Anwendbarkeit von modellgestützten Safety-Analysen

Torge Hinrichs

University of Applied Sciences,
Dept. Computer Science,
Berliner Tor 7
20099 Hamburg, Germany

I. Einleitung

"Safety can be described as a characteristic of the system of not endangering, or causing harm to human lives or the environment in which the equipment or plant operates. [...] safety evaluates system operation in terms of freedom from occurrence of catastrophic failures." -Bozzano und Villaorita[2010] [1]

Der Begriff „Sicherheit"(Safety) spielt eine entscheidende Rolle bei der Verwendung von vielen technischen Systemen. Dieser Begriff sagt aus, dass die Gefährdung eines Menschen oder Schaden an der Umwelt in der das System eingesetzt wird zu keiner Zeit auftreten darf. Um Sicherheitsaspekte bei der Entwicklung sicherzustellen, werden im Vorfeld so genannte Safety- Requirements definiert. Diese dokumentieren alle Anforderungen, die ein technisches System erfüllen muss, damit es nicht zu einem Schaden kommen kann. Grundlegende Sicherheitsanforderungen werden aus Anwendungsbereichsspezifischen und generellen Normen und Richtlinien, wie zum Beispiel die Normen der International Electrotechnical Commission für den Bereich Elektrotechnik und Elektronik [2]. Kann es durch ein technisches System zur Gefährdung Dritter kommen, so sind diese Anforderungen sogar durch den Gesetzgeber vorgeschrieben. So muss zum Beispiel im Automobilbereich die funktionale Sicherheit nach der ISO Norm 26262 gewährleistet sein [3]. In Deutschland kann dies zum Beispiel durch den TÜV überprüft werden.

In dieser Arbeit werden zunächst weitere grundlegende Begrifflichkeiten eingeführt. Der Hauptteil befasst sich mit einem Überblick über eine Auswahl verschiedener Techniken, die im Bereich der Safety-Analysen im Einsatz sind. Des Weiteren wird ein Bezug zum aktuellen Stand der Forschung hergestellt. Abschließend werden die Erkenntnisse zusammengefasst und in einem Ausblick die weiteren Arbeitspakete für das Grundprojekt vorgestellt.

II. Motivation

In sicherheitskritischen Systemen muss die ordnungsgemäße Funktion belegt oder sogar bewiesen werden. Dies gilt besonders, wenn durch die Benutzung des Systems der jeweilige Anwender oder sogar Dritte zu schaden kommen können. In der Industrie gestalten sich solche Belege aber als schwierig. Sie sind sehr zeit- und ressourcenaufwendig. In Bereichen wie der Luft- und Raumfahrt oder dem Bahnsektor müssen diese Belege durch externe Unternehmen erbracht werden, ansonsten kann das Produkt nicht zertifiziert und zugelassen werden.

Warum also Safety-Analysen? Safety - Analysen sollen dabei helfen zu prüfen, ob die ordnungsgemäße Funktionalität gegeben ist, also die Requirements eingehalten werden. Des Weiteren können durch diese Analysen mögliche Fehlerquellen und kritische Bereiche identifiziert werden, die durch normale Tests nicht entdeckt worden wären.

III. Problemstellung

Safety Analysen sind sehr Zeitaufwendig und erfordern ein hohes Aufkommen an Ressourcen. Obwohl diese Analysen meist durch Tools unterstützt werden müssen sie manuell durchgeführt werden. Des Weiteren sind die zu überprüfenden Systeme in der Regel groß und oder in einer Vielzahl vorhanden.

Das Ziel dieser Arbeit ist es daher den enormen Aufwand dieser Safety Analysen zu senken. Dies kann entweder durch eine Vereinfachung der Verfahren oder durch eine Automatisierung von Abschnitten innerhalb des Analyseprozesses erfolgen.

Um jedoch einen Einstieg in dieses Themengebiet zu ermöglichen befasst sich der folgende Abschnitt mit der Terminologie, die für dieses Thema essenziell sind.

IV. Grundlegendes

Dieser Abschnitt werden bereits benutzte und für die weitere Arbeit wichtige Begriffe erläutert.

