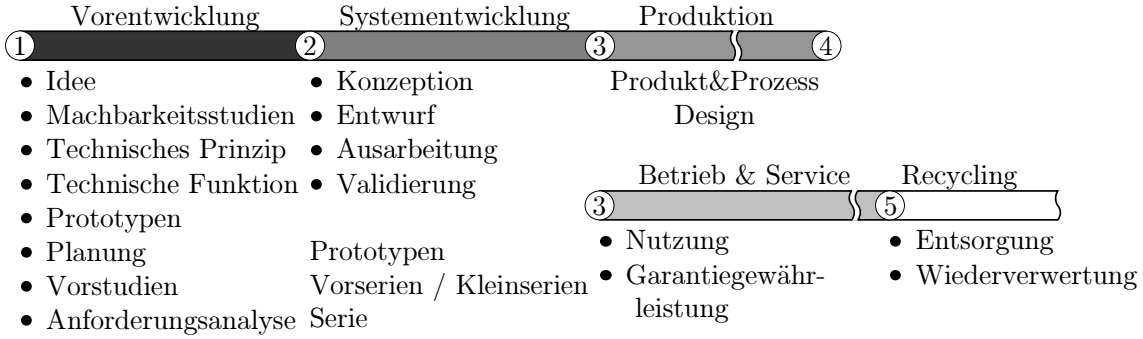

Abbildung 2.3: Produktlebenszyklus im Detail
(1) Idee, (2) Projektstart, (3) Produktionsstart, (4) Produktionsende, (5) Außerbetriebnahme

Gordon Moore formulierte 1965 das nach ihm benannte Moorsche Gesetz, das die Komplexitätszunahme in Abhängigkeit der Zeit beschreibt. Es prognostiziert eine Verdopplung der Integrationsdichte von Schaltkreisen auf Substratflächen innerhalb von ca. 20 Monaten. Aufgrund dieser bis heute anhaltenden Technologiefortschritte in der Hardware sind die Intervalle in der Elektronikentwicklung heute wesentlich kürzer als vor einigen Jahren. Dies stellt beispielsweise die langfristige Versorgung des Marktes mit Elektronikersatzteilen vor große Herausforderungen und muss bereits während der Entwicklungsphase mit berücksichtigt werden (vgl. [SZ10]).

In diesem Zusammenhang sind in Abbildung 2.4 die einzelnen Phasen des Produktlebenszyklus im Detail dargestellt. Für den Hersteller ist jedoch nach der Produktion des Produktes das Ende für das Produkt noch lange nicht erreicht. Für ihn beginnt mit der Markteinführung eine Gewährleistungszeit. In dieser muss sichergestellt sein, dass das Produkt sicher funktioniert und instand gesetzt werden kann. Der Lebenszyklus des Produktes endet somit für den Hersteller eigentlich erst nach der Gewährleistungsphase oder der Nutzungsphase.

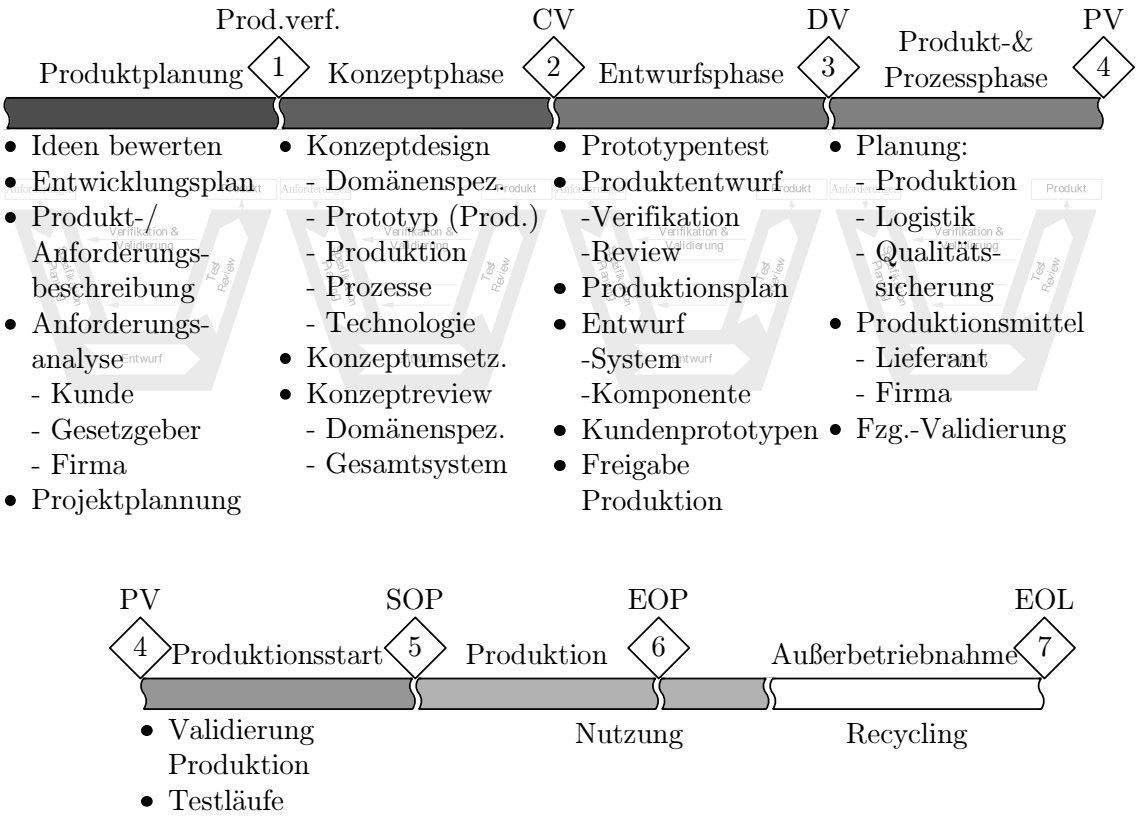

Abbildung 2.4: Die einzelnen Phasen im Produktlebenszyklus mit
1: Produktverifikation,
2: Konzeptverifikation (CV),
3: Designverifikation (DV),
4: Produkt und Prozessverifikation (PV),
5: Produktionsstart („Start Of Production“; SOP),
6: Produktzyklusende („End of Produktion“; EOP),
7: Außerbetriebnahme („End of Life“; EOL).

Wird der Produktlebenszyklus in Bezug auf den Reifegrad[2] betrachtet, ist ein steigender Reifegrad des Produktes in Abhängigkeit von der Lebenszyklusphase des Produktes erkennbar (Abbildung 2.5). Die Möglichkeit auf Änderungswünsche des gesamten Produktes zu reagieren, besteht bis zum Beginn der Produktion. Nach dem Produktionsstart ist eine Anpassung nur noch im Bereich der Software möglich. Diese Tatsache stellt für die Produktentwicklung, insbesondere der Hardwareentwicklung, besondere Anforderungen an den Entwicklungsprozess, denn in diesen Domänen ist eine Konzeptentscheidung in frühen Entwicklungsphasen wichtig und sinnvoll. Die

[2]Der Reifegrad bezieht sich hier auf den Umsetzungsgrad der globalen Anforderungen während der Entwicklung. Je fortgeschrittener der Lebenszyklus, desto mehr Anforderungen wurden umgesetzt. Änderungen lassen sich in späten Phasen schwieriger integrieren, da zu diesem Zeitpunkt die Hardware des Systems weitestgehend definiert ist.

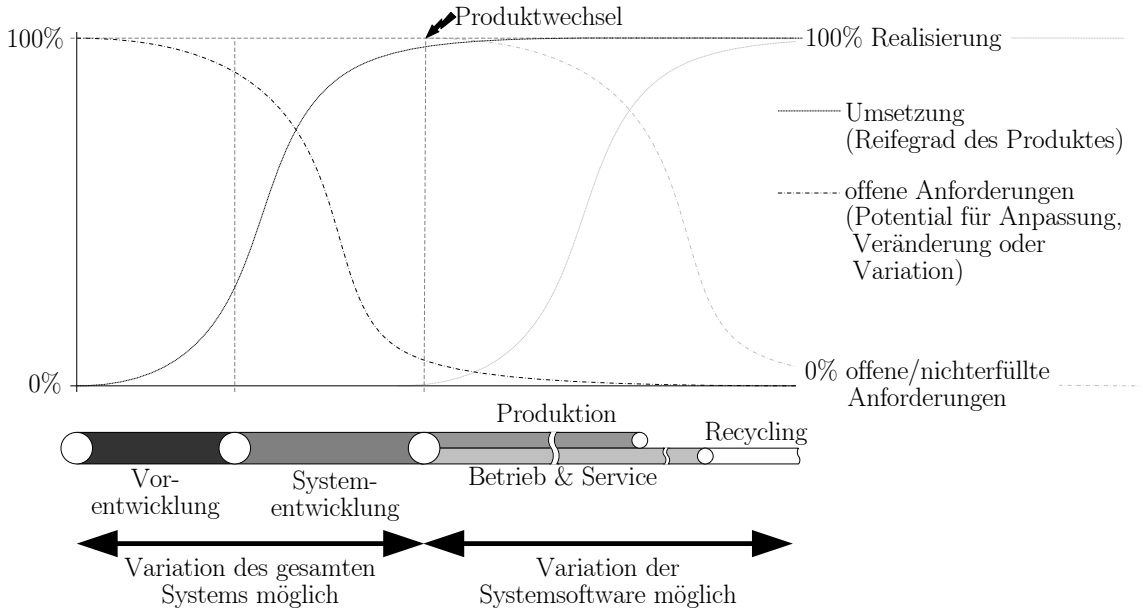

Abbildung 2.5: Änderungspotential und Umsetzungsgrad in Abhängigkeit von der Produktphase

Kosten für Anpassungen eines Produktes im frühen Produktentwicklungsprozess sind relativ gering, verglichen mit sehr späten Produktveränderungen. Diese Kostenzunahme begründet sich mit einer Komplexitätssteigerung bei der Implementierung von Änderungen innerhalb einer späten Phase. Änderungen innerhalb benachbarter Systeme, notwendige Fertigungsumstellungen oder erneute Freigaben können die Ursachen für ergänzende Kosten sein. Dieser Sachverhalt begründet auch die Bedeutung von frühen Konzeptbewertungen während der Entwicklung. Wird der Sachverhalt in frühen Entwicklungsphasen in Bezug auf die Produktlebenszeit beachtet, bietet sich durch geschickte Auslegung der Hardwarestruktur die Möglichkeit Defizite, die durch die Anpassungsproblematik der Hardware entstehen, durch die Einführung von virtuellen Sensoren zu minimieren. Virtuelle Sensoren basieren auf der Datenerfassung der vorhandenen Hardware, werten die erfassten Daten jedoch nach anderen Verfahren aus. Virtuelle Sensoren können auch durch die Kombination von unterschiedlichen erfassten physikalischen Messwerten generiert werden. Die Realisierung von virtuellen Sensoren bietet die Möglichkeit, ohne Hardwareänderungen eine Redundanz in das System zu integrieren. Diese Redundanz ist nur eine virtuelle Redundanz. Virtuelle Sensoren können auch zur Überwachung des Systems genutzt werden und bieten neue Diagnosemöglichkeiten. Mechatronische Systeme liefern die Grundlage für virtuelle Sensoren. In rein mechanischen Systemen können diese Sensoren nicht integriert werden, da die informationsverarbeitenden Komponenten fehlen.

2.2 Mechatronik im Kraftfahrzeug

Mechanik, ehedem „die Domäne“ im Automobil, ist in heutigen Fahrzeugen durch einen sehr hohen Anteil an elektronischen Systemen, die den Fahrer unterstützen oder völlig autonom arbeiten, ersetzt worden. Hierzu ist die Mechanik eine regelrechte Symbiose mit der Elektronik und Informatik eingegangen, was diese Systeme, aufgrund ihres Funktionsumfangs, häufig sehr komplex werden lässt. Die technischen Zusammenhänge zwischen den Systemkomponenten, dem Menschen und der Umwelt sind in Abbildung 2.6 dargestellt. Bei Betrachtung eines Automobils wird der Mensch als Fahrer dargestellt. Eine Energieversorgung wird als Komponente innerhalb des mechatronischen Systems integriert. Das System kann durch den Fahrer oder die Umwelt beeinflusst werden. Rückmeldungen an den Fahrer werden über die Umwelt oder das Fahrzeug gegeben. Die meisten Informationen sind jedoch über die Umwelt mit dem Fahrer verknüpft.

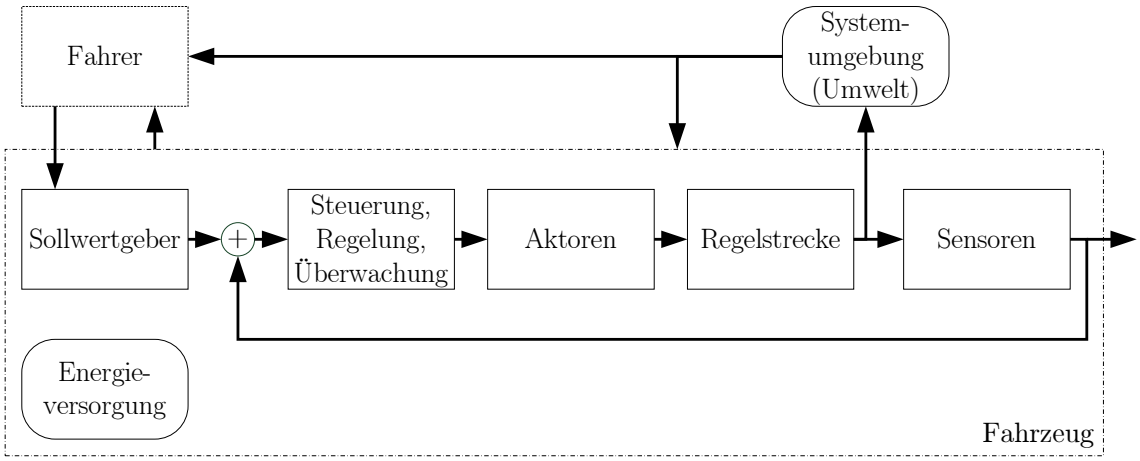

Abbildung 2.6: System-Mensch-Umgebung (vgl. [SZ10])

Allgemein werden mechatronische Systeme durch die in Abbildung 2.7 gezeigte Grundstruktur beschrieben. Grundsystem, Sensoren, Aktoren und die Informationsverarbeitung bilden die wesentlichen Bestandteile dieser Struktur. Deren Randbedingungen von der Betriebsumgebung vorgegeben werden. Der Mensch und die Kommunikationssysteme sind optionale Elemente der Darstellung und nicht immer notwendig. Nach [Mar04] sind die Elemente in der Black-Box-Darstellung mittels Energie-, Stoff-, Informationsflüssen miteinander verknüpft. Das Grundsystem ist ein technologieunabhängiges physikalisches System, es kann z. B. aus mechanischen, elektromechanischen, hydraulischen oder pneumatischen Strukturen bestehen. Hier sind auch Mischformen der Strukturen denkbar. Das Grundsystem ist über Energie- und Stoffflüsse über die Systemgrenzen hinaus mit anderen Systemen, anderen mechatronischen (Teil-)Systemen oder auch anderen Grundsystemen verbunden. Die Sensoren ermitteln - meist analog - die notwendigen Zustandsgrößen und Zustandsveränderungen des Grundsystems, der Umgebung oder des Benutzers, in der das System

eingesetzt wird. Diese können als konventionelle Messwertaufnehmer physikalisch real vorhanden sein oder durch Software-Sensoren (virtuelle Sensoren) implementiert werden. Diese Zustandsgrößen bilden die Eingangsgrößen der Informationsverarbeitungseinheit. Nach der Erfassung der Messwerte werden die Informationen in der Regel rein digital weiterverarbeitet. Mikroprozessoren verarbeiten die Eingangsgrößen zeit- und wertdiskret. Hierdurch wird auch die analoge oder analog/digitale (A/D) Verarbeitung der Signale mittels Elektronik ermöglicht. „Intelligente Sensoren“ liefern zudem bereits gefilterte und A/D gewandelte Signale. Die Verarbeitungseinheit bestimmt aufgrund der Zustandsgrößen am Eingang die Zustandsgrößen für die gewünschte Manipulation des Grundsystems. Optional ist diese zum Datenaustausch über ein Kommunikationssystem mit anderen logischen Einheiten verbunden. Über die Mensch-Maschine-Schnittstelle kann dem Benutzer die Möglichkeit zum Informationsaustausch und zur Interaktion mit dem System gegeben werden. Die Umsetzungen der von der Informationsverarbeitung bestimmten Einwirkungen erfolgen durch die Aktoren direkt am Grundsystem. Die Leistungsversorgung der Systemelemente kann über die Systemgrenzen hinaus durch externe Energiequellen, wie dargestellt, oder auch intern durch das Grundsystem erfolgen.

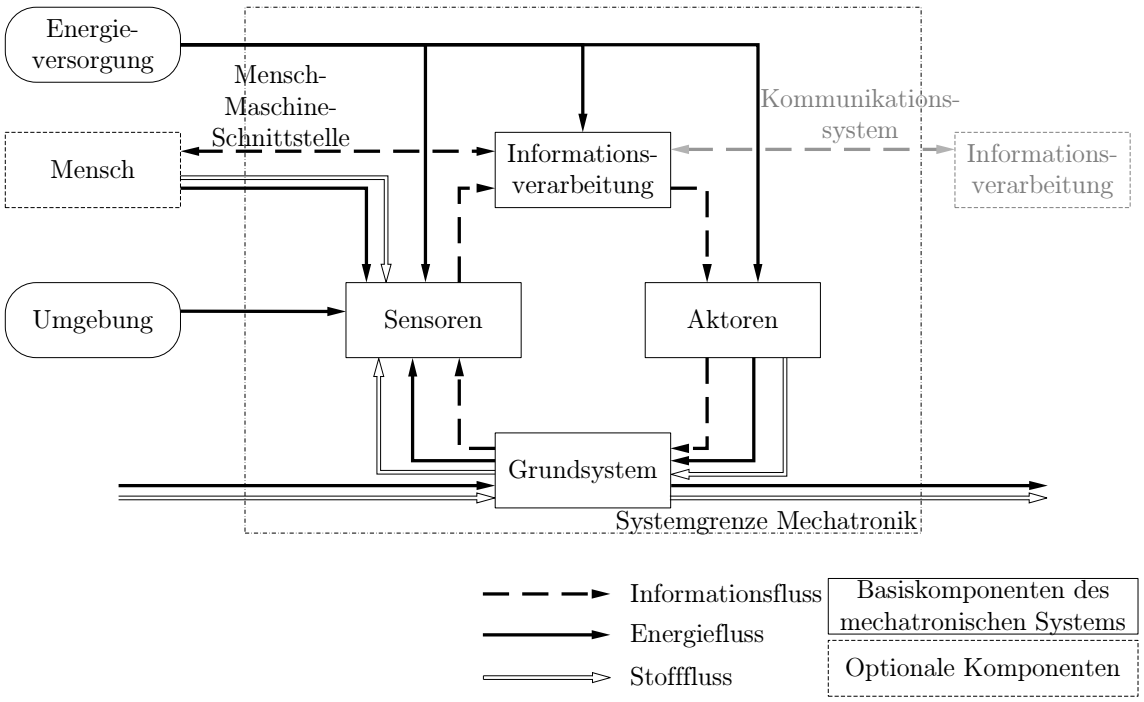

Abbildung 2.7: Verallgemeinerte Darstellung mechatronischer Systeme

Integrierte mechanisch-elektronische System werden als mechatronische Systeme bezeichnet. Mechatronik ist ein Kunstwort aus Mechanik, Elektronik und Informatik. In [Ise08] liefert Isermann eine zusammengefasste Definition der zu entnehmen

ist, dass Mechatronik (im engeren Sinne) ein interdisziplinäres Gebiet ist, in dem nachfolgend aufgeführte Disziplinen zusammenwirken:

- Maschinenbau (Maschinenelemente, Maschinen, Feingerätetechnik),
- Elektrotechnik/Elektronik (Mikroelektronik, Leistungselektronik, Messtechnik, Aktorik, Energieversorgung) sowie
- Informatik (Systemtheorie, Regelungs- und Automatisierungstechnik, Software-Gestaltung, künstliche Intelligenz).

Bei mechatronischen Systemen erfolgt die Lösung einer Aufgabe sowohl auf mechanischem als auch auf digital-elektronischem Weg. Hierbei spielen die Wechselwirkungen zwischen den Domänen der Produktentwicklung eine wichtige Rolle. Die einzelnen Domänen bedingen sich aufgrund ihrer Abhängigkeit gegenseitig. Durch das enge Zusammenwirken von Maschinenbau, Elektrotechnik und Informationstechnik werden innovative Lösungen und erweiterte Funktionen möglich. Hierdurch kann das Kosten-Nutzen-Verhältnis heute bekannter Produkte verbessert und technologisch neuartige Ideen umgesetzt werden. Mechatronische Systeme sind aufgrund des vernetzten Zusammenspiels verschiedener Wissensdomänen durch eine hohe Komplexität gekennzeichnet und stellen hohe Anforderungen an einen integrativen Entwicklungsprozess. Die Mechatronik basiert auf der Synergie der klassischen Ingenieurwissenschaften Maschinenbau, Elektrotechnik und Informationstechnik (vgl. [VDI04]).

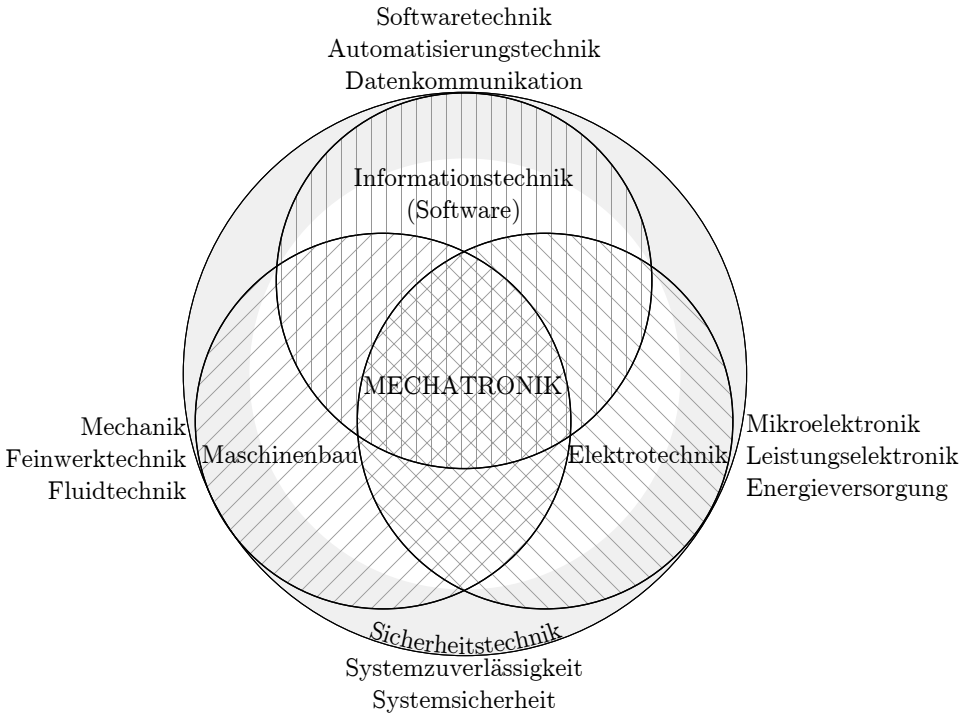

Abbildung 2.8: Mechatronische Domänen und ihre Abhängigkeiten

Nach Isermann [Ise08] entstehen mechatronische Systeme durch simultanes Entwerfen und die Integration von folgenden Komponenten oder Prozessen:

- Mechanische Komponenten/Prozesse,
- Elektronische Komponenten/Prozesse und
- Informationstechnik (einschließlich Automatisierungstechnik).

Anschließend kann die Integration von Mechanik und Elektronik durch Komponenten (Hardware: technische Struktur) oder informationsverarbeitende Funktionen (Software: funktionale Struktur) erfolgen. Ziel ist dabei, eine optimale Lösung zwischen der mechanischen Struktur, Sensor- und Aktor-Implementierung, automatischer digitaler Informationsverarbeitung und der Regelung zu finden.

In Abbildung 2.8 wird der Synergie-Effekt der Ingenieurwissenschaften durch die Schnittmengen der einzelnen Domänen verdeutlicht. Der Aspekt der Sicherheitstechnik während der Entwicklung von mechatronischen Systemen wird immer bedeutender und die Bestimmung der Systemzuverlässigkeit wird immer häufiger vom Kunden gefordert. Bisher werden die ingenieurwissenschaftlichen Domänen der Mechatronik separat bezüglich ihrer Zuverlässigkeit bewertet oder die Bewertung wird am Gesamtsystem durchgeführt. Für die ganzheitliche Bestimmung der Zuverlässigkeit von Systemen ist es jedoch wichtig die Bewertung der einzelnen Domänen zu kombinieren. Das Systemverhalten aller Domänen und ihrer Schnittstellen ist in einer einheitlichen Methode[3] zu beschreiben.

Das generische Vorgehen bei der Entwicklung mechatronischer Systeme beruht auf dem V-Modell[4] (Abbildung 2.9). Zu Beginn werden die Anforderungen an das System erarbeitet und zusammengefasst. Diese können unter anderem funktionale Anforderungen, Anforderungen an die Kosten oder Anforderungen an die Sicherheit beinhalten. Innerhalb des V-Modells startet die Systementwicklung mit der Analyse der Systemanforderungen und endet mit der Produktfreigabe. Das Systemdesign geht über in eine domänenspezifische Entwicklung und wird später wieder zu einem System zusammengefügt (Systemintegration). Die Einhaltung der spezifizierten Eigenschaften wird ständig überwacht. Unter der Beachtung dieser Aspekte kombiniert der in dieser Arbeit eingeführte Systembewertungsprozess die Entwicklung innerhalb der einzelnen Domänen und liefert die Basis für den anschließenden Quantifizierungsprozess.

[3] Eine Methode beschreibt die systematische Vorgehensweise zur Beschreibung und Quantifizierung von technischen Systemen. Für die Quantifizierung und Informationsgewinnung können systematisierte Verfahren genutzt werden.

[4] V-Modell: Bei der Entwicklung nach dem V-Modell [IAB] sind den einzelnen Entwurfsphasen die Integrations-, Prüf- und Testphasen gegenübergestellt. Das V-Modell wurde aus dem Wasserfallmodell abgeleitet und bildete ursprünglich die Grundlage für den Softwareentwicklungsprozess. Aktuell ist es ebenso für die eher traditionellen Ingenieursdisziplinen Maschinenbau und Elektrotechnik als Leitmethode in der Entwicklung akzeptiert und übernommen. Somit wurde das ursprünglich aus der Softwareentwicklung stammende Modell im Laufe der Zeit an die Bedürfnisse der Entwicklung von mechatronischen Systemen angepasst.

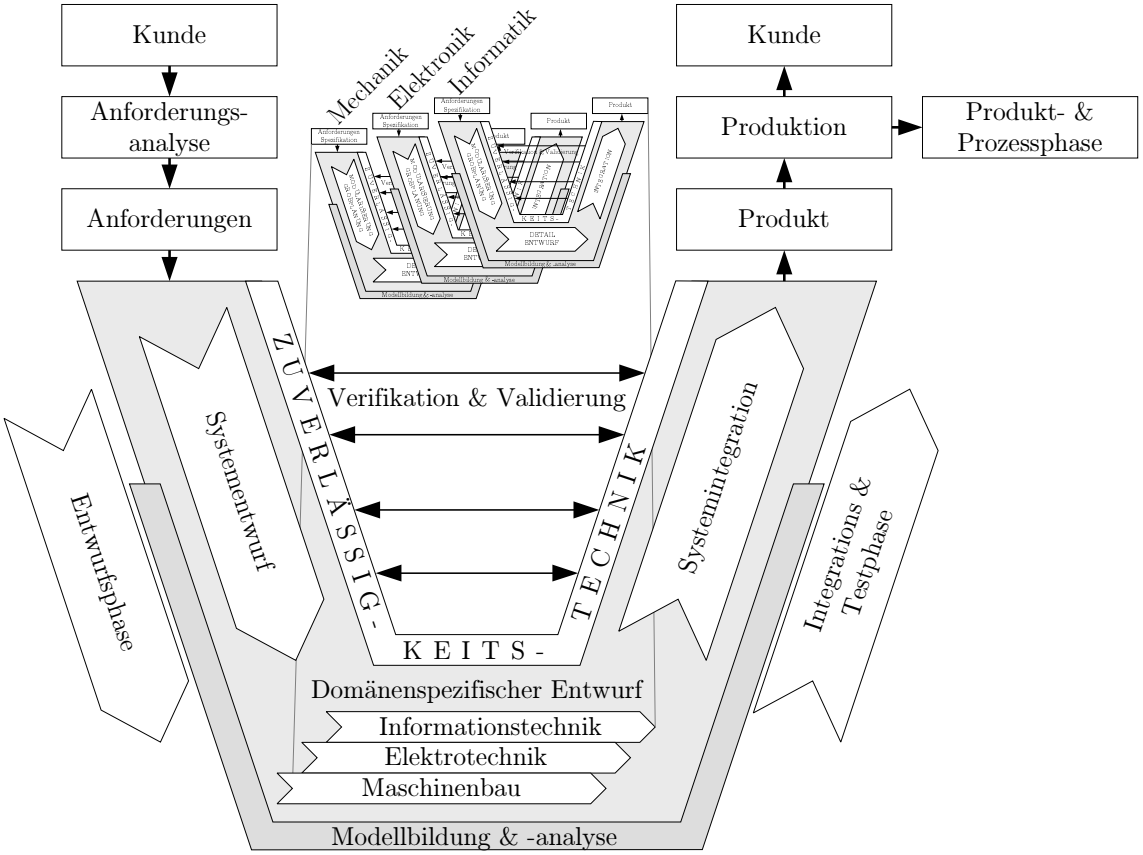

Abbildung 2.9: Erweitertes V-Modell im Entwicklungsprozess (vgl. [VDI04])

In der Richtlinie VDI 2206 [VDI04] wird eine methodische Unterstützung domänenübergreifender Entwicklung mechatronischer Systeme beschrieben. Hier liegen die Schwerpunkte auf Vorgehensweisen, Methoden und Werkzeugen. Die Grundlagen zum Entwickeln mechatronischer Systeme werden vermittelt und eine ganzheitliche Sichtweise über die Fachdisziplinen hinaus anregt. Die Grundlage für das methodische Vorgehen innerhalb der Entwicklung von mechatronischen Systemen bildet das V-Modell. Dieses Modell beschreibt die wesentlichen Schritte die zur Entwicklung von Systemen notwendig sind in ihrem logischen Zusammenhang. Die wesentlichen Teilschritte des V-Modells (Abbildung 2.9) bilden

- Leistungsbeschreibung,
- Systementwurf,
- domänenspezifischer Entwurf,
- Systemintegration und
- Leistungsnachweis.

Der Entwicklungszyklus beginnt mit der Produktidee, die aus einem bestehenden Sachverhalt resultiert oder sich aus einer Spezifikation ergibt. Dieser wird im günstigsten

Fall mit einem Entwicklungsauftrag fortgesetzt. Die anschließende Anforderungsanalyse liefert eine Anforderungsliste. Diese bildet die Grundlage für den Systementwurf und die Überwachung der Entwicklungsziele. Die Forderungen und Wünsche der Anforderungsliste werden auf wesentliche Aussagen und lösungsneutrale Formulierungen reduziert.

Der Systementwurf legt, basierend auf der Anforderungsliste, ein domänenübergreifendes Lösungskonzept fest. Dieses Konzept beschreibt die wesentlichen physikalischen und logischen Wirkweisen des technischen Systems und liefert eine erste abstrakte Strukturbeschreibung. Die Beschreibung der Gesamtfunktion ist im Allgemeinen sehr komplex, daher ist es sinnvoll, diese in Teilfunktionen zu unterteilen. Diesen Teilfunktionen werden dann die notwendigen Lösungs- oder Wirkprinzipien zugeordnet. Die Teilfunktionen werden durch eine Funktionsstruktur beschrieben, die in Form von Blockdiagrammen dargestellt werden können. Diese visualisieren den Zusammenhang zwischen den Eingangs- und Ausgangsgrößen und ermöglichen die Darstellung von Verknüpfungen in Form von Material-, Energie- und Informationsflüssen. Die Funktionserfüllung der Teilfunktionen wird anschließend im Systemzusammenhang überprüft. Die einzelnen Wirkprinzipien oder Lösungselemente werden über eine Wirkstruktur zu der Gesamtsystemfunktion zusammengefasst und auf Funktionserfüllung überprüft [PBFG07].

Der nächste Abschnitt innerhalb des V-Modells wird als domänenspezifischer Entwurf bezeichnet. Hier werden die Lösungselemente oder Wirkprinzipien der Teilfunktionen den Domänen zugeordnet. Auf der Domänenebene findet eine weitere Konkretisierung statt: Es werden detaillierte Auslegungen und Berechnungen benötigt, die insbesondere bei sicherheitskritischen Funktionen die Funktionserfüllung sicherstellen. Die Entwicklung innerhalb der einzelnen Domänen erfolgt nach domänenspezifischen Entwicklungsmethoden, die durch eigene Denkweisen, Begriffswelten und Erfahrungen geprägt sind. Eine Entwicklungsmethodik für den Maschinenbau wird in der VDI 2221 [VDI93] vorgestellt. Für die Software beschreiben Phasenmodelle, das Wasserfallmodell, das Spiralmodell und insbesondere das V-Modell die Vorgehensweise für die Entwicklung. Elektronik wird nach hierarchischen Entwurfsmethoden und Phasenmodellen entwickelt.

Die Systemintegration umfasst den Zusammenschluss der in den domänenspezifischen Entwürfen realisierten Funktionen, Komponenten oder Teilsysteme zu einem übergeordneten Produkt. Es wird nach [VDI04] zwischen verschiedenen Integrationsarten differenziert. Die Integration verteilter Komponenten beschreibt die Verbindung zwischen den Komponenten, wobei die Komponenten über Energie-, Stoff- und Informationsflüsse miteinander verbunden und in Abhängigkeit zueinander gebracht werden. Wird das Gesamtsystem aus Modulen definierter Funktionalität und standardisierter Schnittstellen zusammengesetzt, wird dies als modulare Integration bezeichnet. Bei der räumlichen Integration werden komplexe Funktionseinheiten durch die räumliche Zusammenfassung der Komponenten gebildet. Kommt es während

des domänenspezifischen Entwurfs zu Veränderungen der Wirkstruktur, müssen eventuelle Inkompatibilitäten während der Systemintegration beseitigt werden. Das zusammengesetzte System repräsentiert je nach Reifegrad oder durchlaufenem Zyklus des V-Modells ein Funktions-/Labormuster, ein Prototyp, ein Vorserienprodukt (Nullserie) oder ein Serienprodukt. Das System ist erst in diesem Stadium bezüglich seines Zusammenwirkens überprüfbar. Im Normalfall werden Zuverlässigkeitsanalysen des Systems auch erst in dieser Phase durchgeführt oder aus der domänenspezifischen Entwicklung zusammengeführt.

Die Eigenschaftsabsicherung überprüft den Entwurfsfortschritt anhand des festgelegten Lösungskonzeptes und der systemspezifischen Anforderungen. Es soll sichergestellt werden, dass die tatsächlichen und geforderten Systemeigenschaften übereinstimmen. Lösungsvarianten können miteinander verglichen und die optimale Variante selektiert werden. Hierzu werden die Eigenschaften der Lösungsvarianten der Anforderungsliste gegenübergestellt.

Die Phasen Systementwurf, domänenspezifischer Entwurf und Systemintegration werden durch die Modellbildung und -analyse begleitet. Mit Hilfe von Modellen und rechnergestützten Werkzeugen wird das System simuliert und der Entwurf überwacht.

Das Resultat der durch das V-Modell beschriebenen Vorgehensweise ist nicht ausschließlich ein fertiges, real existierendes Produkt, sondern eher das konkretisierte zukünftige Produkt oder die Produktfreigabe. Ist der Reifegrad für ein Vorserienprodukt erreicht wird das Produkt für die Produktion freigegeben. Die Produktfreigabe geht dann in die Produkt- & Prozessphase über.

Für den domänenspezifischen Entwurf werden keine konkreten Vorgehensweisen festgelegt. Hier empfiehlt sich, die Methodik des generischen V-Modells anzuwenden. Diese Vorgehensweise ermöglicht die Sicherstellung der Einhaltung der Anforderungen und Spezifikationen die an das System gestellt werden. Die Zuverlässigkeitsbewertung der Systeme wird im V-Modell erst in der Systemintegrationsphase möglich. Um im Entwicklungsprozess frühzeitig Entscheidungen bezüglich der Zuverlässigkeit zu treffen, ist ein angepasster, entwicklungsbegleitender Bewertungsprozess notwendig. Dieser Zusammenhang ist in Abbildung 2.9 durch das um die Zuverlässigkeitstechnik erweiterte V-Modell dargestellt. In diesem Zusammenhang wird in dieser Arbeit eine Strukturbeschreibung durch Matrizen eingeführt, die eine frühe Bewertung der Teilsysteme ermöglicht und Rückschlüsse auf das Gesamtsystem erlaubt. Hierbei wird die Abbildung und Untersuchung der Systemeigenschaften wird in den Vordergrund gestellt.

2.3 Zuverlässigkeit und Sicherheit

Zur Beschreibung der zuverlässigkeitstechnischen Zusammenhänge und zum einheitlichen Verständnis werden folgend die wichtigsten Begriffe und Kenngrößen als

Arbeitsdefinition erläutert. Hierdurch wird eine Abgrenzung der Begriffe Zuverlässigkeit, Verfügbarkeit und Sicherheit erreicht. Die Darstellung dieser Zusammenhänge erfolgt in Anlehnung an [VDI07, DIN90].

Fehler Der Fehler (engl. „fault, defect“) bezeichnet einen anormalen Zustand, der eine Verminderung oder den Verlust der Fähigkeit einer Funktionseinheit verursachen kann, eine geforderte Funktion auszuführen.

Ausfall Ein Ausfall (engl. „failure“) führt zur Beendigung der Fähigkeit einer Funktionseinheit, eine geforderte Funktion auszuführen. Die Begriffe Versagen, Funktionsverlust und Totalausfall werden in diesem Zusammenhang synonym verwendet.

Zuverlässigkeit Unter der Zuverlässigkeit wird die Wahrscheinlichkeit verstanden, dass ein System seiner Spezifikation entsprechend, während einer bestimmten Zeitdauer funktioniert. Nach DIN 40041 [DIN90] ist Zuverlässigkeit (engl. „dependability“) die Eigenschaft eines Systems bezüglich seines Anwendungsgebietes, bei vorgegebenen Anwendungsbedingungen die Zuverlässigkeitsanforderung zu erfüllen. In diesem Zusammenhang wird in der DIN der Begriff Zuverlässigkeit übergeordnet definiert, er fasst die Funktionsfähigkeit und Überlebenswahrscheinlichkeit des Systems zusammen,

Funktionsfähigkeit Als Funktionsfähigkeit (engl. „reliability“) wird die Fähigkeit eines Systems bezeichnet, die geforderten Funktionen unter den vorgegebenen Anwendungsbedingungen zu erfüllen. Der Begriff „reliability“ wird häufig synonym zur Beschreibung der Zuverlässigkeit verwendet.

Überlebenswahrscheinlichkeit Die Wahrscheinlichkeit, dass ein System seine Betriebsdauer ab Anwendungsbeginn erreicht, wird als Überlebenswahrscheinlichkeit bezeichnet.

Verfügbarkeit Als Verfügbarkeit wird die Wahrscheinlichkeit, ein System an einem vorgegebenen Zeitpunkt in einem funktionsfähigen Zustand anzutreffen, definiert.

Sicherheit Als Sicherheit wird die Freiheit von unvertretbaren Risiken bezeichnet.

Risiko Das Risiko wird im Allgemeinen nicht quantitativ erfasst. Es wird über die Kombination der Häufigkeit eines Schadenseintrittes und seines Schadensausmaßes dargestellt.

Abweichung Die Abweichung (engl. „error“) ist die Differenz zwischen einem berechneten, beobachteten oder gemessenen Wert und dem wahren, spezifizierten oder theoretisch korrekten Wert.

Gefahr Eine Gefahr entsteht, wenn das Risiko größer als das vertretbare Risiko wird. Ein vertretbares Risiko wird nach den gültigen Wertvorstellungen der Gesellschaft akzeptiert.

Innerhalb der Systementwicklung werden den Systemen Eigenschaften zugeordnet (u. a. Funktionsstruktur, technisches Prinzip), anhand derer eine Systemauswahl

getroffen werden kann. Die charakterisierenden Eigenschaften aus Sicht der Zuverlässigkeitstechnik werden durch die Zuverlässigkeit, Verfügbarkeit und Sicherheit dargestellt. Diese Eigenschaften haben einen direkten Einfluss auf die Kosten und entscheiden indirekt über die Marktakzeptanz. Diese werden direkt oder indirekt vom Kunden wahrgenommen.

Das Management von Sicherheit und Zuverlässigkeit wird aufgrund von Kundenforderungen, gesetzlichen Forderungen und firmeneigenen Zielen vorausgesetzt. An technische Systeme im Automobil werden Anforderungen gestellt, die über die rein technischen Anforderungen hinausgehen. Dies sind unter anderem

- sicherheitstechnische Anforderungen,
- zuverlässigkeitstechnische Anforderungen,
- ökonomische Anforderungen,
- ergonomische Anforderungen,
- soziale/humane Anforderungen und
- umwelttechnische Anforderungen.

Der Nachweis von Zuverlässigkeit wird aufgrund der zunehmenden Komplexität der Systeme immer schwieriger. In [MP03] werden Forderungen hinsichtlich der Zuverlässigkeit und Sicherheit unter der Einbeziehung der Verfügbarkeit und Wartbarkeit als RAMS-Prozess (Reliability, Availabililty, Maintainability, Safety) [DIN00], den es unter der Berücksichtigung von Lebenslaufkosten (Kosten über den gesamten Produktlebenszyklus hinweg) zu optimieren gilt, eingeführt. Das Sicherheits- und Zuverlässigkeitsmanagement - häufig auch im „Systems Engineering“ integriert - stimmt den RAMS-Prozess auf die Produktphasen Planung, Entwicklung, Produktion und Anwendung ab. In diesem Zusammenhang werden Vorgaben definiert, deren Entwicklung überwacht und alle Aktivitäten zur Optimierung, Demonstration, Beurteilung, Vorausbestimmung etc. unternehmensspezifisch umgesetzt. Diese Vorgehensweise wird in der Regel durch einen RAMS-Programmplan unterstützt und häufig eingesetzt, um Fehler schon in der Planungs- bzw. Entwicklungsphase von Projekten zu vermeiden [MP03].

Die Anforderungen an die Sicherheit von Fahrzeugfunktionen sind, verglichen mit anderen Branchen wie beispielsweise dem Maschinenbau oder der Telekommunikation, besonders hoch. Bei Fahrzeugfunktionen erfolgt meist eine Einstufung in eine hohe Sicherheitsklasse. Die grundlegenden Sicherheitsbetrachtungen sind in Normen, wie DIN 19250 [DIN89], IEC 61508 [IEC06], ISO 26262 [ISO09], Straßenverkehrsordnung und in den ECE-Regelungen, wie [ECE04, ECE98], festgelegt [SZ10]. Die Analyse der Funktionssicherheit hat somit einen großen Einfluss auf die Entwicklung von Funktionen und Sicherheitskonzepten.

Die Bedeutung von Zuverlässigkeits- und Sicherheitsanalysen technischer Systemen ist im Bereich der Automobilindustrie gleichstellt.

2.3.1 Zuverlässigkeitstechnische Eigenschaften der Mechatronik

Der Entwicklungsprozess von mechatronischen Systemen ist stets mehrdeutig und unbestimmt. Dementsprechend wird nach Beurteilungskriterien gesucht, die die unterschiedlichen Entscheidungsprozesse unterstützen. Ein an Bedeutung zunehmendes Kriterium ist die Systemzuverlässigkeit, da ein Ausfall oder Fehler auf breiter Kundenebene schlichtweg als inakzeptabel betrachtet wird.

Bei Fehlern kann zwischen den Fehlerarten physikalisch, inhärent und nicht-inhärent unterschieden werden. Physikalische Fehler beruhen auf physikalischen oder chemischen Ausfallmechanismen oder Effekten (Verschleiß, Alterung, etc.). Fehler, die schon von Beginn der Nutzung vorhanden sind, werden als inhärente Fehler bezeichnet. Die Ursachen dieser liegen in menschlichem Versagen wie z. B. Spezifikationsfehler, Entwurfsfehler, Denkfehler, Kommunikationsfehler, mangelnde Erfahrung, Kundenfehler oder mangelndes Wissen [Ber09]. Inhärente Fehler werden häufig erst bei der Inbetriebnahme entdeckt. Tritt ein Fehler erst mit der Inbetriebnahme auf, wird dieser als nicht-inhärent bezeichnet. Die Ursachen dieser entsprechen denen inhärenter Fehler. Den einzelnen Domänen der Mechatronik können unterschiedliche Fehlerarten zugeordnet werden. Im Folgenden werden eine Abgrenzung der Domänen und deren Fehlerarten durchgeführt.

Die Zuverlässigkeitsbewertung von mechanischen Komponenten beginnt mit der Anforderungsanalyse auf Komponentenebene. Technische Dokumente (z. B. Stücklisten, Lastenhefte oder Skizzen) mit phasenabhängigem Reifegrad liefern die notwendigen Informationen für die Analyse. In einem Funktionsblockdiagramm werden die aus der Analyse ermittelten Funktionen zu einander in Bezug gebracht. Anhand des Funktionsblockdiagramms werden die Wechselwirkungen zwischen den Elementen dargestellt. Die anschließende qualitative Untersuchung wird durch qualitative und quantitative Methoden der Zuverlässigkeitstechnik unterstützt. Es werden kritische Elemente identifiziert und einer quantitativen Analyse unterzogen. Dies setzt jedoch ausreichend Informationen über die Komponente voraus. Felddaten und Tests können hier wichtige Informationsquellen sein. In frühen Entwicklungsphasen können die Zuverlässigkeitsdaten aus sogenannten Ausfallratenkatalogen entnommen werden. Diese Daten liefern jedoch nur eine grobe Abschätzung. Für konkrete Zuverlässigkeitsaussagen müssen mechanische Komponenten zuverlässigkeitstechnisch untersucht werden. Mechanische Komponenten werden durch schlagartige Ausfälle oder Driftausfälle charakterisiert. Ein schlagartiger Ausfall kann durch Überbeanspruchung, falsche Dimensionierung oder falsche Bedienung erfolgen. Dieser geschieht in der Regel ohne Ankündigung und führt meist zu einem völligen Funktionsverlust. Ausfälle infolge einer zeitveränderlichen partiellen Abnahme der Funktionalität werden als Driftausfälle definiert. Diese Ausfallart beruht auf physikalischen Phänomenen und kann nicht direkt durch konstruktive Maßnahmen optimiert werden. Zur Optimierung sind indirekte konstruktive Maßnahmen notwendig.

Im Entwicklungsprozess kann die Zuverlässigkeit elektronischer Komponenten erst spät quantifiziert werden. Die Ausfälle treten meist ohne Vorwarnung auf, sind daher scheinbar zufällig und werden mathematisch als Exponentialverteilungen betrachtet. Die Elektronik unterliegt wie die Mechanik Alterungsprozessen, nur ist diese in der Regel unsichtbar im Vergleich zur Alterung in Form von Abrieb. Fehlerraten können durch geraffte Tests ermittelt werden. Um Zuverlässigkeitsaussagen während des Entwurfs treffen zu können, wird auf Ausfallratenkataloge (u. a. Military Handbook [Dep91]) zurückgegriffen. Aufgrund der umfangreichen Zuverlässigkeitsdaten ist eine Zuverlässigkeitsprognose für elektronische Komponenten sehr präzise und liefert somit bereits in der Entwurfsphase wichtige Informationen.

Informationsverarbeitende Komponenten eines mechatronischen Systems werden durch die Kombination von Elektronik und Software realisiert. Das Versagen einer Softwarekomponente basiert nicht auf physikalischen oder chemischen Ursachen. Es wird hervorgerufen, wenn ein Fehler bereits in der Software vorhanden ist oder der Fehler bei Ausführung der Software aktiviert wird. Hier ist die Anzahl von Fehlern, die ein Versagen des Systems hervorrufen können, eine unveränderliche Größe. Abhängig vom Zeitpunkt wird zwischen Spezifikationsfehlern und Implementierungsfehler unterschieden. Softwarefehler sind inhärent und systematischer Natur. Diese entstehen durch den Entwickler während des Entwurfs oder der Umsetzung (Programmierung), infolge falscher Programmierung, Unachtsamkeit, mangelnder Erfahrung oder falscher Spezifikation. Die Bewertung der Zuverlässigkeit erfolgt mit Hilfe von Modellen und empirischen Daten über das Versagen der Software. Besteht keine Möglichkeit auf empirische Daten zurückzugreifen, werden indirekte Größen für die Einschätzung der Zuverlässigkeit verwendet. Dabei unterstützen qualitative Methoden die Bewertung der Zuverlässigkeit.

Eine eindeutige Trennung zwischen den Teilgebieten der Mechatronik ist nicht möglich. Domänenspezifische Analyse- und Bewertungsmethoden sind zwar separat anwendbar, können jedoch nur mit großem Aufwand auf das Gesamtsystem übertragen werden. Häufig wird, das aus den einzelnen Domänen integrierte Gesamtsystem analysiert. Diese Vorgehensweise hat den Nachteil, dass das System abstrahiert werden muss. Mechatronische Systeme erfordern eine domänenübergreifende einheitliche Analyse- und Bewertungsmethode, die die Eigenschaften der einzelnen Domänen berücksichtigt. Eine methodische Durchdringung des gesamten Produktentwicklungsprozesses ist für alle mechatronischen Teildisziplinen notwendig.

2.3.2 Methoden zur Ermittlung der System-Zuverlässigkeit

Die Ermittlung der Systemzuverlässigkeit ist Teil der Entwicklungsmethodik und für den Produktentstehungsprozess als Freigabekriterium vorgeschrieben. Nach [VDA00] zählen

– Fehler-Möglichkeits- und Einfluss-Analyse (FMEA),

- ABC-Analyse,
- Fehlerbaum-Analyse (FTA) (qualitativ/quantitativ),
- Importanzanalyse,
- Boolesche Theorie und
- Markov Theorie

zu den heute üblichen Methoden zur Ermittlung der Systemzuverlässigkeit.

Die FMEA unterstützt die systematische Suche nach Schwachstellen und ihren systemischen Auswirkungen. Diese Methoden werden durch die ABC-Analyse und die qualitative FTA unterstützt und liefern nur eine qualitative Aussage über die System-Zuverlässigkeit. Die quantitative FTA, die Markov Theorie und die Boolesche Theorie liefern quantitative Prognosen über das zu erwartende Ausfallverhalten des Systems.

Allen Methoden ist gemein, dass die Genauigkeit der Analyse sehr stark vom Detaillierungsgrad und von der Abbildungsgenauigkeit der Strukturbeschreibung abhängig ist. Jede Methode setzt eine Analyse der Systemstruktur voraus und basiert auf einer formalen oder adäquaten Strukturbeschreibung. Eine domänenübergreifende Strukturbeschreibung ist schwierig. Gerade bei einer komplexen Systemstruktur, die informationsverarbeitende Komponenten beinhaltet, sind die Systemstruktur und die Abhängigkeiten innerhalb der Struktur nicht sichtbar.

Der folgende Abschnitt behandelt die Zuverlässigkeitsanalyse von Systemen und die Anwendung von Zuverlässigkeitsmethoden. Aufgrund von Funktionsmodellen bzw. physikalisch-technischen Modellen eines technischen Systems ist es möglich, die Zuverlässigkeits- bzw. Sicherheitskenngrößen des Systems zu ermitteln. Für die quantitative Auswertung sind die Zuverlässigkeitskenngrößen der Komponenten notwendig. Somit ist es möglich, alternative Konzepte miteinander zu vergleichen und bei vorhandenen Zuverlässigkeitskenngrößen der Komponenten das System zu quantifizieren.

In Anlehnung an [Fre73] ist in Abbildung 2.10 die abstrakte Darstellung der Zuverlässigkeits-/Sicherheitsanalyse dargestellt. Die quantitative Zuverlässigkeitsbewertung wird durch eine Modellbildung unterstützt. Die drei Hauptphasen der Zuverlässigkeitsmodellbildung sind in Abbildung 2.10 in abstrakter Form dargestellt. Darin wird das technische System zunächst in ein technisches Modell überführt. Mit Informationen aus einer Komponentendatenbank ist es folglich möglich, das technische Modell in ein mathematisches Modell zur Berechnung der Zuverlässigkeit zu überführen. Die quantifizierten Kenngrößen werden bezüglich der Anforderungen bewertet. Ist die Systemanforderung nicht erfüllt, findet zyklisch eine Systemoptimierung/-variation statt und die Bewertung wird erneut durchgeführt.

Werden bei der Analyse des Systems potentielle Fehler identifiziert, wird nach den Fehlerursachen gesucht. Unter Berücksichtigung der Kosten sowie der Vermeidbarkeit/Beherrschbarkeit wird eine Fehlerbehebung durchgeführt. Der Zusammenhang

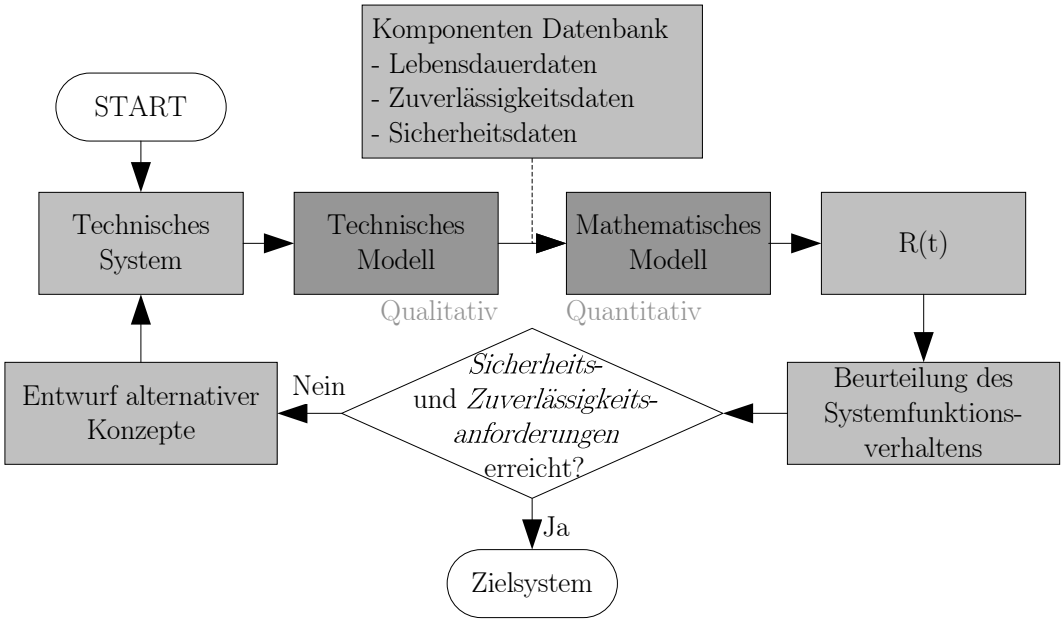

Abbildung 2.10: Zuverlässigkeitsmodellbildung (vgl. [Fre73])

zwischen Kosten für die Beseitigung von Fehlern und dem Zeitpunkt der Entdeckung ist in Abbildung 2.11 verdeutlicht. Die Zehnerregel [TM03] zeigt auf, das sich die Kosten für einen Fehler bei jedem Prozessschritt verzehnfachen. Aus diesem Grund

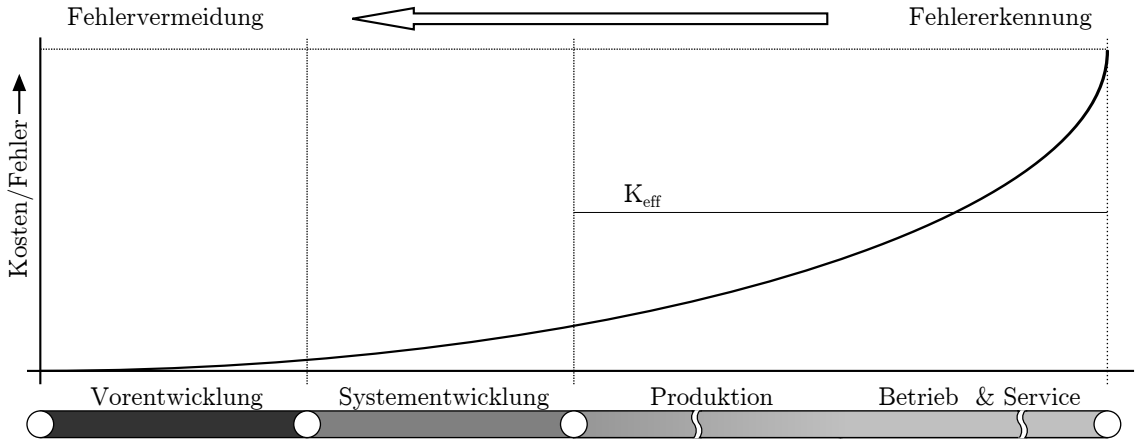

Abbildung 2.11: Fehlerkostenentwicklung über den Lebenszyklus (vgl. [TM03])
Kostenschwelle K_{eff}: effektive Kosten-pro-Fehler-Schwelle

ist eine präventive Fehlervermeidung immer einer nachträglichen Fehlerbehebung vorzuziehen. Eine effektive Kostenschwelle K_{eff} wird für die Planung festgelegt. Wird diese bei der Planung überschritten, ist ein Gewinn bei der Vermarktung des Produktes nicht zu erwarten. Die Kosten für die Fehlerbehebung während der Laufzeit wären größer als der Gewinn.

Die Analyse der Zuverlässigkeit basiert auf einer Strukturanalyse des Systems. Hierdurch werden neben der Zuverlässigkeitsanalyse des Gesamtsystems auch Schwachstellen und kausale Zusammenhänge, die zu einem Fehler führen können, aufgedeckt. Neben den systembedingten Faktoren können auch äußere Faktoren berücksichtigt werden. Äußere Faktoren sind zum Beispiel die Einwirkungen von Benutzern oder der Umwelt. Die Verfahren der Zuverlässigkeitsanalyse werden in der Regel während der Entwicklungs- und Konstruktionsphase durchgeführt, um entdeckte Fehlermöglichkeiten je nach Risiko rechtzeitig beseitigen zu können. Für alle Verfahren sind die genaue Kenntnis des Zusammenspiels und der Wechselwirkungen der Systemkomponenten sowie das Verhalten angrenzender Systeme, der Umgebung und des Benutzers unerlässlich. Die üblichen Methoden der Zuverlässigkeitstechnik unterstützen die Strukturanalyse, die wiederum wichtige Informationen für diese Methoden bereitstellt.

Die graphische Darstellung von Systemkonfigurationen kann durch Zuverlässigkeitsblockdiagramme, Fehler- oder Funktionsbäume oder Zustandsdiagramme erfolgen. Das Zuverlässigkeitsblockdiagramm (engl. „reliability block diagram“, RBD) ist eine sehr gebräuchliche zuverlässigkeitstechnische Darstellungsform von Systemen. Jede Systemkomponente wird durch eine spezifische Zuverlässigkeitseigenschaft charakterisiert. Besitzt eine Komponente beispielsweise mehrere Ausfallarten, muss jede Ausfallart separat durch einen Block dargestellt werden. Die Blöcke werden als eine „Black-Box“ betrachtet. Das entstehende RBD besitzt einen Eingang und einen Ausgang und besteht aus einer Mischstruktur von seriellen und parallelen Verknüpfungen. Die Darstellung bezieht sich stets auf die Funktionsfähigkeit des Systems. Die hierdurch generierte Struktur weißt erhebliche Unterschiede zu der Darstellung in Form eines Funktionsblockdiagramms (FBD[5]) auf. In einem RBD werden die Funktionen und Komponenten, die zu einem fehlerfreien Systemzustand führen, dargestellt. Wie später noch detailliert erläutert, beschreibt das Funktionsblockdiagramm die Abhängigkeiten der Funktionen untereinander aus funktionaler Sicht.

Komplexe Systeme werden zur zuverlässigkeitstechnischen Quantifizierung häufig als Fehlerbäume oder Funktionsbäume dargestellt. Diese Methoden stellen eine Top-Down-Analyse dar. Das Systemverhalten wird bezüglich eines kritischen Ausfalls (Top[6]) analysiert. Von diesem kritischen Ausfall ausgehend, werden die Komponentenzustände (Basisereignisse) durch logische Symbole dargestellt. Der Fehlerbaum ist die graphische Darstellung des Ereignisses „Systemausfalls“ eines Systems oder

[5]Funktionsblockdiagramme liefern einen leicht verständlichen Einblick in die Arbeitsweise eines Systems und lassen sich relativ einfach in Fehlerbäume überführen. Neben den Betriebs- und Funktionsabläufen sind Redundanzen oder Abhängigkeiten innerhalb des Systems erkennbar. Zu erfüllende Nebenbedingungen sind ebenso ersichtlich. Parallel zu dieser graphischen Beschreibung sind die ergänzenden verbalen Beschreibungen der Konstrukteure von Bedeutung.

[6]Top: Kritischer Ausfall, kritisches Ereignis oder kritischer Zustand der näher analysiert wird. Die Fehlerbaumanalyse bezeichnet dieses unerwünschte Ereignis auch als Top.

Untersystems (Subsystems) als Boolesche Funktion unter der Benutzung der logischen Symbole Konjunktion, Disjunktion und Negation. Der Funktionsbaum charakterisiert hingegen das Ereignis „Systemfunktion“. Bei der Darstellung mittels Zustandsdiagrammen wird das System in endlich viele Zustände zerlegt. Die einzelnen Zustände werden durch Kreise und die Übergänge zwischen den Zuständen durch Pfeile (Kanten) dargestellt. Im einfachsten Fall besitzt das System zwei Zustände, das System ist funktionsfähig oder das System ist ausgefallen. Sind mehr als zwei Zustände zu berücksichtigen, ist die Darstellung mittels Zustandsdiagrammen möglich [MP03, Rak02]. Hierbei wird die zeitliche Reihenfolge der Zustände berücksichtigt.

Zur Identifizierung von kritischen Ausfällen oder unerwünschten Ereignissen stehen unterschiedliche Methoden der Zuverlässigkeitstechnik zur Verfügung. Die Ereignisbaumanalyse untersucht die Auswirkung eines Teilsystem- oder Komponentenausfalls auf die Systemfunktion. Diese Methode ist jedoch bei komplexen Systemen sehr aufwendig. Aus diesem Grund wird bei komplexen Systemen auf die häufig verwendete und anschließend näher beschriebene FMEA zurückgegriffen.

2.3.3 Fehler-Möglichkeits und Einfluss-Analyse

Als Methode des präventiven Qualitätsmangements zur strukturierten und qualitativen Analyse von Systemen und Prozessen verfolgt die formalisierte FMEA das Ziel, potenzielle Fehlermöglichkeiten, deren Ursachen sowie deren Folgen zu entdecken, zu analysieren und zu vermeiden. Deren Anwendung im Qualitätsmanagement dient der produkt- und prozessbezogenen Risikominimierung in Folge auftretender Fehler. Anwendung findet die FMEA in der Entwicklung von neuen Konzepten, sowie in der Weiterentwicklung oder Änderung von Produkten und Prozessen.

Mit der FMEA wird die Idee einer präventiven Fehlererkennung und Fehlervermeidung, anstelle einer nachträglichen Korrektur, verfolgt. Zum Erreichen dieses Ziels werden geeignete Maßnahmen bereits in frühen Phasen des Produktentstehungsprozesses definiert.

Es können die drei Hauptformen Konstruktions-FMEA (Design), Prozess-FMEA (Produktion) und Sytem-FMEA (Produkt und Funktionalität), mit jeweils unterschiedlichen Anwendungsgebieten, unterschieden werden. Diese Hauptformen werden in Anlehnung an [MP03] wie folgt beschrieben.

Konstruktions-FMEA Die Konstruktions-FMEA untersucht die spezifikationsgerechte Gestaltung und Auslegung der Komponenten zur Vermeidung von Entwicklungsfehlern und konstruktiv beeinflussbaren Prozessfehlern.

Prozess-FMEA Die zeichnungs- und fertigungsgerechte Prozessplanung und -ausführung der Komponenten zur Vermeidung von Planungs- und Fertigungsfehlern wird durch die Prozess-FMEA untersucht. Es soll dabei sichergestellt werden, dass die Qualität des Endproduktes den Erwartungen des Kunden entspricht.

System-FMEA Auf Grundlage der Systemspezifikation untersucht die System-FMEA das funktionsgerechte Zusammenwirken der Systemkomponenten und ihrer Verbindungen zur Vermeidung von Fehlern bei Systemauswahl und -auslegung sowie Feldrisiken.

Aufgrund von Erfahrungen, aus der Anwendung in der Automobilindustrie, wird die FMEA durch den Verband der Automobilindustrie (VDA) [VDA06] automobilspezifisch angepasst und erweitert. Es werden lediglich zwei Varianten, die gesteigerten Wert auf die ganzheitliche und systematische Betrachtung legen, in einer System-FMEA zusammengefasst. Es wird nicht mehr aufgrund der Abstraktionsgrade oder der Detaillierungsniveaus untergliedert sondern aufgrund der Art der Vorgehensweise der Untersuchung in funktionale Betrachtungen (Produkte) und Betrachtung nach Abläufen (Prozesse). In Anlehnung an [VDA06] werden die beiden Varianten wie folgt beschrieben:

Produkt-FMEA Die Produkt-FMEA betrachtet die geforderten Funktionen von Produkten und Systemen bis auf die Auslegung der Eigenschaften und Merkmale. Dabei werden die möglichen Abweichungen betrachtet und die Maßnahmen zur Sicherstellung der Forderungen definiert.

Prozess-FMEA Werden die die Abläufe zur Herstellung von Produkten und Systemen bis zu den Anforderungen an die Prozesseinflussfaktoren betrachtet, wird dies der Prozess-FMEA zugeordnet. Mögliche Abweichungen werden betrachtet und Maßnahmen zur Sicherstellung der Abläufe und der Produktmerkmale definiert. Die Prozess-FMEA betrachtet unter anderem die Einflussgrößen Mensch, Maschine, Methode, Material und Umwelt.

Die FMEA ist universell ausgelegt und kann auch auf nichttechnische Systeme angewendet werden. Im Betrachtungsumfang befinden sich Systeme, Softwarefunktionen, Schnittstellen, Konstruktion, Komponenten, Fertigungsabläufe, inline Tests, inline Prüfungen, Montageabläufe, Logistik, Transport und Maschinen/Werkzeuge.

Die methodische Anwendung der FMEA ist schon in frühen Konzeptphasen sinnvoll und wird unter der Berücksichtigung folgender Ziele (vgl. [VDA06]) durchgeführt, die wiederum das Erreichen von unternehmerischen Zielen unterstützen:

- Erfassen von Forderungen die an das Produkt in Bezug auf Vollständigkeit, Verifizierbarkeit und Validierbarkeit aus Sicht der Qualität gestellt werden,
- Ermittlung von Forderungen bezogen auf die Fehlererkennung während der Entwicklungsphasen,
- Bestimmung von Forderungen zur Fehlererkennung im Kundenbetrieb,
- Identifikation von möglichen Fehlern und weiterhin
- Fehlerabstellung sowie Neubewertung.

Die Identifikation und Betrachtung von möglichen Fehlern ist von besonderer Bedeutung, denn diese basiert auf einer strukturierten und detaillierten Systembeschreibung

und ermöglicht die Verbesserung des Systems sowie die Generierung von alternativen Konzepten.

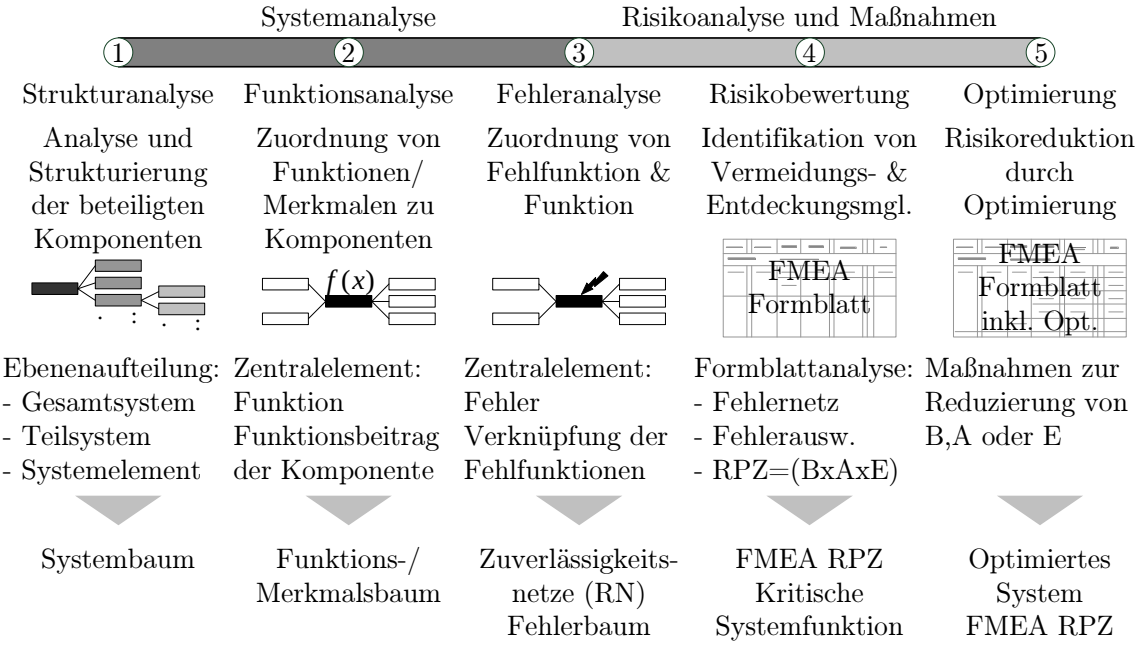

Abbildung 2.12: Vorgehensweise bei der Durchführung einer FMEA
RPZ = Risikoprioritätszahl, B = Bedeutung, E = Entdeckungswahrscheinlichkeit, A = Auftretenswahrscheinlichkeit

Für die Durchführung schlägt der VDA eine systematische Vorgehensweise vor und gliedert die FMEA in fünf Schritte (Abbildung 2.12):

1. Strukturanalyse (komponenten-/bauteilorientiert),
2. Funktionsanalyse (funktionsorientiert),
3. Fehleranalyse,
4. Risikobewertung und
5. Optimierung.

Die Reihenfolge der Struktur- und Funktionsanalyse richtet sich nach dem verfolgten Ziel. Eine bauteilorientierte Analyse beginnt mit der Strukturanalyse und geht in die Funktionsanalyse über. Im Gegensatz hierzu wird bei der funktionsorientierten Analyse die Reihenfolge vertauscht.

Durch die Strukturanalyse wird das Gesamtsystem in Teilsysteme aufgeteilt und der zu analysierende Systemumfang abgegrenzt. Die Systemelemente werden hierarchisch in einer Systemelementstruktur (Strukturbaum) angeordnet.

Die Funktionsanalyse ordnet jedem Teilsystem mindestens eine Funktion zu. Die Beschreibung dieser Funktion muss eindeutig, verifizierbar und validierbar sein. Für

die Analyse sind umfassende Kenntnisse über das System und die Umgebungsbedingungen notwendig. Das Zusammenwirken der Funktionen mehrerer Systemelemente wird als Funktionsstruktur (Funktionsbaum, Funktionsnetz) dargestellt. Das Funktionsnetz stellt die logischen Ursache-Wirkung-Beziehungen dar.

Die Identifikation aller potentiellen Fehlfunktionen wird durch die Fehleranalayse erreicht. Die Zusammenhänge werden durch die Fehlerart, die Fehlerursache und die Fehlerfolgen beschrieben.

Die Bedeutung der Fehlerfolge (B), die Auftretenswahrscheinlichkeit der Fehlerursache (A) und die Entdeckungswahrscheinlichkeit der aufgetretenen Fehlerursache (E) werden in der Risikoprioritätszahl (RPZ) als Produkt zusammengefasst. Die RPZ dient als Entscheidungskriterium zur Einleitung von Optimierungsmaßnahmen oder detaillierteren Analysen der betrachteten Funktion. Wurde das System optimiert ist eine erneute Bewertung durch die FMEA erforderlich.

Zur frühzeitigen Vermeidung von potenziellen Systemfehlern ist die System-FMEA ein zentraler Punkt im sicherheitsgerichteten Entwicklungsprozess. Diese wird im Rahmen der Fehlerbaumanalyse durch die Mehrfachfehlerbetrachtung und logische Verknüpfung erweitert.

2.3.4 Fehlerbaumanalyse

Bei der Fehlerbaumanalyse (engl. „fault tree analysis“, FTA) handelt es sich um ein deduktives Verfahren zur Bestimmung der Ausfallwahrscheinlichkeit. Dabei wird ein vorgegebenes unerwünschtes Ereignis auf eine Kombination von Primärereignissen zurückgeführt. Die Auswertung beruht auf der Grundlage der Booleschen Algebra. Die Fehlerbaumanalyse eignet sich zur Sicherheits- und Zuverlässigkeitsanalyse für Anlagen und Systeme aller Art. Ausfälle gemeinsamer Ursache (engl. „common mode failure“) und menschliche Fehler (engl. „human error“) können einschließend betrachtet werden. Die Ergebnisse ermöglichen eine Systembeurteilung im Hinblick auf Zuverlässigkeit, Verfügbarkeit und Sicherheit (vgl. [DIN81, MP03]).

Die Fehlerbaumanalyse ist eine Top-Down-Analyse. Von einem unerwünschten Ereignis (Top) ausgehend, wird untersucht, welche Ursachen oder Ursachenkombinationen zu diesem Ereignis führen. Die Beziehungen zwischen den Ausfallursachen werden durch Boolesche Operatoren beschrieben. Die Fehlerbaumanalyse eignet sich zur qualitativen und quantitativen Zuverlässigkeitsbewertung. Die qualitative Bewertung erlaubt die Gegenüberstellung von Entwurfsalternativen und unterstützt die Produktverbesserung. Die quantitative Form der Fehlerbaumanalyse unterstützt die Abschätzung der Produktzuverlässigkeit.

Mit der Fehlerbaumanalyse können folgende Ziele erreicht werden:

- Systematische Identifizierung aller möglichen Ausfallursachen und Ausfallkombinationen, die zu dem unerwünschten Ereignis führen,
- Vergleich von Entwurfsvorschlägen durch probabilistische Vorhersagen über die Zuverlässigkeit und Sicherheit,
- Ermittlung von Zuverlässigkeitskenngrößen (u. a. Ausfallwahrscheinlichkeit, Verfügbarkeit),
- Grafische Darstellung der Strukturfunktion (siehe 2.3.5),
- Nachweis geforderter Zuverlässigkeits- und Sicherheitsanforderungen und
- Aufzeigen von potenziellen Schwachstellen im System.

Die Voraussetzung für die Fehlerbaumanalyse ist die exakte und eindeutige Beschreibung des Systems (Systemanalyse) und die Definition von unerwünschten Ereignissen (z. B. mit Hilfe der FMEA). Ergebnisse der System-FMEA können wiederverwendet werden und liefern Informationen über die möglichen Ausfallarten als Basis für die Fehlerbaumanalyse.

Für die Darstellung eines Fehlerbaumes nach [DIN81] werden die in Abbildung 2.13 gezeigten Symbole verwendet.

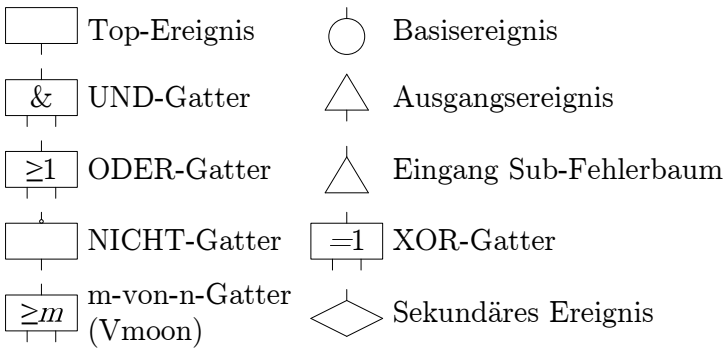

Abbildung 2.13: Grafische Notation für die Fehlerbaumanalyse nach [DIN81]

Ausgehend von einem unerwünschten Ereignis wird untersucht, welche Ausfälle auf der nächst tiefer gelegenen Ebene zu diesem Ereignis führen. Bei konsequenter Fortsetzung bis zum Erreichen eines Basisereignisses (Komponentenausfall) entsteht ein Fehlerbaum. Eine detaillierte Beschreibung zur Vorgehensweise für die Erstellung von Fehlerbäumen ist in DIN 25424 festgelegt [DIN81]. In Abbildung 2.14 ist ein exemplarischer Fehlerbaum für ein System dargestellt.

Die Fehlerbaumanalyse eignet sich sehr gut zur zuverlässigkeits- und sicherheitsrelevanten Darstellung und Analyse von großen komplexen Systemen, die in der Regel aus vielen Minimalschnitten (Ereigniskombinationen die zum TOP-Ereignis führen)

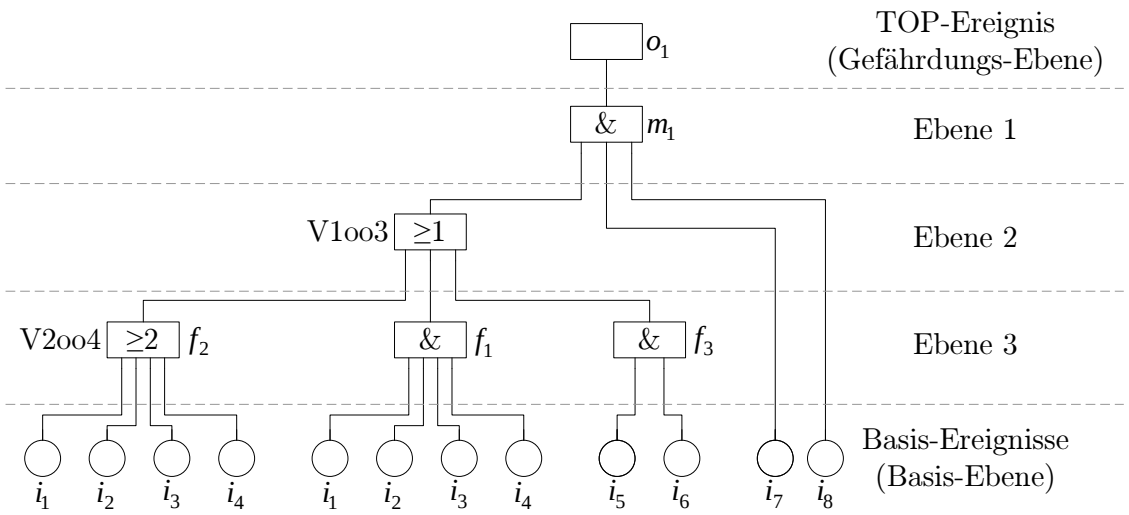

Abbildung 2.14: Fehlerbaum für ein Beispielsystem

bestehen können. Weiterhin eignet sich diese zur Identifikation von kritischen Fehlerpfaden und ermöglicht eine Betrachtung der Auswirkung von Mehrfachfehlern. Die methodische Grundlage für die Auswertung von Fehlerbäumen bildet die Boolesche Algebra. Der Fehlerbaum wird zunächst in eine Boolesche Funktion überführt und anschließend zuverlässigkeitstechnisch ausgewertet.

2.3.5 Boolesche Modellbildung (Boolesche Theorie)

Unter der Voraussetzung das ein technisches System entweder *ausgefallen* oder *funktionsfähig* ist, lässt sich das System in Abhängigkeit von den Komponenten, die ebenfalls auf die Zustände *ausgefallen* oder *funktionsfähig* beschränkt sind, mit Hilfe der Booleschen Algebra beschreiben. Die Boolesche Theorie ermöglicht die Ermittlung des Systemausfallverhaltens unter folgenden Bedingungen:

- Bei dem zu beschreibenden System darf es sich nicht um ein reparierbares System handeln. Der erste Systemausfall beendet somit die Lebensdauer des Systems. Werden reparierbare Systeme betrachtet, ist dies nur bis zum Zeitpunkt des ersten Ausfalles möglich.
- Die Systemkomponenten und das Gesamtsystem können nur die Zustände *ausgefallen* oder *funktionsfähig* annehmen.
- Eine weitere Voraussetzung für die Anwendung der Booleschen Theorie ist die Unabhängigkeit der Systemkomponenten.

Unter Beachtung dieser Bedingungen können mechatronische Systeme bezüglich des Ausfallverhaltens analysiert werden. Die Darstellung der Funktion lässt sich mit Zuverlässigkeitsblockdiagrammen verdeutlichen. Aus diesen Diagrammen ist die

Zuverlässigkeitsstruktur ersichtlich. Im Zuverlässigkeitsblockdiagramm wird gezeigt, wie sich der Ausfall einer Komponente auf das System auswirkt.

Aufgrund der Definition von Fehlerbäumen wird bei Anwendung der Booleschen Theorie die Negativ-Logik verwendet. Diese ordnet dem Ausfall-Zustand den Wert 1 und dem Funktions(fähigen)-Zustand, der als Soll-Zustand angenommen wird, den Wert 0 zu. Die Zustandsmenge $Z_{BM} = \{0, 1\}$ mit $i \in Z_{BM}$ besteht somit aus nur zwei Zuständen [Rak02].

Die Systemkomponenten des zu untersuchenden Systems werden in einer endlichen Menge $C = \{c_1, \ldots, c_i, \ldots, c_n\}$ mit $i \in \{1, 2, \ldots, n\}$, $n \in \mathbb{N}$ zusammengefasst. Die Komponenteneigenschaften werden durch Boolesche Variablen für alle $c_i \in C$ beschrieben. Es gilt folgende Definition:

$$x_i = \begin{cases} 1 & \text{Komponente } c_i \text{ im } \textit{ausgefallenen} \text{ Zustand,} \\ 0 & \text{Komponente } c_i \text{ im } \textit{funktionsfähigen} \text{ Zustand (Soll-Zustand).} \end{cases} \tag{2.1}$$

Der Zustand der Systemkomponenten wird in einem Zustandsvektor $\underline{x} = (x_1, x_2, \ldots, x_n)$ zusammengefasst. Der Boolesche Term ist ein Ausdruck, der die Zustände der Komponenten endlich oft verknüpft. Die dadurch entstehende Abbildung

$$f_{BM} : \{0, 1\}^n \to \{0, 1\} \text{ mit } y = f_{BM}(\underline{x}). \tag{2.2}$$

wird als Boolesche Funktion y definiert. Wobei diese durch unterschiedliche Boolesche Terme dargestellt werden kann.

Boolesche Modelle basieren auf einer algebraischen Struktur mit den zwei Verknüpfungsarten Disjunktion und Konjunktion. Hierfür werden die Operatoren $\vee$ (ODER) sowie $\wedge$ (UND) verwendet. Es besteht die Möglichkeit die Variablen $x_1, x_2, \ldots, x_n$ für $c_1, c_2, \ldots, c_n \in C$ disjunktiv zu y_D mit

$$y_D = x_1 \vee x_2 \vee \ldots \vee x_n = \bigvee_{i=1}^{n} x_i \tag{2.3}$$

zu verknüpfen. Die Funktion y_D hat somit den Wert 1, wenn für x_1 ODER x_2 ODER ein anderes x_i der Wert 1 vorliegt. Die Disjunktion der n Variablen in y_D wird als Maxterm bezeichnet. Die Variablen $x_1, x_2, \ldots, x_n$ können konjunktiv zu

$$y_K = x_1 \wedge x_2 \wedge \ldots \wedge x_n = \bigwedge_{i=1}^{n} x_i \tag{2.4}$$

verknüpft werden. Hier nimmt y_K den Wert 1 nur an, wenn für x_1 UND x_2 UND alle x_i der Wert 1 vorliegt. Um die Schreibweise zu verkürzen wird häufig der $\wedge$-Operator weggelassen und die Variablen einfach hintereinander geschrieben. Die Konjunktion y_K wird im Allgemeinen auch als Minterm bezeichnet. Hat eine Boolesche Variable

den Wert 1, dann ist die negierte Variable 0 und umgekehrt. Dies gilt unter der Voraussetzung, dass für x_i ein komplementäres Element $\overline{x_i}$ existiert. Es gilt

$$y_N = \overline{x_i}. \tag{2.5}$$

Dieser Zusammenhang beschreibt die Negation einer Variablen x_i. Die Funktion y_N hat den Wert 1, wenn für x_i NICHT der Wert 1 vorliegt.

Für die Boolesche Algebra sind folgende Axiome gültig:

- Kommutativgesetz (Vertauschungsregel),
- Assoziativgesetz (Anreihungsregel),
- Distributivgesetz (Mischungsregel) und
- Postulate.

Diese werden in [MP03] näher beschrieben und um die Gesetze

- Idempotenzgesetz,
- Absorptionsgesetz und
- De Morgan'sches Gesetz

erweitert. Weiterhin gilt:

$$0 \vee x_i = x_i \quad 0 \wedge x_i = 0 \quad \text{(kleinstes Element)} \tag{2.6}$$
$$1 \vee x_i = 1 \quad 1 \wedge x_i = x_i \quad \text{(größtes Element)} \tag{2.7}$$
$$x_i \vee \overline{x_i} = 1 \quad x_i \wedge \overline{x_i} = 0 \quad \text{(komplementäres Element)} \tag{2.8}$$

Die Idempotenzgesetze und Absorptionsgesetze bilden eine besondere Eigenschaft der Booleschen Algebra und sind vor allem bei der quantitativen Auswertung von Funktions- und Fehlerbäumen von Bedeutung.

Für die Booleschen Funktionen gibt es einige übersichtliche Darstellungsformen die als kanonische Darstellungen bezeichnet werden. Mit deren Hilfe ist es möglich, anhand einer Funktionstabelle eine äquivalente Boolesche Gleichung zu entwickeln.

Zur Analyse eines Fehlerbaumes ist die Darstellung einer Booleschen Funktion in der Normalform sinnvoll. Die Formulierung von Normalformen bedingt die Definition der Begriffe Maxterm und Minterm. Diese Terme bestehen aus einer Verknüpfung von Literalen, d.h. von nicht-negierten („einfachen“) oder negierten Booleschen Variablen. Wobei auch beide Arten innerhalb eines Terms auftreten können [Rak02].

Die disjunktive Normalform (DN) ist die disjunktive Verknüpfung von Mintermen zu

$$y_{DN} = y_{min_1} \vee y_{min_2} \vee \ldots \vee y_{min_n} = \bigvee_{i=1}^{n} y_{min_i}. \tag{2.9}$$

Die konjunktive Normalform (KN) ist die konjunktive Verknüpfung von Maxtermen mit

$$y_{KN} = y_{max_1} \wedge y_{max_2} \wedge \ldots \wedge y_{max_n} = \bigwedge_{i=1}^{n} y_{max_i}. \tag{2.10}$$

Die ausgezeichnete oder kanonische disjunktive Normalform (ADN) sowie die ausgezeichnete konjunktive Normalform (AKN) sind weitere Normalformen zu Darstellung der Booleschen Funktion. Hier muss jedoch in jedem Minterm der ADN jede Variable des Vektors $\underline{x}$ genau einmal in negierter oder einfacher Form auftreten. Dies hat auch Gültigkeit für den Maxterm der AKN. Die ADN und die AKN unterstützen die Umsetzung einer Wertetabelle, die die Systemeigenschaften wiedergibt, in eine Boolesche Funktion (vgl. [MP03]).

Für die Analyse der Zuverlässigkeit des Systems, ist die Auswertung der Booleschen Gleichungen mit reellen Variablen notwendig. Jede Boolesche Gleichung lässt sich in einen Ausdruck mit reellen Variablen überführen, wenn lediglich die reellen Zahlen 0 und 1 verwendet werden und alle Variablen linear auftreten (Multilinearform). Für die Analyse von Wahrscheinlichkeiten eines Systems werden folgend die Operatoren der Booleschen Funktionen ersetzt. Für die Disjunktion folgt

$$x_i \vee x_j \rightarrow 1 - (1 - x_i) \cdot (1 - x_j) = x_i + x_j - x_i \cdot x_j. \tag{2.11}$$

Für den allgemeinen Fall wird die Disjunktion durch

$$x_1 \vee x_2 \vee \ldots \vee x_n = \bigvee_{i=1}^{n} x_i \rightarrow 1 - \prod_{i=1}^{n} (1 - x_i) \tag{2.12}$$

dargestellt. Die Konjunktion wird durch

$$x_i \wedge x_j \rightarrow x_i \cdot x_j \tag{2.13}$$

beschrieben. Im allgemeinen Fall entsteht durch die Verknüpfung von mehreren Variablen

$$x_1 \wedge x_2 \wedge \ldots \wedge x_n = \bigwedge_{i=1}^{n} x_i \rightarrow \prod_{i=1}^{n} (1 - x_i). \tag{2.14}$$

Ein Negation wird durch

$$\overline{x_i} \rightarrow 1 - x_i \tag{2.15}$$

beschrieben. Alle zuvor genannten Axiome gelten auch für diese Operatoren. Es ist jedoch zu beachten, dass die Gleichungen (2.12) und (2.14) in allen Variablen linear sein müssen (Multilinearform). Dies bedeutet in der Regel, dass das Idempotenzgesetz und das Absorptionsgesetz zu berücksichtigen sind. Mit diesen Zusammenhängen

ist es möglich, von einer Booleschen Funktion in die probabilistische Funktion überzugehen.

Ein technisches System kann in Abhängigkeit von den Zuständen seiner Komponenten mit Hilfe der Booleschen Algebra zuverlässigkeitstechnisch abgebildet werden. Hierzu werden den Komponenten die Zustände *ausgefallen* oder *funktionsfähig* zugeordnet. Das Ergebnis, die Boolesche Funktion, wird zur weiteren Analyse in die Strukturfunktion überführt und zuverlässigkeitstechnisch ausgewertet.