- **Safety**
  Safety beschreibt die Freiheit eines Systems von unvertretbaren Risiken (DIN EN 61508-4)
- **Safety-Requirements**
  Safety-Requirements sind Spezifikationen, die die Anforderungen an die Sicherheitsfunktionen eines Systems beinhalten (IEC 61511).
- **Safety-Critical / Safety-Related System**
  Ein Safety-Critical bzw. Safety-Related System bezeichnet ein System, dass die Sicherheit der Umgebung und beteiligter Personen, sowie Dritter im Sinne von Safety, garantiert. [4]
- **Funktionanale Sicherheit**
  Funktionale Sicherheit beschreibt die Abwesenheit von unzumutbarem Risiko geschuldet durch ein Fehlverhalten eines elektronischen oder elektrischen Systems (ISO 26262).
- **Funktionanale Sicherheit nach Storey [4]**
  „ *Teil der Gesamtsicherheit, bezogen auf das EUC (Equipment Under Control) [...], die von der korrekten Funktion des elektischen/elektronischen/programmierbaren*

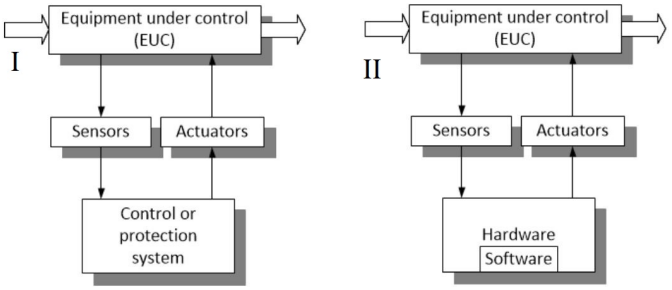

Abbildung 1. Beispiel EUC Sicherheit
Quelle: [4]

elektronischensicherheitsbezogenen Systems,[...] zur Risikominderung."

Die Abbildung 1 zeigt ein „Equipment Under Control"(EUC) [4] und abstrakte Form die funktionale Sicherheit des EUC zu gewährleisten, außerdem wird noch eine konkrete Umsetzung in einem System gezeigt. Dieses EUC ist in seiner Ausprägung nicht definiert und daher ein abstraktes Modell. Das hat zur Folge, dass es sich bei einem EUC um ein beliebiges technisches Modell handeln kann. Angefangen bei einfachen Schaltungen bis hin zu komplexen Teilsystemen. Sensoren übermitteln dabei den aktuellen Zustand des EUC an eine Kontrolleinheit. In dieser Kontrolleinheit wird entschieden, ob Aktionen zur Sicherung oder Steuerung des EUC getätigt werden müssen. Muss eine solche Aktion durchgeführt werden, so wird diese über einen oder mehrere Aktoren ausgeführt. Die Abbildung 1.II zeigt, dass die Aufgaben der Kontrolleinheit von Hardware in Kombination mit Software übernommen werden können.

Wie stark eine Funktion gesichert werden muss ergibt sich zum Beispiel aus dem Sicherheits-Integritätslevel(SIL), die in der internationalen Normung IEC 61508/IEC61511 festgelegt wird:

„*Vier wohlunterschiedene Stufen zur Spezifizierung der Anforderung für die Sicherheitsintegrität von Sicherheitsfunktionen, die dem E/E/PE-sicherheitsbezogenen System zugeordnet werden, wobei der Sicherheits-Integritätslevel 4 die höchste Stufe der Sicherheitsintegrität und der Sicherheits-Integritätslevel 1 die niedrigste darstellt.*"

Wichtig hierbei ist, dass Gerätehersteller bis zum Level 2 ihre Geräte selbst beurteilen können. Ab Level 3 muss dies durch eine unabhängige dritte Instanz geprüft werden. Ist die Prüfung erfolgreich, so wird ein entsprechendes Zertifikat ausgestellt.

Zur Definition und Prüfung der Safety spielt der Begriff des Risikos eine zentrale Rolle. Risiko einer bestimmten Ausnahme ist definiert als: [5]

$$Risiko_{Ausnahme} = Auftrittswahrscheinlichkeit_{Ausnahme} X Schweregrad_{Ausnah}$$