2.3.6 Strukturfunktion

Die im zuverlässigkeitstechnischen Sinne relevanten Booleschen Funktionen werden als Strukturfunktion ϕ bezeichnet. Eine Boolesche Funktion heißt Strukturfunktion $\phi(\underline{x})$ mit

$$\phi(\underline{x}) = \begin{cases} 1 & \text{System ist } \textit{ausgefallen}, \\ 0 & \text{System ist } \textit{funktionsfähig}, \end{cases} \tag{2.16}$$

für alle Variablen $\underline{x} = (x_1, x_2, \ldots, x_n)$ mit dem in Gleichung (2.1) beschriebenen Zusammenhang. Weiterhin sind die Randbedingungen

$$\phi(\underline{x}_a) \leq \phi(\underline{x}_b) \text{ für alle } x_a \leq x_b \tag{2.17}$$

$$\phi(x_1, x_2, \ldots, x_j = 0, \ldots, x_n) \leq \phi(x_1, x_2, \ldots, x_j = 1, \ldots, x_n) \quad \text{für alle } x_i, i \neq j \tag{2.18}$$

$$\phi(\underline{x}) = 0 \text{ , wenn } \underline{x} = (0, 0, \ldots, 0) \tag{2.19}$$

$$\phi(\underline{x}) = 1 \text{ , wenn } \underline{x} = (1, 1, \ldots, 1) \tag{2.20}$$

zu beachten. Eine wichtige Voraussetzung, die die meisten Systeme erfüllen, ist die Eigenschaft der Monotonie, die durch die Gleichungen (2.17) und (2.18) beschrieben wird. Der System-Zustand verbessert sich nicht, wenn sich der Zustand einer Komponente verschlechtert und umgekehrt. Durch diese Voraussetzung wird die Anzahl der in der Zuverlässigkeit anwendbaren Booleschen Funktionen stark eingeschränkt. Es sind lediglich die monoton abnehmenden Funktionen von Interesse. Allgemein gilt ein System als *ausgefallen*, wenn alle Komponenten des Systems *ausgefallen* sind (vgl. Gleichung (2.19)). Umgekehrt gilt Gleichung (2.20), hier ist das System *funktionsfähig*, wenn alle seine Komponenten *funktionsfähig* sind.

Bei vorliegender Strukturfunktion des zu analysierenden Systems kann die Ausfall- und Überlebenswahrscheinlichkeit in Abhängigkeit von den Kenngrößen der Komponenten bestimmt werden. Die Komponentenausfälle dürfen hierbei jedoch nicht in statistischer Abhängigkeit stehen, es dürfen keine zeitlichen Abhängigkeiten existieren. Wenn Reparaturen zugelassen sind, müssen diese ebenfalls unabhängig voneinander erfolgen.

Die Wahrscheinlichkeit die zu einem Ausfall einer Komponente führt, wird mit q_i bezeichnet. Der Erwartungswert der Ausfallwahrscheinlichkeit q_i der Variablen x_i ist durch

$$q_i = P(x_i = 1) = E(x_i) \tag{2.21}$$

definiert. Die Überlebenswahrscheinlichkeit p_i einer Komponenten i wird durch

$$p_i = 1 - q_i \tag{2.22}$$

beschrieben. Die Ausfallwahrscheinlichkeiten werden in $\underline{q} = (q_1, q_2, \ldots, q_n)$ und die Überlebenswahrscheinlichkeiten in $\underline{p} = (p_1, p_2, \ldots, p_n)$ zusammengefasst. Für die Ausfallwahrscheinlichkeit F des Systems gilt

$$F(\underline{q}) = P(\phi(\underline{x} = 1) = E(\phi(\underline{x})) \tag{2.23}$$

und für die Überlebenswahrscheinlichkeit R

$$R(\underline{p}) = 1 - F(\underline{q}). \tag{2.24}$$

Durch diese Betrachtungen wird der Übergang von der Strukturfunktion zur Wahrscheinlichkeitsbetrachtung möglich. Vereinfachend gilt

$$\begin{aligned} x_i &:= q_i(t), \\ \overline{x}_i &:= 1 - q_i(t) = p_i(t), \\ \phi(\underline{x}) &:= F(t), \\ \overline{\phi}(\underline{x}) &:= 1 - F(t) = R(t), \end{aligned} \tag{2.25}$$

wobei anzunehmen ist, dass die Strukturfunktion in Multilinearform[7] vorliegt.

Zuverlässigkeitsblockdiagramme bieten die Möglichkeit, ein System bzw. die Strukturfunktion graphisch darzustellen. Die einfachsten Verknüpfungen von Komponenten sind die Serien- und Parallelverknüpfung.

Für die Berechnung einer Serienstruktur mit n-Komponenten, nach

$$y = \bigvee_{i=1}^{n} x_i \ ; \ \phi(\underline{x}) = 1 - \prod_{i=1}^{n}(1 - x_i), \tag{2.26}$$

ist die Bildung der Booleschen Funktion nicht notwendig, da die Systemkenngrößen

$$F(q) = 1 - \prod_{i=1}^{n}(1 - q_i) \ ; \ R(p) = \prod_{i=1}^{n}(p_i) \tag{2.27}$$

[7]Multilinearform: In der Strukturfunktion $\phi(\underline{x})$ sind die Komponenten häufig mehrfach vorhanden. Aus diesem Grund wird diese unter Anwendung des Idempotenzgesetzes ($x_i^a = x_i$ für alle $i = 1, 2, \ldots, n$ mit $a \in \mathbb{N}$) ausmultipliziert. Nach Anwendung dieser Vorgehensweise liegt die Strukturfunktion in der Multilinearform vor. Die Strukturfunktion $\phi(\underline{x}) = 1 - (1 - x_1x_2)(1 - x_1x_3)$ lautet in der Multilinearform $\phi(\underline{x}) = x_1x_2 + x_1x_3 - x_1x_2x_3$.

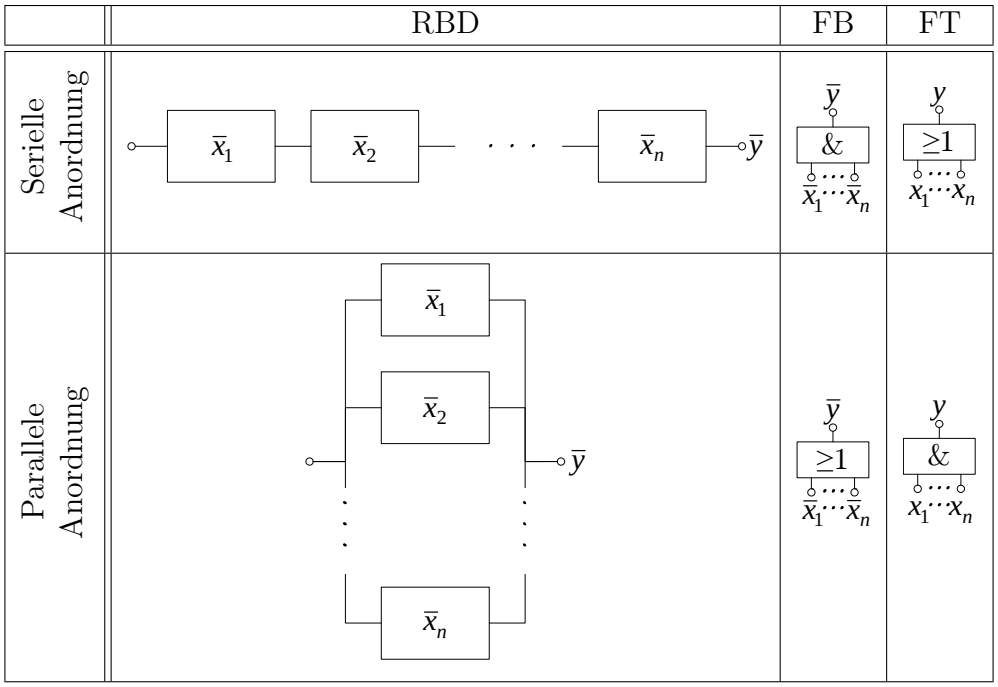

Abbildung 2.15: Darstellung der zuverlässigkeitstechnischen Grundstrukturen Zuverlässigkeitsblockdiagramm (RBD), Funktionsbaum (FB), Fehlerbaum (FT)

direkt bestimmt werden können. Für die Boolesche Funktion y und die Strukturfunktion ϕ einer Parallelschaltung gilt

$$y = \bigwedge_{i=1}^{n} x_i \; ; \; \phi(\underline{x}) = \prod_{i=1}^{n} (x_i). \tag{2.28}$$

Für die Systemkenngrößen folgt:

$$F(q) = \prod_{i=1}^{n} (q_i) \qquad R(p) = 1 - \prod_{i=1}^{n} (1 - p_i) \tag{2.29}$$

Neben der seriellen und parallelen Darstellung sind auch vermaschte Strukturen möglich (z. B. die Brückenstruktur). Bei der Brückenstruktur lässt sich die Zuverlässigkeit nicht mit elementaren Grundgleichungen beschreiben. Bei Systemen mit geringer Anzahl von Komponenten lässt sich die Boolesche Funktion durch die Bildung der disjunktiven Normalform herleiten. Ist das System jedoch durch eine größere Anzahl von Komponenten charakterisiert, erhöht sich der Aufwand für die Bestimmung der Systemfunktion enorm, da die Systemgleichung in Abhängigkeit von der Anzahl der Komponenten n 2^n Terme besitzt. Zur Beherrschung solch komplexer Systeme werden die Methoden

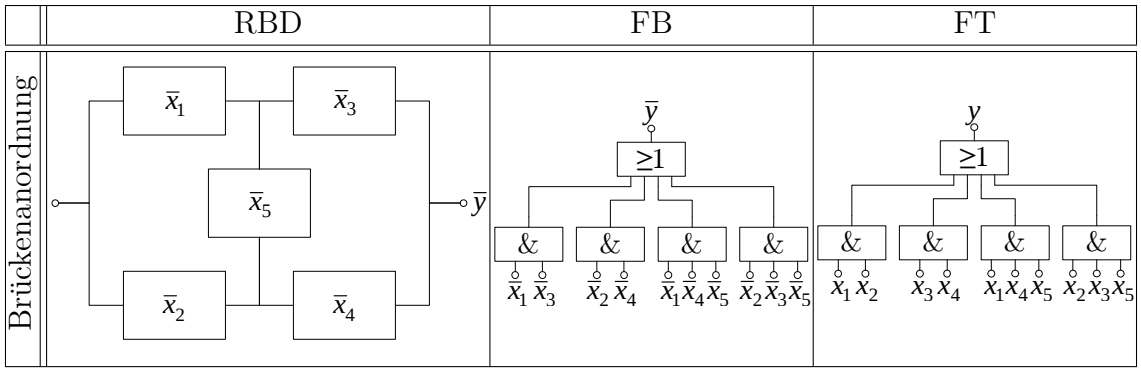

Abbildung 2.16: Zuverlässigkeitstechnische Darstellungen der Brückenanordnung

- minimale Ausfallschnitte (Minimalschnitte),
- minimale Erfolgspfade (Minimalpfade) und
- Separation (Shannonscher Zerlegungssatz)

angewandt. Eine detaillierte Beschreibung dieser Methoden findet sich beispielsweise in [MP03, Rak02, BL04].

Die Grundlage für die Separation bildet der Shannonsche Zerlegungssatz. Für die Strukturfunktion ergibt sich

$$\begin{aligned}\phi(\underline{x}) = x_i\phi(x_1, x_2, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_n) \\ + (1 - x_i)\phi(x_1, x_2, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_n).\end{aligned} \tag{2.30}$$

Hierdurch ist es möglich, die Boolesche Funktionen systematisch zu entwickeln. Unter der Berücksichtigung von $F(q) = E(\phi(\underline{x}))$ gilt für die Ausfallwahrscheinlichkeit F

$$\begin{aligned}F(q) = q_iF(q_1, q_2, \ldots, q_{i-1}, 1, q_{i+1}, \ldots, q_n) \\ + (1 - q_i)F(q_1, q_2, \ldots, q_{i-1}, 0, q_{i+1}, \ldots, q_n).\end{aligned} \tag{2.31}$$

Eine weitere Methode zur Herleitung der Strukturfunktion basiert auf den sogenannten minimalen Ausfallschnitten. Ein Minimalschnitt ist gekennzeichnet durch alle Kombinationen von Komponenten, die bei Versagen oder Ausfall zu einem Versagen des Systems führen. Alle Komponenten werden aufgrund der Negativ-Logik innerhalb der Schnitte durch ein UND, die einzelnen Schnitte werden durch ein ODER verknüpft. Ein Minimalschnitt ist eine Teilmenge aller Komponenten $C_{Sch} \subseteq C$, für die gilt:

$$x_i = 1 \forall c_i \in C_{Sch} \text{ und } x_j = 0 \forall c_j \notin C_{Sch} \Rightarrow \phi(\underline{x}) = 1. \tag{2.32}$$

Das System ist somit *ausgefallen*, wenn alle Komponenten des Minimalschnitts *ausgefallen* sind. Alle restlichen Komponenten, die nicht zu diesem Schnitt gehören, können

funktionsfähig sein und beeinflussen den beschriebenen Systemzustand nicht. Ein Minimalpfad wird hierzu analog beschrieben und ist die Teilmenge aller Komponenten $C_{Pf} \subseteq C$ für die gilt:

$$x_i = 0 \forall c_i \in C_{Pf} \text{ und } x_j = 1 \forall c_j \notin C_{Pf} \Rightarrow \phi(\underline{x}) = 0. \tag{2.33}$$

Das System ist *funktionsfähig*, wenn alle Komponenten eines Minimalpfades *funktionsfähig* sind. Alle restlichen, die nicht zu diesem Pfad gehören, beeinflussen den Systemzustand nicht. Für die Strukturfunktion ϕ bei der Beschreibung des Systemverhaltens durch Minimalschnitte oder Minimalpfade gilt mit dem Laufindex k über die Schnitte oder Pfade

$$\phi(\underline{x}) = 1 - \prod_{C_{Sch}} (1 - \prod_{c_i \in C_{Sch_k}} x_i) = \prod_{C_{Pf}} (1 - \prod_{c_i \in C_{Pf_k}} (1 - x_i)). \tag{2.34}$$

Die Beschreibung der Systemkenngrößen in Abhängigkeit der Komponentenkenngrößen und die Zerlegung des Systems in seine Komponenten wird durch die Boolesche Modellbildung realisiert. Die Komponenten selbst können jedoch einen unterschiedlichen Einfluss auf die Systemstruktur haben. Bei der Serienstruktur ist die Bedeutung jeder einzelnen Komponente wichtiger als bei der Parallelstruktur. Hierzu wurden einige Methoden entwickelt, die die Importanz (Bedeutung) jeder einzelnen Komponenten innerhalb der Struktur beschreiben und quantifizieren. Die wichtigste Komponente ist diejenige, für die sich Verbesserungsmaßnahmen am effektivsten auswirken. Je höher der Importanzwert einer Komponente, desto höher die Bedeutung innerhalb der Struktur.

Obwohl die Boolesche Modellbildung nur die beiden Systemzustände *ausgefallen/funktionsfähig* in der Betrachtung zulässt, ist diese bei der Analyse von Fehlerbäumen von entscheidender Bedeutung.

2.3.7 Importanzkenngrößen

Nach [MP03] ergibt sich in der zuverlässigkeitstechnischen Praxis häufig die Fragestellung welchen Einfluss eine bestimmte Systemkomponente auf die Sicherheit bzw. Zuverlässigkeit des technischen Systems hat. Sind die Einflüsse bekannt, lassen sich Fragestellungen hinsichtlich einer Optimierung, Schwachstellenanalyse, Fehlererkennung, Diagnose, Wartungsstrategien u. a. objektivieren und quantifizieren. Für die Bewertung wurden sogenannte Importanzkenngrößen eingeführt. Importanzkenngrößen quantifizieren die Wichtigkeit einer Komponente innerhalb des Systems hinsichtlich der Zuverlässigkeit bzw. Sicherheit in Bezug auf das Gesamtsystem. Voraussetzung für die Beschreibung der Importanz ist die Analyse der Strukturfunktion ϕ und eventuell das Vorhandensein der Ausfallraten der Systemkomponenten. Es werden verschiedene Typen von Importanzkenngrößen, z. B.

- marginale Importanz (Birnbaum-Importanz),
- strukturelle Importanz,
- fraktionale Importanz,
- diagnostische Importanz (Vesely-Fussel-Importanz),
- kompetitive Importanz (Barlow-Proschman-Importanz),
- sequentielle kontributive Importanz,
- Link Importanz,
- Pfade-Schnitte Importanz,
- Joint Reliability Importanz.
- usw.

für verschiedene Aussagen definiert. Die strukturelle Importanz bestimmt die Bedeutung von Komponenten aufgrund ihrer logischen Anordnung innerhalb des Systems. Gilt für eine Komponente i der Zusammenhang

$$\phi(x_1, x_2, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_n) - \phi(x_1, x_2, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_n) = 1, \quad (2.35)$$

so bestimmt der Zustand dieser den Zustand des Gesamtsystems. Der Vektor $(x_1, x_2, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_n)$ wird als kritischer Vektor des Systems und somit der Strukturfunktion bezeichnet. Hat diese Gleichung keine Gültigkeit, ist der Zustand der Komponente i nicht für den Zustand des Gesamtsystems relevant. Die Anzahl der kritischen Vektoren einer Komponente i ist $n(i)$. Die strukturelle Importanz wird über

$$I_{strukt}(i) = \frac{n(i)}{2^{n-1}} \quad (2.36)$$

definiert. Diese beschreibt den Anteil an kritischen Vektoren einer Komponenten an allen 2^{n-1} möglichen Vektoren des Systems.

Eine ausführliche Beschreibung der Methoden zur Bestimmung der Importanzkenngrößen ist in [MP03, Rak02, WS04] zu finden.

Die bisher beschriebenen Methoden bilden die Grundlage für die Bewertung der Zuverlässigkeit, die für Systeme im automotiven Bereich immer häufiger gefordert wird. Insbesondere wenn es sich um Systeme handelt, die sicherheitskritische Funktionen beinhalten, wird der Nachweis vom Gesetzgeber gefordert. Im folgenden Abschnitt werden die Anforderungen an die Systementwicklung aus dem Bereich der Automobilindustrie näher erläutert.

2.4 Kraftfahrzeugtechnische Sicherheitsanforderungen

In der Vergangenheit diente die internationale Norm IEC 61508 [IEC06] als Grundlage für die Entwicklung von sicherheitsbezogenen elektrischen/elektronischen/programmierbaren

elektronischen Systemen. Diese findet Anwendung bei Systemen die Funktionen ausführen, bei denen ein Ausfall dieser Funktion zu einem Risiko für den Menschen oder die Umwelt führen kann und ist als Sicherheitsgrundnorm ausgelegt. Ursprünglich wurde diese für die Prozessindustrie definiert und bildet die Basis für anwendungsspezifische Sicherheitsnormen. Für die Automobilbranche wird die ISO 26262 [ISO09] als neuer Standard eingeführt, die die IEC 61508 ablöst. Die Norm

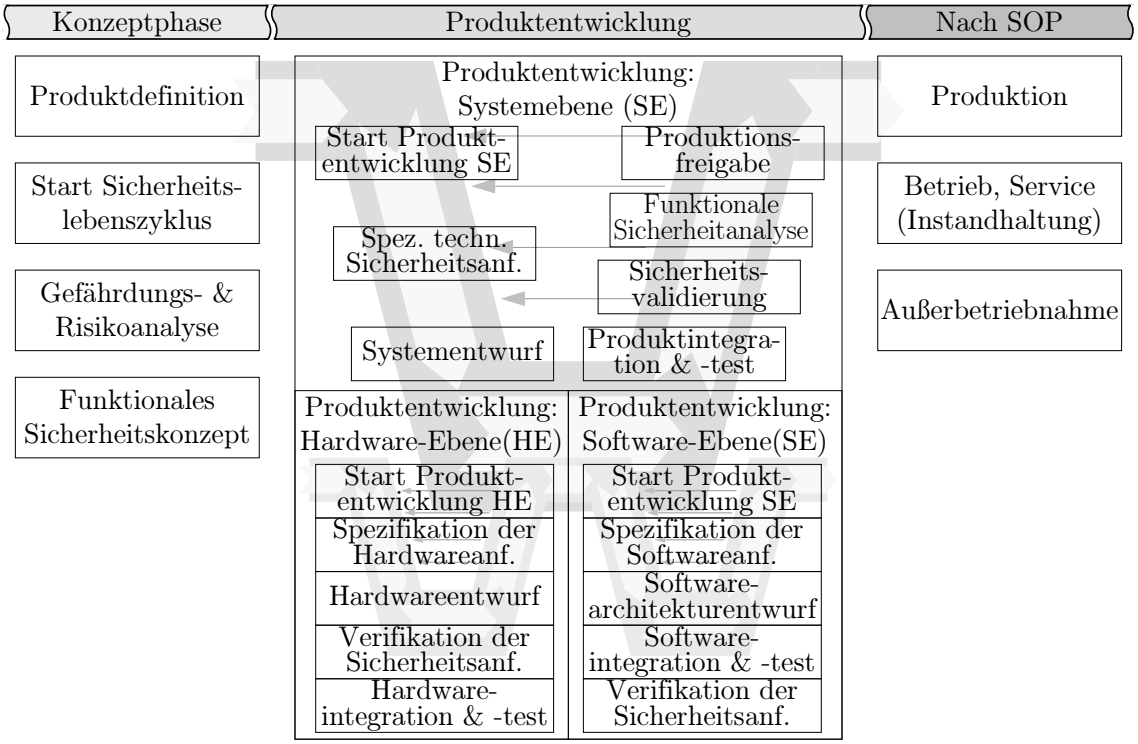

Abbildung 2.17: Kernprozess bei der Entwicklung sicherheitskritischer Systeme nach ISO 26262

berücksichtigt den für die Automobilbranche typischen Lebenszyklus, insbesondere die durchgängige Verifikation bzw. Validation, und die intrinsische Eigensicherheit automobiler Systeme. Des Weiteren werden die für die Automobilindustrie als Standard angenommene Massenfertigung und verteilte globale Entwicklung von Systemen beachtet. Eine wichtige Erweiterung innerhalb dieser Norm ist die Betrachtung von konfigurierbaren Systemen mittels Software (Funktionen).

Die ISO 26262 beschreibt die Anforderungen für den gesamten Produktlebenszyklus sicherheitsrelevanter elektrischer/elektronischer Systeme und definiert einen Sicherheitslebenszyklus. Hierzu wurde ein Kernprozess (Abbildung 2.17) für die Entwicklung sicherheitskritischer Systeme festgelegt. Dieser Zyklus beginnt mit der Konzeptphase und geht über die Systementwicklung, die sich wiederum in Hardware- und Softwareentwicklung aufteilt, über die Produktion hinweg bis zum Service und der Außerbetriebnahme. Unter der Verwendung eines verschachtelten V-Modells

durchläuft jede Ebene (Systemebene, Hardwareebene, Softwareebene) den Entwicklungsprozess.

Der Sicherheitslebenszyklus beginnt mit der Definition des Systems bzw. der Fahrzeugfunktion. Anschließend wird mittels der Gefährdungs- und Risikoanalyse dem System ein „Automotive Safety Integrity Level“ (ASIL) zugeordnet und ein funktionales Sicherheitskonzept erstellt. Die Systementwicklung verfeinert das funktionale Sicherheitskonzept zu einem technischen Sicherheitskonzept. Hieraus werden die Anforderungen an die Hard- und Softwareentwicklung abgeleitet. Das Gesamtsystem wird auf Systemebene zusammengesetzt und für die Produktion freigegeben. Die Entwicklung des Systems erfolgt stets unter Berücksichtigung der Fertigung, des Betriebs, der Instandhaltung und des Services. Diese Phase beginnt mit der Produktion und wird durch die Außerbetriebnahme beendet. Die Außerbetriebnahme beendet den Sicherheitslebenszyklus des Systems.

Der ASIL wird als Maß für die Sicherheitsrelevanz einer Fehlerfunktion eingeführt und über die Parameter „Exposure“, „Controllability“ und „Serverity“ definiert. „Exposure (E)“ beschreibt die Häufigkeit von Situationen, in denen die Fehlfunktion relevant ist. Mit „Controllability (C)“ wird beschrieben, wie gut sich die Situation, im Falle einer Fehlfunktion, beherrschen lässt. Die „Serverity (S)“ beschreibt die Schwere der Auswirkung, wenn die Fehlfunktion in der angenommenen Situation auftritt und nicht beherrscht werden kann. Aus diesen Parametern ergibt sich eine Einstufung für den ASIL auf einer Skala von A bis D. Der ASIL A stellt die niedrigste Einstufung dar und erfordert nur geringe Sicherheitsanforderungen. Die höchsten Sicherheitsanforderungen werden an den ASIL D gestellt. Ist ein System oder Teilsystem in einen ASIL eingestuft, sind die Anforderungen entsprechend der ISO 26262 umzusetzen. Die Einstufung für nicht sicherheitsrelevante Systeme wird durch „Quality Management“ QM dargestellt.

2.5 Konsequenzen für die Bewertung technischer Systeme

Alle Methoden zur Analyse und Bewertung von Zuverlässigkeit und Sicherheit basieren auf einer detaillierten Strukturbeschreibung des Systems. Diese Strukturbeschreibung war in der Vergangenheit häufig nur domänenspezifisch vorhanden und wurde erst nach der Integration zu einem Gesamtsystem geschlossen betrachtet. Die Zuverlässigkeitsanalyse in frühen Entwicklungsphasen wird u. a. in [GJBJ07, JWBG05, NWV03, Gru07] betrachtet. Diese Analysen beziehen sich häufig auf domänenspezifische Aspekte, wodurch die Betrachtung des Gesamtsystems im Verbund der Domänen unter Berücksichtigung der Teilsystemzusammenhänge häufig unberücksichtigt bleibt. Obwohl die Betrachtung von Qualitätsmerkmalen in frühen Entwicklungsphasen notwendig ist, darf die lebenszyklusbegleitende Analyse und Bewertung von Systemen nicht unterschätzt werden. In Garantiefällen oder bei Haftungsfragen ist es wichtig, auf solche Daten zurückzugreifen. Die Analyse

und Bewertung von Systemen unterstützt somit nicht nur den Entscheidungsprozess, sondern kann auch zur Bewertung des Produktes über die Lebenszeit hinweg genutzt werden. Aufgrund von Softwareänderungen kann die funktionale Struktur im Nachhinein verändert werden. Dieser Sachverhalt muss in der Analyse des Systems berücksichtigt werden. Die Anpassung von Software ist jedoch nicht nur in den frühen Phasen der Entwicklung möglich. Diese Änderungen werden im Produkt auch während des gesamten Produktionszeitraumes vorgesehen und ermöglichen so die Änderung des Systems infolge variierender Umfeldbedingungen. Im folgenden Kapitel wird ein allgemeingültiger Ansatz zur domänenübergreifenden Strukturbeschreibung von Systemen dargestellt und die Vorgehensweise durch einen Prozess näher erläutert.

3 Systembewertungsprozess technischer Systeme

Die Bedeutung der Quantifizierung und der Strukturbeschreibung von Systemen wurde im vorherigen Kapitel beschrieben. Die Methoden der Bewertung der Zuverlässigkeit wurden eingeführt und erläutert. Allen Methoden ist eine Systemanalyse, die sich unter anderem auf die Systemstruktur bezieht, vorangestellt. Die Strukturanalyse kann hinsichtlich deren funktionalen, räumlichen und zuverlässigkeitstechnischen Zusammenhänge erfolgen. Die räumlichen Zusammenhänge liefern lediglich einen Überblick über die räumliche Verteilung der Systemkomponenten und ist somit für die hier dargestellte Bewertung des Systems nicht von Bedeutung. Die funktionale Strukturbeschreibung liefert in Verbindung mit der zuverlässigkeitstechnsichen Beschreibung alle notwendigen Informationen für den im Folgenden dargestellten Bewertungsprozess.

Für die Optimierung der Systemkosten ist es wichtig, Fehler in der Systemstruktur so früh wie möglich zu entdecken (siehe Zehnerregel Abbildung 2.11). Hier kann eine präventive Fehleranalyse die Kosten für die Fehlerbehebung minimieren. Durch die Formalisierung des Entscheidungsprozesses ist eine zuverlässigkeitsorientierte automatisierte Analyse möglich. Die Analyse liefert den verantwortlichen Entwicklern und Qualitätsingenieuren notwendige Informationen über den gesamten Produktentstehungsprozesses hinweg. Die Importanzanalyse liefert in Verbindung mit der Sensitivitätsanalyse Informationen über das Optimierungspotential der Komponenten oder der Struktur. In Verbindung mit der Optimierung können Aussagen getroffen werden, die das System verbessern und so zu einer erhöhten Zuverlässigkeit führen.

Ein Vorschlag zur Integration des Bewertungsprozesses in den standardisierten Entwicklungsprozess sicherheitskritischer Produkte der Automobilindustrie wird in Kapitel 5 vorgestellt. In diesem Zusammenhang wird der Optimierungs- und Entscheidungsprozess definiert. Die erwähnte Vorgehensweise etabliert einen adäquaten Systementscheidungsprozess während der Systementwurfsphase und liefert umfassende zuverlässigkeitsorientierte Informationen über die Produktentwicklung.

Wie bereits gezeigt, ist die Qualität eines Produktes entscheidend für den erfolgreichen Absatz. Zur Unterstützung der Produktentwicklung wird im Folgenden ein Systembewertungsprozess eingeführt. Die in Abbildung 3.1 kompakt dargestellte Vorgehensweise zeigt die wesentlichen Elemente des in dieser Arbeit entwickelten Prozesses von der Synthese der Anforderungen bis zur Bewertung der Qualitätsmerkmale.

Der Systembewertungsprozess lässt sich in die Schritte

- Anforderungsanalyse,
- Strukturanalyse,
- Zuverlässigkeitsmodellbildung und
- Quantifizierung

unterteilen.

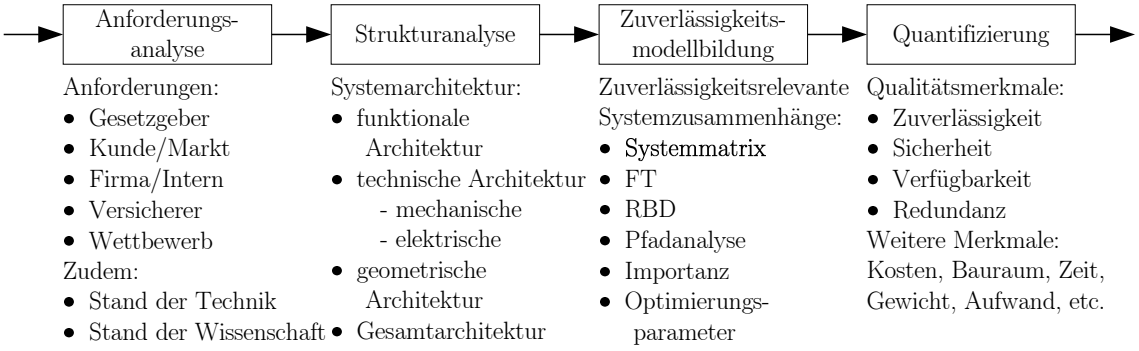

Abbildung 3.1: Darstellung des Systembewertungsprozesses mit den Schwerpunkten: Anforderungsanalyse, Strukturanalyse, Modellbildung und Quantifizierung

Der Prozess beginnt mit der Synthese der Anforderungen und Spezifikationen. Diese Informationen werden anschließend in eine Architektur überführt. Davon ausgehend wird im nächsten Schritt die Systemmatrix generiert. Eine detaillierte Beschreibung dieser Matrix wird in Abschnitt 4.2 durchgeführt. Diese bildet die Basis für den nächsten Schritt, der Quantifizierung. In diesem Schritt wird die Systemmatrix ausgewertet und die den Prozess unterstützenden Informationen generiert. Die Informationen über das quantifizierte System werden als Datensatz in einer Wissensdatenbank abgelegt. Die Wissensdatenbank ermöglicht den Zugriff auf bereits erhobene Daten. Es entsteht ein Baukasten (Kapitel 5) für Systeme, der bei Wiederverwendung von Systemen oder Teilsystemen die notwendigen Informationen liefert.

3.1 Anforderungsanalyse

Der Entwicklung von Systemen ist stets die Anforderungsanalyse vorangestellt. Diese beinhaltet die Synthese der Ansprüche, die von unterschiedlichen Seiten an das System gestellt werden.In der Automobilindustrie werden die Forderungen unter anderem vom Gesetzgeber, Kunden und Kfz-Versicherer gestellt. Weiter werden interne Anforderungen (z. B. Firmenziele), Marktziele (z. B. wirtschaftliche Ziele, Imageziele, Kostenziele) und der Stand der Technik (z. B. Technologieauswahl) berücksichtigt. Alle Forderungen werden in der Anforderungsliste zusammengefasst.

Durch die Anforderungen werden, insbesondere bei sicherheitskritischen Systemen, Vorgaben bezüglich der Zuverlässigkeit festgelegt und Zuverlässigkeitsnachweise gefordert, die sich meist auf das Gesamtsystem oder die Gesamtfunktion beziehen.

Der Systembewertungsprozesses beginnt mit der Anforderungsanalyse und geht anschließend in die Strukturanalyse über . Die Anforderungsanalyse bildet die Grundlage für die Strukturanalyse und legt die wesentlichen Parameter für das System fest.

3.2 Strukturanalyse

Für die zuverlässigkeitstechnische Analyse ist eine Strukturbeschreibung des Systems erforderlich. Das Ergebnis der Strukturanalyse ist eine domänenübergreifende Architekturbeschreibung. Eine erste abstrakte Systemarchitekturbeschreibung kann aus den Systemanforderungen abgeleitet werden. Diese Architektur wird dann im weiteren Lebenszyklus mit Hilfe von Spezialisten detailliert und erweitert.

Die Darstellung und weitere Synthese der Systemarchitektur erfolgt stets unter Berücksichtigung der zuverlässigkeitstechnischen Zusammenhänge (Bewertungsziel). Die Systemarchitektur lässt sich in die

- mechanische Hardwarearchitektur,
- elektronische Hardwarearchitektur,
- Softwarearchitektur und
- Funktionsarchitektur

unterteilen. Unter Berücksichtigung der Systemgrenzen ist es notwendig, zwischen der Gesamtarchitektur, z. B. des Fahrzeugs, und der internen Architektur zu differenzieren. Besteht bei dem zu entwickelnden System keine Schnittstelle nach außen, kann die Systemgrenze vernachlässigt werden. Anders verhält es sich, wenn Schnittstellen nach außen bestehen, wie bei Fahrzeugsystemen: Schnittstellen nach außen sind zu berücksichtigen und in die Analyse und Architekturerstellung mit einzubeziehen. Hierbei kann es sich z. B. um ausgelagerte oder gemeinsam genutzte Sensorik handeln.

Das Gesamtsystem mit allen Komponenten und allen Informationen die notwendig sind, um die funktionalen Zusammenhänge zu analysieren, wird durch die Systemarchitektur beschrieben. Die funktionale Architektur setzt sich aus Hardware- und Softwarekomponenten, die für die Funktionserfüllung notwendig sind, zusammen. Die zuverlässigkeitstechnischen Zusammenhänge für Softwarearchitekturen sind schwieriger zu ermitteln als für Hardwarearchitekturen. Die innere Struktur der programmierbaren Komponenten, die in der Regel als „Black-Box“[1] betrachtet und hinsichtlich einer Strukturanalyse untersucht wird in der Softwarearchitektur beschrieben. Obwohl nach [AUT] und [ISO09] die Architekturbeschreibung über die funktionalen Aspekte hinausgehende Informationen beinhaltet, ist für die Bewertung der Zuverlässigkeit nur die funktionale Beschreibung von Bedeutung. Aus zuverlässigkeitstechnischer Sicht ist die Software als Komponente mit systematischen intrinsischen Fehlern behaftet. Diese Fehler entstehen in der Entwurfsphase oder bei der Umsetzung der Anforderungen. Softwarefehler entstehen nicht während der

[1] Eine „Black-Box“ wird durch die generelle Systemdarstellung beschrieben. Hier sind die Eingabe-Verarbeitung- Ausgabe die Hauptkomponenten. Eingang und Ausgang können als Vektoren auftreten. Es werden die von außen wirkenden und nach außen wirkenden Vektoren betrachtet. Im Gegensatz zum „white box“ Verfahren wird beim diesem Verfahren nicht auf die innere Struktur des Systems geachtet. Eine „white box“ Darstellung ist somit eine Detaildarstellung des Systems.

Betriebsphase des Systems, sondern sind schon bei der Inbetriebnahme im System vorhanden. Aus diesem Grund wird im Folgenden nicht näher auf die Zuverlässigkeitsbewertung von Software eingegangen. Bei der Analyse von Softwarekomponenten werden nur die funktionalen Aspekte berücksichtigt. Es wird die Annahme getroffen, dass Software schon im Vorfeld validiert und fehlerfrei ist. In diesem Zusammenhang wird die funktionale Architektur als Grundlage für die Bewertung der Zuverlässigkeit verwendet.

Die Strukturanalyse wird durch Experten unterstützt. Sie liefern die notwendigen Informationen, die für die funktionale formale Beschreibung benötigt werden. Das Expertenwissen ist insbesondere dann notwendig, wenn das Bewertungsziel nicht mit der Systembeschreibung erfasst werden kann. Dies ist bei zuverlässigkeitstechnischen Informationen der Fall, denn diese werden generell nicht bei der Systembeschreibung erfasst. Die Bewertung der Zuverlässigkeit und Sicherheit der Systemstruktur (vgl. [JKES09]) stehen im Vordergrund. Eine Strukturanalyse ist unter Berücksichtigung der Bewertungskriterien durchzuführen.

Für die Beschreibung von Architekturen werden Architekturbeschreibungssprachen (ADL) verwendet. Diese dienen zur Darstellung der funktionalen Zusammenhänge und berücksichtigen zurzeit keine zuverlässigkeitstechnischen Zusammenhänge. Mit Hilfe von Kommentarblöcken besteht die Möglichkeit, fehlende Informationen der Zuverlässigkeit zu ergänzen. Die anschließende Auswertung erfordert wiederum spezielle Entwicklungswerkzeuge, die die Kosten der Produktentwicklung erhöhen. Ein allgemeingültiger und einfacher Ansatz, der die formale Architekturbeschreibung innerhalb der Spezifikationsfestlegung ermöglicht, ist zu bevorzugen. Zudem erlaubt ein formaler Ansatz die Reproduzierbarkeit der Analyse und den Aufbau einer Wissensbasis. Die Systemarchitekturbeschreibung ist bisher keine standardisierte Vorgehensweise. Zur Systembeschreibung werden häufig Architekturbeschreibungssprachen wie die „System Modeling Language“ (SysML, [Obj10a]) verwendet. Für die Standardisierung der Prozesse im Bereich „Systems Engineering“ wurde 2007 diese einheitliche Modellierungssprache eingeführt. Ein wichtiger Aspekt ist dabei die Unabhängigkeit von den spezifischen Domänen der Mechatronik (Software, Hardware und Mechanik). Der Grund hierfür lag in den Problemen der Kommunikation und des Austauschs der verschiedenen Fach- und Entwicklungsabteilungen. Ziel war es, die einzelnen Domänen (Teillösungen) getrennt über spezifizierte Schnittstellen zu betrachten und später zu einem Gesamtsystem (Gesamtlösung) zusammenzufügen. Voraussetzung hierfür ist ein einheitliches Systemverständnis.

Die Systembeschreibungssprache SysML bildet eine standardisierte Erweiterung der „Unified Modeling Language“ (UML, [Obj10b]) zur Spezifikation, zum Design und zur Verifikation von komplexen Systemen. Die Erweiterung war notwendig, um unter anderem die Modellierung von Anforderungen und Dekompositionsstrukturen, insbesondere der softwareintensiven Komponenten, welche für das „Systems Engineering“ zu detailliert sind, zu unterstützen.

Die Beschreibung von Systemen kann auch mit der „Architecture Analysis and Description Language“(AADL, [Soc10]) erfolgen. Die AADL beschreibt ergänzend zur den funktionalen Zusammenhängen die zeitlichen Abhängigkeiten von Systemen. Im Vergleich zu SysML bietet AADL die Möglichkeit der Simulation während des Systementwurfs. Eine Gegenüberstellung von Vor- und Nachteilen der Sprachen AADL und UML/SysML wird in [Niz07] durchgeführt.

Die Anwendung von Architekturbeschreibungssprachen ist nicht sehr verbreitet und bietet immer noch eine große Hürde für die Anwendung innerhalb aller Entwicklungsdomänen der Mechatronik. Dies ist darin begründet, dass keine einheitliche domänenübergreifende formale Beschreibungssprache existiert. Jede Domäne verwendet eigene Methoden zur Darstellung des Systems. Zudem erlauben die Beschreibungssprachen bisher keine unmittelbare Bewertung der Zuverlässigkeit. Hierzu sind Erweiterungen der Sprache notwendig. Wie in Abschnitt 2.2 erwähnt, ist die Beschreibung von funktionalen Strukturen für die Bewertung der Zuverlässigkeit von Bedeutung. Die Architekturbeschreibungssprachen unterstützen dies und ermöglichen durch die Erweiterungen die Bewertung zuverlässigkeitstechnischer Zusammenhänge.

Diese Zusammenhänge und die Systemstruktur lassen sich durch die Verwendung einer speziellen Matrixdarstellung erfassen (Abschnitt 4.2). Diese Matrix stellt die Eingangs- und Ausgangsgrößen des Systems dar und beschreibt ihre Zusammenhänge aus funktionaler und zuverlässigkeitstechnischer Sicht.

3.3 Zuverlässigkeitstechnische Modellbildung

In der ISO 26262 wird für die Entwicklung von Systemen die Verwendung eines V-Modell vorgeschlagen. Des Weiteren wird eine Vorgehensweise für die Umsetzung und Integration von Funktionen beschrieben. Die einzelnen Entwicklungsstufen werden ebenfalls durch das V-Modell dargestellt und eine generelle Vorgehensweise zur Bewertung und Klassifizierung von Systemen eingeführt. Die Grundlage für die Bewertung der Zuverlässigkeit von Systemen liefert eine Beschreibung der funktionalen Zusammenhänge. Hierzu werden in den jeweiligen Normen keine Vorgaben gemacht. Aufgrund der starken Vernetzung der einzelnen Domänen mechatronischer Systeme ist es schwierig, die jeweiligen Domänen einzeln zu betrachten. Um die Komplexität der Systemanalyse zu minimieren, ist eine Selektion der zu betrachtenden Funktionen sinnvoll. Hierzu bietet sich die in der ISO 26262 beschriebene Gefährdungs- und Risikoanalyse an. Mit der Vorauswahl ist es möglich, den Analyseaufwand innerhalb des Systembewertungsprozesses zu reduzieren.

Unabhängig von der Reduktion der Betrachtungsfälle besteht die Möglichkeit, die Systemarchitektur aufzuteilen. Die Modularisierung kann nach funktionalen oder technischen Aspekten erfolgen. Dies erfolgt stets unter Berücksichtigung der zu betrachtenden Funktion und mit Fokus auf das Bewertungsziel. Für die Beschreibung

der Funktionen ist es erforderlich Hard- und Softwarekomponenten zu berücksichtigen. Die Gesamtfunktion wird durch die Integration der Teilfunktionen erreicht werden. Bei der technischen Aufteilung des Systems wird die Struktur auf Basis der technischen Zusammenhänge in Teilsysteme aufgeteilt. Diese Teilsysteme können später wieder zu einem Gesamtsystem zusammengefügt werden.

Die Systemmatrix bildet den zentralen Punkt des Systembewertungsprozesses. Diese liefert Informationen für die Zuverlässigkeitsbewertung und eine subjektive formale Darstellung des Systems. Gegenüber der informativen Darstellung, wird hiermit der Interpretationsspielraum eingeschränkt und eine semantische Beschreibung ist nur noch vereinzelt notwendig. In Abbildung 3.2 ist die Matrix als zentrales Element der Strukturbeschreibung mit allen notwenidgen Schnittstellenabhängigkeiten dargestellt.

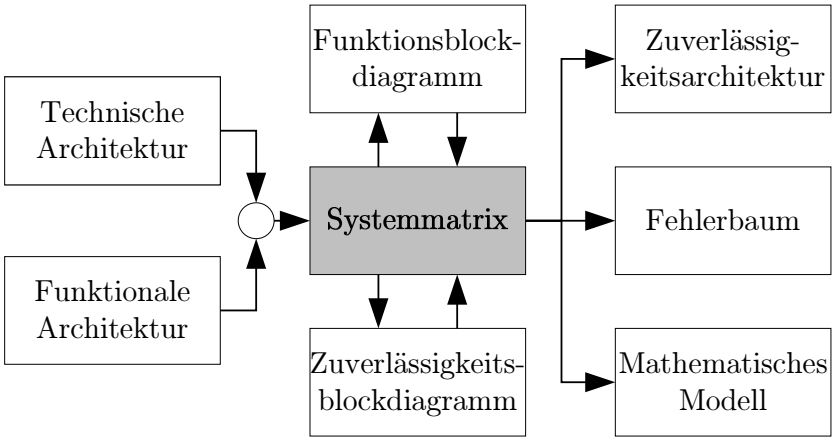

Abbildung 3.2: Die Systemmatrix als zentrales Element im Systembewertungsprozess und deren Überführung in Zuverlässigkeitsmodelle

Informationen zur Ableitung der Systemmatrix können aus verschiedenen Quellen gewonnen werden. Eine Variante erlaubt die Generierung der Systemmatrix durch Nutzung der Architekturbeschreibungen. Sind weitere strukturbeschreibende Dokumente vorhanden, können diese Informationen in die Systemmatrix mit einfließen und die Abbildungsgenauigkeit der Matrix erhöhen. Eine weitere Möglichkeit zum Entwurf der Systemmatrix bietet die Nutzung von Funktionsblockdiagrammen und Zuverlässigkeitsdarstellungen. Zu den Zuverlässigkeitsdarstellungen zählen Fehlerbäume und Zuverlässigkeitsblockdiagramme. Zuverlässigkeitsblockdiagramme und Fehlerbäume lassen sich direkt in die Systemmatrix überführen. Funktionsblockdiagramme bedürfen einer Interpretation der Funktionen und ihren Abhängigkeiten untereinander.

Sind die Dokumente der Zuverlässigkeitsanalyse und die Architekturbeschreibungen bzw. Systembeschreibungen in formaler Form vorhanden, kann der Prozess bei der Überführung der Architektur in die Systemmatrix gestartet werden. Bei einer formalen Beschreibung der Zusammenhänge ist es möglich, eine automatisierte Generierung

der Systemmatrix zu nutzen. In diesem Fall ist die Spezifikationsanalyse bezüglich der zuverlässigkeitstechnischen Zusammenhänge nicht zwingend erforderlich.

Die Systemmatrix basiert auf der Systembeschreibung und kann als Ergebnis der Strukturanalyse gesehen werden. Da diese aufgrund der Analyse mit den Experten alle zuverlässigkeitstechnischen Zusammenhänge enthält, ist eine zuverlässigkeitstechnische Auswertung mit der Matrix möglich. Für die Systemmatrix sind die funktionalen Strukturen von besonderer Bedeutung. Diese Strukturen bringen die einzelnen Systemkomponenten (Hardware, Software) in Bezug zueinander. In der Systemmatrix werden die Ein-/Ausgangsbeziehungen des Systems dargestellt. Eine detaillierte Beschreibung der Systemmatrix wird in Abschnitt 4.2 durchgeführt. Aus dieser werden die Modelle zur Bewertung der Zuverlässigkeit abgeleitet und in ein Zuverlässigkeitsnetz überführt.

Zudem liefern diese Darstellungsformen einen kompakteren Systemüberblick und können für weitere Methoden der Zuverlässigkeitsbewertung herangezogen werden. Mathematisches Modelle zur Ermittlung festgelegter Merkmale, die wiederum den Entscheidungsprozess unterstützen, können basierend auf der Systemmatrix generiert werden. Ein „abstrakter“ Fehlerbaum, aus der Matrix abgeleitet, dient als Vorlage für die detaillierte Fehlerbaumanalyse, die durch das Nutzen von Expertenwissen in eine erweiterte Fehlerbaumanalyse überführt wird. Die hieraus zusätzlich gewonnen Informationen werden in den Entwicklungsprozess zurückgeführt und ermöglichen so eine detailliertere Beschreibung des Systems durch die Systemmatrix. Die Abbildungsgenauigkeit des Systems durch die Systemmatrix wird somit immer präziser. Diese Vorgehensweise ist ein iterativer Prozess innerhalb des Systembewertungsprozesses und wird kontinuierlich durchgeführt.

Sind alle Informationen für die Bewertung der Merkmale erfasst und die Modelle für die Bewertung der Merkmale erzeugt, kann der Systembewertungsprozess mit der Quantifizierung abgeschlossen werden.

3.4 Quantifizierung technischer Systeme

Durch die Beschreibung des Systems mittels der Systemmatrix ist es möglich, eine Analyse bezüglich vorher definierter und in der Systemmatrix erfasster Merkmale durchzuführen. Aus der Systemmatrix lassen sich die notwendigen Modelle zur Quantifizierung dieser Merkmale ableiten. Mathematische Modelle bilden die Grundlage für die Bewertung. Hier ist die Aufteilung in Teilsysteme von Vorteil, da so deren Überführung in das Modell vereinfacht wird und die Merkmale direkt bewertet werden können.

Es können alle Merkmale, die in der Systemmatrix hinterlegt oder aus dieser bestimmbar sind, quantifiziert werden. Zuverlässigkeit und Kosten sind Merkmale, die sich sehr einfach quantitativ bewerten lassen. Andere Merkmale lassen sich jedoch

nur qualitativ oder nur schwer quantitativ bewerten, da hier die Zusammenhänge komplexer und zusätzliche Informationen notwendig sind. Für die Auswahl von Systemkonzepten sind die Merkmale

- Zuverlässigkeit,
- Sicherheit,
- Kosten,
- Bauraum und
- Gewicht

von Bedeutung. Diese können durch den Systembewertungsprozess, wie später in Abbildung 5.2 dargestellt, gegenübergestellt werden und somit die Konzeptauswahl vereinfachen. Durch die qualitative und quantitative Auswertung der Merkmale

- Maß für Optimierbarkeit,
- Kritische Pfade,
- Importanzkenngrößen oder
- Pfadabdeckung

wird der Entwicklungsprozess weiter unterstützt. Die Gegenüberstellung der Merkmale erlaubt den Vergleich von Systemkonzepten und die Identifikation von Änderungen (Delta-Analyse).

3.5 Analyse- und Quantifizierungsprozess im Überblick

Der gesamte Entwicklungsprozess zur Analyse und Bewertung des Systems und seine kausalen Zusammenhänge ist in Abbildung 3.3 dargestellt. Dieser Prozess lässt sich in die drei Phasen

- Synthese → Architekturbeschreibung (Phase 1),
- Architekturbeschreibung → Zuverlässigkeitsmodelle (Phase 2) und
- Zuverlässigkeitsmodelle → Quantifizierung (Phase 3)

unterteilen. Die Phasen sind in sich nicht abgeschlossen und überschneiden sich teilweise. Jede Phase trägt einen Teil zum Quantifizierungsprozess des Systems bei. Phase 1 liefert die Architekturbeschreibung unter der Berücksichtigung der Systemanforderungen. Die Systemmatrix, wie schon in Abbildung 3.2 dargestellt, bildet den zentralen Punkt in Phase 2 und liefert die Informationen für die Zuverlässigkeitsmodelle. Ausgehend von diesen Zuverlässigkeitsmodellen ist es möglich, in Phase 3 eine Quantifizierung des Systems durchzuführen. Informationen dieser Phase werden dem Prozess wieder zugeführt und unterstützen die Quantifizierung von weiteren Systemen.

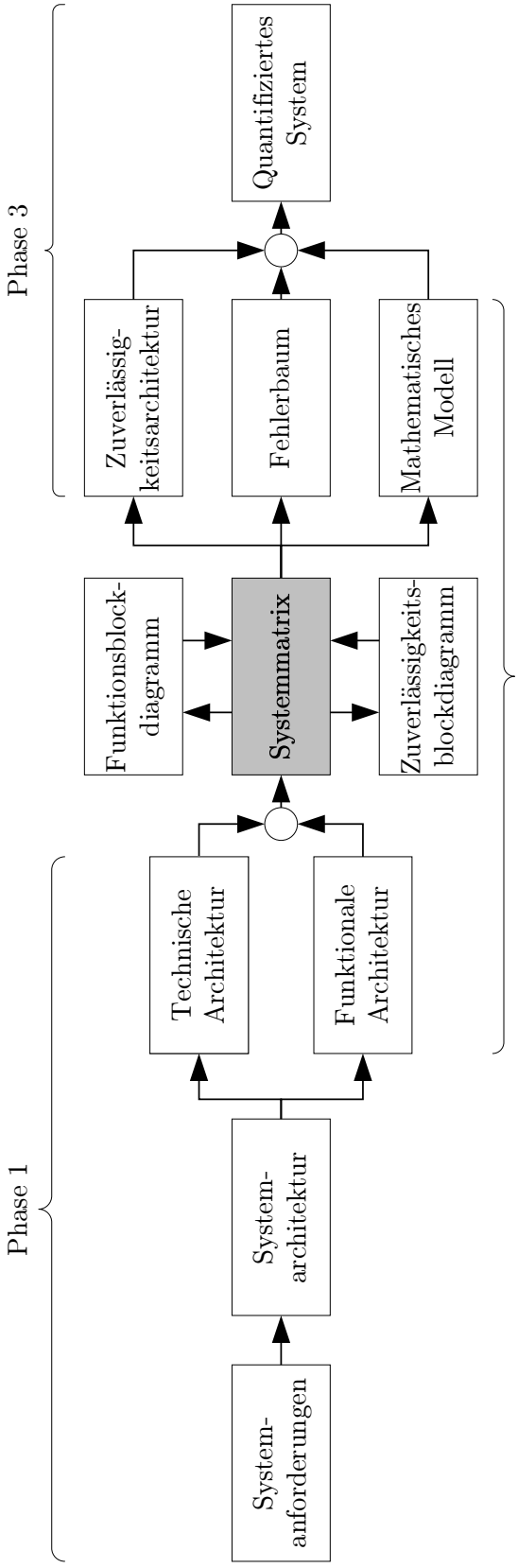

Abbildung 3.3: Synthese, Analyse und Quantifizierung als durchgehender Prozess (Systembewertungsprozess) und ihre kausalen Zusammenhänge
Phase 1: Synthese und Architektur;
Phase 2: Architektur, Systemmatrix und Zuverlässigkeitsmodelle;
Phase 3: Quantifizierung

Die hier eingeführte Vorgehensweise stellt einen iterativen Prozess dar und unterstützt die Strukturanalyse von Systemen. Den zentralen Punkt in diesem Prozess bildet die Systemmatrix. Deren Reifegrad wächst mit jedem Durchlauf der Prozessschritte, die einen Beitrag zur Systemmatrixgenerierung liefern. Besonders die Quantität der Informationen von Experten definiert die Abbildungsgenauigkeit des Systems. Letztendlich liefern die Ergebnisse der Analyse wichtige Informationen für den anschließenden Optimierungs- und Entscheidungsprozess.

4 Analyse und Quantifizierung mechatronischer Systeme

In der Zuverlässigkeitstechnik bildet die detaillierte Strukturbeschreibung die Grundlage für die Analyse und Quantifizierung von Systemen. Die Systemmatrix bildet das zentrale Element des in Kapitel 3 vorgestellten Bewertungsprozesses. Zur Beschreibung von Systemen ist im Vorfeld eine geeignete Darstellung der Systemstruktur notwendig. Im Folgenden wird die in dieser Arbeit verwendete generalisierte Darstellung von funktionalen Zusammenhängen mechatronischer Systeme näher erläutert. Weiterhin wird die Systemmatrix definiert und die Analyse von Systemen unter Verwendung der Systemmatrix dargestellt. Zuverlässigkeitsnetze zur Darstellung der Informationen der Quantifizierung bilden die Grundlage für die weitere Bewertung der Systeme. Unter Verwendung dieser Netze werden die Strukturanalyse und die Pfadanalyse eingeführt.

4.1 Funktionale Darstellung von mechatronischen Systemen

Für die abstrakte domänenübergreifende Darstellung von Systemstrukturen werden sehr häufig Funktionsblockdiagramme verwendet, die eine einfache und übersichtliche Darstellung von komplexen Systemen ermöglichen. Mit Hilfe solcher Diagramme wird das funktionale Verhalten des Systems beschrieben.

Die Grundkomponenten der Systemstruktur werden über Eingangs-, Verarbeitungs- (Modell) und Ausgangsblöcke dargestellt. Ein Eingangsblock ist für die Bereitstellung der Eingabegrößen, die mittels Sensoren erfasst werden können, verantwortlich. Ausgangsblöcke werden durch Ausgabegrößen oder Aktoren repräsentiert. Der Verarbeitungsblock bildet das physikalische Verhalten des Systems ab oder ist für die Regelung/Steuerung des Systems verantwortlich. Dieser ist zentraler Bestandteil des Systems. Das Modell, das im Verarbeitungsblock integriert ist, verknüpft die Eingänge mit den Ausgängen und realisiert die geforderten Funktionen. Funktionsblockdiagramme stellen die Systemfunktionen und ihre Abhängigkeiten untereinander, sowie die globalen Abhängigkeiten übersichtlich dar. Eingang, Verarbeitung und Ausgang bilden auch bei Funktionsblockdiagrammen die zentralen Elemente der funktionalen Systemdarstellung. Ein Vorverarbeitungsblock bildet die Schnittstelle zwischen dem Systemeingang und dem Modellkern (logische Funktionen). Die Signale werden für den Modellkern aufbereitet und angepasst. In der Nachverarbeitungseinheit werden die Signale für den Ausgang vorbereitet.

Für mechatronische Systeme ist dieser Zusammenhang in Abbildung 4.1 dargestellt. Dieses System besteht aus mindestens einem Sensor, einer informationsverarbeitenden Komponente und einem Aktor. Das Modell des Systems wird in den informationsverarbeitenden Block integriert und meist als programmierbare elektronische

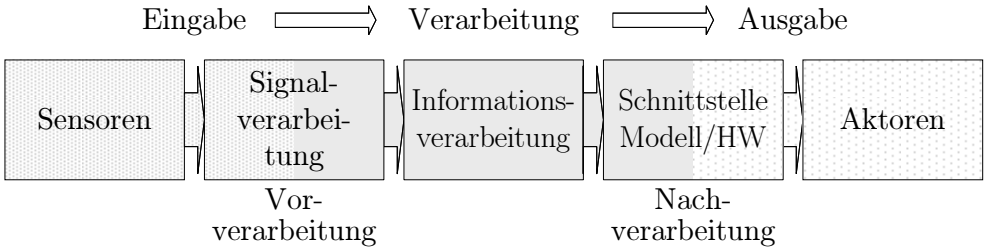

Abbildung 4.1: Verallgemeinerte Darstellung der Systemstruktur eines mechatronischen Systems (Eingang - Verarbeitung - Ausgang)

Komponente umgesetzt. Zudem wird die Software als fester oder variabler (programmierbarer) Bestandteil hierin integriert. Die Eingangsgrößen werden von Sensoren erfasst und an das System weitergeleitet. Im Automobil werden Sensoren zur Erfassung der Umgebung, des aktuellen Fahrzeugzustandes, der Zustandsänderungen und des Fahrerwunsches (nach Abbildung 2.6, 2.7) eingesetzt. Diese Informationen werden an ein Steuergerät weitergeleitet und zur Ansteuerung der Aktoren verwendet. Wird in diesem Zusammenhang eine Bremse betrachtet, so generiert der Fahrer bei Pedalbetätigung einen Verzögerungswunsch. Die Verarbeitungseinheit wertet den Fahrerwunsch, der über die Sensoren erfasst wird, aus und bestimmt unter Berücksichtigung der Umgebungsbedingungen und des Fahrzeugzustandes die notwendige Bremskraft. Aufbauend auf dieser Information regelt die Verarbeitungseinheit die Aktoren und überwacht die Anforderung des Fahrers unter Berücksichtigung des Fahrzeugzustandes.

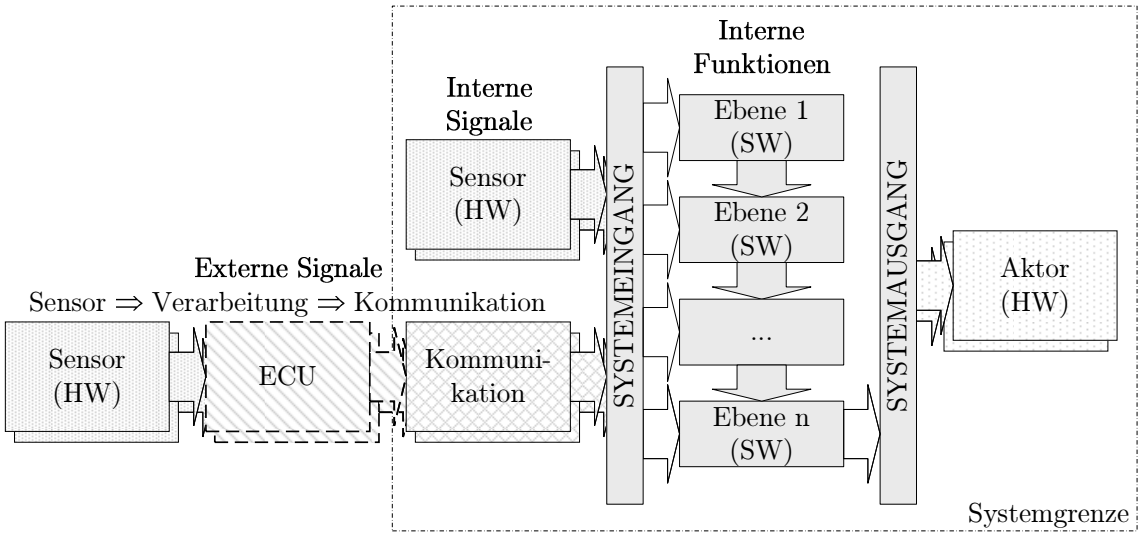

Abbildung 4.2: Generelle Darstellung von Funktionsblockdiagrammen und der inneren funktionalen Struktur

Ein Vergleich mit dem mechatronischen System (Abbildung 2.7) zeigt, dass im Inneren des Systems Energie- und Informationsflüsse als Verknüpfungen der Komponenten dominieren. Da innerhalb der Informationsverarbeitung (interne Funktionen) nur Informationsflüsse auftreten können, benötigt diese Komponente Schnittstellen nach außen. Diese Schnittstellen sind für die Umsetzung des Informationsflusses in Energie- oder Stoffflüsse verantwortlich. Sensoren bilden hier die Eingangsschnittstelle und setzen Energie- oder Stoffflüsse in Informationsflüsse um. Aktoren stellen eine Ausgangsschnittstelle dar, die Informationsflüsse in Energie- oder Stoffflüsse umsetzen. Die informatorischen Flüsse der Informationsverarbeitung werden hierbei durch Aktoren in die jeweilige Größe zur Manipulation des Grundsystems umgesetzt.

In heutigen Systemen werden häufig Informationskanäle für die Übertragung von Sensordaten verwendet. Der Sensor ist nicht mehr in das System integriert sondern ausgelagert. Das von Sensoren gemessene Signal wird in diesem Fall durch eine externe Verarbeitungseinheit verarbeitet (ausgelagerte Signalverarbeitung). Diese Einheit übernimmt die gesamte Auswertung (Vorverarbeitung der Rohdaten, A/D-Wandlung und Plausibilisierung) und übermittelt die vorverarbeitete Information über ein Kommunikationsnetz. Innerhalb der Systemgrenze befindet sich eine Kommunikationskomponente die die Informationen des Kommunikationsnetzes aufnimmt und zur Weiterverarbeitung bereitstellt.

Wird die Software als Komponente eines Systems betrachtet, kann diese als Funktionsblockdiagramm dargestellt werden. Die einzelnen Softwarefunktionen werden als Softwarekomponenten visualisiert. In dieser Arbeit wird die Software als innere funktionale Struktur einer programmierbaren Einheit betrachtet. Diese bringt Eingänge und Ausgänge in Relation zueinander und bildet „virtuelle Abhängigkeiten“ im System. Diese Abhängigkeiten können als logische Verknüpfungen dargestellt werden. Die innere funktionale Struktur wird, um die Komplexität des Systems besser visualisieren und beschreiben zu können, in Ebenen unterteilt (Abbildung 4.2).

Aus zuverlässigkeitstechnischer Sicht ist es schwierig, der inneren funktionalen Struktur (hier: Software) Ausfallraten zuzuordnen. Da die Software auf einer programmierbaren elektronischen Komponente integriert wird, wird die Ausfallrate der Integrationskomponenten vererbt. Die innere funktionale Struktur kann fester oder variabler Bestandteil der Komponente sein. Ein variabler Bestandteil wird in einer variabel programmierbaren Komponente, beispielsweise in einem Prozessor, integriert, erlaubt die Anpassung der Software und ist jederzeit neu programmierbar. Eine fest programmierte Komponente wird durch einen ASIC (engl. „Application Specific Integrated Circuit“, Anwendungsspezifische Integrierte Schaltung) repräsentiert. Ein ASIC ist eine Komponente mit fest integrierten spezifischen Funktionen. Werden logische programmierbare Komponenten betrachtet, steht das funktionale Verhalten im Vordergrund. Die darin integrierte Software wird in diesem Zusammenhang als fehlerfrei angesehen. Fehler, die aufgrund einer falschen Implementierung der

Anforderung entstehen, können durch das Fehlerverhalten gegenüber der Spezifikation entdeckt werden. Systematische Fehler[1] können durch die Beachtung von Programmierstandards und detaillierte Spezifikationen reduziert oder sogar vermieden werden. Ist eine Bewertung der Softwarezuverlässigkeit notwendig gibt es auch die Möglichkeit eine Softwarezuverlässigkeitsanalyse durchzuführen. Hierzu beschreiben Leveson et. al. in [Lev86, LCS91, LH83] Möglichkeiten zur Quantifizierung von Softwarezuverlässigkeit und der Bewertung der Sicherheit von Software.

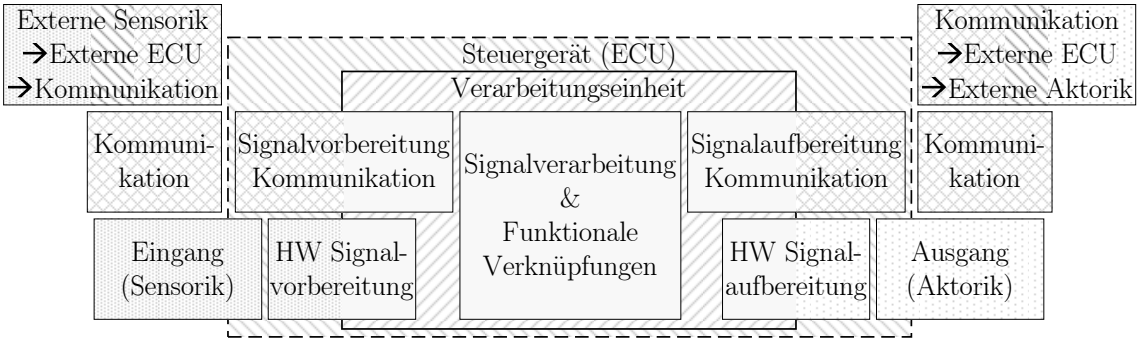

Abbildung 4.3: Symbolik zur Darstellung von Funktionsblockdiagrammen

Nachdem die grundlegenden Eigenschaften der Komponenten eingeführt wurden, wird im Folgenden die allgemeine Darstellung der funktionalen Systemstruktur durch Funktionsblockdiagramme beschrieben. Hierzu ist eine allgemeine funktionale Darstellung von automobilen Systemen zu berücksichtigen. Für die Systemstruktur von Fahrzeugsystemen ergibt sich das in Abbildung 4.4 exemplarisch dargestellte allgemeine Funktionsblockdiagramm. Die Eigenschaftsdarstellung der einzelnen Komponenten wird durch die in Abbildung 4.3 dargestellte Symbolik unterstützt. Eingangskomponenten werden mit i_x gekennzeichnet und bestehen ausschließlich aus Hardware. Wie in Abbildung 4.3 gezeigt, werden Steuergeräte durch Blöcke (Systemgrenze) dargestellt, die die zu dem System gehörenden Komponenten umranden. Da das betrachtete Steuergerät im Detail beschrieben wird, ist hier keine spezielle Kennzeichnung erforderlich. Externe Steuergeräte werden hingegen mit u_x gekennzeichnet. Hier dienen Kommunikationsschnittstellen c_x als Übertragungskanäle. Verarbeitungseinheiten, die wiederum auf den Steuergeräten integriert sind, beinhalten den funktionalen Teil des Systems. Diese Blöcke bilden die Systemfunktionen ab und verknüpfen die Eingänge und Ausgänge miteinander. Funktionen, die auf diesen Verarbeitungseinheiten integriert werden, sind mit f_x gekennzeichnet. Eine Kommunikationsschnittstelle zwischen dem betrachteten System und seiner Umwelt ist in Abbildung 4.4 oben rechts dargestellt. Von besonderem Interesse ist die Wechselwirkung des Systems mit seiner Umwelt. Hierzu wird die Ausgangshardware verwendet. Die Ausgangsgrößen werden mit o_x gekennzeichnet und ausschließlich durch Hardware realisiert. Durch

[1] Als systematische Fehler werden Fehler bezeichnet, die grundsätzlich eine bestimmbare und reproduzierbare Ursache haben.

Blöcke mit der Bezeichnung externe Hardware, können ausgelagerte Aktoren oder Sensoren dargestellt werden, die sich nicht innerhalb des Steuergerätes befinden.

Das Steuergerät (ECU[2]) grenzt das System nach außen ab und bildet so die Betrachtungsgrenze des Systems. Innerhalb der ECU wird das System meist durch elektronische Hardware und informationsverarbeitende Komponenten repräsentiert. Die Hardware außerhalb der ECU wird häufig durch elektromechanische Komponenten realisiert. Eine Systembetrachtung ist daher nur innerhalb der Systemgrenzen notwendig. Sind jedoch Kommunikationsnetze oder komplexe äußere Systemarchitekturen vorhanden, kann es sinnvoll sein, das System über diese Systemgrenze hinaus zu analysieren.

Der innere Teil, von der Verarbeitungseinheit umrandet, bildet die funktionale Einheit des Systems, der meist in Form von Software realisiert wird. Bei der Verarbeitungseinheit handelt es sich in der Regel um eine programmierbare Komponente (Prozessor).

Interne Sensoren sind in der ECU integriert und besitzen eine direkte Schnittstelle zur Verarbeitungseinheit. In manchen Fällen ist eine Signalanpassung notwendig. Hierzu stehen Signalvorverarbeitungsblöcke zur Verfügung.

Alle Knoten und Kanten, die von der ECU eingeschlossen werden, sind Bestandteil des Systems. Diese Bestandteile sind alle in ein Gehäuse und somit einem Steuergerät integriert. Knoten repräsentieren die Funktionen, Eingänge und Ausgänge des Systems. Kanten stellen die Abhängigkeiten untereinander dar. Im mechatronischen Sinne werden die Flüsse innerhalb des Systems beschrieben.

Externe Einheiten können als abgeschlossenes eigenes System betrachtet und separat bewertet werden. Meist liegen die Kenngrößen des externen Systems mit allen Informationen bezüglich der weiteren Merkmalbestimmung vor. Die externe Einheit kann somit als Sensor für das System angesehen werden. Der einzige Unterschied zu einem internen Sensor ist hier die Schnittstelle. Externe Sensoren haben meist eine Kommunikationsschnittstelle. In Einzelfällen kann es wie oben erwähnt sinnvoll sein, detaillierter auf die Systemstruktur des externen Systems einzugehen. Insbesondere wird dies notwendig, wenn für diese Einheiten keine zuverlässigkeitstechnische Informationen vorliegen. Die Darstellung dieser Systeme erfolgt jedoch nach dem gleichen Prinzip, wie hier dargestellt.

Jedem Knoten und somit jeder Komponente können Quantifizierungskenngrößen zugeordnet werden. Für die Bewertung der Zuverlässigkeit kann, mit Ausnahme des

[2]Electronic Control Unit (Steuergerät): Steuergeräte werden im Kraftfahrzeugbereich (Kfz) zur Steuerung und Regelung verwendet. Diese werden in allen erdenklichen elektronischen Bereichen eingesetzt. In Maschinen, Anlagen und anderen technischen Prozessen werden ebenso Steuergeräte eingesetzt. Diese stellen ein eingebettetes System dar und arbeiten nach dem Eingabe-Verarbeitung-Ausgabe Prinzip.

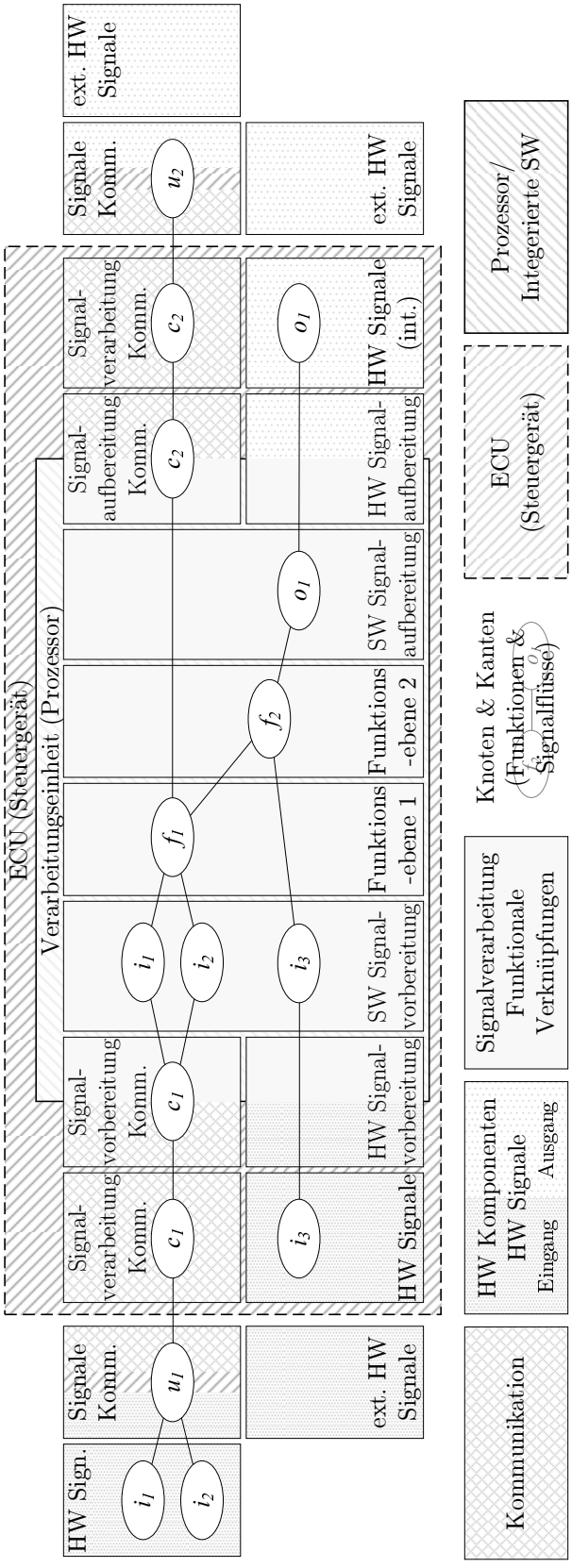

Abbildung 4.4: Exemplarisches Funktionsblockdiagramm zur Veranschaulichung der Komponentenzuordnung

funktionalen Anteils, jedem Knoten eine Ausfallrate zugeordnet werden. Der funktionale Anteil des Systems besitzt kein eigenes Ausfallverhalten sondern ist durch das Ausfallverhalten der Verarbeitungseinheit geprägt. Die zuverlässigkeitstechnischen Kenngrößen von externen Steuergeräten, Informationen von Kommunikationsschnittstellen oder externen Sensoren werden als unabhängige Eingangsgrößen des Systems betrachtet.

Die Darstellung von Funktionsblockdiagrammen ist aufgrund der unterschiedlichen Darstellungarten nicht direkt in Zuverlässigkeitsmodelle überführbar. Die Überführung von Funktionsblockdiagrammen in Zuverlässigkeitsmodelle erfordert aus diesem Grund die Unterstützung von Systemexperten, die die notwendigen und fehlenden Informationen über die zuverlässigkeitstechnischen Zusammenhänge der Struktur liefern. Hierdurch wird es möglich, die funktionalen Zusammenhänge in die zuverlässigkeitstechnischen Zusammenhänge zu überführen.

Aus zuverlässigkeitstechnischer Sicht ist die Analyse der Systempfade ein wichtiges Kriterium für die Bewertung der Redundanz innerhalb des Systems. Die in Abbildung 4.4 dargestellte Kommunikationsstrecke ($u_1 \rightarrow c_1$)zwischen dem externen und dem betrachteten Steuergerät ist nicht redundant ausgeführt. Ist das System auf die Informationen von den Sensoren i_1 und i_2 angewiesen, ist im Falle eines hohen Sicherheitsanspruches mit sicherheitskritischen Folgen zu rechnen. Unter der Annahme, dass die Verbindung zwischen dem Steuergerät u_1 und dem betrachteten Steuergerät fehleranfällig ist, stellt der Pfad $u_1 \rightarrow c_1$ einen kritischen Pfad dar. Es muss sichergestellt werden, dass die Verbindung den sicherheitstechnischen Anforderungen genügt. Die Verbindung kann hierzu redundant ausgelegt oder eine adäquate sichere Information über den Messwert eingeführt werden. Dieser Zusammenhang sollte, bevor weitere Aussagen getroffen werden, bezüglich seiner Kritikalität[3] näher untersucht werden. Es könnte möglich sein, dass die interne Sensorik (im Steuergerät integrierte Sensorik) aus sicherheitstechnischer Sicht ausreichend ist, um die geforderte Information zu generieren.

4.2 Strukturbeschreibung mittels der Systemmatrix

Nachdem die Systemmatrix bereits in Abschnitt 3.3 eingeführt wurde, wird im Folgenden detaillierter auf die Zusammenhänge, die durch diese beschrieben werden, eingegangen. Die Systemmatrix erlaubt eine einfache und leicht verständliche Strukturbeschreibung des Systems. Zudem wird die Bewertung und Analyse des Systems über den gesamten Produktentwicklungszyklus hinweg und über diesen hinaus unterstützt. Mit mechatronischen Systemen verglichen, beschreibt die Systemmatrix die Flüsse innerhalb der Struktur. Hierzu zählen Energieflüsse, Informationsflüsse

[3]Kritikalität: Durch die Kritikalität kann beispielsweise der Einfluss eines Ausfalls einer Komponenten auf das System bewertet werden.

und Stoffflüsse. Die Systemmatrix berücksichtigt durch ihre abstrakte Darstellung alle drei Hauptdomänen der Mechatronik. Zuverlässigkeitstechnische Abhängigkeiten, Kosten sowie qualitative oder wirtschaftliche Merkmale des Systems können durch die Systemmatrix erfasst werden. Hierzu sind lediglich die entsprechenden Informationen zu den Komponenten zuzuordnen. Die Auswertung der Systemmatrix in diesem Kontext bezieht sich hauptsächlich auf die Bewertung von zuverlässigkeitstechnischen Informationen. Aus diesem Grund werden die Kosten und andere Faktoren für die Bewertung vorerst nicht berücksichtigt.

4.2.1 Informationsquellen für die Erstellung

Die Informationsquellen zur Erstellung der Systemmatrix setzen sich aus verschiedenen Wissensdomänen zusammen. Abbildung 4.5 verdeutlicht diesen Zusammenhang.

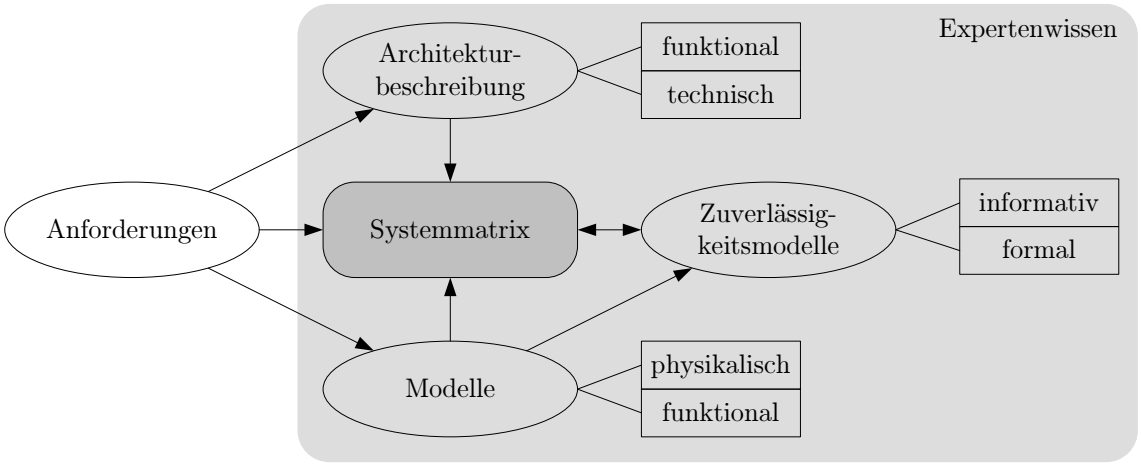

Abbildung 4.5: Informationsquellen zur Erstellung der Systemmatrix

Die Hauptinformationsträger sind die Anforderungen und die Architekturbeschreibungen.

Werden die Architekturbeschreibungen betrachtet, sind für die Analyse der Zuverlässigkeit Ergänzungen durch Experten notwendig. Steht für die Analyse ein Modell zur Verfügung, ist es möglich, die funktionalen Zusammenhänge direkt aus dem Modell zu analysieren und eine Systemmatrix zu erstellen. Ergänzend hierzu lassen sich die Modelle durch geeignete und bekannte Methoden in Zuverlässigkeitsmodelle überführen, die sich wiederum durch eine Systemmatrix abbilden lassen. Hierbei ist das Expertenwissen nicht zwingend notwendig. Die Darstellung der Architektur kann rein funktional oder rein technisch erfolgen. Die funktionale Darstellung beschreibt die Systemfunktionen und ihre Abhängigkeiten, während die technische Architektur die geometrische Verteilung des Systems berücksichtigt. Innerhalb der Modellbildung wird zwischen funktionalen und physikalischen Modellen unterschieden. Die funktionalen Modelle bilden die Systemfunktionen ab. Die Systemfunktion

ermittelt aufgrund der Informationen von Sensoren die Stellgrößen für die Aktoren. Das physikalische Modell bildet das Grundsystem und dessen physikalischen Wirkprinzipien ab. Informative oder formale Darstellungen sind bei der Erstellung von Zuverlässigkeitsmodellen möglich. Informative Darstellungen werden z. B. durch die FMEA repräsentiert. Eine eher formale Darstellung liefert die FTA. Für die Überführung von Zuverlässigkeitsmodellen in die Systemmatrix sind die formalen Beschreibungen von Systemen zu bevorzugen, da diese sich ohne zusätzliches Expertenwissen in die Systemmatrix überführen lassen.

4.2.2 Nomenklatur der Matrix

Zur Beschreibung der zuverlässigkeitstechnischen Zusammenhänge durch eine Systemmatrix wird die in Tabelle 4.1 gezeigte Nomenklatur verwendet. Die Tabelle beschreibt in der linken Spalte die logischen Verknüpfungen bzw. Abhängigkeiten der Komponenten untereinander. Die rechte Seite beschreibt die Art der Komponenten. Diese lassen sich in funktionale Komponenten und Hardwarekomponenten untergliedern. Wird die Systemmatrix zur Beschreibung mechatronischer Systeme

Tabelle 4.1: Nomenklatur der Systemmatrix

Zuverlässigkeitstechn. Beziehungen Logische Verknüpfungen		Komponenten Einheiten	
Symbol	Beschreibung	Symbol	Beschreibung
s	SIGNALPFAD	i_x	Eingang (Sensor)
e	AUSFALL	o_x	Ausgang (Aktor)
p	ENERGIE	f_x	Funktion (Logische)
i	INTEGRIERT	u_x	Einheit
a	UND	m_x	Modul (HW/SW)
o	ODER	c_x	Kommunikation
b	BRÜCKE	p_x	Spannungsversorgung
v	VOTER	e_x	Ausfall(z. B. Steckverbindung)
c	LINK/VERKNÜPFUNG		

verwendet, sind Energieversorgungen der einzelnen Komponenten zu berücksichtigen. Bei der Integration funktionaler Strukturen in informationsverarbeitende Komponenten, werden die logischen funktionalen Strukturen den entsprechenden Komponenten zugeordnet. Die Verknüpfung i beschreibt, in welcher Komponente welche Funktion integriert ist. Besteht das System aus mehreren verteilten Systemen, ist die Kommunikationsstruktur zu analysieren und mit der Systemmatrix zu erfassen.

4.2.3 Erläuterung der Grundstrukturen

Funktionale Strukturbeschreibungen erfordern keine detaillierte Darstellung der Abhängigkeiten zwischen den Komponenten. Wird der Systemzustand aus zuverlässigkeitstechnischer Sicht bewertet, ist eine Beschreibung der funktionalen zuverlässigkeitstechnischen Abhängigkeiten notwendig. Die Systematrix geht von einem funktionsfähigen System aus und beschreibt, welche Komponenten zur Funktionserfüllung notwendig sind. In diesem Zusammenhang werden im Folgenden die Grundstrukturen

- ODER-Struktur,
- UND-Struktur,
- VOTER-Struktur (Diagnose) und
- BRÜCKEN-Stuktur

der Systemmatrix eingeführt. Der generelle Aufbau der Systemmatrix ist in Abbildung 4.6 dargestellt. In der linken Spalte werden die Komponenten mit dem allgemeingültigen Bezeichner beschrieben. Eine Zuordnung entsprechend der Matrixnomenklatur ist in der rechten Spalte gezeigt. Die oberste Zeile beinhaltet wiederum die allgemeingültigen Bezeichnungen, wohingegen in der untersten Zeile die definierte Zuordnung zur Matrixnomenklatur dargestellt wird. Im Inneren dieser Bereiche werden Abhängigkeiten der Komponenten durch die in der Nomenklatur verwendeten Verknüpfungen beschrieben. Diese Verknüpfung spiegelt die zuverlässigkeitstechnischen Zusammenhänge des Systems dar und kann auch zur Darstellung der funktionalen Zusammenhänge genutzt werden.

Die ODER-Struktur wird in der Matrix durch ein o gekennzeichnet. Aus zuverlässigkeitstechnischer Sicht muss mindestens eine Komponente funktionsfähig sein, damit das System funktionsfähig ist. In der Darstellung als Zuverlässigkeitsblockdiagramm wird die ODER-Verknüpfung durch eine parallele Anordnung der Komponenten abgebildet.

Die Verknüpfung der Komponenten durch ein a beschreibt die UND-Verknüpfung. Diese Struktur wird durch die in Abbildung 4.8 dargestellte Matrix beschrieben. Eine Serien-Struktur beschreibt diesen zuverlässigkeitstechnischen Zusammenhang. Alle Komponenten müssen intakt sein, damit das System funktioniert. Dieser Zusammenhang ist nicht zwingend aus der funktionalen Struktur ersichtlich. Dies lässt sich am Beispiel eines Spannungsteilers darstellen, denn hier sind alle Komponenten zur Funktionserfüllung notwendig.

Eine etwas komplexere Struktur wird durch einen Voter (engl. „Voter“= Entscheider oder Mehrheitsentscheider) dargestellt. Der Mehrheitsentscheider vergleicht die Eingangssignale miteinander und entscheidet anschließend, ob die Eingangsinformationen korrekt oder fehlerhaft sind. Aus zuverlässigkeitstechnischer Sicht ist hierbei

Eingang/Signal	Signal/Ausgang	
	Interne Signale und Ausgangssignale (Beschreibung)	in
Interne Signale und Eingagssignale (Beschreibung)	Beschreibung der funktionalen und zuverlässigkeitstechnischen Zusammenhänge unter Verwendung der Grundverknüpfungen	Bezeichner nach Nomenklatur
out	Bezeichner nach Nomenklatur	

Abbildung 4.6: Genereller Aufbau einer Systemmatrix

	y	in
x_1	o	i_1
x_2	o	i_2
...	o	...
x_n	o	i_n
out	o_1	

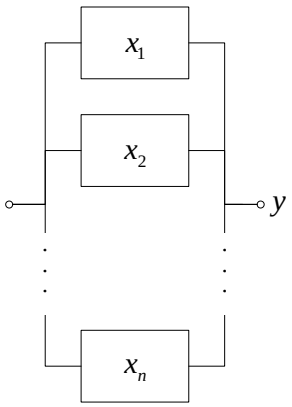

Abbildung 4.7: Systemmatrix einer parallelen Struktur (ODER-Verknüpfung)

eine Aussage zu treffen, wie viele Eingangssignale fehlerfrei sein müssen, damit der Ausgang fehlerfrei ist und somit das System funktionsfähig. Ein Voter wird in der Systemmatrix mit der Verknüpfung vk beschrieben. Wird der Voter mit n Eingangssignalen beschaltet, ergibt sich ein $Vkoon$-Voter. Dies bedeutet, dass mindestens k von n Eingänge fehlerfrei sein müssen, damit das System funktionsfähig ist.

Eine weitere Grundstruktur bildet die Brückenstruktur. Hier gibt es eine Querverbindung innerhalb der Struktur, die sich nicht durch parallele oder serielle Strukturen erfassen lässt. Eine Brückenstruktur wird in der Systemmatrix durch die Variable b_j

	y	in
x_1	a	i_1
x_2	a	i_2
...	a	...
x_n	a	i_n
out	o_1	

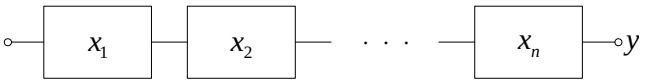

Abbildung 4.8: Systemmatrix einer seriellen Struktur (UND-Verknüpfung)

	y	in
x_1	vk	i_1
x_2	vk	i_2
x_3	vk	i_3
...	vk	...
x_n	vk	i_n
out	o_1	

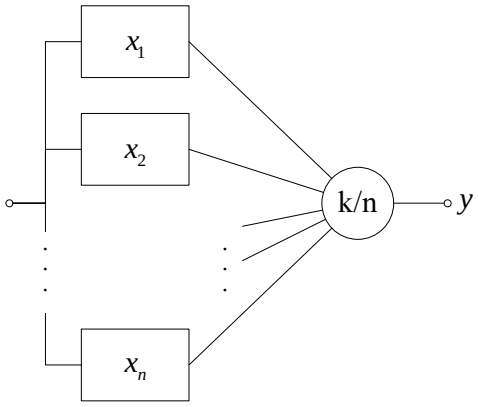

Abbildung 4.9: Systemmatrix der Voterstruktur

	y	in
x_1	b_1	i_1
x_2	b_2	i_2
x_3	b_3	i_3
x_4	b_4	i_4
x_5	b_5	i_5
out	o_1	

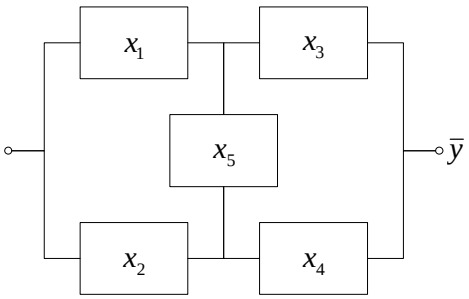

Abbildung 4.10: Systemmatrix der Brückenstruktur

beschrieben. Der Index j beschreibt die Anordnung innerhalb der standardisierten Brückenstruktur.

Zur Reduktion der Komplexität der Strukturbeschreibung wird das System in Teilsysteme aufgeteilt. Diese Teilsysteme können innerhalb der Matrix durch ein Modul erfasst werden. Besitzt das Modul ausschließlich unabhängige Eingangsvariablen, ist eine Separierung des Moduls und eine separate Analyse möglich. Ist dies nicht der Fall, muss das Modul in der Systemgesamtanordnung betrachtet werden. Eine Kombination aus einer seriellen und einer parallelen Struktur ist in Abbildung 4.11

dargestellt. Das Modul m_1 beschreibt in dieser Strukturdarstellung den seriellen Anteil der Zuverlässigkeitsstruktur.

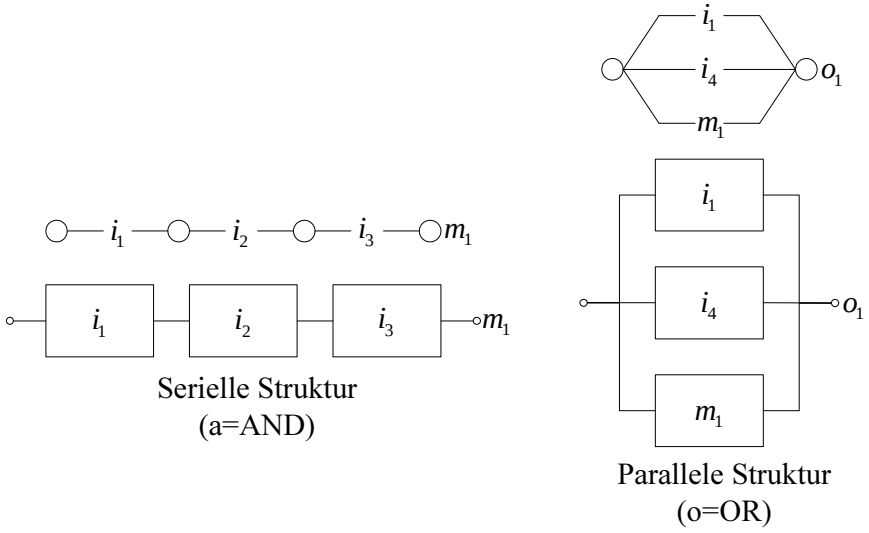

Abbildung 4.11: Dekomposition mit der Systemmatrix

Tabelle 4.2: Systemmatrix für Abbildung 4.11

	m_1	o_1	in
i_1	a	o	i_1
i_2	a		i_2
i_3	a		i_3
i_4		o	i_4
m_1		o	m_1
out	m_1	o_1	

Unter der Verwendung der hier dargestellten Grundstrukturen lassen sich beinahe alle Systemstrukturen durch die Systemmatrix abbilden. Ergänzend hierzu ist es möglich, Übertragungspfade innerhalb der Systemstruktur zu definieren, Energiequellen festzulegen und Fehler zu injizieren.

4.3 Analyse der Systemstruktur

Die Systemstruktur wird wie zuvor beschrieben durch die Systemmatrix beschrieben. Die Systemmatrix bietet den großen Vorteil, dass diese sich einfach analysieren lässt und daraus weitere Darstellungsformen (z. B. Fehlerbäume, RBD's) der Systemstruktur gewonnen werden können.

4.3.1 Überführung in die Strukturfunktion

Mit Hilfe der beschriebenen Grundstrukturen lassen sich Systeme und ihre zuverlässigkeitstechnischen Zusammenhänge darstellen. Die erstellte Systemmatrix erlaubt die Quantifizierung der im Vorfeld festgelegten Merkmalen. Hierzu wird diese in die Boolesche Funktion und anschließend in die Strukturfunktion überführt. Die Grundlagen zur Herleitung von Booleschen Funktionen sind in Kapitel 2.3.5 dargestellt.

Für die Systemmatrix der ODER-Verknüpfung (siehe Abbildung 4.7) ergibt sich die Boolesche Funktion durch

$$y_{\text{or}} = x_1 \wedge x_2 \wedge \ldots \wedge x_n. \tag{4.1}$$

Diese Boolesche Funktion wird in die Strukturfunktion überführt und kann zur Bestimmung der zuverlässigkeitstechnischen Kenngröße

$$\phi(\underline{x})_{\text{or}} = x_1 \cdot x_2 \cdot \ldots \cdot x_n \tag{4.2}$$

genutzt werden.

Die UND-Verknüpfung, die durch die Systemmatrix in Abbildung 4.8 beschrieben wird, bildet mit

$$y_{\text{and}} = x_1 \vee x_2 \vee \ldots \vee x_n \tag{4.3}$$

die Boolesche Funktion. Für die Strukturfunktion ergibt sich

$$\phi(\underline{x})_{\text{and}} = 1 - (1 - x_1)(1 - x_2)\ldots(1 - x_n). \tag{4.4}$$

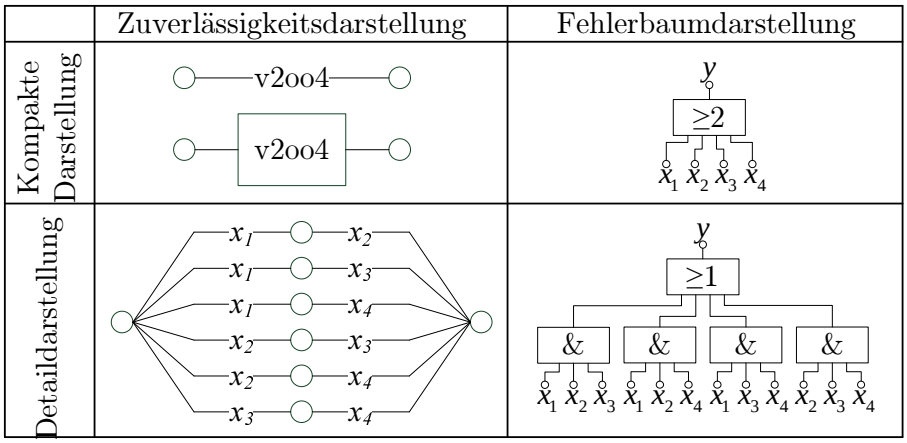

Abbildung 4.12: Zuverlässigkeitstechnische Strukturdarstellungen eines 2-aus-4 Mehrheitsentscheiders (V2oo4)

Einfache Voterstrukturen, wie in Abbildung 4.12 dargestellt, lassen sich auch durch die Grundverknüpfungsarten darstellen. Die Bestimmungsgleichungen der Voterstrukturen aus der Literatur setzen identische Eingangskomponenten voraus. Diese Voraussetzung ist in realen Systemen nicht immer erfüllt und stellt somit einen Sonderfall der Betrachtung dar. Die Entwicklung der Strukturfunktion für die allgemeine Betrachtung von Voterstrukturen ist notwendig. Eine etwas komplexere Voterstruktur kann allgemein durch die Boolesche Funktion unter Verwendung von

$$C_n^j := \left\{ (x_1, x_2, x_3, \ldots, x_n) | x_i \in \{0,1\} \,; \sum_{i=1}^{n} x_i = j \right\} \tag{4.5}$$

beschrieben werden. C_n^j liefert die Kombinationen der Komponenten, die ausfallen müssen, damit es zu einem Systemausfall kommt. Bei einem *koon*-Voter steht k für die Anzahl der mindestens funktionsfähigen Komponenten. Unter Verwendung der Negativ-Logik gilt für j:

$$j = n - k + 1 \tag{4.6}$$

und für die allgemeine Darstellung der Booleschen Funktion eines Voters wird

$$y_{\text{Voter}} = \bigvee_{i=1}^{o} \left(\bigwedge_{x_m \in C_{ni}^j} x_m \right) \quad \text{mit} \quad o = \frac{n!}{k!(n-k)!} \tag{4.7}$$

definiert. Die Strukturfunktion für die Voter-Struktur kann allgemein mit

$$\phi(\underline{x})_{\text{Voter}} = 1 - \left(\prod_{i=1}^{o} \left(1 - \prod_{x_m \in C_{ni}^j} x_m \right) \right) \quad \text{mit} \quad o = \frac{n!}{k!(n-k)!} \tag{4.8}$$

beschrieben werden.

Eine Sonderform der Voterstruktur stellt ein System mit Diagnose dar. In diesem Fall sind die Zusammenhänge nicht mehr in der einfachen Form zu beschreiben. Zur Analyse werden die aus der Literatur bekannten Methoden (z. B. FTA, FMEA) zur Bewertung der Zuverlässigkeit angewendet. Das System wird in diesem Fall bezüglich Wahrscheinlichkeit eines Ausfalls bei Anforderung der Funktion ausgewertet. Hierzu stehen die Wahrscheinlichkeiten für die Entdeckbarkeit eines Fehlers zur Verfügung und werden zur Berechnung der Zuverlässigkeit verwendet. Beinhaltet ein System solche überwachten Funktionen, ist dies meist in einem unabhängigen Teilsystem realisiert. Dies hat den Vorteil, dass die ermittelten Merkmale als vorhanden angenommen werden können.

Eine weitere Sonderform bilden Voterstrukturen mit kalter oder heißer Redundanz. Bei heißer Redundanz sind alle Komponenten der Struktur unter Belastung. In

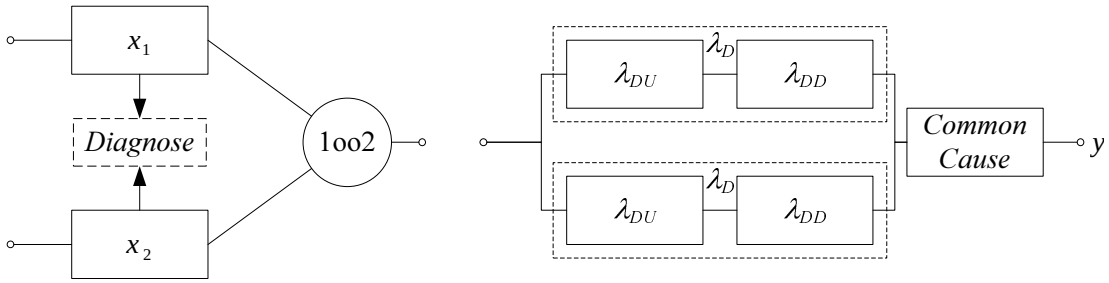

Abbildung 4.13: Voter mit Diagnose
λ_{DU} := Gefährliche unentdeckte Ausfälle
λ_{DD} := Gefährliche entdeckte Ausfälle
λ_D := Gefährliche Ausfälle
„*Common Cause*“ := Weitere Ausfälle aufgrund gemeinsamer Ursache

diesem Fall ist keine besondere Quantifizierung des Systems durchzuführen. Bei kalter Redundanz wird erst bei der Fehlererkennung auf die inaktive Komponente umgeschaltet. Zur Quantifizierung dieses Systems ist es notwendig, die Systeme aufzuteilen und die durch die kalte Redundanz aufgespaltenen Teilsysteme separat zu betrachten. Für die Bewertung der Gesamtausfallwahrscheinlichkeit werden die Ergebnisse der Teilsystembetrachtungen geeignet zusammengeführt.

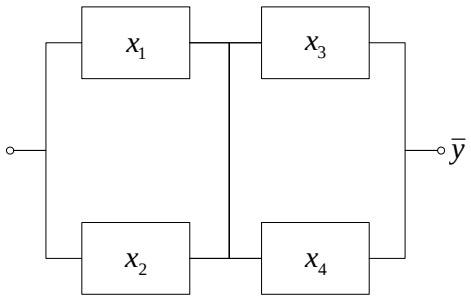

Abbildung 4.14: Separation der Brückenstruktur mit $x_5 = 0$

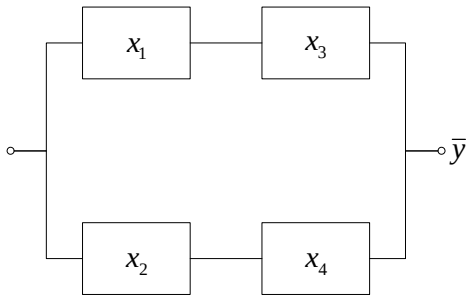

Abbildung 4.15: Separation der Brückenstruktur mit $x_5 = 1$

Die Matrix der Brückenstruktur bildet einen Sonderfall und kann nicht direkt durch eine Boolesche Funktion beschrieben werden. Hier wird die Methode der Separation angewendet und die Strukturfunktion über eine Fallunterscheidung hergeleitet. Im ersten Fall wird die Querkomponente x_5 (siehe Abbildung 4.10) als funktionsfähig und im zweiten als ausgefallen betrachtet. Für die beiden Fälle ergeben sich unterschiedliche Zuverlässigkeitsblockdiagramme. Das Blockdiagramm für den ersten Fall ist in Abbildung 4.15 dargestellt. Für den zweiten Fall ist der Zusammenhang in Abbildung 4.14 dargestellt.

Für die Boolesche Funktion ergibt sich

$$x_5 = 0 : y_{x_5=0} = (x_1 \wedge x_2) \vee (x_3 \wedge x_4) \Rightarrow \phi(x_1, x_2, x_3, x_4, 1) \tag{4.9}$$

$$x_5 = 1 : y_{x_5=1} = (x_1 \vee x_3) \wedge (x_2 \vee x_4) \Rightarrow \phi(x_1, x_2, x_3, x_4, 0) \tag{4.10}$$

Hieraus wird die Strukturfunktion ϕ für die Brückenstruktur nach Gleichung (2.30) über die Strukturfunktion der Brückengleichung

$$\phi(\underline{x}) = x_5\phi(x_1, x_2, x_3, x_4, 1) + (1 - x_5)\phi(x_1, x_2, x_3, x_4, 0) \tag{4.11}$$

beschrieben. Unter Verwendung der Zusammenhänge aus Gleichung (4.10) ergibt sich hiermit

$$\begin{aligned}\phi(\underline{x})_{\text{Brücke}} =& x_5(1 - (1 - x_1)(1 - x_3))(1 - (1 - x_2)(1 - x_4)) \\ &+ (1 - x_5)(1 - (1 - x_1x_2)(1 - x_3x_4)).\end{aligned} \tag{4.12}$$

Bei der Dekomposition der Systemmatrix nach Abbildung 4.11 wird das System in die parallele und serielle Struktur aufgeteilt. Hier bildet das Modul m_1 die serielle Struktur des Systems ab und wird in die parallele Struktur integriert. Für das dargestellte System gilt:

$$y_{\text{Dekomposition}} = o_1 = i_1 \wedge i_4 \wedge m_1 = i_1 \wedge i_4 \wedge (i_1 \vee i_2 \vee i_3). \tag{4.13}$$

Die Boolesche Funktion kann aufgrund der einfachen Struktur schnell in die Strukturfunktion überführt werden. Die Strukturfunktion ist mit

$$\phi(\underline{x})_{\text{Dekomposition}} = i_1i_4m_1 = i_1i_4(1 - (1 - i_1)(1 - i_2)(1 - i_3)) \tag{4.14}$$

beschrieben. Die zuverlässigkeitstechnische Auswertung dieser Strukturfunktion ist durch die doppelt vorhandene Komponente i_1 nicht einfach zu lösen. Hierzu wird die Strukturfunktion unter Anwendung des Idempotenzgesetzes ausmultipliziert.

Die Überführung in die Wahrscheinlichkeitsrechnung $\phi(\underline{x}) \rightarrow F(q)$ ergibt die Ausfallrate des Systems. In manchen Betrachtungsfällen ist es günstiger, die Strukturfunktion in der Positiv-Logik zu betrachten. In diesem Fall ergeben sich folgende Zusammenhänge:

$$\overline{\phi}(\underline{x}) = \begin{cases} 1 & \text{System ist } \textit{funktionsfähig} \text{ (Soll-Zustand)}, \\ 0 & \text{System ist } \textit{ausgefallen}, \end{cases} \tag{4.15}$$

für alle Variablen $\underline{x} = (x_1, x_2, \ldots, x_n)$ mit

$$\overline{x_i} = \begin{cases} 1 & \text{Komponente } c_i \text{ ist } \textit{funktionsfähig} \text{ (Soll-Zustand)}, \\ 0 & \text{Komponente } c_i \text{ ist } \textit{ausgefallen}. \end{cases} \tag{4.16}$$

Mit den in Gleichung (2.25) beschriebenen Ersetzungen folgt für $\overline{\phi}(\underline{x}) := 1 - F(t) = R(t)$ mit $\overline{x}_i := 1 - q_i(t) = p_i(t)$ und die Überführung in die Wahrscheinlichkeitsrechnung die Quantifizierung der Zuverlässigkeit des Systems. Die Strukturfunktionen der einfachen Grundverknüpfungen werden durch

$$\overline{y_{\text{or}}} = \overline{x_1 \wedge x_2 \wedge \ldots \wedge x_n} = \overline{x_1} \vee \overline{x_2} \vee \ldots \vee \overline{x_n} \tag{4.17}$$

$$\overline{\phi(\underline{x})_{\text{or}}} = 1 - (1 - \overline{x_1})(1 - \overline{x_2}) \ldots (1 - \overline{x_n}) \tag{4.18}$$

$$\overline{y_{\text{and}}} = \overline{x_1 \vee x_2 \vee \ldots \vee x_n} = \overline{x_1} \wedge \overline{x_2} \wedge \ldots \wedge \overline{x_n} \tag{4.19}$$

$$\overline{\phi(\underline{x})_{\text{and}}} = \overline{x_1} \cdot \overline{x_2} \cdot \ldots \cdot \overline{x_n} \tag{4.20}$$

beschrieben. Werden diese Gleichungen in die Wahrscheinlichkeitsrechnung überführt, ergeben sich

$$R(p)_{\text{or}} = 1 - (1 - p_1)(1 - p_2) \ldots (1 - p_n) \tag{4.21}$$

$$R(p)_{\text{and}} = p_1 \cdot p_2 \cdot \ldots \cdot p_n \tag{4.22}$$

zur Bestimmung der Überlebenswahrscheinlichkeiten unter Beachtung der Positiv-Logik.

4.3.2 Pfad- und Schnittanalyse

Eine weitere Möglichkeit zur Ermittlung der Strukturfunktion besteht über die Analyse der Minimalschnitte oder Minimalpfade des Systems. Wie in Kapitel 2.3.5 beschrieben, setzt sich die Strukturfunktion, bei Verwendung der Schnitte oder Pfade, nach Gleichung (2.34) zusammen. In diesem Zusammenhang gilt für die Darstellung in der Positiv-Logik folgende Gleichung

$$\overline{\phi(\underline{x})} = 1 - \prod_{C_{Pf}} (1 - \prod_{c_i \in C_{Pf_k}} x_i) = \prod_{C_{Sch}} (1 - \prod_{c_i \in C_{Sch_k}} (1 - x_i)), \tag{4.23}$$

mit $k :=$ Laufindex über die Schnitte oder Pfade. Mit Hilfe der Systemmatrix und der daraus generierten Booleschen Funktion ist es durch die Bildung der konjunktiven Normalform möglich, die Minimalschnitte oder -pfade des Systems zu ermitteln. Hier ist zwischen der Positiv-Logik und Negativ-Logik zu unterscheiden. Bei der Bestimmung über die Positiv-Logik werden durch die konjunktive Normalform die Pfade C_{Pf} des Systems dargestellt und durch die Negativ-Logik werden die Schnitte C_{Sch} des Systems dargestellt. Sind die Zusammenhänge im System nicht mehr durch einfache Strukturen zu beschreiben, besteht die Möglichkeit, das Systemverhalten zu simulieren. Zur Veranschaulichung der zuverlässigkeitstechnischen Zusammenhänge wird die graphische Darstellung durch Zuverlässigkeitsnetze verwendet.

4.4 Strukturbeschreibung durch Zuverlässigkeitsnetze

Mit Hilfe der Grundverknüpfungen lassen sich nun annähernd alle Systeme detailliert beschreiben und auswerten. Die Ablage dieser neu gewonnen Informationen in einer Datenstruktur stellt sich als sinnvoll dar. In dieser Struktur werden zwei Arten von Informationen abgelegt: Diejenigen, die zur Beschreibung des Systems notwendig sind sowie diejenigen, die das Zuverlässigkeitsnetz des Systems beschreiben.

Für die Analyse wird die Matrix soweit iterativ durchlaufen, bis sich die Teilsysteme nicht weiter aufteilen lassen oder der betrachtete Zweig in einem Ereignis / einer Komponente endet. Bei Betrachtung einer Fehlerbaumanalyse werden diese Ereignisse als Basisereignisse bezeichnet. In einer Systemmatrix werden diese Ereignisse durch Eingangsgrößen oder Ausfälle dargestellt. Nicht weiter aufteilbare Teilsysteme werden als „atomare Einheiten“ bezeichnet und können durch die Grundverknüpfungen dargestellt werden.

4.4.1 Informationsstruktur eines Zuverlässigkeitsnetzes

Informationen, die während der Analyse des Systems generiert werden, können zur Wiederverwendung in einer Informationsstruktur abgelegt werden. Hierzu wurde eine Datenstruktur definiert, die alle wesentlichen Informationen über das System enthält. Im Folgenden soll auf die Zusammenhänge innerhalb der Struktur und auf die grundlegenden Annahmen, die für die Berechnung des Zuverlässigkeitsnetzes notwendig sind, eingegangen werden. Ein Zuverlässigkeitsnetz RN fasst alle Einheiten i der in der Systemmatrix beschriebenen Architektur zusammen.

Die einzelnen Elemente der Datenstruktur werden in Tabelle 4.3 dargestellt. Diese Tabelle beschreibt die Datenstruktur, die für jede Komponente i des Systems, die in der Systemmatrix definiert ist, angelegt wird. Der Strukturindex $RN(i).index$ stellt eine eindeutige Kennzeichnung der Elemente innerhalb der Gesamtstruktur dar. Die Zuverlässigkeit der beschriebenen Komponente wird in $RN(i).p$ hinterlegt. Ist für diese in der Systemmatrix keine Zuverlässigkeit hinterlegt, wird diese als $p = 1,0$ angenommen. Dies ist z. B. bei einer rein funktionalen Komponente der Fall. Die während der Kalkulation ermittelte Zuverlässigkeit wird in $RN(i).rel$ hinterlegt. Ist die Ermittlung über die Fehlerrate effizienter, wird entsprechend der Umformung die Zuverlässigkeit durch $1 - RN(i).f = RN(i).rel$ gebildet. In $RN(i).imp$ wird die ermittelte Pfad-Importanz der Komponente abgelegt. Die Kosten von Komponenten und weitere Klassifizierungsmerkmale können in den entsprechenden Feldern hinterlegt werden. Handelt es sich bei der betrachteten Komponente um eine funktionale Einheit (Funktion) die in Software realisiert ist, wird durch $RN(i).int$ die Hardware-Komponente, in der Regel eine Verarbeitungseinheit, beschrieben in der diese integriert ist. Einer Softwarefunktion wird keine Ausfallrate zugeordnet sondern die Ausfallrate des Integrationselements vererbt. Mittels $RN(i).err$ können Fehler

Tabelle 4.3: Datenstruktur der Komponenten innerhalb des Zuverlässigkeitsnetzwerks

Element der Struktur	Beschreibung
$RN(i).sig$	Knotenname
$RN(i).index$	Strukturindex
$RN(i).p$	Zuverlässigkeit der Komponente oder Funktion
$RN(i).rel$	Analysierte Zuverlässigkeit
$RN(i).f$	Analysierte Fehlerrate
$RN(i).imp$	Importanz des Pfades
$RN(i).cost$	Kosten der Komponente oder der Funktion
$RN(i).par_1$	Erweitertes Entscheidungsmerkmal 1
$RN(i).par_2$	Erweitertes Entscheidungsmerkmal 2
...	...
$RN(i).par_n$	Erweitertes Entscheidungsmerkmal n
$RN(i).int$	Funktionale Integration im System
$RN(i).pow$	Energieversorgung
$RN(i).error$	Fehlerinjektion
$RN(i).down$	Informationen über die untergeordnete Ebene (zum Knoten)
$RN(i).up$	Informationen über die übergeordnete Ebene (vom Knoten)
$RN(i).link$	Zuverlässigkeitsrelation (Logische Relation)
$RN(i).path$	Signalpfade (Redundanz)

dargestellt und näher betrachtet werden. Die Systemstruktur ist meist hierarchisch aufgebaut. Jeder Knoten besitzt somit mindestens einen Nachfolger ($RN(i).down$) und einen Vorgänger ($RN(i).up$). Ausnahmen bilden hier die betrachtete Systemfunktion und die Eingänge des Systems. Die Systemfunktion besitzt nur Nachfolger und steht an der Wurzel bzw. Kopf der Datenstruktur. Diese Zusammenhänge werden in Abschnitt 4.4.2 anhand einer graphischen Darstellung deutlich erkennbar. Die Abhängigkeiten zwischen den Komponenten und Funktionen werden aus zuverlässigkeitstechnischer Sicht durch $RN(i).link$ beschrieben. Handelt es sich bei dem betrachteten System um ein verteiltes System, sind ergänzend die Kommunikations-/Informationspfade von zu betrachten. Hierbei werden die Pfade der Signale vom Ursprungsort bis zum Verwendungsort identifiziert. Ergänzende Informationen, die für die Bewertung des Systems notwendig sind, werden von der gewünschten Merkmalanalyse vorgegeben und sind in die Struktur aufzunehmen ($RN(i).par_n$).

Die Informationen der Systemmatrix sind zur Auswertung der Qualitätsmerkmale und für die Beschreibung des Systems ausreichend. Angaben zu quantifizierten Kenngrößen und die Ergebnisse der Analyse erweitern die Systemmatrix und werden in der dargestellten Struktur zusammengefasst. Die hinzugewonnenen Informationen stehen ohne erneute Berechnungen in einer Datenbasis für die weitere Analyse zur Verfügung.

Die Datenstruktur zur formalen, mathematischen Beschreibung der Zuverlässigkeitsnetze bietet sich aufgrund ihrer kompakten Darstellung auch zur Archivierung in einer Datenbank an. Die Ergebnisse der Analyse können in dieser Datenbank abgelegt werden, um bei späteren Auswertungen wieder auf diese Informationen zurückzugreifen. Es entsteht eine Wissensdatenbank, die Informationen über bereits quantifizierte Systeme enthält und die Entwicklung insbesondere bei Wiederverwendung von Teilsystemen unterstützt.

4.4.2 Graphische Darstellung der Zusammenhänge

Tabellen und Datenbanken sind jedoch für eine einfache Darstellung und Beschreibung des Systems ungeeignet, da diese sehr komplex und unlesbar wirken. Eine Überführung der Zuverlässigkeitsnetze in eine grafische Darstellung unterstützt die Lesbarkeit.

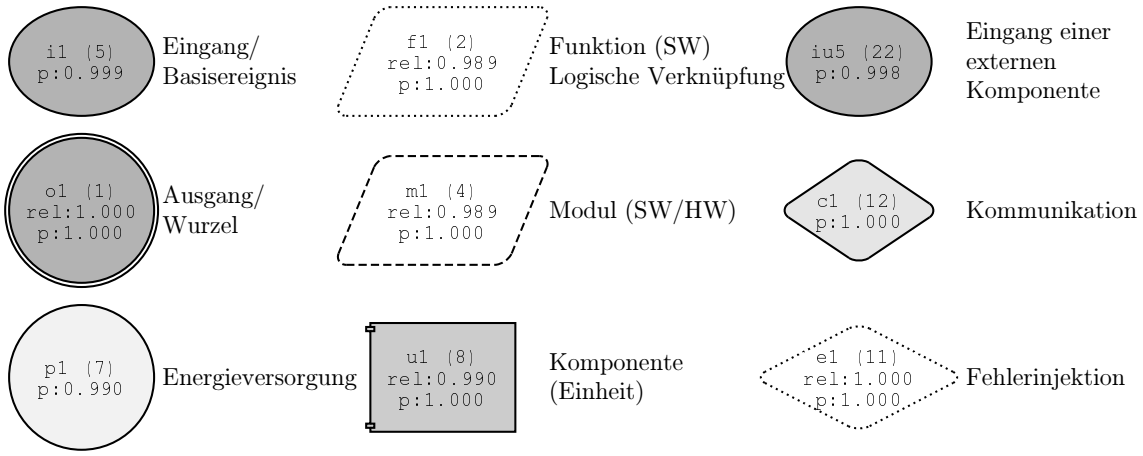

Abbildung 4.16: Nomenklatur der Zuverlässigkeitsnetze

Das Zuverlässigkeitsnetz bildet eine in sich geschlossene Struktur und kann allein durch eine Systemmatrix beschrieben werden. Die Auswertung des Zuverlässigkeitsnetzes erfolgt durch die Auswertung der Informationen der Systemmatrix. Die Ergebnisse der Auswertung werden in einer Struktur abgelegt. Die graphische Darstellung erfolgt mit der in Abbildung 4.16 gezeigten Nomenklatur. Vorerst steht die Beschreibung der zuverlässigkeitstechnischen Zusammenhänge im Vordergrund.

Für die Grundstrukturen ergeben sich die im Folgenden dargestellten Zuverlässigkeitsnetze (Abbildung 4.17 bis 4.20). Die Knoten bilden die Komponenten des Systems ab. Diese sind über die Kanten, die die Abhängigkeiten der Knoten untereinander beschreiben, miteinander verknüpft. Die graphische Darstellung des Zuverlässigkeitsnetzes ist direkt aus der Systemmatrix abgeleitet und dient der Übersichtlichkeit. Jeder Knoten, der in diesem Netz dargestellt ist, wird während der Analyse bezüglich der definierten Merkmale bewertet. Die Information zu jedem Knoten ist in der

Datenstruktur des Zuverlässigkeitsnetzes hinterlegt. Die drei unterschiedlichen logischen Strukturen sowie die Brückenstruktur unterscheiden sich dabei hinsichtlich der Kanten.

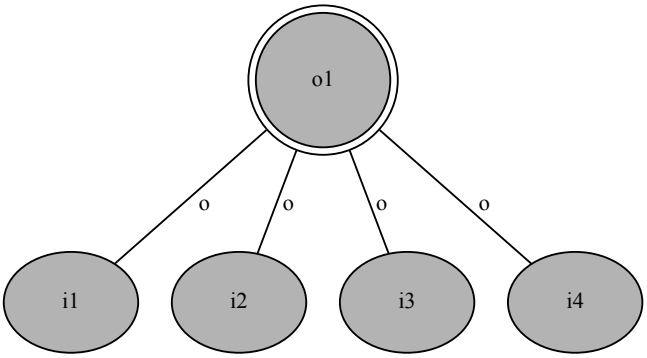

Abbildung 4.17: Zuverlässigkeitsnetz der ODER-Struktur

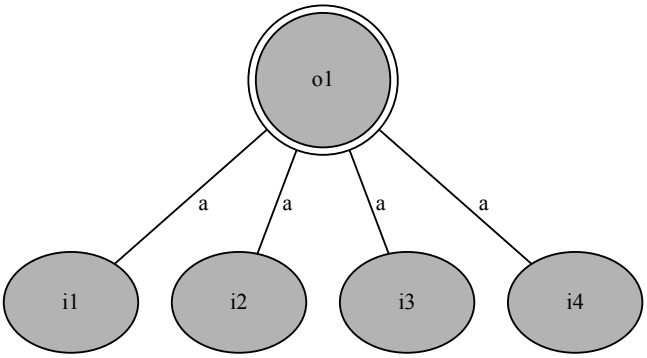

Abbildung 4.18: Zuverlässigkeitsnetz der UND-Struktur

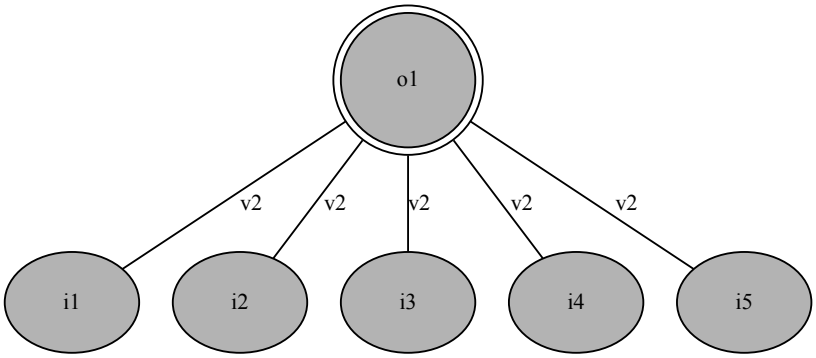

Abbildung 4.19: Zuverlässigkeitsnetz der VOTER-Struktur

Die graphische Darstellung ist selbst bei sehr komplexen Systemen intuitiv und sehr gut überschaubar. Die hier dargestellten Zuverlässigkeitsnetze sind mit der

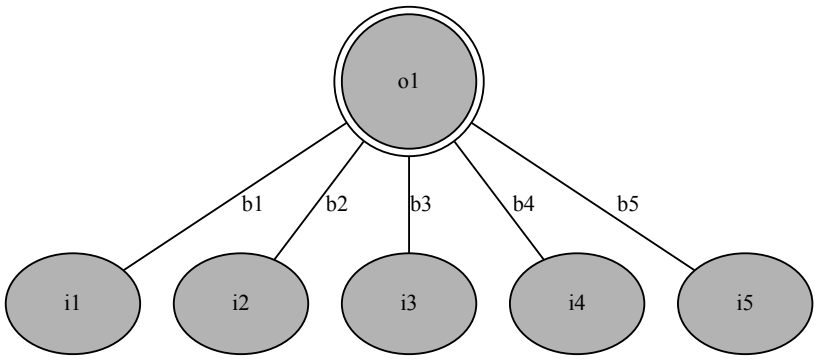

Abbildung 4.20: Zuverlässigkeitsnetz der BRÜCKEN-Struktur

Beschreibung durch einen Fehlerbaum vergleichbar. Aus diesem Grund lassen sich diese Netze direkt in ein Fehlernetz (invertierte Logik) oder einen Fehlerbaum überführen.

Zuverlässigkeitsnetze bieten aus zuverlässigkeitstechnischer Sicht einen guten Überblick über die Systemstruktur, die Abhängigkeiten der Komponenten, das Ausfallverhalten und die quantifizierten Merkmale. Zuverlässigkeitsnetze erweitern somit die Darstellungsmöglichkeiten der Systemstruktur. Die Art der Veranschaulichung kann sowohl zur Visualisierung von komplexen Strukturen oder von Teilsystemen verwendet werden. Bei der Aufteilung der Systemfunktion in Teilfunktionen und somit -systeme wird jedes einzelne System in einem Zuverlässigkeitsnetz beschrieben und später wieder zu einem Gesamtsystem integriert.

Tabelle 4.4: Dekomposition von Zuverlässigkeitsnetzen

	RN 1	RN 2	RN 3	RN4	out2	out4	out8	in
$in1$	f							i_1
$in2$		f						i_2
$in3$		f						i_3
$in4$	f							i_4
$in5$	f							i_5
$RN1$		f_{m_1}	f_{m_1}, f_{m_2}					rn_1
$RN2$				f_{m_3}, f_{m_4}				rn_2
$RN3$				f_{m_5}	f_{o_2}	f_{o_4}		rn_3
$RN4$							f_{o_8}	rn_4
out	rn_1	rn_2	rn_3	rn_4	o_2	o_4	o_8	

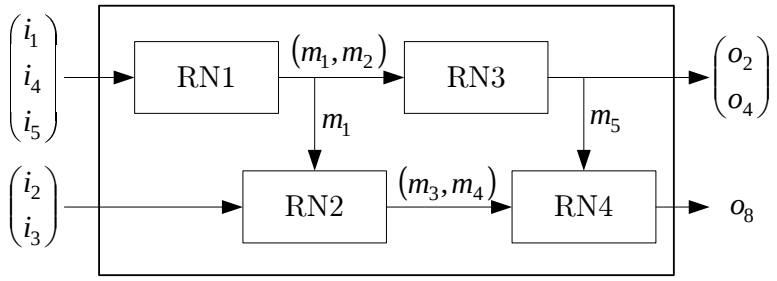

Abbildung 4.21: Dekomposition von Zuverlässigkeitsnetzen

Die Zusammenhänge aus Tabelle 4.4 lassen sich formal wie folgt beschreiben.

$$RN1 = f(i_1, i_4, i_5) \quad \text{liefert } (m_1, m_2) \tag{4.24}$$
$$RN2 = f(i_2, i_3, m_1) \quad \text{liefert } (m_3, m_4) \tag{4.25}$$
$$RN3 = f(m_1, m_2) \quad \text{liefert } (o_1, o_4, m_5) \tag{4.26}$$
$$RN4 = f(m_3, m_4, m_5) \quad \text{liefert } (o_8) \tag{4.27}$$

mit

$$m_1 = f(i_1, i_4) \quad \rightarrow \text{RN 1}, \tag{4.28}$$
$$m_2 = f(i_1, i_4, i_5) \quad \rightarrow \text{RN 1}, \tag{4.29}$$
$$m_3 = f(i_1, i_4, i_5, i_2, i_3) \quad \rightarrow \text{RN 2}, \tag{4.30}$$
$$m_4 = f(i_1, i_4, i_5, i_2, i_3) \quad \rightarrow \text{RN 2} \tag{4.31}$$
$$m_5 = f(i_1, i_4, i_5, i_2, i_3) \quad \rightarrow \text{RN 3} \tag{4.32}$$

Für die zusammengesetzen Systeme ergibt sich eine Gesamtstrukturfunktion, die sich aus der Strukturfunktion der Einzelsysteme zusammensetzt. Die Abhängigkeiten der Teilsysteme und eine mögliche Beschreibung ist in Abbildung 4.21 und in Tabelle 4.4 aufgezeigt. In der Tabelle werden die Abhängigkeiten der einzelnen Teilsysteme durch ein f symbolisiert. Das Zuverlässigkeitsnetz ist somit abhängig von den Größen i_1, i_4, i_5. Das Modul m_1 ist jedoch nur von den Größen i_1, i_4 abhängig. Dieser Zusammenhang wird aus der detaillierteren Beschreibung der Abhängigkeiten aus den Gleichungen (4.29) bis (4.32) ersichtlich. Die Tabelle ist folgendermaßen zu lesen. Das Netz rn_1 stellt eine Funktion von i_1, i_4 und i_5 dar. Dieses Netz stellt die Module m_1 und m_2 bereit und liefert die Informationen an die Netzwerke rn_2 und rn_3. Das Netzwerk rn_2 verwendet nur die Größen aus Modul m_1 (f_{m_1}), das Netzwerk rn_3 hingegen die Ergebnisse aus den Modulen m_1 und m_2 (f_{m_1}, f_{m_2}). Mit Hilfe dieser Darstellungsform ist es möglich, große komplexe Netze zu modularisieren, die Komplexität abzubilden und formal zu beschreiben.

4.5 Empirische Ermittlung des Ausfallverhaltens

Für die Darstellung der Struktur durch Fehlerbäume werden die Minimalschnitte der Struktur ermittelt und durch die Methoden aus Kapitel 2.3.5 in entsprechende mathematische Beschreibungen umgewandelt. In diesem Zusammenhang werden Entscheidungsbäume häufig zur grafischen Darstellung von Systemen verwendet. Diese Form der Darstellung wird durch eine empirische Ermittlung des Ausfallverhaltens des Systems hergeleitet. Ausfallkombinationen, die zum Ausfall des Gesamtsystems führen, werden in der Struktur der Zuverlässigkeitsnetze zusammengefasst.

Die zuverlässigkeitstechnischen Zusammenhänge der Strukturfunktion lassen sich in Fehlerbäume und Zuverlässigkeitsblockdiagramme überführen. Zuverlässigkeitsblockdiagramme bilden die Grundlage für die Optimierung nach [KJS10]. Bei komplexen Strukturen kann es sinnvoll sein, die Zuverlässigkeitsstruktur empirisch zu ermitteln. Hierbei wird von dem schlechtesten anzunehmenden Fall ausgegangen. Alle Komponenten werden durch eine UND-Verknüpfung (seriell) miteinander verbunden. Durch die gezielte Einprägung von Fehlern und der Analyse der Systemreaktion ist es möglich, die Struktur in Bezug auf ihre zuverlässigkeitstechnischen Zusammenhänge zu bewerten. Mit steigendem Detaillierungsgrad (j-fach-Fehler, Fehlerkombinationen) der Analyse wächst der Reifegrad der Zuverlässigkeitsstruktur. Die initiale Struktur wird durch die Systemmatrix beschrieben und durch das Zuverlässigkeitsnetz abgebildet. Für die empirische Ermittlung der Strukturfunktion werden kombinatorisch alle Komponenten zum Ausfall gebracht und die Systemauswirkung beobachtet. Führt eine Ausfallkombination zu einem Systemausfall, ist dies in der Strukturfunktion zu berücksichtigen. Werden alle Kombination von Ausfällen des Systems simuliert, wird die Strukturfunktion vollständig abgebildet. Sind, wie in der Automobilindustrie üblich, nur Doppelfehler zu betrachten, vereinfacht sich die Analyse, da nur Zweierkombinationen (zwei-fach-Fehler) von Ausfällen zu berücksichtigen sind.

Die Ausfallkombinationen werden durch

$$X_n^j := \left\{ (x_1, x_2, x_3, \ldots, x_n) | x_i \in \{0,1\} \, ; x_i \notin \{m, f, o\} \, ; \sum_{i=1}^{n} x_i = j \right\} \tag{4.33}$$

beschrieben. Alle Ausfallkombinationen der n Komponenten (exklusive der Module m, Funktionen f und Ausgänge o) werden für j-fach Fehler durch X_n^j berücksichtigt. Wie beschrieben, wird das Ausfallverhalten mit den Fehlerkombinationen simuliert. Für den Spezialfall, dass diese zu einem Systemausfall ($\phi(\underline{x}) = 1$) führen, werden diese Ausfallkombinationen in C_{Sch} abgelegt. Unter Verwendung der Schnitte C_{Sch} lässt sich die reduzierte Strukturfunktion, die reduzierte Systemmatrix sowie das vereinfachte RBD für den Optimierungsprozess herleiten. In diesen Darstellungen werden nur die notwendigen Komponenten zur Ausfallbeschreibung des Systems dargestellt. Somit ist zu beachten, dass diese Darstellungsform nicht das System in seinem vollen Funktionsumfang darstellt.

Das Ergebnis dieser Methode ist vergleichbar mit der Pfad- oder Schnittanalyse. Der Vorteil besteht hier in der Möglichkeit, das System aus Teilsystemen zusammenzusetzen und nur die Ausfallfortpflanzung der Teilsysteme zu beschreiben. Die Überführung in die Strukturfunktion erfolgt erst am Ende der Analyse. Die Komplexität der Analyse des Systems wird in diesem Fall enorm reduziert, da nur noch die Ausfallkombinationen der Teilsysteme betrachtet werden und das Resultat für Gesamtsystem aus denen der Teilsystemene zusammengesetzt wird.

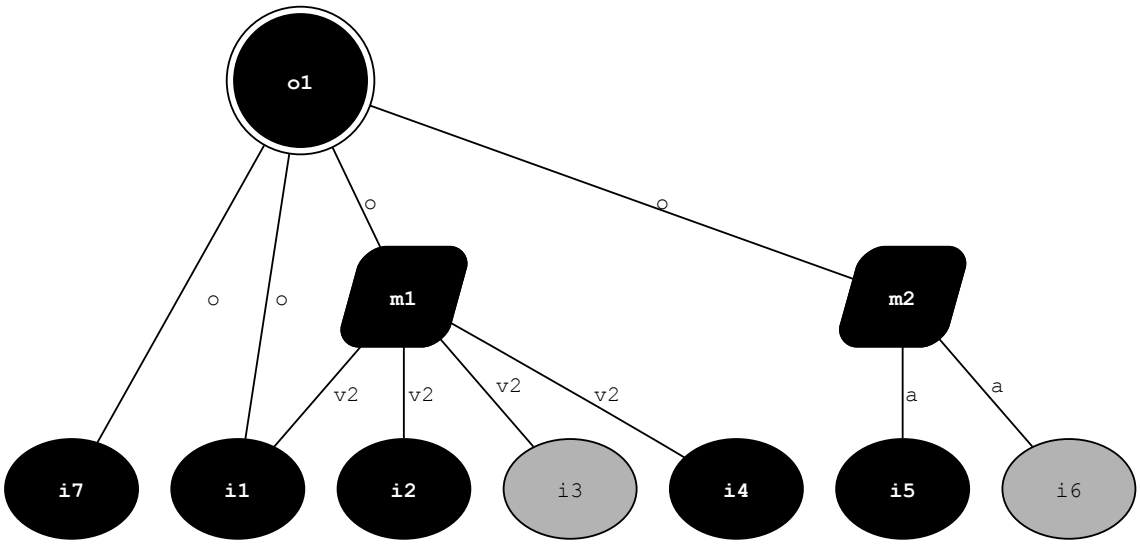

Abbildung 4.22: Exemplarische Darstellung einer Fehlerkombinationseinprägung zur empirischen Strukturermittlung

Die empirische Ermittlung der Pfade ist aufgrund der vereinfachten optimierten Bestimmung des Systemzustandes sehr effizient. Ein Beispiel zur empirischen Strukturermittlung aus Zuverlässigkeitsnetzen ist in Abbildung 4.22 dargestellt. Die schwarz hinterlegten Knoten sind in diesem Fall ausgefallen. Die Simulation wurde hierbei mit der Fehlerkombination $e = \{i1, i2, i4, i5, i7\}$ durchgeführt. Alle Fehlerkombinationen, die zu einem Systemausfall führen werden in der Menge der minimalen Schnitte C_{Sch} festgehalten und für eine statistische Auswertung (Histogramm) zur Verfügung gestellt. Die Häufigkeit, mit der eine Komponente/Ausfallkombination für einen Systemausfall verantwortlich ist, wird für die Importanzermittlung der Komponente/Ausfallkombination genutzt. Hierdurch wird es möglich, kritische Pfade und Komponenten oder kritische Teilsysteme des Systems zu identifizieren. Zusätzliche Informationen über die Importanz von Komponenten/Pfade liefert die Pfadimportanzanalyse.

4.6 Analyse und Bewertung der Pfade - Pfadimportanz

Weitere Quantifizierungsmerkmale können für den Entscheidungsprozess des Systems von Bedeutung sein. Importanzanalysen bieten die Möglichkeit, den Einfluss von Komponenten auf das System zu bewerten. Die Systemmatrix unterstützt die Analyse der Importanz durch die Beschreibung der Systemstruktur aus zuverlässigkeitstechnischer Sicht. Die notwendigen Informationen für die Ermittlung der Importanz können aus dem Zuverlässigkeitsnetz gewonnen werden. Die in der Literatur beschriebenen Importanzkenngrößen dienen ausschließlich der Bewertung von zuverlässigkeitstechnischen Aspekten. Für die Quantifizierung der Struktur wird in diesem Zusammenhang die Importanzanalyse um die Pfadimportanz erweitert. Hierzu werden die Pfade von der Wurzel[4] des Zuverlässigkeitsnetzes bis zum Basisereignis[5] gewichtet. Die unterschiedlichen Gewichtungen sind von der Bedeutung der Verknüpfung aus struktureller Sicht abhängig.

Tabelle 4.5: Gewichtung der Verknüpfungen

Verknüpfung	Gewichtung	
	Variante 1	Variante 2
UND	$\frac{n}{(n-1)\cdot d}$	$w_{basis} + (n-1) \cdot w_{and}$
ODER	$\frac{n-1}{n)\cdot d}$	$w_{basis} - (n-1) \cdot w_{or}$
VOTER	$\frac{n\cdot(n-1)}{n\cdot d}$	$w_{basis} + \left(\binom{n}{k} - 1\right) \cdot w_{and} - \left(\binom{n}{k} - 1\right) \cdot w_{or}$

mit

$$
\begin{aligned}
n &:= \sum \text{Kanten} \\
d &:= \text{Distanz von der Wurzelebene} \\
k &:= \text{Votergrad} \\
w_{basis} = 0,8 &:= \text{Basisgewichtung;} \\
w_{and} = 0,02 &:= \text{Gewichtung UND-Struktur} \\
w_{or} = 0,01 &:= \text{Gewichtung ODER-Struktur}
\end{aligned}
$$

Besteht in diesem Zusammenhang nicht die Möglichkeit, die Gewichtung bezüglich der Grundstruktur vorzunehmen, ist eine Überführung der Strukturfunktion für das Teilsystem sinnvoll.

Die Auswertung erfolgt innerhalb des Zuverlässigkeitsnetzes und kann wie in Abbildung 4.23 gezeigt dargestellt werden. Für die Gewichtung nach Tabelle 4.5 ist die Anzahl der Kanten, die von dem Knoten in die nächste Ebene gehen, entscheidend. Für die Gewichtung wird zusätzlich noch die Ebene (Distanz zur Wurzel) als Gewichtungskriterium hinzugefügt. Die Gewichtung ist so gewählt, dass eine a-Kante

[4]Wurzel: Von der Wurzel sind alle Knoten des Graphen erreichbar und sie besitzt keinen Vorgänger.

[5]Basisereignis: Von der Wurzel ausgehend wird ein Graph mit allen Verzweigungen entwickelt. Ein Zweig endet in einem Basisereignis.

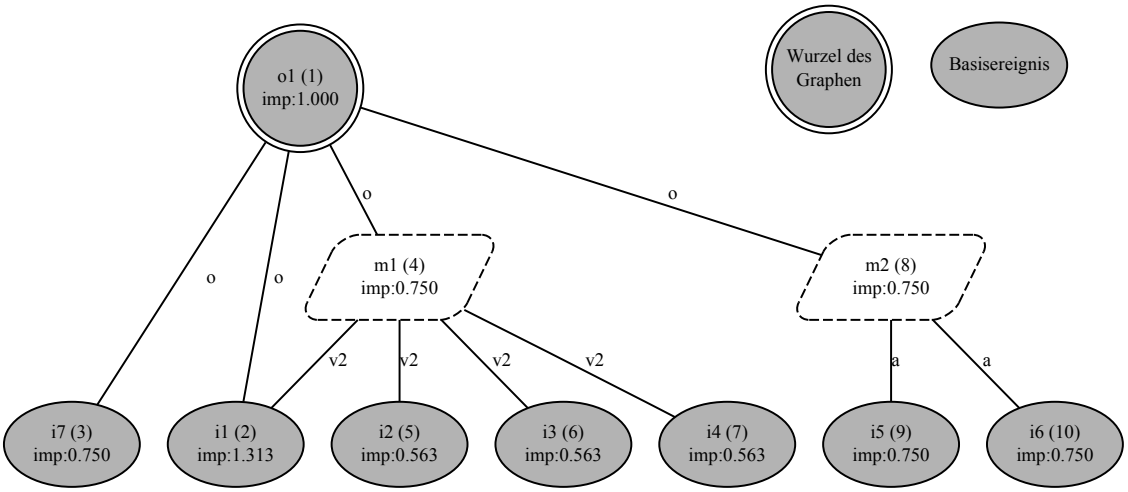

Abbildung 4.23: Exemplarische Darstellung der Gewichtung von Pfaden nach Tab. 4.5 (Variante 1)

höher gewichtet wird als z. B. eine *o*-Kante. Dieser Zusammenhang ist durch die Beschreibung der Zuverlässigkeitstechnik bedeutend. Eine *a*-Kante steht für eine Serienstruktur. Alle Komponenten der Struktur müssen funktionsfähig sein, damit die Systemfunktion erfüllt wird. Dieser Zusammenhang ist bei der *o*-Kante genau umgekehrt. Mindestens eine Komponente muss funktionieren, damit die Systemfunktion erfüllt wird.

Für die Analyse des Systems besteht des Weiteren die Möglichkeit, gezielt Ausfälle in das System zu injizieren. Es kann hiermit geprüft werden, ob das System sich gegenüber Ausfällen „tolerant“[6] oder „sicher“[7] verhält. Redundanzen und ihre Wirkmechanismen können ebenso überprüft werden.

4.7 Automatisierte Strukturanalyse aus Modellen

Systemstrukturen werden durch die Anforderungen definiert und in der Konzeptphase häufig durch Modelle nachgebildet oder simuliert. Die Bewertung von zuverlässigkeitstechnischen Zusammenhängen basiert auf der Strukturbeschreibung des Systems. Die Überführung der Anforderungen oder Modelle in eine Strukturbeschreibung ist sehr aufwändig und fehlerbehaftet. Eine Automatisierung der Strukturanalyse sowie eine formale Beschreibung der Systemzusammenhänge ermöglichen es somit, sowohl

[6] Besitzt ein System die Eigenschaft, trotz auftretender Fehler voll funktionsfähig zu sein, wird dieses System als „fehlertolerant“ bezeichnet.

[7] Als „ausfallsicher“ wird in diesem Zusammenhang die Eigenschaft betrachtet, dass das System im Fehlerfall in einen sicheren Zustand versetzt wird. Hierbei wird jedoch der volle Funktionsumfang eingeschränkt.

die Komplexität des Systems zu beherrschen als auch dabei Fehler zu vermeiden und den Aufwand zu reduzieren.

Im Systembewertungsprozess bildet die Analyse von Anforderungen und Spezifikationen den Ausgangspunkt für die Analyse. Anforderungen und Spezifikationen beschreiben die Systeme meist verbal und in nicht wiederverwendbarer Form. Durch die Erweiterung von Spezifikationen oder Anforderungen mit formalen Beschreibungen ist eine Automatisierung der Auswertung möglich. Ergänzende Zuverlässigkeitsinformationen und eine die Beschreibung der Wirkzusammenhänge in formaler Form ermöglichen die automatisierte Strukturanalyse des Systems. Hierzu ist jedoch eine geeignete Darstellungsform der Zusammenhänge notwendig. Einfache Beschreibungssprachen können dies unterstützen und eine einfache Interpretation des Systemverhaltens ermöglichen. Durch die Abbildung der Wirkzusammenhänge mittels Beschreibungssprachen ist es möglich, erste abstrakte Modelle zu generieren. Des Weiteren können aus den Ergänzungen die Testfälle abgeleitet werden.

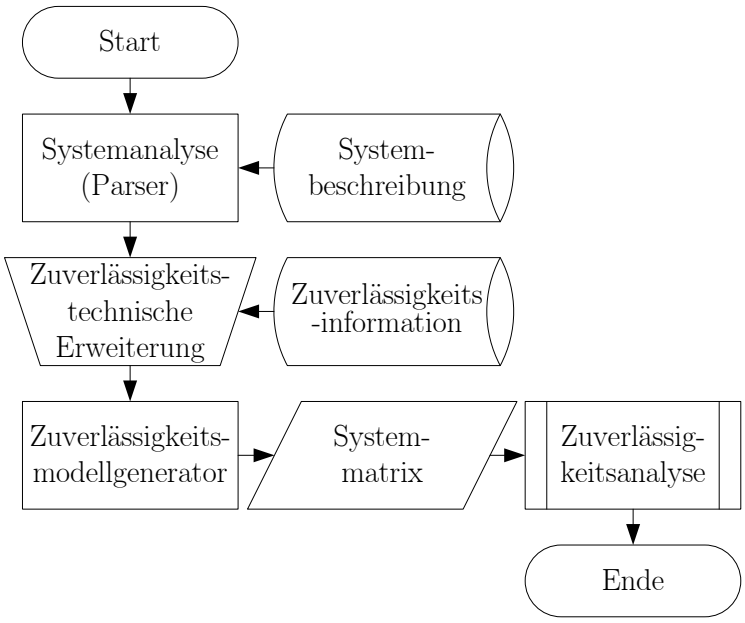

Abbildung 4.24: Allgemeine Vorgehensweise zur automatisierten Architekturanalyse

In Abbildung 4.24 sind die allgemeinen Zusammenhänge für eine automatisierte Analyse dargestellt. Die Systembeschreibung kann in Form von Spezifikationen oder Modellen vorliegen. Ist die Systembeschreibung in Form einer Spezifikation vorhanden, wird diese um die zuverlässigkeitstechnischen Informationen, insbesondere der Abhängigkeiten der Komponenten untereinander, erweitert. Durch diese Erweiterung ist es möglich, aus der Spezifikation ein erstes Zuverlässigkeitsmodell abzuleiten. Dieses Modell kann in die Systemmatrix überführt werden und steht dann zur weiteren Analyse des Systems zur Verfügung.

Wie im einführenden Abschnitt erwähnt, werden in der Produktentwicklung zur Beschreibung von Architekturen Beschreibungssprachen (ADL[8]) verwendet. Die bekanntesten Architekturbeschreibungssprachen sind SysML, UML und AADL. Obwohl diese sehr häufig für die Systembeschreibung eingesetzt werden, sind für die Zuverlässigkeitsbewertung Erweiterungen notwendig. Hierzu werden durch [GH08] einige Methoden beschrieben. Die Problematik besteht jedoch in der Durchgängigkeit der Beschreibungsformen - diese sind nicht für alle Domänen anwendbar. Die Modellierungssprachen sind hauptsächlich auf die programmierbaren Komponenten ausgelegt und Hardwarearchitekturen werden nicht global betrachtet. Für die Zuverlässigkeitsanalyse oder die Generierung der Systemmatrix sind die Modelle geeignet zu erweitern.

Eine automatisierte Analyse der Systemstruktur aus Modellen ist grundsätzlich möglich. Hierzu beschreibt Papadopoulos in [PM01] einen Ansatz zur automatisierten Strukturanalyse. Ein ähnlicher Ansatz wird durch Tajarrod in [TLS08] verfolgt. In beiden Ansätzen wird gezeigt, dass es grundsätzlich möglich ist, die Struktur zur zuverlässigkeitstechnischen Bewertung automatisiert zu analysieren und in einen Fehlerbaum zu überführen. Jedoch basieren beide Ansätze auf einer detaillierten Strukturbeschreibung in Form von Modellen. Diese Modelle benötigen ergänzende Informationen, die die zuverlässigkeitstechnischen Zusammenhänge der einzelnen Komponenten beschreiben.

Die funktionale Struktur kann aus Modellen direkt abgeleitet werden. Da sich das funktionale Systemverhalten und das zuverlässigkeitstechnische Systemverhalten grundsätzlich unterscheiden, müssen der funktionalen Systembeschreibung noch Informationen, die das zuverlässsigkeitstechnische Verhalten beschreiben, hinzugefügt werden. Das funktionale Systemverhalten beschreibt die funktionalen Abhängigkeiten im System, hier liefert das Modell die notwendigen Informationen. Bei der zuverlässigkeitstechnischen Betrachtung von Systemen wird die Systemverfügbarkeit als Ausgangspunkt der Betrachtung festgelegt. Nach [PM01, PPG04] lässt sich durch die Erweiterung des Modells direkt ein Fehlerbaum ableiten. Dieser Fehlerbaum wird für die weiteren Analysen verwendet und kann als Informationsquelle zur Beschreibung der zuverlässigkeitstechnischen Zusammenhänge dienen. Eine Methode zur formalen Risikoanalyse und automatisierter Generierung von Fehlerbäumen wird von Liggesmeyer in [LR98] beschrieben. Die Darstellung des Systems durch Zustandsautomaten wird vorgeschlagen und hierauf aufbauend der Fehlerbaum entwickelt. Diese Methode hat den Nachteil, von Zustandsautomaten abgeleitet zu sein, die jedoch zunächst entworfen werden müssen. Für die Systementwicklung ist dies sehr zeitintensiv und nicht immer möglich.

Methoden, die die Generierung von Fehlerbäumen unterstützen, liefern einen wertvollen Beitrag zu der Beschreibung von Systemen durch die Systemmatrix. Fehlerbäume lassen sich aufgrund ihrer formalen Beschreibung einfach in die Systemma-

[8] Architecture Description Language (ADL)

trix überführen und können somit einen direkten Beitrag zur Bewertung des Systems liefern. Das Ziel, einen Fehlerbaum aus Systemmodellen zu generieren, wird von vielen Arbeiten verfolgt. Allerdings basieren die meisten auf einem Modell, das einen weiteren Entwicklungsschritt voraussetzt und nur das funktionale Verhalten abbildet. Zuverlässigkeitstechnische Informationen sind in einem weiteren Schritt zu ergänzen.

Ein weitere Variante die Systemmatrix zu generieren, liefern die Zuverlässigkeitsmodelle. Aus diesen kann die Zuverlässigkeitsstruktur des Systems direkt abgeleitet und in die Systemmatrix überführt werden. Ansätze zur Beschreibung der zuverlässigkeitstechnischen Systemzusammenhänge verfolgen die Beschreibung von Systemkomponenten durch FTPN, SEFT oder CFT ([Gru07, Gru03]). Diese berücksichtigen nur teilweise die funktionalen Zusammenhänge des Systems, meist auch nur aus Sicht der Software. Die Ansätze liefern Methoden zur Darstellung und Auswertung der Systemzuverlässigkeit, jedoch sind die Beschreibungsformen sehr komplex und werden nicht domänenübergreifend angewendet. Eine, aus mechatronischer Sicht, domänenübergreifende Systemstrukturbeschreibung, unter Berücksichtigung der funktionalen und zuverlässigkeitstechnischen Zusammenhänge, ist notwendig und für die Bewertung des Gesamtsystems von Vorteil. Dieses Ziel wird durch die Verwendung der Systemmatrix zur Abbildung der Struktur erreicht.

4.8 Analyse und Quantifizierung mittels der Systemmatrix - Überblick

Die Strukturanalyse und Datenauswertung unter Verwendung der Systemmatrix ist eine systematische Methode mit strikter Vorgehensweise. Diese umfasst die folgenden Arbeitsschritte:

1. Komponentenidentifikation und Aufteilung in Teilsysteme (Modularisierung),
2. Erzeugung einer initialen Struktur und Beschreibung unter Verwendung der Systemmatrix,
3. Analyse der Komponentenzusammenhänge (Funktional, Zuverlässigkeitstechnisch),
4. Pfadanalyse (Strukturanalyse: Kommunikationspfade etc.),
5. Modellbildung - Überführung der Systemmatrix in Zuverlässigkeitsmodelle (Analyse und Quantifizierung mit standardisierten Methoden),
6. Auswertung der Qualitätsmerkmale,
7. Simulation von Fehlzuständen,
8. Datenablage in Zuverlässigkeitsnetzen,
9. Reduktion der Strukturfunktion durch empirische Ausfallanalyse und
10. Bereitstellung einer Optimierungsstruktur für den Optimierungsprozess

Die Strukturanalyse beginnt mit der Identifikation der Komponenten des Systems. Für jede Systemkomponente wird ein Element in der Systemmatrix angelegt. Infor-

mationen über die Komponenten sind in der Notation der Systemmatrix festgelegt. Zur übersichtlicheren Darstellung der Struktur kann es sinnvoll sein, das System zu modularisieren. Hierzu werden die Komponenten zu Funktionsgruppen zusammengefasst und in Teilsysteme gruppiert. Die entstandene Matrix bildet die Grundlage für die Auswertung der Quantifizierungsmerkmale. Im nächsten Schritt werden die funktionalen und zuverlässigkeitstechnischen Zusammenhänge innerhalb der Struktur analysiert. Jede Komponente, die die Systemfunktion beeinflusst, steht in einem zuverlässigkeitstechnischen Zusammenhang zu anderen Systemkomponenten. In diesem Schritt entsteht, ausgehend von der Systemmatrix, eine hierarchische Struktur des Systems. Bei verteilten Systemen ist die Pfadanalyse von entscheidender Bedeutung. Hier werden alle Informationswege von Daten betrachtet. Es werden alle Pfadvarianten, die das Signal von der Quelle bis zur Senke zurücklegt, analysiert. Die qualitativen Zusammenhänge des Systems sind nach diesem Arbeitsschritt bekannt und die Modellbildung zur quantitativen Bestimmung der Zuverlässigkeit kann erfolgen. Die Systemstruktur wird in eine Boolesche Funktion überführt und die Zuverlässigkeit bestimmt. Mit der Systemmatrix und der erweiterten Strukturdarstellung kann das Ausfallverhalten der Komponenten und die Auswirkung auf das System simuliert werden. Die Informationen des quantifizierten Systems werden in einem Zuverlässigkeitsnetz abgelegt. Des Weiteren ist es möglich, eine Systembeschreibung zu generieren, die für das Optimierungsverfahren nach [KJS09] angepasst ist. Eine Modularisierung des Systems mit anschließender Klassifizierung der Systemkomponenten ermöglicht die Abbildung durch einen Systembaukasten.

5 Konzeptentscheidung unter Anwendung eines Systembaukastens

Zu Beginn der Produktentwicklung ist es wichtig, Konzepte miteinander zu vergleichen und wissensbasierte Prognosen für Systeme durchzuführen. Informationen hierzu stehen in Form von Zuverlässigkeitsnetzen zur Verfügung und werden in einem folgend beschriebenen Baukasten[1] abgelegt. Der Optimierungsprozess dient der kontinuierlichen Verbesserung im Bezug auf die Anforderungsumsetzung durch das Konzept und wird durch die vorgestellte Methode unterstützt.

5.1 Systemkomposition unter Verwendung eines Systembaukastens

Innerhalb der Konzeptphase des Produktlebenszyklus wird die domänenspezifische Entwicklung durchgeführt. In dieser Phase werden Konzepte gegenübergestellt und ein favorisiertes Konzept für die Entwurfsphase freigegeben. Da es sich um domänenspezifische Konzeptentwürfe handelt, ist eine Bewertung des Gesamtsystems schwierig.

Zur Beherrschung der Komplexität wurden Methoden vorgestellt, die den Betrachtungsumfang reduzieren oder das System in Teilsysteme unterteilen. Die Aufteilung des Systems wurde nach räumlichen oder funktionalen Aspekten durchgeführt. Somit können die Teilsysteme aus reiner Hardware oder einer Kombination aus Soft- und Hardware bestehen, die durch den Quantifizierungsprozess separat bewertet werden. Die hierdurch gewonnenen Informationen werden in einem Zuverlässigkeitsnetz abgelegt. Das im Konzept erarbeitete Gesamtsystem lässt sich durch die Komposition der Teilsysteme darstellen und bewerten. Teilsysteme, die bei der Konzepterstellung häufig wiederverwendet werden, müssen den Quantifizierungsprozess nicht erneut durchlaufen. Hier bietet sich für die Konzeptphase ein Systembaukasten an, der es ermöglicht, Systeme aus bereits bekannten und bewerteten Teilsystemen zusammenzusetzen. Ein Systembaukasten stellt somit eine Wissensdatenbank dar, der die Informationen aller bisher bewerteten Teilsysteme in Form von Zuverlässigkeitsnetzen zusammenfasst.

[1] Baukasten: Die Konstruktion von technischen Produkten unter Verwendung von Baukästen wird in [PBFG07, Kra00] näher erläutert. In diesem Zusammenhang findet eine Klassifizierung von Baugruppen und deren Schnittstellen statt. Diese werden durch Kombination und Variation zu einem Gesamtsystem zusammengefasst. In der hier vorliegenden Arbeit wird eine abstrakte Klassifizierung unter Berücksichtigung der strukturellen Eigenschaften durchgeführt, um die generelle Vorgehensweise näher darzustellen. Eine weitere Detaillierung der Baugruppen ist nicht ausgeschlossen.

Die Komponenten werden für die Erstellung eines Baukastens klassifiziert und folgenden Gruppen zugeordnet:

- Sensoren (Eingangssignale),
- Aktoren (Ausgangssignale),
- Hardware und
- Software (Systemfunktionen).

Die Hardware beinhaltet in diesem Zusammenhang häufig nur elektronische Hardware, da die mechanische Hardware meist innerhalb der Aktorik umgesetzt wird.

Die Systemmatrix erlaubt eine einfache Aufteilung in Teilsysteme und unterstützt die Bewertung der festgelegten Qualitätsmerkmale. Die Matrix berücksichtigt bereits die Einteilung der Komponenten in die unterschiedlichen Gruppen. Somit ist eine Aufteilung und Einordnung der Teilsysteme in den Baukasten möglich.

Für die Bewertung des Gesamtkonzeptes ist eine Komposition des Systems aus den einzelnen Teilsystemen durchzuführen. Wie in Abbildung 5.1 dargestellt, ist es möglich, die Gesamtarchitektur aus den Komponentengruppen des Baukastens zu erstellen. Für das Gesamtsystem wird aus jeder Gruppe (Sensorik, Aktorik, Hardware, Software) jeweils eine Variante ausgewählt und zu einer Gesamtarchitektur zusammengefügt. Diese Architektur wird dann durch die Anwendung des Quantifizierungsprozesses bewertet. Es wird vorausgesetzt, dass alle Varianten der Gruppen, die für die Auswahl zur Verfügung stehen, identische Ein- und Ausgangsmerkmale besitzen, sich jedoch in ihrer Umsetzung oder ihren Parametern unterscheiden (Unbestimmtheit und Mehrdeutigkeit der Entwicklung). Für die Bewertung der Teilsysteme und die Gegenüberstellung der Systemkonzepte können unterschiedliche produktspezifische Quantifizierungsmerkmale festgelegt werden. Die Auswahl dieser Merkmale ist jedoch vom Umfeld abhängig, in dem das System entwickelt wird. In Abbildung 5.2 werden beispielhaft Qualitätsmerkmale, die die Konzeptentscheidung in der Automobilindustrie unterstützen können, dargestellt.

Werden im Laufe der Entwicklung mehrere Systemkonzepte betrachtet, wird der Baukasten stetig erweitert. Mit der fortlaufenden Quantifizierung der Systeme wächst die Wissensdatenbank stetig. Der Baukasten ermöglicht durch die Variation der Komponenten und der zugrunde liegenden Wissensdatenbank eine Gegenüberstellung unterschiedlicher Systemarchitekturen. Hierbei kann das System bezüglich seiner

- Architektur (inkl. Redundanzstrategien),
  - Funktion (Software),
  - elektrischen/elektronischen Hardware und
  - mechanischen Hardware,
- Sensorik und
- Aktorik

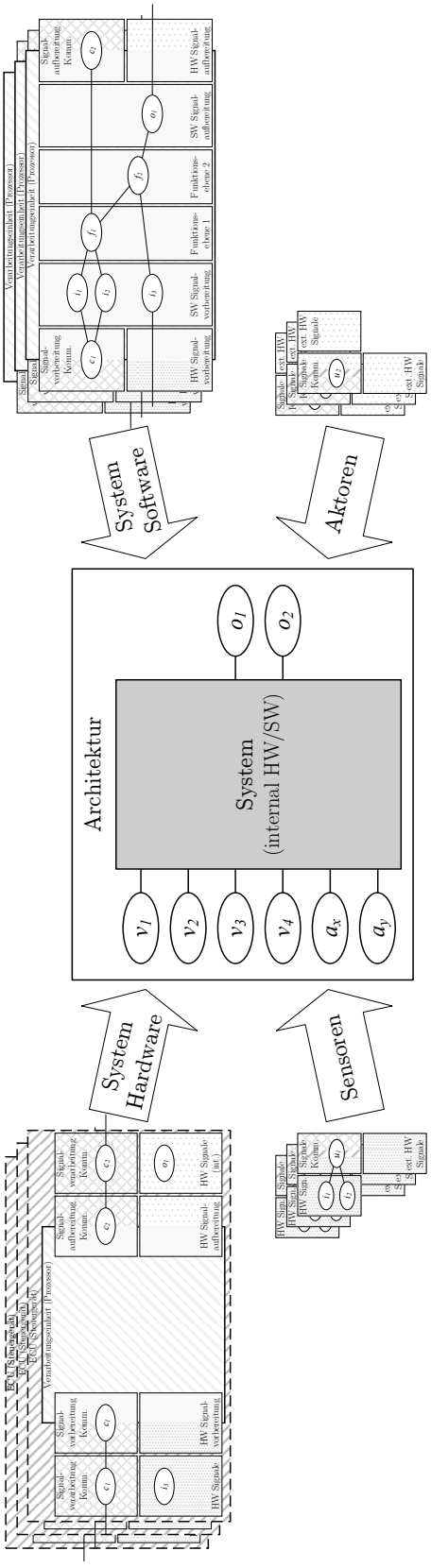

Abbildung 5.1: Übersicht zur modularen Komposition von Systemen

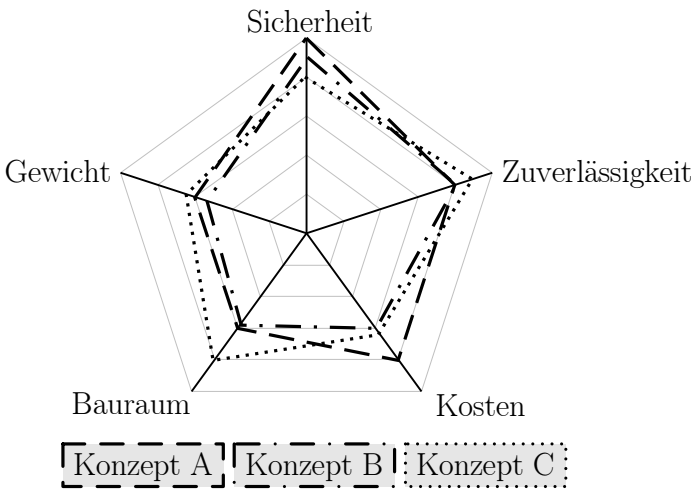

Abbildung 5.2: Beispiel für Quantifizierungsmerkmale zum Konzeptvergleich in der Automobilindustrie

variiert werden. Bei der Quantifizierung des Gesamtsystems kann auf Informationen des Baukastens zurückgegriffen werden. Ergibt sich eine neue Variante, wird diese quantifiziert und damit der Baukasten erweitert. Es ist zu erkennen, dass ein modularer Aufbau des Systems von Vorteil und Voraussetzung für eine Gruppierung in den Baukasten ist. Bei der Variation der Systemstruktur ist jede Variante durch den Quantifizierungsprozess zu bewerten. Werden die Quantifizierungsmerkmale unterschiedlicher Architekturen gegenübergestellt, ist ein direkter Vergleich der Systeme möglich. Sind beispielsweise für die Entwicklung unterschiedliche Architekturen konzipiert worden, können diese durch Auswertung und Gegenüberstellung der Merkmale miteinander verglichen werden. Abbildung 5.3 zeigt exemplarisch eine Gegenüberstellung unterschiedlicher Architekturkonzepte. In diesem Beispiel wurden für die unterschiedlichen Konzepte verschiedene Baugruppen variiert. Architektur A stellt eine Referenzarchitektur und das ursprüngliche Konzept dar. Ausgehend von dieser Architektur wurden für die Architektur B die Sensorik und für die Architektur C die System-Hardware variiert. Da hier auf bereits bewertete Komponentengruppen zurückgegriffen werden konnte, liefert der Baukasten alle notwendigen Informationen. Der Analyseaufwand wird hierdurch reduziert, da eine erneute Bewertung der Teilsysteme nicht erforderlich ist. Eine vereinfachte Systemgegenüberstellung wird zudem durch den Systembaukasten möglich. Ist in diesem Fall das Entscheidungskriterium für das Systemkonzept die Zuverlässigkeit, ist Variante C die beste Wahl. Sind mehrere Kriterien zu erfüllen, ist eine Gewichtung der Kriterien durchzuführen und das passende Konzept zu wählen.

Durch die Einführung eines Systembaukastens wird eine erwartungskonforme Umsetzung und Komposition von Systemen möglich. Die Bewertung der einzelnen Teilsysteme und die Ablage dieser Informationen in einem Baukasten tragen hierzu bei und unterstützten die Konzeptentwicklung, indem auf bewährte Konzepte

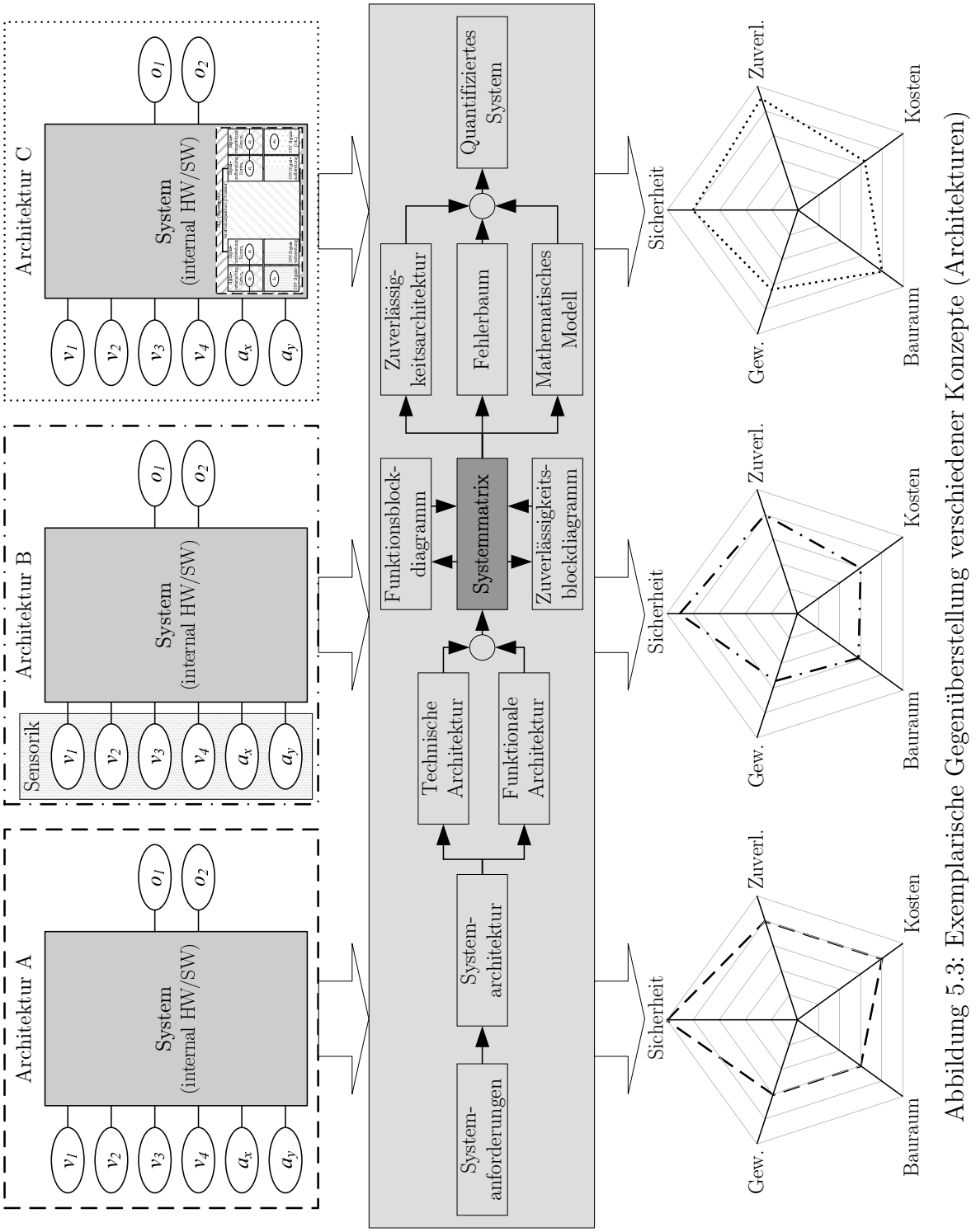

Abbildung 5.3: Exemplarische Gegenüberstellung verschiedener Konzepte (Architekturen)

zurückgegriffen werden kann. Während der Systementwicklung lassen sich Konzeptvergleiche einfach und schnell realisieren. Hierdurch wird eine Entscheidungsfindung in frühen Produktphasen ermöglicht. Der Systembaukasten ist nicht ausschließlich für die Konzeptphase bedeutend, denn dieser liefert auch Informationen für die Systementwicklung in darauf folgenden Produktphasen. Da in dem Baukasten alle Informationen über das analysierte System abgelegt sind, können Garantiefragen effizient und zeitnah bearbeitet werden. Können durch die Variation nicht alle geforderten Merkmale erfüllt werden, kann eine Optimierung der Komponenten oder der Systemstruktur den Konzeptentwurf unterstützen.

5.2 Systemquantifzierungs- und Optimierungsprozess

Wird die Konzeptphase durch einen umfangreichen Systembaukasten unterstützt, können Konzepte durch die Variation der Komponenten generiert werden. Sind nicht alle Anforderungen durch die generierte Architektur zu erfüllen, besteht die Notwendigkeit, eine Optimierung durchzuführen. Eine Vorgehensweise zur Optimierung von Systemstrukturen unter Berücksichtigung von Strukturparametern wird in [KJS09,KJS10] beschrieben. Hierzu ist ein Systemmodell auf Basis von Zuverlässigkeitsblockdiagrammen notwendig. Zuverlässigkeitsnetze können zur Strukturbeschreibung verwendet werden und bilden die Schnittstelle zwischen dem Quantifizierungs- und Optimierungsprozess. Die Optimierung erfolgt mittels bekannter Methoden und basiert auf der Zuverlässigkeitsstruktur des Systems.

Die Vorgehensweise der Zuverlässigkeitsmodellbildung (Abbildung 2.10) steht in engem Zusammenhang mit dem vorgestellten Quantifizierungs- und Optimierungsprozess. In der hier vorgestellten Methode bildet die Systemmatrix die Grundlage für die Zuverlässigkeitsmodelle. Der Systemquantifizierung- und Optimierungsprozess stellt eine Anpassung der Zuverlässigkeitsmodellbildung an den hier vorgestellten Ansatz dar.

Der Entscheidungsprozess ist in Abbildung 5.5 als zusammenhängender Prozess dargestellt. Er gliedert sich in die Unterprozesse zur Quantifizierung, Konzeptentscheidung und Optimierung des Systems. Der erste Schritt in diesem Prozess ist die Quantifizierung. Die ermittelten Ergebnisse bilden die Entscheidungsgrundlage für oder gegen das Systemkonzept. Ist dieses nicht zufriedenstellend, ist eine Optimierung des Systems notwendig. Eine Optimierung lässt sich nach dem in Abbildung 5.4 und in [KJS10] vorgestellten Ansatz durchführen. Der geforderten Optimierung werden durch den vorangestellten Quantifizierungsprozess wichtige Informationen bereitgestellt und ermöglichen so eine Effizienzeinschätzung für die Optimierung des Gesamtsystems. Strukturinformationen in Form von Zuverlässigkeitsnetzen bieten den Vorteil, dass die Komplexität der Struktur beherrscht werden kann und nur Teilsysteme betrachtet werden müssen. Der Quantifizierungsprozess liefert zudem wichtige Informationen über die Komponenten und Teilsysteme. Diese Informationen

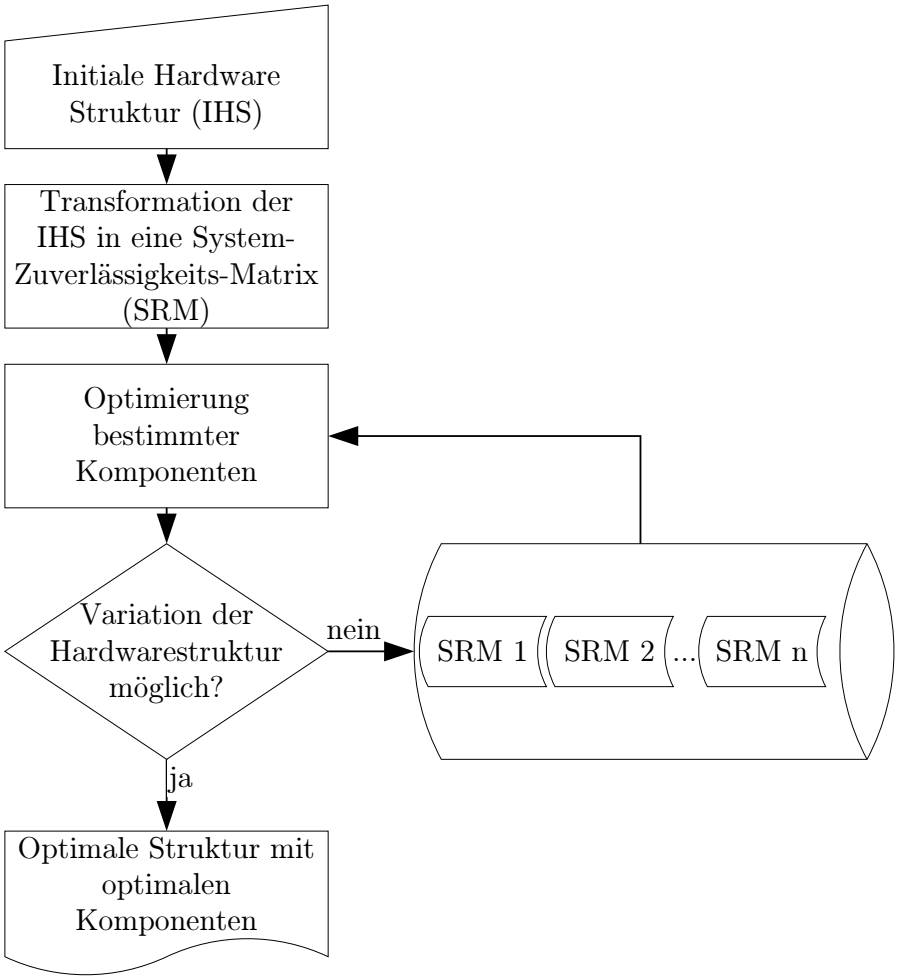

Abbildung 5.4: Optimierungsprozess nach [KJS10]

betreffen entweder die Bedeutung (Importanz) oder das Optimierungspotential der Komponenten. Die Optimierung kann auf der Strukturebene oder Komponentenebene durchgeführt werden, dient damit der Produktverbesserung oder wird zum Entwurf einer optimalen Architektur genutzt. Wurde innerhalb des Optimierungsprozesses eine Strukturvariation durchgeführt, ist diese erneut durch den Quantifizierungsprozess zu bewerten. Die resultierende Veränderung der Merkmale wird mit den ursprünglichen verglichen (Delta-Analyse) und dient als Entscheidungskriterium für weitere Optimierungsschritte.

5.2.1 Delta-Analyse

Eine Delta-Analyse beschreibt die systematische Vorgehensweise zur Untersuchung von Abweichungen und Fehlern. Da eine Überprüfung der Übereinstimmung zwischen

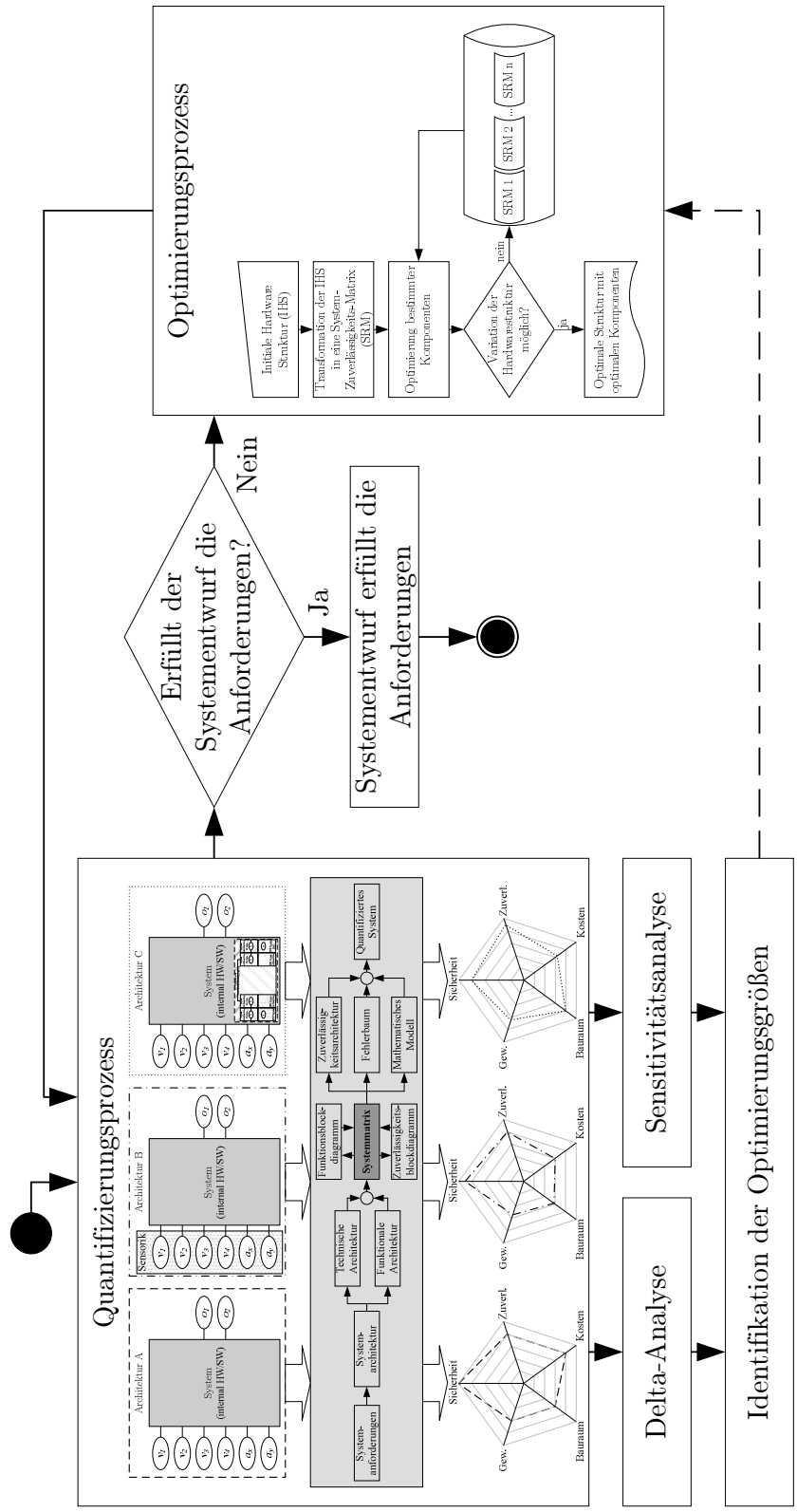

Abbildung 5.5: Systemquantifizierungs- und Optimierungsprozess als zusammenhängender Entscheidungsprozess

spezifizierten, idealen und realen System mit den Methoden des Qualitätsmanagements (FMEA, FTA, etc.) nicht möglich ist, werden die Unterschiede zwischen den Systemen mit Hilfe der Delta-Analyse ermittelt. Liegt ein nicht ideales System vor, werden dessen Abweichungen als Unterschiede zum idealen System festgehalten und versucht zu minimieren. In diesem Zusammenhang wird hier die Delta-Analyse nicht zur Identifizierung der Abweichung von einem idealen System verwendet, sondern zur Darstellung der Unterschiede zwischen den betrachteten Systemen. Stehen für das Systemkonzept mehrere Varianten zur Verfügung, ist ein Vergleich der Qualitätsmerkmale hilfreich. Wurde eine Systemoptimierung durch die Variation oder den Optimierungsprozess durchgeführt, liefert die Delta-Analyse wichtige Anhaltspunkte über die Effizienz (Verbesserung bezogen auf den Aufwand) der Optimierung.

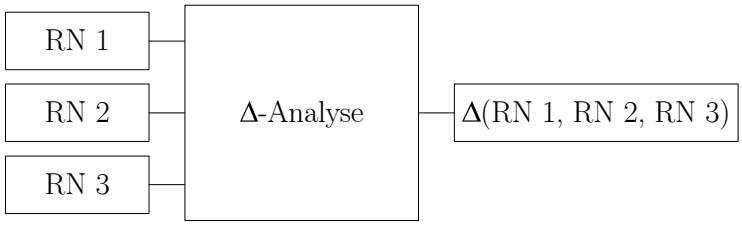

Abbildung 5.6: Systemvergleich durch Anwendung einer Delta-Analyse auf Zuverlässigkeitsnetze

Die Delta-Analyse lässt sich, wie in Abbildung 5.6 dargestellt, auf verschiedenen Ebenen der Systementwicklung anwenden und Veränderungen durch die Optimierung oder Variation lassen sich tendenziell erkennen. Stellen die Qualitätsmerkmale eine Entscheidungsgrundlage für das System dar, sind die durch die Optimierung generierten Verbesserungen hervorzuheben und gegenüberzustellen. Die Delta-Analyse ermöglicht den Vergleich der Systemvarianten und kann die relativen Änderungen gegenüber einem Referenzsystem verdeutlichen.

Durch die Variation der Systemstruktur entstehen verschiedene Varianten der Systemmatrix. Diese Unterschiede können durch die Anwendung einer Delta-Analyse aufgezeigt werden und liefern Hinweise auf Bereiche, in denen die Strukturoptimierung durchgeführt wurde. Dies kann bei automatisierten Strukturvariationen notwendig sein, um den Unterschied zwischen den Strukturen (Netzen) hervorzuheben und den Optimierungsbereich einzuschränken.

Die Systemstruktur wird durch die Systemmatrix vorgegeben. Wird eine Variation der Komponentenparameter durchgeführt ohne die Struktur zu verändern, lässt sich dies innerhalb des Zuverlässigkeitsnetzes realisieren. Hier lassen sich durch die Delta-Analyse die Komponentenmerkmale und die Auswirkung der Variation dieser näher analysieren. Die Zuverlässigkeitskenngrößen und Qualitätsmerkmale der Komponenten stehen hierbei im Vordergrund. Durch die Variation der Komponenten ist ein Vergleich der Systemvarianten möglich und kritische Komponenten können identifiziert werden.

Eine Delta-Analyse ermöglicht somit den Vergleich von Systemen auf verschiedenen Betrachtungsebenen. Es können Qualitätsmerkmale auf System- und Komponentenebene miteinander verglichen und Systemstrukturen näher analysiert werden. Der Erfolg oder Misserfolg einer Systemvariation kann hierdurch dargestellt und bewertet werden.

Können für die Optimierung keine eindeutigen kritischen Komponenten ermittelt werden, kann die Delta-Analyse zur Identifikation von Optimierungspotenzialen genutzt werden. Hierzu wird die Architektur empirisch variiert und die so entstehenden Varianten sind miteinander zu vergleichen. Das Ergebnis zeigt die Auswirkung der Strukturänderung bezüglich der Optimierungskriterien. Durch den Vergleich der Architekturvarianten kann auf den Einfluss der Variation geschlossen werden. Diese Vorgehensweise ermöglicht die Identifikation von kritischen Teilsystemen oder Komponenten.

5.2.2 Sensitivitätsanalyse

Komponenten oder Teilsysteme, die für eine Optimierung geeignet sind, können auch direkt aus der Struktur des Systems identifiziert werden. Hierzu wird eine Sensitivitätsanalyse durchgeführt. Die Sensitivitätsanalyse hat zum Ziel, die Identifikation von kritischen Komponenten oder die Ermittlung der Bedeutung einer Komponente innerhalb der Struktur. Hierzu werden die Informationen aus der

- Pfad- und Schnittanalyse,
- Pfadimportanzanalyse und
- empirischen Ausfallanalyse

verwendet.

Die Pfad- und Schnittanalyse aus der Booleschen Funktion ermöglicht die Identifikation von kritischen Pfaden innerhalb der Systemstruktur. In diesem Zusammenhang sind Redundanzen und das Ausfallverhalten des Systems zu erkennen.

Die Pfadimportanzanalyse bewertet die Bedeutung der Komponenten innerhalb der Struktur und liefert wichtige Informationen über den Einfluss der Komponenten auf das Systemverhalten. Es besteht hierdurch die Möglichkeit, das Potential einer Optimierung zu ermitteln.

Im Fall einer empirischen Ausfallanalyse kann die Komplexität der Strukturfunktion reduziert werden, da nur noch die notwendigen Schnitte (Ausfallkombinationen) betrachtet werden. Diese Anzahl der Schnitte ist abhängig von der Anzahl der Ausfallkombinationen. Diese ist wiederum abhängig von der erlaubten Anzahl gleichzeitig auftretender Ausfälle. Eine empirische Ausfallanalyse ist für die Ermittlung der Strukturfunktion sinnvoll, da häufig nur Zweifach- oder Dreifachfehler von Bedeutung

sind. Das hat eine direkte Auswirkung auf die Optimierung, denn eine vereinfachte Strukturfunktion reduziert die Komplexität bei der Optimierung.

Durch den Systemquantifizierungs- und Optimierungsprozess werden alle in dieser Arbeit vorgestellten Prozesse zusammengefasst. Der Optimierungsprozess wird als externer Prozess in den Entscheidungsprozess integriert. Die Systemmatrix, die Delta-Analyse und die Sensititvitätsanalyse liefern für den Optimierungsprozess alle notwendigen Informationen über das System. Durch die Identifikation von Optimierungspotenzialen wird der Optimierungsraum eingeschränkt und die Effizienz der Optimierung gesteigert. Durch die iterative Anwendung dieses Entscheidungsprozesses wird eine stetige Verbesserung des Systems erreicht, also die Abweichung zu den geforderten Zielen minimiert. Das Ergebnis ist ein anforderungsgerechtes System.

6 Beispiele mechatronischer Anwendungen

Nachdem in den vorherigen Abschnitten die Herleitung der Systemmatrix eingeführt wurde, wird folgend auf einige Anwendungsbeispiele eingegangen. Die Anwendung und Herleitung der Systemmatrix wird aufgezeigt und die Darstellung durch ein Zuverlässigkeitsnetz verdeutlicht. Die Auswertung der empirischen Ausfallanalyse wird unter Verwendung der Darstellung durch die Zuverlässigkeitsnetze detaillierter beschrieben. Die gezeigten Beispiele sind vom Abstraktionsgrad so gewählt, dass die universelle Einsetzbarkeit der in dieser Arbeit vorgestellten Vorgehensweise deutlich wird.

6.1 Beispiel 1: Physikalische und funktionale Darstellungen

In Abbildung 6.1 ist ein Beispielsystem aus der Automobilindustrie dargestellt. Dieses System erzeugt geschwindigkeitsabhängig eine Freigabe für andere, mit diesem verbundene, Systeme. Eine zusätzliche Warnmetrik, die ab bestimmten Beschleunigungswerten eine Warnmeldung abgibt, erweitert die Funktionalität. Das hier beschriebene System kann zur automatisierten geschwindigkeitsabhängigen Türverriegelung genutzt werden und mittels einer integrierten Sitzbelegungserkennung einen Anschnallhinweis generieren. Es ist an ein System aus der Fahrzeugtechnik angelehnt und hat keinen Anspruch auf Vollständigkeit. Die Vorgehensweise für die Systemanalyse wird folgend an diesem Beispiel verdeutlicht. In der ersten abstrakten

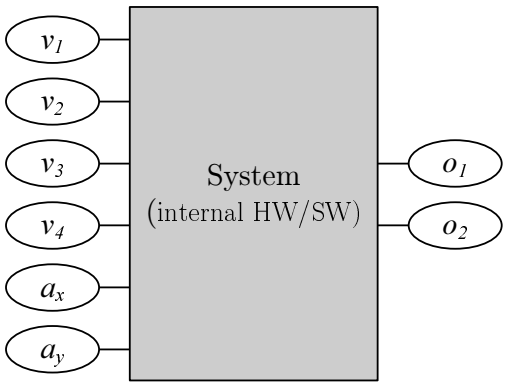

Abbildung 6.1: Black-Box-Darstellung eines geschwindigkeitsgesteuerten Freigabesystems (Beispiel 1)

Betrachtung des System wird das Ein-/Ausgangsverhalten analysiert. Aus der Systemspezifikation und der Anforderungsbeschreibung ist das Soll-Verhalten bekannt. Durch die Unterstützung der Experten kann das System aus zuverlässigkeitstechnischer Sicht genauer beschrieben werden. Hier sind die Abhängigkeiten zwischen den Ein- und Ausgängen im Detail zu betrachten. Das in Abbildung 6.1 dargestellte

System führt eine Geschwindigkeitsauswertung durch und bedient sich hierzu dreier unterschiedlicher Methoden. Die erste Methode ist über eine Mittelwertbildung der vier getrennt erfassten Geschwindigkeiten realisiert. Das zweite Verfahren führt eine, für die Geschwindigkeitsschwellenbetrachtung ausreichende, min/max-Betrachtung durch. Die letzte Variante für die Bestimmung der Geschwindigkeitsschwelle ist eine direkte Geschwindigkeitsermittlung aus den Beschleunigungen des Systems. Physikalisch betrachtet ergeben sich für die verschiedenen Methoden die folgenden Zusammenhänge: In der ersten Methode wird der Mittelwert der Geschwindigkeiten

$$v(v_1, v_2, v_3, v_4) = \frac{v_1 + v_2 + v_3 + v_4}{4} \tag{6.1}$$

ermittelt. Die zweite Methode wird durch eine Voter-Struktur $V2oo4$ (sprich „2 out of 4 Voter“) realisiert. Es handelt sich hier um einen min/max-Vergleich nach

$$\begin{aligned} v_{max_1}(v_1, v_2, v_3, v_4) &= max(max(v_1, v_2), max(v_3, v_4)) && (6.2)\\ v_{max_2}(v_1, v_2, v_3, v_4) &= min(max(v_1, v_2), max(v_3, v_4)) && (6.3)\\ v_{max_3}(v_1, v_2, v_3, v_4) &= max(min(v_1, v_2), min(v_3, v_4)) && (6.4)\\ v_{min}(v_1, v_2, v_3, v_4) &= min(min(v_1, v_2), min(v_3, v_4)). && (6.5) \end{aligned}$$

Für die Schwellenbestimmung werden zwei gültige Geschwindigkeiten vorausgesetzt. Die dritte Methode erfolgt nach der Geschwindigkeitsbestimmung über die Beschleunigungen

$$v = \int a \; dt. \tag{6.6}$$

Aus der Systembeschreibung ist zu entnehmen, dass die Informationen der Geschwindigkeitssensoren von einem externen System zur Verfügung gestellt werden. Aus diesem Grund ist für die Geschwindigkeitssensoren die Kommunikationsstrecke und das Ausfallverhalten des externen Systems zu berücksichtigen. Die Auswertung der Beschleunigungssensoren erfolgt innerhalb des betrachteten Systems.

Für die Systemstruktur ergibt sich das in Abbildung 6.2 dargestellte Funktionsblockdiagramm. Die Beschreibung durch das Funktionsblockdiagramm und der Geschwindigkeitsermittlung kann nicht direkt in die zuverlässigkeitstechnische Beschreibung überführt werden. Hierzu ist das Expertenwissen, mit dem es möglich ist die funktionale Struktur aus Abbildung 6.2 in eine Systemmatrix zu überführen, von entscheidender Bedeutung. Die in Tabelle 6.1 dargestellte Matrix ermöglicht die zuverlässigkeitstechnische Auswertung. Einige Informationen sind nicht direkt aus dem Funktionsblockdiagramm ersichtlich. Hierzu wird die Systemmatrix um Informationen zu bidirektional ausgelegten Kommunikationspfaden s, Kommunikationskomponenten c_i, Spannungsversorgungen p_i und Fehlerbetrachtungen e_i erweitert. Bei der Fehlerbetrachtung kann es sich beispielsweise um einen Steckerfehler des

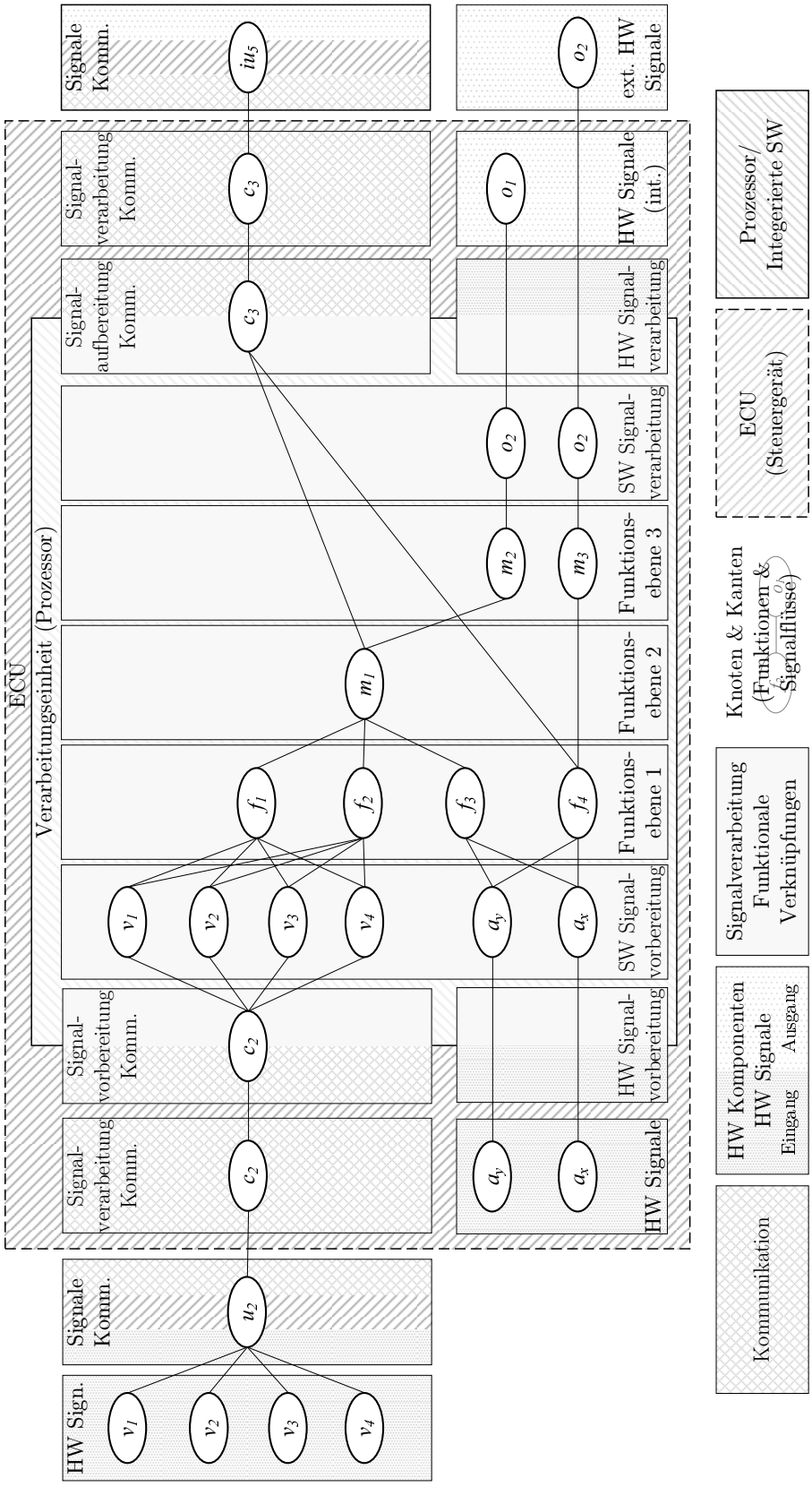

Abbildung 6.2: Funktionsblockdiagramm für Beispiel 1

Steuergerätes handeln. Im Fehlerfall sind die Spannungsversorgung, die Kommunikation und das Steuergerät selbst betroffen. Die funktionale Struktur des Systems wird durch die Systemmatrix abgebildet und die Funktionsverteilung innerhalb der Steuergeräte wird berücksichtigt. Funktionen, die in eine Komponente integriert werden, sind über die Kante i miteinander verbunden. Für die Analyse von Systemen sind die Signalpfade innerhalb der Struktur von Interesse. Hierzu werden die möglichen Signalpfade innerhalb des Systems durch eine mit s gekennzeichneten Kante dargestellt.

Tabelle 6.1: Systemmatrix von Beispiel 1

Eingang/Signal	Signal/Ausgang										
	Geschwindigkeit 1	Geschwindigkeit 2	Geschwindigkeit 3	v	a	Kommunikation	Einheit 1	Einheit 2	Systemfunktion	in	Zuverlässigkeit(p)
Spannungsversorgung							p			p_1	1
Spannungsversorgung								p		p_2	1
v_1	a	v2					s			i_1	0,8
v_2	a	v2					s			i_2	0,8
v_3	a	v2					s			i_3	0,8
v_4	a	v2					s			i_4	0,8
a_y			a		a		s			i_5	0,995
a_x			a		a		s			i_6	0,995
Kommunikation								s		c_1	1
Einheit 1	i	i	i	i		s				u_1	1
Einheit 2					i					u_2	0,99
Geschwindigkeit 1				v1						f_1	1
Geschwindigkeit 2				v1						f_2	1
Geschwindigkeit 3				v1						f_3	1
v									a	m_1	1
a										m_2	1
Fehlerinjektion					e	e	e	e		e1	1
out	f_1	f_2	f_3	m_1	m_2	c_1	u_1	u_2	o_1		

Das Zuverlässigkeitsnetz für Beispiel 1 ist in Abbildung 6.3 dargestellt. Die ermittelten Importanzkenngrößen wurden in diese Abbildung integriert. In diesem Beispiel ist deutlich zu erkennen, dass der injizierte Fehler e_1 eine hohe Importanz besitzt. Dies bedeutet, dass der Fehler die Systemfunktion stark beeinflusst und innerhalb der Systemstruktur genauer zu untersuchen ist. Die Komponenten u_1 und p_1 besitzen

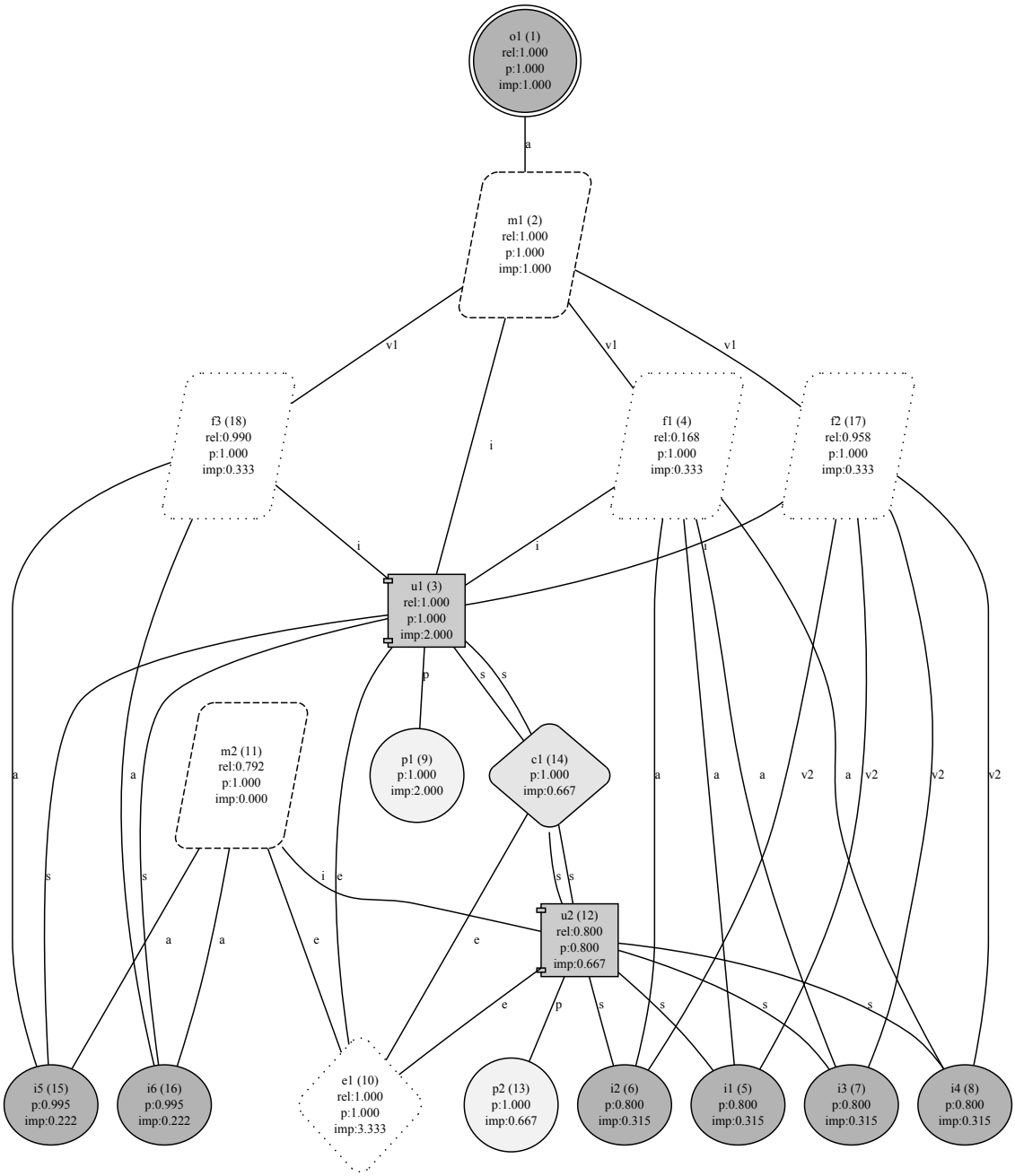

Abbildung 6.3: Zuverlässigkeitsnetz von Beispiel 1

ebenfalls eine hohe Importanz und bilden den Hauptbestandteil des Systems. Deren Ausfall bewirkt einen Ausfall des Gesamtsystems. Bei der empirischen Strukturermittlung werden diese Komponenten am häufigsten für den Ausfall des Systems verantwortlich sein. Die Kommunikationsstrecke für die Sensorsignale $i_1 \dots i_4$ sind für die Funktionalität der Funktionen f_1 und f_2 von großer Bedeutung. Aus der

Darstellung des Funktionsblockdiagrammes ist zu entnehmen, dass der Ausfall des externen Systems oder der Ausfall der Kommunikationsstrecke einen großen Einfluss auf das Systemausfallverhalten hat. Hier wird gezeigt, dass aber der Ausfall dieser Strecke keinen unmittelbaren Ausfall des Gesamtsystems bewirkt. Allerdings kann dies in Kombination mit einem weiteren Fehler zu einem Systemausfall führen. Informationen zu Fehlerkombinationen sind aus der Simulation und Einprägung von Fehlern (vergleiche Abschnitt 4.5) oder der Pfadanalyse ersichtlich.

6.2 Beispiel 2: Zerlegung von Systemen in Module

In einem weiteren Beispiel soll die Modularisierung der Systemmatrix verdeutlicht und die Überführung der Strukturbeschreibung in die Systemmatrix erläutert werden. Das

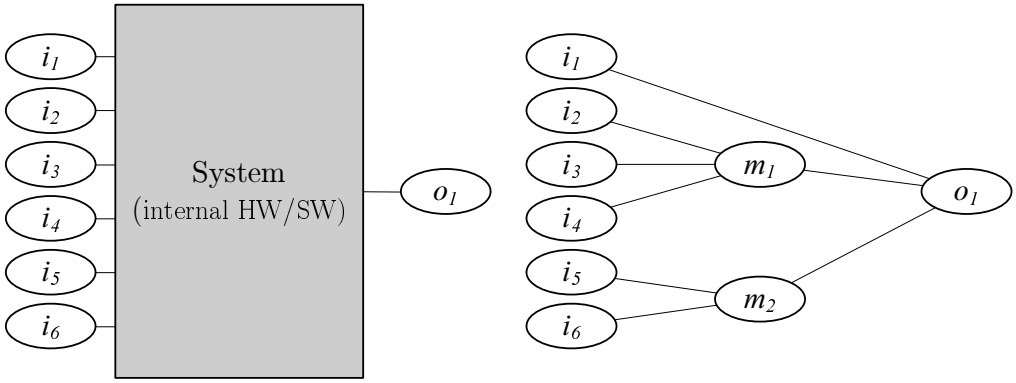

Abbildung 6.4: Funktionsblockdiagramm für Beispiel 2

hier betrachtete Funktionsblockdiagramm (Abbildung 6.4) ist abstrakt dargestellt und es findet keine Zuordnung der lokalen Anordnung der Komponenten statt. In diesem Zusammenhang sind nur die Module und die Abhängigkeiten von Bedeutung. Im linken Teil des Bildes ist das System als „Black-Box“ dargestellt. Der rechte Teil gibt einen detaillierteren Einblick in das System und stellt die funktionalen Abhängigkeiten der einzelnen Module dar.

In Abbildung 6.5 ist das System aus zuverlässigkeitstechnischer Sicht dargestellt. In dieser Darstellung sind der Zuverlässigkeitsgraph und der Fehlerbaum des Systems dem Zuverlässigkeitsblockdiagramm gegenübergestellt. Die Module des gezeigten Systems bestehen aus den beiden am häufigsten verwendeten Grundstrukturen (UND- und ODER-Verknüpfung). Für die Beschreibung durch einen Zuverlässigkeitsgraphen wird die Positiv-Logik verwendet. Der Fehlerbaum verwendet im Gegensatz hierzu eine Negativ-Logik. Diese entspricht der Beschreibung durch die Boolesche Modellbildung (vgl. Abschnitt 2.3.5). Die Beschreibung in der Positiv-Logik entspricht der Darstellungsform der Systemmatrix und kann daher direkt in die Systemmatrix überführt werden.

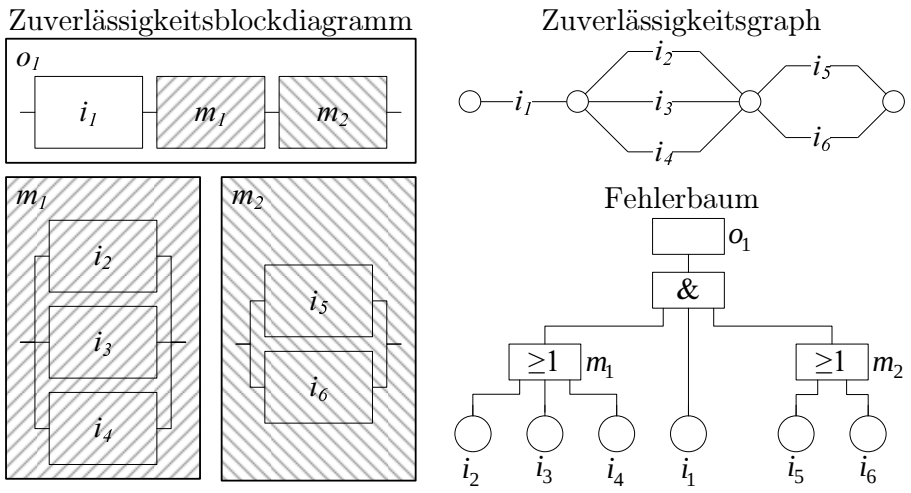

Abbildung 6.5: Zuverlässigkeitstechnische Darstellungen von Beispiel 2

Tabelle 6.2: Systemmatrix von Beispiel 2 (vgl. Abbildung 6.5)

Eingang/Signal	Signal/Ausgang			
	Modul 1	Modul 2	Ausgang 1	in
Eingang 1			a	i_1
Eingang 2	o			i_2
Eingang 3	o			i_3
Eingang 4	o			i_4
Eingang 5		o		i_5
Eingang 6		o		i_6
Modul 1			a	m_1
Modul 2			a	m_2
out	m_1	m_2	o_1	

In der Abbildung 6.5 ist die Analogie zwischen der Systemdarstellung als Funktionsblockdiagramm und der Systembeschreibung durch das Zuverlässigkeitsblockdiagramm deutlich zu erkennen. Die zu dem System gehörige Systemmatrix ist in Tabelle 6.2 dargestellt. Die Modularisierung ist in dieser Matrix durch die Module eins und zwei angedeutet. Diese Darstellungsform setzt sich in der Analyse und den Zuverlässigkeitsdarstellungen fort. Alle notwendigen Informationen für die Dekomposition können aus der Tabelle 6.3 entnommen werden.

Tabelle 6.3: RN-Dekomposition von Beispiel 2

	RN 1	RN 2	RN 3	Ausgang 1	in
$in1$			f		i_1
$in2$	f				i_2
$in3$	f				i_3
$in4$	f				i_4
$in5$		f			i_5
$in5$		f			i_6
$RN1$			f_{m_1}		rn_1
$RN2$			f_{m_2}		rn_2
$RN3$				f_{o_1}	rn_3
out	rn_1	rn_2	rn_3	o_1	

Durch die in dieser Tabelle dargestellten Zusammenhänge der Modularisierung ergeben sich für

$$RN1 = f(i_2, i_3, i_4) \quad \rightarrow(m_1) \tag{6.7}$$
$$RN2 = f(i_5, i_6) \quad \rightarrow(m_2) \tag{6.8}$$
$$RN3 = f(i_1, m_1, m_2) \quad \rightarrow(o_1) \tag{6.9}$$

und mit

$$m_1 = f(i_2, i_3, i_4) \quad \rightarrow\text{RN 1} \tag{6.10}$$
$$m_2 = f(i_5, i_6) \quad \rightarrow\text{RN 2} \tag{6.11}$$
$$o_1 = f(i_1, i_2, i_3, i_4, i_5, i_6) \quad \rightarrow\text{RN 3} \tag{6.12}$$

die Systemabhängigkeiten für die Dekomposition.

Für diese Darstellung des Systems ist eine Aufteilung in Teilsysteme durchgeführt worden. Die Aufteilung erfolgt mit dem Ziel, atomare Einheiten zu erhalten, die nur durch eine Verknüpfungsart gekennzeichnet sind. Eine Analogie zwischen der Darstellung eines Funktionsblockdiagramms und eines Zuverlässigkeitsblockdiagramms wird deutlich erkennbar.

6.3 Beispiel 3: Ausfall, Pfad- und Schnittanalyse

Wie bereits erwähnt, ist ein Zuverlässigkeitsblockdiagramm direkt in eine Systemmatrix übertragbar. Für das in Abbildung 6.7 gezeigte RBD ergibt sich die in Tabelle 6.4 dargestellte Systemmatrix. Das hier dargestellte System enthält nur die Grundverknüpfungsarten, besteht aus 11 Eingangskomponenten $a \ldots k$ $(i_1 \ldots i_{11})$ und ist

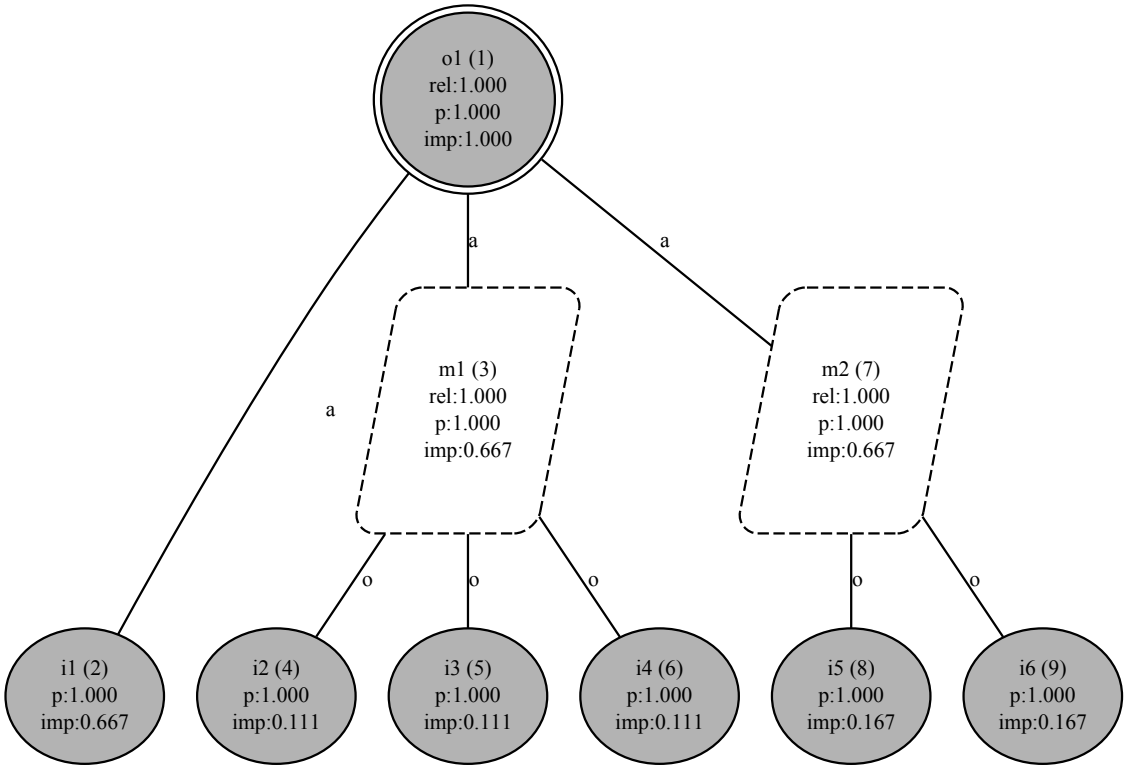

Abbildung 6.6: Zuverlässigkeitsnetz von Beispiel 2

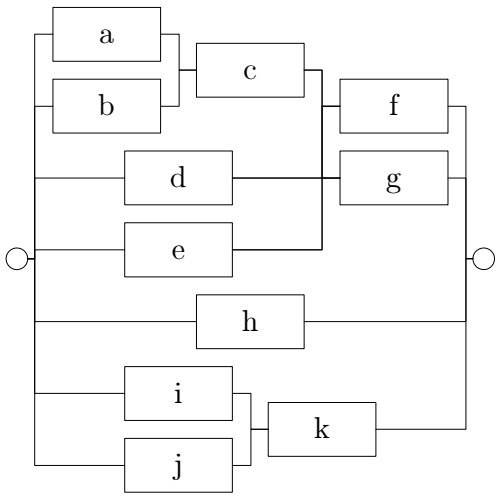

Abbildung 6.7: Zuverlässigkeitsblockdiagramm für Beispiel 3

in sieben Module $m_1 \ldots m_7$ aufgeteilt. Anhand dieses Systems wird die Analyse des empirischen Ausfallverhaltens näher erläutert.

Das Ergebnis der Auswertung ist in Abbildung 6.8 in Form eines Zuverlässigkeitsnetzes visualisiert.

Tabelle 6.4: Systemmatrix von Beispiel 3

Eingang/Signal	Signal/Ausang									
	mod1	mod2	mod3	mod4	mod5	mod6	mod7	out1	in	Zuverlässigkeit (p)
a	o								i_1	0,8
b	o								i_2	0,8
c		a							i_3	0,8
d			o						i_4	0,8
e			o						i_5	0,8
f				o					i_6	0,8
g				o					i_7	0,8
h								o	i_8	0,8
i						o			i_9	0,8
j						o			i_{10}	0,8
k							a		i_{11}	0,8
mod1		a							m_1	
mod2			o						m_2	
mod3					a				m_3	
mod4					a				m_4	
mod5								o	m_5	
mod6							a		m_6	
mod7								o	m_7	
out	m_1	m_2	m_3	m_4	m_5	m_6	m_7	o_1		

Durch die Simulation von Ausfällen ist es möglich, das Systemausfallverhalten näher zu beschreiben und die Systemfunktion für diesen speziellen Fall herzuleiten. Die Häufigkeit der Systemausfälle für o_1 ist in Abbildung 6.9 dargestellt. Die Analyse erfolgte unter der Berücksichtigung von 5-fach-Fehlern. Diese Betrachtung ist für den automobilen Bereich eher unüblich, aber für das System dennoch notwendig, da es bei dem gegebenen System erst ab 4-fach-Fehler zu Systemfunktionsausfällen kommt. Aus diesen Darstellungen sind die Schnitte des Systems direkt ablesbar und die Strukturfunktion für das System kann gebildet werden. Es ist jedoch zu

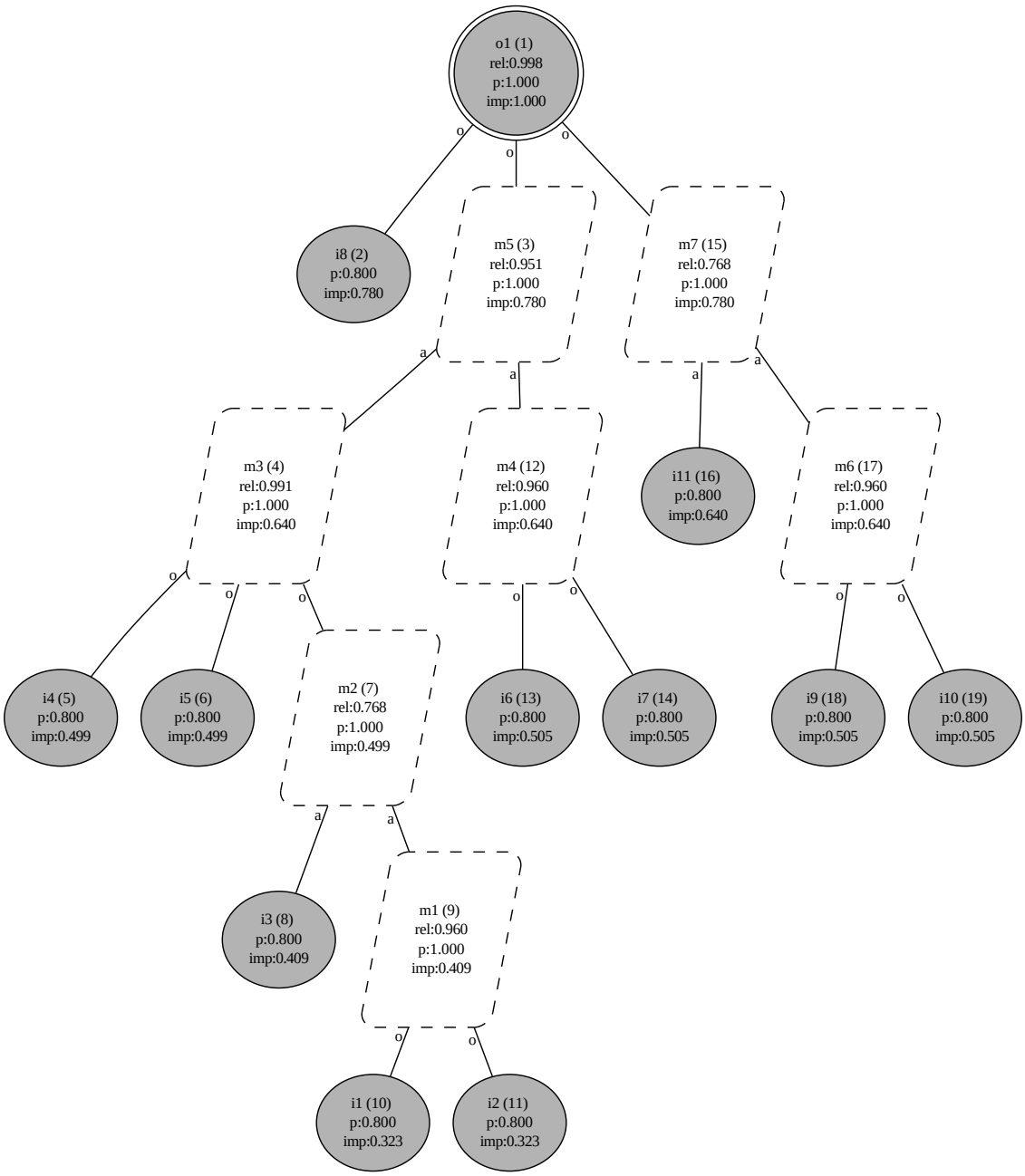

Abbildung 6.8: Zuverlässigkeitsnetz von Beispiel 3

beachten, dass die Simulation zunächst nur 5-fach-Fehler berücksichtigt und somit die Strukturfunktion noch eine gewisse Ungenauigkeit hat. Mit den Gleichungen

$$\begin{aligned} C_{Sch_1} &= \{2, 13, 14, 16\}, \\ C_{Sch_2} &= \{2, 5, 6, 8, 16\} \text{ und} \\ C_{Sch_3} &= \{2, 13, 14, 18, 19\} \end{aligned} \tag{6.13}$$

gilt für die Strukturfunktion

$$\phi(\underline{x}) = 1-(1-x_2 \cdot x_{13} \cdot x_{14} \cdot x_{16})(1-x_2 \cdot x_{13} \cdot x_{14} \cdot x_{18} \cdot x_{19})(1-x_2 \cdot x_5 \cdot x_6 \cdot x_8 \cdot x_{16}). \quad (6.14)$$

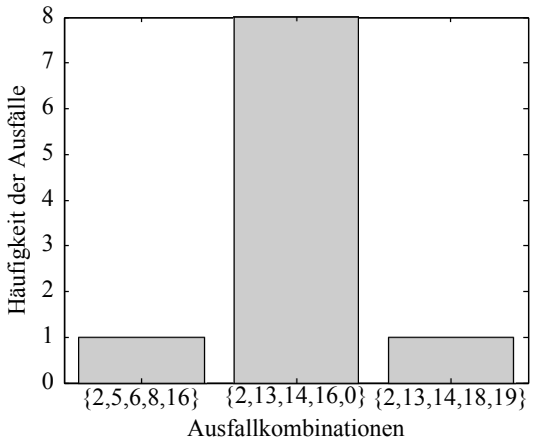

Abbildung 6.9: Ausfallkombination für die Systemfunktion o_1 aus Beispiel 3

Die relevanten Zuverlässigkeitsnetze, die die Ausfälle und somit die Schnitte näher beschreiben, sind in den Abbildungen 6.10 bis 6.12 dargestellt. In diesen Abbildungen sind nur die relevanten Schnitte die zum Systemausfall führen dargestellt.

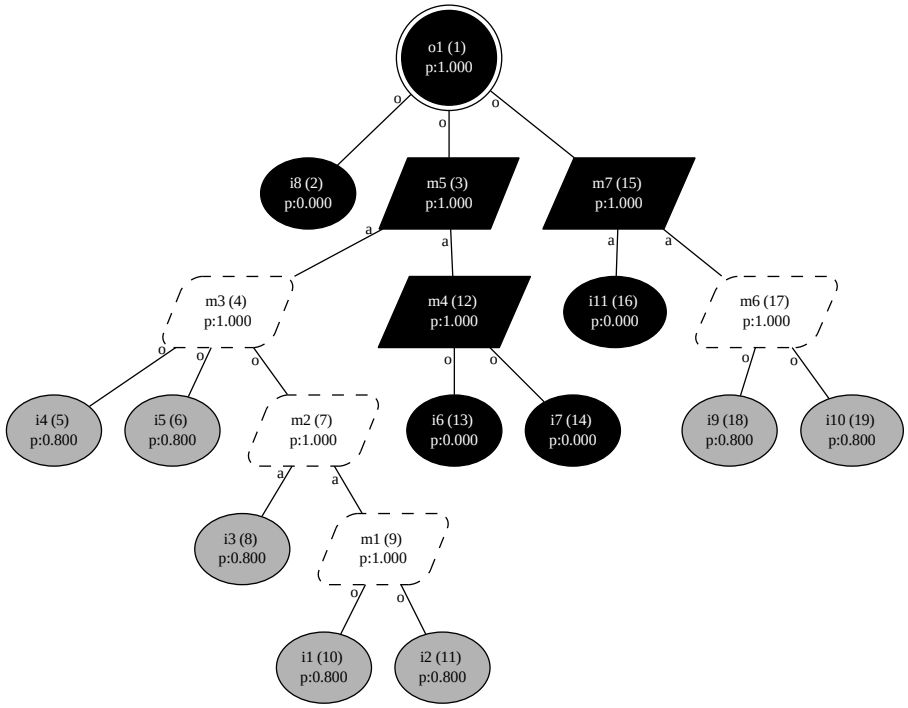

Abbildung 6.10: Fehlernetz für die Ausfallkombination $C_{Sch_1} = \{2, 13, 14, 16\}$

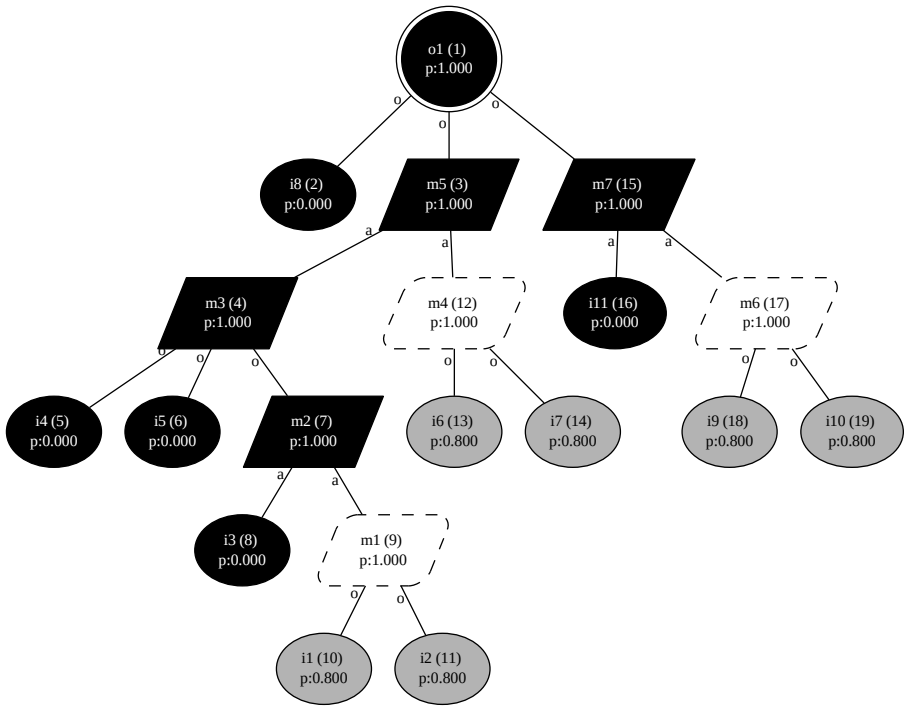

Abbildung 6.11: Fehlernetz für die Ausfallkombination $C_{Sch_2} = \{2, 5, 6, 8, 16\}$

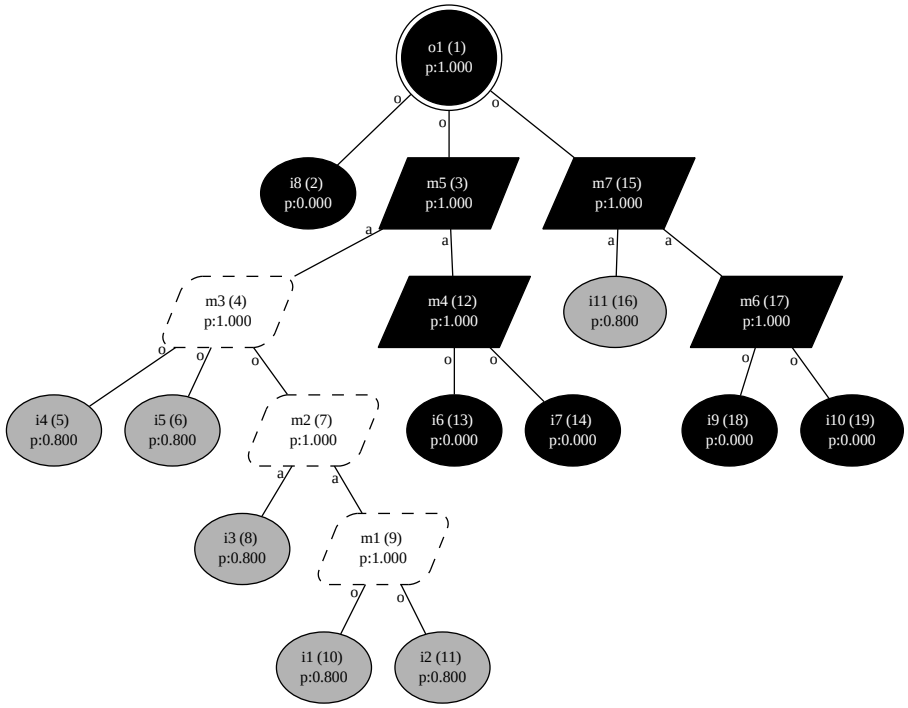

Abbildung 6.12: Fehlernetz für die Ausfallkombination $C_{Sch_3} = \{2, 13, 14, 18, 19\}$

Durch die automatisierte Analyse der Zuverlässigkeitsnetze ist es möglich, die Betrachtung auf 7-fach-Fehlern ($k = 7$) zu erweitern. Bei dem betrachteten System gibt es unter der Annahme, dass 11 Eingänge ($n = 11$) ausfallen können, mit

$$\binom{n}{k} = \binom{11}{7} = 330, \tag{6.15}$$

330 mögliche Ausfallkombinationen. In Abbildung 6.13 sind die Ausfallkombinationen dargestellt, die einen Systemausfall verursachen. Von allen möglichen Ausfallkombination führen 68 Ausfallkombinationen zum Ausfall der Systemfunktion o_1. Eine Analyse der Daten liefert die minimalen Schnitte die zum Ausfall führen und erhöht die Abbildungsgenauigkeit des Systems durch die Strukturfunktion. Mit den Schnitten

$$\left.\begin{array}{l} C_{Sch_1} = \{2, 13, 14, 16\},\ C_{Sch_2} = \{2, 5, 6, 8, 16\}\ , \\ C_{Sch_3} = \{2, 13, 14, 18, 19\}\ ,\ C_{Sch_4} = \{2, 5, 6, 8, 18, 19\}\ , \\ C_{Sch_5} = \{2, 5, 6, 10, 11, 18, 19\} \text{ und } C_{Sch_6} = \{2, 5, 6, 10, 11, 16\} \end{array}\right\} \tag{6.16}$$

ergibt sich für die 7-fach Fehlerbetrachtung die Strukturfunktion

$$\begin{aligned} \phi(\underline{x}) = 1 - & (1 - x_2 \cdot x_{13} \cdot x_{14} \cdot x_{16})(1 - x_2 \cdot x_{13} \cdot x_{14} \cdot x_{18} \cdot x_{19}) \\ & (1 - x_2 \cdot x_5 \cdot x_6 \cdot x_8 \cdot x_{16})(1 - x_2 \cdot x_5 \cdot x_6 \cdot x_8 \cdot x_{18} \cdot x_{19}) \\ & (1 - x_2 \cdot x_5 \cdot x_6 \cdot x_{10} \cdot x_{11} \cdot x_{18} \cdot x_{19}) \\ & (1 - x_2 \cdot x_5 \cdot x_6 \cdot x_{10} \cdot x_{11} \cdot x_{16}). \end{aligned} \tag{6.17}$$

6.4 Beispiel 4: Betrachtung interner und externer Signalpfade

In einem weiteren Beispiel wird die Analyse von Signalpfaden näher beschrieben. Das hier dargestellte System stellt ein Teilsystem aus dem Anwendungsbereich verteilter Bremssysteme eines Automobils dar. Das in Abbildung 6.14 dargestellte Funktionsblockdiagramm zeigt, dass das System über mehrere Signalpfade an die Informationen der Sensoren v, a, i_4, i_4 gekoppelt ist.

Für betrachtete Systemfunktion o_1 entsteht durch die direkte Kopplung mit einer externen Komponente u_5 eine zusätzliche Abhängigkeit. Die für die Bewertung relevanten funktionalen Zusammenhänge werden ausgehend von der Systembeschreibung und des FBD's in die Systemmatrix (Tabelle 6.5) überführt. Eine ergänzende Fehlerbetrachtung für das System wird durch die Betrachtung der Fehler e_1 und e_2 in die Matrix aufgenommen. Der Fehler e_1 beschreibt einen Fehler, der dem Steckverbinder 1 zugeordnet wird und zur Folge hat, dass die Komponenten c_1, c_4 und u_1 ausfallen. Ein

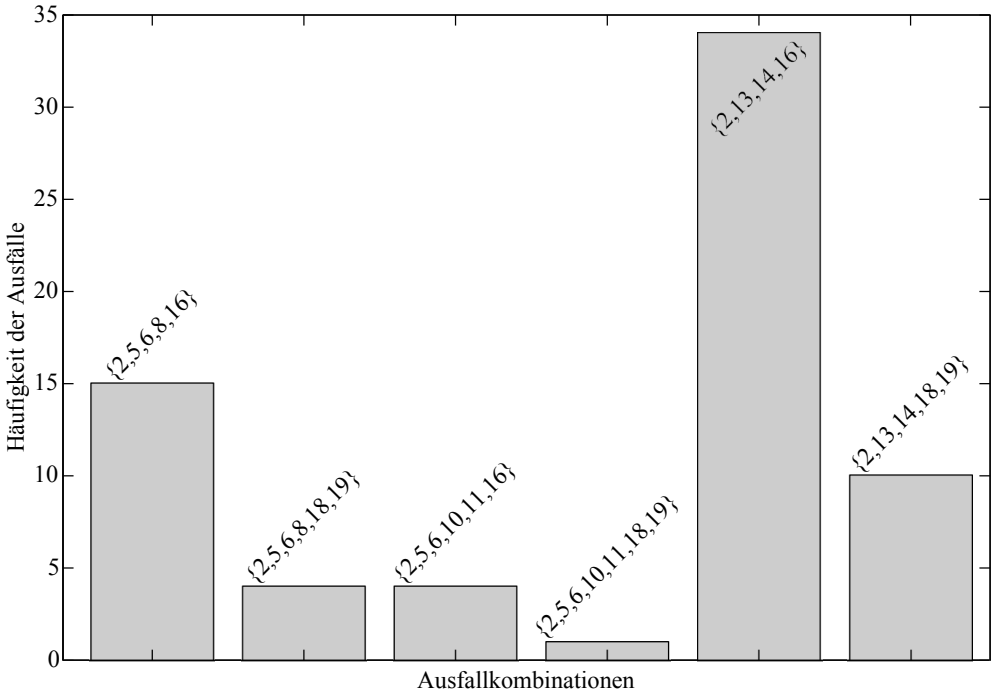

Abbildung 6.13: Ausfallkombination für 7-fach Fehler (Beispiel 3)

Fehler des zweiten Steckverbinders (e_2) wirkt sich nur auf die Kommunikationsstrecke c_3 aus.

Die Auswertung der Systemmatrix liefert das in Abbildung 6.15 dargestellte Zuverlässigkeitsnetz. In diesem Netz sind nur die relevanten Komponenten dargestellt. Die Komponenten, die keinen Beitrag zum Ausfallverhalten des Systems leisten, sind durch die Importanzkenngröße $RN(i).imp = 0$ gekennzeichnet.

Für die Betrachtung des Ausfallverhaltens des Systems sind 14 Komponenten von Bedeutung. Die Anzahl der Betrachtungsfälle kann weiter um die Komponenten, die keinen Beitrag zur Systemfunktion liefern ($RN(i).imp = 0$), reduziert werden.

Für eine 3-fach-Fehlerbetrachtung entstehen somit

$$\binom{n}{k} = \binom{14}{3} = 364 \tag{6.18}$$

Ausfallkombinationen die näher untersucht werden. Die empirische Ausfallverhaltensermittlung liefert die Schnitte

$$\left.\begin{array}{l} C_{Sch_1} = \{3, 22\},\ C_{Sch_2} = \{4, 22\},\ C_{Sch_3} = \{5, 22\},\ C_{Sch_4} = \{6, 22\}, \\ C_{Sch_5} = \{7, 8, 22\},\ C_{Sch_6} = \{8, 12, 22\},\ C_{Sch_7} = \{8, 13, 22\}, \\ C_{Sch_8} = \{8, 14, 22\},\ C_{Sch_9} = \{8, 15, 22\},\ C_{Sch_10} = \{9, 22\}, \\ C_{Sch_11} = \{10, 22\} \text{ und } C_{Sch_12} = \{20, 22\} \end{array}\right\} \tag{6.19}$$

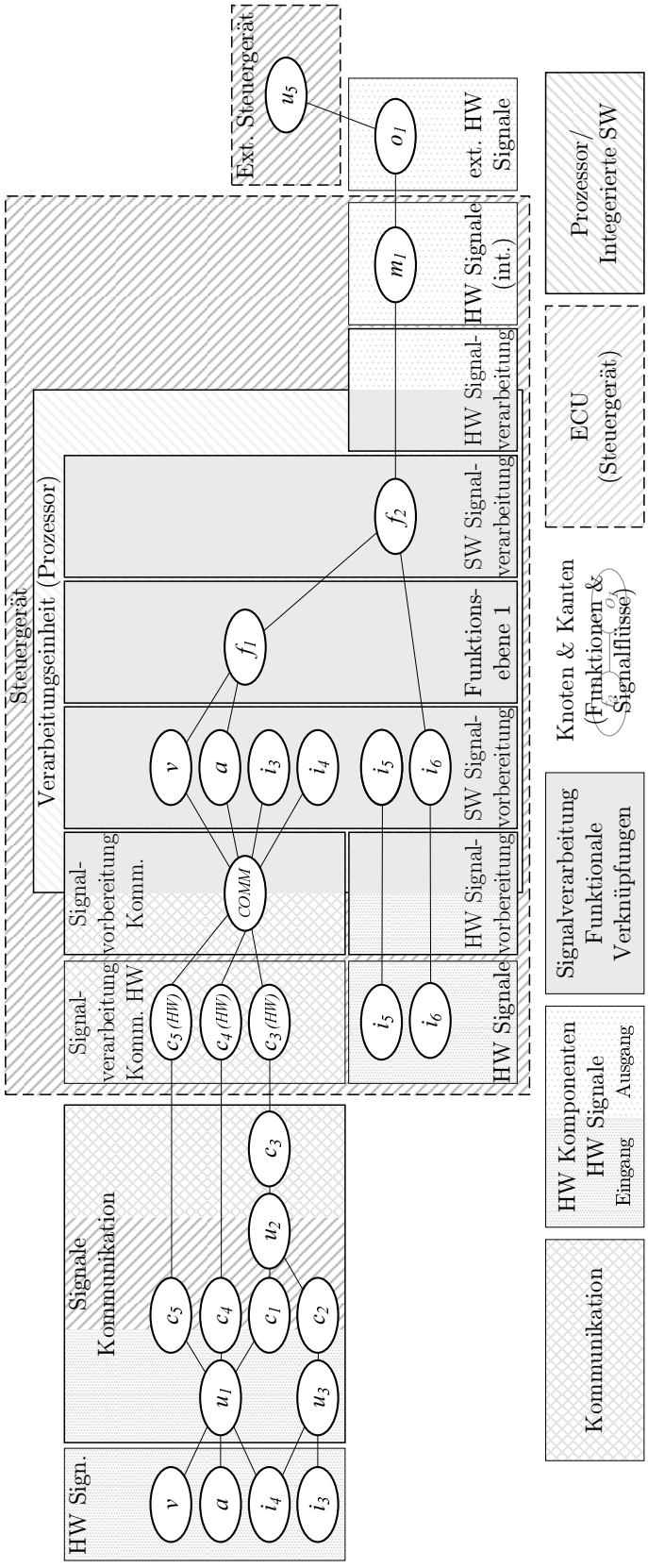

Abbildung 6.14: Funktionsblockdiagramm für Beispiel 4

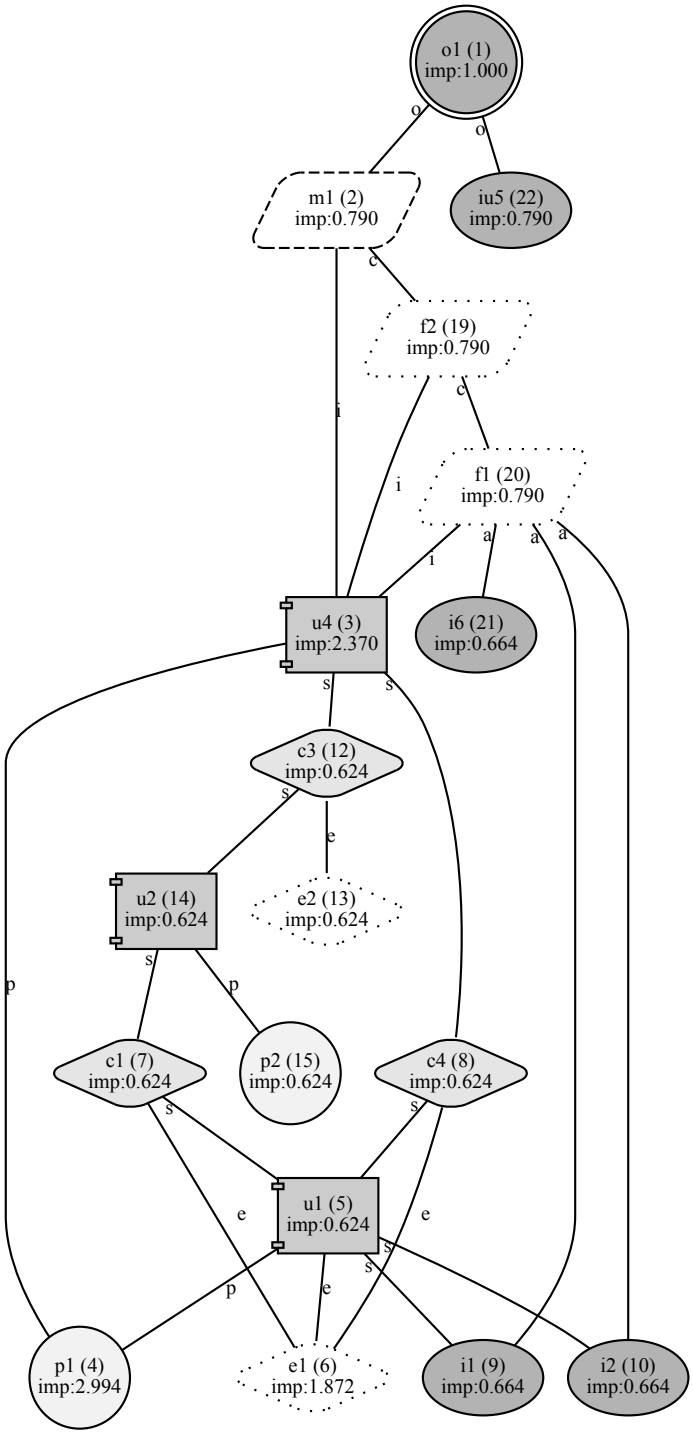

Abbildung 6.15: Zuverlässigkeitsnetz von Beispiel 4

Tabelle 6.5: Systemmatrix von Beispiel 4

Eingang/Signal	Signal/Ausgang													
	Kommunikation 1	Kommunikation 2	Kommunikation 3	Kommunikation 4	Steuergerät 1	Steuergerät 2	Steuergerät 3	Steuergerät 4	Funktion 1 (u_4)	Funktion 2 (u_4)	Modul 1 (u_4)	Sys-Funktion	in	Zuverlässigkeit (p)
Energievers. 1					p			p					p_1	0,99
Energievers. 2						p	p						p_2	0,99
v					s				a				i_1	0,999
a					s				a				i_2	0,9999
Sensor 3							s						i_3	0,999
Sensor 4					s		s						i_4	0,9999
Sensor 5													i_5	1
Sensor 6										a			i_6	1
Kommunikation 1						s							c_1	1
Kommunikation 2						s							c_2	1
Kommunikation 3								s					c_3	1
Kommunikation 4								s					c_4	1
Steuergerät 1	s			s									u_1	1
Steuergerät 2			s										u_2	1
Steuergerät 3		s											u_3	1
Steuergerät 4									i	i	i		u_4	
Steckverbinder 1	e			e	e								e_1	1
Steckverbinder 4			e										e_2	1
Funktion 1 (u_4)								i		a			f_1	
Funktion 2 (u_4)								i			a		f_2	
Modul 1 (u_4)								i				o	m_1	
Modul 5 (u_5)												o	iu_5	0,9981
out	c_1	c_2	c_3	c_4	u_1	u_2	u_3	u_4	f_1	f_2	m_1	o_1		

für das System. Die Strukturfunktion lautet

$$\begin{aligned}\phi(\underline{x}) = 1-&(1-x_3\cdot x_{22})(1-x_4\cdot x_{22})(1-x_5\cdot x_{22})\\ &(1-x_6\cdot x_{22})(1-x_9\cdot x_{22})(1-x_{10}\cdot x_{22})\\ &(1-x_{20}\cdot x_{22})(1-x_7\cdot x_8\cdot x_{22})\\ &(1-x_8\cdot x_{12}\cdot x_{22})(1-x_8\cdot x_{13}\cdot x_{22})\\ &(1-x_8\cdot x_{14}\cdot x_{22})(1-x_8\cdot x_{15}\cdot x_{22}).\end{aligned} \tag{6.20}$$

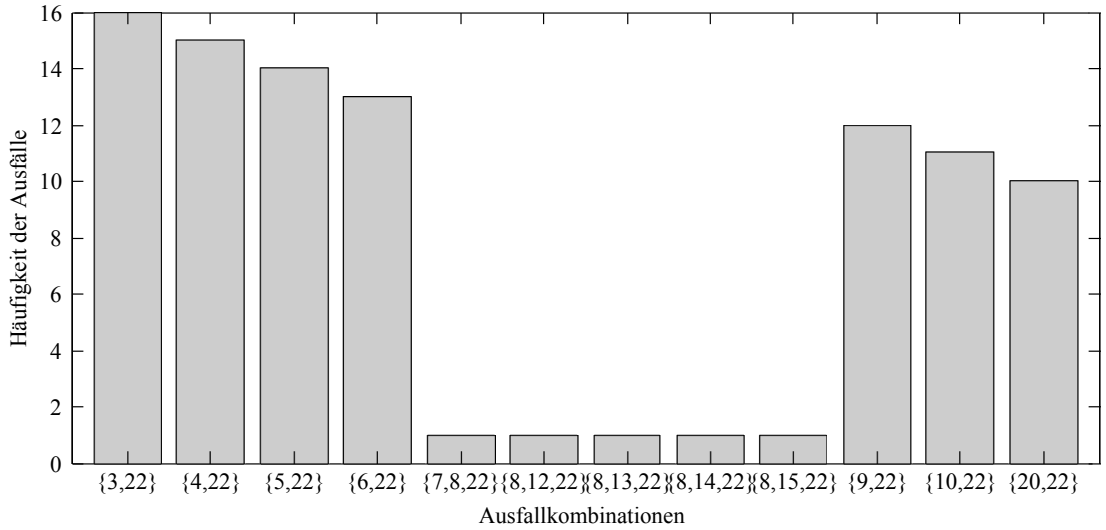

Abbildung 6.16: Ausfallkombinationen Beispiel 4 (3-fach Fehler)

Es ist zu erkennen, dass Einzelfehler in diesem System zu keinem Systemausfall führen. Aufgrund der ODER-Verknüpfung in der obersten Ebene des Zuverlässigkeitsnetzes führt erst ein Mehrfachausfall zu einem Systemausfall. Die Anzahl der von der Wurzel verzweigenden Kanten liefert die Information, dass mindestens zwei Komponenten ausgefallen sein müssen, damit es zu einem Systemausfall kommt. Der rechte Knoten, der mit der Wurzel verknüpft ist, ist nicht weiter verzweigt und stellt ein Basisereignis dar. Aus diesem Grund ist der Ausfall der Komponente iu_5 Voraussetzung für einen Systemausfall. Der linke Zweig (Komponente m_1) stellt kein Basisereignis dar und ist zur Auswertung der Systemmatrix detaillierter zu analysieren. Hierzu kann der Knoten m_1 separat oder in gesamten Kontext des Systems betrachtet werden.

7 Zusammenfassung und Ausblick

Der hier vorgestellte Ansatz unterstützt die Entwicklung von mechatronischen Systemen von der frühen Entwicklung (Konzeptphase) bis hin zur Produktion und über diese hinaus. Hierdurch wird es möglich, effizient auf Produktvariationen (Agilität) zu reagieren und kurze Entwicklungszeiten einzuhalten. Zudem werden durch die hier definierte Vorgehensweise wichtige Entscheidungsmerkmale gegenübergestellt und eine Konzeptentscheidung auf Grundlage definierter Qualitätsmerkmale ermöglicht.

Ein Systembewertungsprozess, der den Produktlebenszyklus, insbesondere den Entwicklungsprozess, von Produkten unterstützt und die Bewertung von Qualitätsmerkmalen komplexer Systeme vereinfacht, definiert die generelle Vorgehensweise zur Bewertung von mechatronischen Systemen. Dieser liefert, aufbauend auf den Anforderungsbeschreibungen, eine erste abstrakte Strukturbeschreibung des Systems, welche zyklisch über die Entwicklung hinweg unter Anwendung des Systembewertungsprozesses weiter detailliert wird.

Die Strukturbeschreibung wird in eine Systemmatrix überführt und mit bekannten Methoden der Zuverlässigkeitstechnik (z. B. Boolesche Modellbildung) analysiert. Mit der innovativ eingeführten Systemmatrix ist die formale domänenübergreifende Beschreibung der Systemstruktur unter Berücksichtigung der funktionalen und darüber hinaus der zuverlässigkeitstechnischen Zusammenhänge möglich. Hierdurch werden Untersuchungen ermöglicht, die die standardisierten zuverlässigkeitstechnischen Methoden unterstützen und erweitern.

Die Abbildungsgenauigkeit des Systems durch die Systemmatrix wächst mit dem Detaillierungsgrad der funktionalen Architektur. Zur Abbildung des Systems mittels einer solchen Matrix, wird das System in atomare Teilsysteme unterteilt. Diese Teilsysteme werden unter Anwendung des Systembewertungsprozesses quantifiziert und später wieder zu einem komplex bewerteten Gesamtsystem zusammengefügt. Die durch die Analyse erweiterte Information über das System und dessen Teilsysteme werden jeweils in einem Zuverlässigkeitsnetz zusammengefasst. Die Zuverlässigkeitsnetze erweitern die Systemmatrix um ergänzende Informationen aus dem Quantifizierungsprozess.

Der durch die Vorgehensweise entstehende Systembaukasten, stellt ein Portfolio an bewerteten Systemen oder Teilsystemen (Bauteilen) zur Realisierung und Variation von technischen Systemen bereit. Dieser Baukasten fasst alle Informationen der Zuverlässigkeitsnetze zusammen. Durch die Wiederverwendung bereits quantifizierter Systeme oder Teilsysteme wird die Konzeption neuer Systeme unterstützt und deren Entwicklungszeit reduziert. Zudem besteht die Möglichkeit, Systeme zu variieren und miteinander zu vergleichen. Hierzu liefern die Delta-Analyse und die Sensitivitätsanalyse wichtige Informationen über den Einfluss der Systemvariation und der einzelnen Komponenten. Somit kann der Einfluss der Komponenten auf das Systemverhalten bewertet und das Optimierungspotential empirisch ermittelt werden.

Die empirische Ausfallanalyse ermöglicht neben der Identifikation der Komponentenimportanz zudem die Reduktion der Komplexität der Strukturfunktion. Durch die Bewertung der Pfadimportanz wird die strukturelle Importanz von Systemkomponenten quantifiziert.

Die detailliert dargelegten Prozessschritte und Hauptpunkte sind hier als Übersicht zusammengefasst:

Systembewertungsprozess Der Systembewertungsprozess beschreibt die gesamte Vorgehensweise zur Bewertung mechatronischer Systeme unter Verwendung der Systemmatrix. Dieser umfasst die Anforderungsanalyse, Strukturanalyse, Zuverlässigkeitsmodellbildung und Quantifizierung technischer Systeme.

Systemmatrix Die Systemmatrix stellt eine domänenübergreifende Strukturbeschreibung für komplexe mechatronische Systeme dar. Diese berücksichtigt die funktionalen Aspekte des Systems und stellt eine formale Beschreibung der Struktur dar. Durch diese Matrix werden neue Möglichkeiten eröffnet Systemstrukturen zu bewerten und neue Analysemethoden anzuwenden.

Systembaukasten Der Systembaukasten unterstützt vor allem die Entwicklung von Innovationen. Bei der Erstellung der initialen abstrakten Struktur können bereits quantifizierte Teilsysteme in das System integriert werden. Zudem liefert der Systembaukasten wichtige Informationen für die Entwicklung neu zusammengesetzter Systeme.

Zuverlässigkeitsnetze Die Ergebnisse der Quantifizierung der Systemmatrix werden in Zuverlässigkeitsnetzen abgelegt und in einem Systembaukasten zusammengefasst. Zuverlässigkeitsnetze enthalten alle Informationen der Teilsysteme und können zum Gesamtsystem zusammengesetzt werden.

Entscheidungsprozess Der Entscheidungsprozess fasst den Systemquantifizierungs- und Optimierungsprozess zusammen. Die Gegenüberstellung verschiedener Systemarchitekturen und ihrer Qualitätsmerkmale unterstützt den Entscheidungsprozess. Ein Vergleich der Systeme wird durch die Deltaanalyse ermöglicht und der Einfluss von Komponenten auf die Qualitätsmerkmale kann durch die Sensitivitätsanalyse bewertet werden.

In der vorliegenden Arbeit werden erstmalig die nachstehenden Probleme

- durchgängige Beschreibung der funktionalen kausalen Zusammenhänge,
- durchgängige Beschreibung der zuverlässigkeitstechnischen Zusammenhänge,
- Strukturbeschreibung unter Berücksichtigung der Zuverlässigkeit,
- Strukturbeschreibung unter Berücksichtigung aller mechatronischen Domänen,
- Bewertung struktureller Importanz auf Komponentenebene (Pfadimportanz),
- empirische Ausfallanalyse zur Vereinfachung der Strukturfunktion und
- Reproduzierbarkeit der Analyseergebnisse über den gesamten Lebenszyklus des Produktes hinweg

definiert und Methoden zur systematischen Lösung dargelegt.

Wie in ausgewählten Beispielen gezeigt, wird so mit dieser Arbeit eine Grundlage für die, aus mechatronischer Sicht, integrative formale Systemstrukturbeschreibung geschaffen und die Etablierung von neuartigen Beschreibungsformen zur automatisierten Systembewertung dargestellt. Die neu eingeführte Systemmatrix und der daraus resultierende Systembaukasten sind die zentralen Elemente, der in dieser Arbeit vorgestellten Methodik. Die Komplexität von Systemen wird beherrschbar und eine formale Strukturbeschreibung möglich. Durch die Anwendung der hier dargestellten Vorgehensweise wird insbesondere während der Konzeptphase die Entscheidungssicherheit erhöht und zudem die Gegenüberstellung von Qualitätsmerkmalen möglich. Eine Einschränkung bezüglich des Anwendungsbereiches dieser Methode liegt nicht vor. Wichtig ist, dass die unterschiedlichen Entwicklungsziele durch die Festlegung der Qualitätsmerkmale angepasst werden. Erstmalig sind auch die Möglichkeiten zur automatisierten Zuverlässigkeitsanalyse basierend auf der Systemspezifikation dargestellt.

Der vorgestellte Ansatz führt eine formale Strukturbeschreibung von Systemen ein und erläutert die Möglichkeiten, diese Struktur aus den Anforderungen abzuleiten. Diese basiert auf dem Expertenwissen und ist vom Detaillierungsgrad der Beschreibung abhängig.

Für die Realisierung einer automatisierten Zuverlässigkeitsanalyse ist die Kombination von bekannten Methoden zur automatisierten Strukturermittlung möglich ([VDS99, PM01, TLS08, MPKW06]). Hierbei liefern diese die notwendigen Informationen über die Zuverlässigkeitsstruktur. Diese Struktur lässt sich in die Systemmatrix überführen und kann durch den Systembewertungsprozess quantifiziert werden. Da diese Methoden bislang noch keine domänenübergreifende Beschreibung von Systemen liefern, war eine Erweiterung notwendig. Die automatisierte Zuverlässigkeitsanalyse aus der Systemspezifikation heraus wird in dieser Arbeit erstmalig in Betracht gezogen. Ähnlich wie bei Modellen zur Beschreibung der technischen Zusammenhänge wird eine Erweiterung um zuverlässigkeitstechnische Informationen durchgeführt. Dies ermöglicht die automatisierte Ermittlung der Zuverlässigkeitsstruktur und die anschließende Überführung in die Systemmatrix.

Die vorgestellte Methode wurde für die Analyse von Booleschen Modellen ausgelegt. Jedoch werden in der Zuverlässigkeistechnik immer häufiger mehrwertige Modelle verwendet, wobei das Markovsche Modell eines der bedeutendsten Modelle darstellt. Durch diesen Ansatz werden detailliertere Ausfallbetrachtungen möglich und der Einfluss von Reparaturen und Servicetätigkeiten kann berücksichtigt werden. Aktuell wird im Hinblick auf mehrwertige Modelle die Systemmatrix derart erweitert, dass eine Zustandsbetrachtung möglich wird.

Für die Analyse moderner Systeme kann es sinnvoll sein, deterministische Aspekte der Zuverlässigkeitsbewertung zu betrachten. In [DP07a, DP07b, DVG97, XXR08] werden

Methoden vorgestellt, die das dynamische Verhalten von Systemen abbilden. Diese Betrachtungen lassen sich durch die Erweiterung der Matrixnomenklatur einfach erfassen und ermöglichen einen noch größeren Anwendungsbereich der Matrix. Mit der Betrachtung von dynamischen Effekten können Redundanzeigenschaften von Systemen besser abgebildet und untersucht werden.

Unterschiedliche Methoden der Zuverlässigkeitsbewertung erfordern unterschiedliche Modelle. Für die Modellbildung ist eine Überführung der Systemmatrix oder des Zuverlässigkeitsnetzes in das gewünschte Zuverlässigkeitsmodell notwendig, was durch die Matrix auf einfache Art unterstützt wird. Hierdurch wird die Anwendung standardisierter Hilfswerkzeuge zur Bewertung von Zuverlässigkeit unterstützt.

Das Potenzial der entwickelten Methode liegt vor allem in der einfachen Adaption an geänderte Voraussetzungen und der leicht realisierbaren Erweiterung der Matrix. Ausfallbetrachtungen und Zuverlässigkeitsbewertungen lassen sich an die Anwendungsbereiche anpassen. Hierbei ist die Beschränkung der ausschließlichen Anwendung auf technische Produkte nicht gegeben. Die Integration in den Entwicklungsprozess unterstützt den Entwickler bei seiner täglichen Arbeit und liefert eine transparente Darstellung der zuverlässigkeitstechnischen Zusammenhänge. Eine kontinuierliche Anwendung dieser Methode ermöglicht es zudem, den immer kürzer werdenden Entwicklungszeiten gerecht zu werden. Durch die Bereitstellung der Strukturinformationen für den in [KJS10] dargestellten Optimierungsprozess und die zyklische Systemverbesserung ist es möglich, den Zielanforderungen gerecht zu werden. Frühe Konzeptentscheidungen werden durch den Quantifizierungsprozess ermöglicht und abgesichert. Durch die Variation der Baugruppen besteht die Möglichkeit, die Systemstruktur zielgerichtet zu variieren. Der vorgestellte Baukastenansatz bietet hierzu das notwendige Erweiterungspotential.

Literaturverzeichnis

[AUT] AUTOSAR: *AUTOSAR standard specification.* `www.autosar.org`, Abruf: 02/18/2010

[Ber09] Bertsche, B.: *Zuverlässigkeit mechatronischer Systeme. Grundlagen und Bewertung in frühen Entwicklungsphasen.* Berlin u.a. : Springer-Verlag, 2009. – ISBN 978–3–540–85089–2

[BL04] Bertsche, B. ; Lechner, G.: *Zuverlässigkeit im Fahrzeug- und Maschinenbau. Ermittlung von Bauteil- und System-Zuverlässigkeiten.* 3., überarb. und erw. Aufl. Berlin u.a. : Springer-Verlag, 2004 (VDI). – ISBN 3–540–20871–2

[Bor10] Borgeest, K.: *Elektronik in der Fahrzeugtechnik. Elektronische Ressource : Hardware, Software, Systeme und Projektmanagement.* Wiesbaden : Vieweg+Teubner Verlag / GWV Fachverlage GmbH, Wiesbaden, 2010. – ISBN 9783834893376

[Dep91] Department of Defense: *Reliability Prediction Of Electronic Equipment, MIL-HDBK-217F.* Washington, DC. : Department of Defense, 1991

[DIN81] DIN: *DIN 25424 Fehlerbaumanalyse, Methode und Bildzeichen.* Berlin : Beuth-Verlag, 1981

[DIN89] DIN: *DIN 19250 Grundlegende Sicherheitsbetrachtungen für MSR-Schutzeinrichtungen.* Berlin : Beuth-Verlag, 1989

[DIN90] DIN: *DIN 40041 Zuverlässigkeit - Begriffe.* Berlin : Beuth-Verlag, 1990

[DIN00] DIN: *DIN EN 50126 Spezifikation und Nachweis der Zuverlässigkeit, Verfügbarkeit, Instandhaltbarkeit, Sicherheit (RAMS).* Berlin : Beuth-Verlag, 2000

[DP07a] Distefano, S. ; Puliafito, A.: Dynamic Reliability Block Diagrams: Overview Of A Methodology. In: *Safety and Reliability Conference (ESREL07)*, Taylor & FrancisStavanger, Norway, Juni 2007, S. 1059–1068

[DP07b] Distefano, S. ; Puliafito, A.: Dynamic Reliability Block Diagrams VS Dynamic Fault Trees. In: *Reliability and Maintainability Symposium, 2007. RAMS '07. Annual*, 2007. – ISSN 0149–144X, S. 71–76

[DVG97] Dugan, J.B. ; Venkataraman, B. ; Gulati, R.: DIFtree: a software package for the analysis of dynamic fault tree models. In: *Reliability and Maintainability Symposium. 1997 Proceedings, Annual*, 1997, S. 64–70

[ECE98] ECE: *ECE R 13 H Harmonisierte Bremsen.* Berlin : BMVBS, 1998

[ECE04] ECE: *ECE R 13 Bremsen - Teil I/II.* Berlin : BMVBS, 2004

[Fre73] Frey, H.: *Computerorientierte Methodik der Systemzuverlässigkeits- und Sicherheitsanalyse; angewandt auf komplexe; technische Systeme.* Zürich, ETH Zürich, Diss., 1973

[GH08] Grunske, L. ; Han, J.: A Comparative Study into Architecture-Based Safety Evaluation Methodologies Using AADL's Error Annex and Failure Propagation Models. In: *High Assurance Systems Engineering Symposium, 2008. HASE 2008. 11Th IEEE*, 2008. – ISSN 1530–2059, S. 283–292

[GJBJ07] Gandy, A. ; Jager, P. ; Bertsche, B. ; Jensen, U.: Decision support in early development phases - A case study from machine engineering. In: *Reliability Engineering & System Safety* 92 (2007), Juli, Nr. 7, S. 921–929. – ISSN 09518320

[Gru03] Grunske, L.: Annotation of component specifications with modular analysis models for safety properties. In: *Proc. of the 1st Int. Workshop on Component Engineering Methodology, Erfurt, Germany* Bd. 110, Citeseer, 2003, S. 31–40

[Gru07] Grunske, L.: Early quality prediction of component-based systems - A generic framework. In: *Journal of Systems and Software* 80 (2007), Nr. 5, S. 678–686. – ISSN 0164–1212

[GW91] Gerpott, T.J. ; Wittkemper, G.: Verkürzung von Produktentwicklungszeiten: Vorgehensweise und Ansatzpunkte zum Erreichen technologischer Sprintfähigkeit. In: *Booz, Allen & Hamilton: Integriertes Technologie-und Innovationsmanagement, E. Schmidt, Berlin* (1991), S. 117–145

[IAB] IABG: *V-Modell XT Version 1.3.* `http://www.v-modell.iabg.de/`, Abruf: 02/18/2010

[IEC06] IEC: *IEC/EN 61508: International Standard 61508 Functional safety: Safety-related System.* Geneva, 2006

[Ise08] Isermann, Rolf: *Mechatronische Systeme. Grundlagen.* 2., vollst. neu bearb. Aufl. Berlin u.a. : Springer-Verlag, 2008. – ISBN 978–3–540–32336–5

[ISO09] ISO: *DRAFT INTERNATIONAL STANDARD ISO/DIS 26262 Road vehicles - Functional safety.* 2009

[JKES09] Junglas, M. ; Kazeminia, A. ; Eick, R. ; Söffker, D.: A practical approach for determination of reliability-oriented system topology in mechatronic systems. In: Bris, R.; Soares, C.;Martorell A. (Hrsg.): *Reliability, Risk and Safety - Theory and application* Bd. 1. Prague (Czech Republic) : Taylor & Francis Group, London, Aug. 2009, S. 1517–1524

[JWBG05] Jäger, P. ; Wedel, M. ; Bertsche, B. ; Göhner, P: Zuverlässigkeitsbewertung softwareintensiver mechatronischer Systeme in frühen Entwicklungsphasen. In: *Mechatronik 2005* (2005)

[KJS09] Kazeminia, A. ; Junglas, M. ; Söffker, D.: Optimization of system component reliability characteristics at early design stage with economically reasonable uncertainty level. In: Bris, R.; Soares, C.;Martorell A. (Hrsg.): *Reliability, Risk and Safety - Theory and application* Bd. 1. Prague (Czech Republic) : Taylor & Francis Group, London, Aug. 2009, S. 1623–1627

[KJS10] Kazeminia, A. ; Junglas, M. ; Söffker, D.: An approach for reliability optimization of mechatronic systems during design phase. In: Ale B.; Papazoglou I.; Zio, E. (Hrsg.): *Reliability, Risk and Safety - Back to the Future* Bd. 1. Rhodos (Greece) : Taylor & Francis Group, London, Sept. 2010. – ISBN 978–0–415–60427–7, S. 1505–1512

[Kra00] Krause, Werner: *Gerätekonstruktion. in Feinwerktechnik und Elektronik*. 3., stark bearb. Aufl. München, Wien : Hanser-Verlag, 2000. – ISBN 3–446–19608–0

[Kra10] Kraftfahrzeugbundesamt (KBA): *Jahresbericht 2010*. 2010

[LCS91] Leveson, N.G. ; Cha, S.S. ; Shimeall, T.J.: Safety verification of Ada programs using software fault trees. In: *IEEE Software* 8 (1991), Juli, Nr. 4, S. 48–59. – ISSN 07407459

[Lev86] Leveson, Nancy G.: Software safety: why, what, and how. In: *ACM Computing Surveys* 18 (1986), Juni, Nr. 2, S. 125–163. – ISSN 03600300

[LH83] Leveson, N.G. ; Harvey, P.R.: Analyzing Software Safety. In: *IEEE Transactions on Software Engineering* SE-9 (1983), Sept., Nr. 5, S. 569–579. – ISSN 0098–5589

[LR98] Liggesmeyer, P. ; Rothfelder, M.: Improving system reliability with automatic fault tree generation. In: *IEEE Comput. Soc* (1998), S. 90–99. ISBN 0–8186–8470–4

[Mar04] Martinus, Marcus: *Funktionale Sicherheit von mechatronischen Systemen bei mobilen Arbeitsmaschinen*, Technische Universität München, Diss., 2004

[MP03] Meyna, A. ; Pauli, B.: *Taschenbuch der Zuverlässigkeits- und Sicherheitstechnik. quantitative Bewertungsverfahren.* München u.a. : Hanser, 2003 (Praxisreihe Qualitätswissen). – ISBN 3–446–21594–8

[MPKW06] McKelvin, M ; Pinello, C ; Kanajan, S ; Wysocki, J: Model-based design of heterogeneous systems for fault tree analysis. In: *J. Gauthier (eds.), System* (2006)

[Niz07] Niz, D. de: *Diagrams and Languages for Model-Based Software Engineering of Embedded Systems: UML and AADL.* 2007

[NWV03] Nagappan, N. ; Williams, L. ; Vouk, M.A.: Towards a Metric Suite for Early Software Reliability Assessment. In: *International Symposium on Software Reliability Engineering FastAbstract Denver CO*, Citeseer, 2003, S. 238–239

[Obj10a] Object Management Group: *Systems Modeling Language (SysML).* `http://www.omgsysml.org/`. Version: 2010, Abruf: 18.02.2010

[Obj10b] Object Management Group: *Unified Modelling Language (UML).* `http://www.uml.org/#UML2.0`. Version: 2010, Abruf: 18.02.2010

[PBFG07] Pahl, G. ; Beitz, W. ; Feldhusen, J. ; Grote, K.H.: *Konstruktionslehre: Grundlagen erfolgreicher Produktentwicklung; Methoden und Anwendung.* 7. Aufl. Berlin u.a. : Springer-Verlag, 2007 (Springer-Lehrbuch). – ISBN 3–540–34060–2

[Pic09] Pickard, Karsten: *Erweiterte qualitative Zuverlässigkeitsanalyse mit Ausfallprognose von Systemen.* Holzgartenstr. 16, 70174 Stuttgart, Universität Stuttgart, Diss., 2009

[PM01] Papadopoulos, Y. ; Maruhn, M.: Model-Based Synthesis of Fault Trees from Matlab-Simulink Models. In: *Dependable Systems and Networks, International Conference on* 0 (2001), S. 77–82

[PPG04] Papadopoulos, Y. ; Parker, D. ; Grante, C.: A method and tool support for model-based semi-automated failure modes and effects analysis of engineering designs. In: *SCS '04: Proceedings of the 9th Australian workshop on Safety critical systems and software.* Darlinghurst, Australia, Australia : Australian Computer Society, Inc., 2004. – ISBN 1–920–68229–5, S. 89–95

[Rak02] Rakowsky, Uwe K.: *System-Zuverlässigkeit : Terminologie, Methoden, Konzepte*. Hagen/Westfalen : LiLoLe-Verl., 2002. – ISBN 3–934447–22–8

[Soc10] Society of Automotive Engineers Standard: *SAE Architecture Analysis and Design Language (AADL)*. `http://www.aadl.info/aadl/currentsite/`. Version: 2010, Abruf: 02/18/2010

[SZ10] Schäuffele, Jörg ; Zurawka, Thomas: *Automotive software engineering. Grundlagen, Prozesse, Methoden und Werkzeuge effizient einsetzen*. 4., überarb. und erw. Aufl. Wiesbaden : Vieweg + Teubner, 2010 (Praxis). – ISBN 978–3–8348–0364–1

[TLS08] Tajarrod, F. ; Latif-Shabgahi, G.: A Novel Methodology for Synthesis of Fault Trees from MATLAB-Simulink Model. In: *Engineering and Technology* 31 (2008), Nr. Juli, S. 631–637

[TM03] Tietjen, T. ; Müller, D.H.: *FMEA-Praxis. das Komplettpaket für Training und Anwendung ; mit 22 Tabellen*. 2., überarb. Aufl. München u.a. : Hanser, 2003. – ISBN 3–446–22322–3

[VDA00] VDA: *VDA 3 Teil 2 Qualitätmanagement in der Automobilindustrie - Zuverlässigkeitssicherung bei Automobilherstellern und Lieferanten*. Bd. 3. Auflage. Frankfurt/Main : Verband der Automobilindustrie, 2000. ISSN 0943–9412

[VDA06] VDA: *VDA 4.2 Qualitätmanagement in der Automobilindustrie - Sicherung der Qualität vor Serieneinsatz - System FMEA*. Bd. 3. Auflage. Frankfurt/Main : Verband der Automobilindustrie, 2006

[VDI93] VDI: *VDI 2221 Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte*. Berlin, Juni 1993

[VDI04] VDI: *VDI 2206 Entwicklungsmethodik für mechatronische Systeme*. Berlin, Juni 2004

[VDI07] VDI: *VDI 2180 Sicherung von Anlagen der Verfahrenstechnik mit Mitteln der Prozessleittechnik (PLT)*. Berlin, Juni 2007

[VDS99] Vemuri, K.K. ; Dugan, J.B. ; Sullivan, K.J.: Automatic synthesis of fault trees for computer-based systems. In: *IEEE Transactions on Reliability* 48 (1999), Nr. 4, S. 394–402. – ISSN 00189529

[Voe99] Voegele, Arno: *Das grosse Handbuch; Konstruktions- und Entwicklungs-Management*. 2., überarb. Aufl. Landsberg/Lech : Verlag Moderne Industrie, 1999. – ISBN 3–478–91692–5

[WS04] Wolters, K. ; Söffker, D.: Diagnoseverfahren und Notlaufkonzepte mechatronischer Systeme / Forschungsvereinigung Antriebstechnik e.V. Frankfurt, 2004 (Forschungsbericht Nr. 408). – Abschlussbericht

[XXR08] Xu, H. ; Xing, L. ; Robidoux, R.: DRBD - Dynamic Reliability Block Diagrams for System Reliability Modelling. In: *International journal of computers & applications* (2008), Nr. 2, S. 132–141. – ISSN 1206–212X