Wie kann jedoch dieses Risiko verringert werden? In dem Ansatz von Storey wird von zwei Arten von Fehlern ausgegangen, die ein unzumutbares Risiko verursachen, diese sind:
Systematische Fehler und zufällige Hardware Fehler. Um das gesamt Risiko zu verringern muss also die Auftrittswahrscheinlichkeit der jeweiligen Fehlerart verringert werden.
Für systematische Fehler kann dies mittels Tests erfolgen. Außerdem können Code und oder Design Reviews dabei helfen das Auftreten dieser Fehler zu verringern.
Hardware Fehler sind dagegen schwieriger zu beheben, da die Zuverlässigkeit einer Komponente von Stück zu Stück variieren kann. Üblicherweise kann man also den Ausfall von Hardware Komponenten nicht verhindern, aber die Konsequenzen, die aus dem Ausfall entstehen können verringert werden. Es folgen zwei Beispiele wie dies erreicht werden kann. Unter anderem kann der redundante Einbau einer Komponente genutzt werden. Fällt eine Komponente aus, so gibt es immer noch einen funktionsfähigen Ersatz und das beschädigte Element kann bei der nächsten Wartung ausgetauscht werden. Eine weitere Möglichkeit ist ein Hardware Selbsttest, der vor der Inbetriebnahme der Komponente durchgeführt wird. Die Komponente prüft dabei selbst ihre ordnungsgemäße Funktionalität und meldet gegebenenfalls eine Störung an das übergeordnete System.

Es bleibt jedoch die Frage, wie kritische Komponenten innerhalb eines Systems ermittelt werden können. Zu diesem Zweck werden in der Praxis Analyseverfahren genutzt, die im folgenden Abschnitt genauer betrachtet werden.

V. Übersicht über Analyseverfahren

Dieser Abschnitt skizziert einen Überblick über eine Auswahl von Analysetechniken, um sicherheitskritische Komponenten innerhalb eines Systems zu identifizieren. Im folgenden werden folgende Verfahren vorgestellt und diskutiert:

- Fehlerbaum Analyse (**F**ault **T**ree **A**nalysis, **FTA**)
- **F**ailure **M**odes and **E**ffects **A**nalysis (**FMEA**)
- **HAZ**ard and **OP**erability Studies (**HAZOP**)

A. Fault Tree Analysis

Die Fehlerbaum Analyse / Fault Tree Analysis (FTA) ist ein grafisches oder auch textuelles Verfahren, um die Zuverlässigkeit eines technischen Systems zu bestimmen. [6] Hierbei wird boolesche Algebra in Kombination mit Wahrscheinlichkeiten dazu genutzt die Gesamtausfallwahrscheinlichkeit eines Systems oder einer Anlage zu bestimmen. Die FTA wird überwiegend im Bereich der Luft- und Raumfahrt, sowie der Nuklear Industrie eingesetzt. Sie ist international unter IEC 61025 standardisiert und ist für die Abnahmen von Systemen für die Öffentlichkeit vorgeschrieben(Flugzeuge, Bahn,...). Das Ziel der FTA ist es alle sequenziellen und parallelen Fehlerereignisse zu identifizieren, die einen Fehlerzustand verursachen können. Auf diese Weise können kritische Pfade ermittelt werden, die zum Gesamtausfall des Systems führen können.Dieser kritische Pfad kann dann genutzt werden besonders sicherheitsrelevante Subsysteme zu ermitteln. Besonders wichtig dabei ist das so genannte „Minimal Cut Set", welches die kleinste Menge an Fehlern aufzeigt, die auftreten müssen, um das Gesamtsystem zum Ausfallen zu bringen.

Die folgende Abbildung zeigt ein einfaches Beispiel für einen Fehlerbaum.

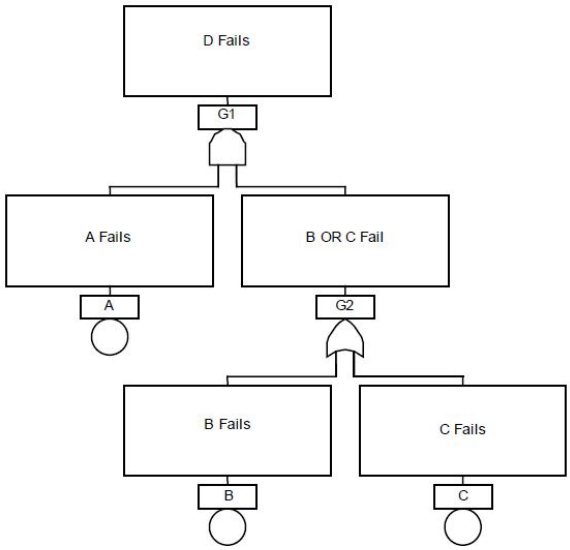

Abbildung 2. Beispiel Fehlerbaum
Quelle: [7]

Die Abbildung 2 zeigt ein einfaches Beispiel für einen Fehlerbaum. Als Wurzel des Baumes steht die zu betrachtende Komponente(vgl. Abbildung 2 D Fails), die davon abgehenden Äste zeigen Teilkomponenten, die zum Versagen der Komponente beitragen. Jede Teilkomponente besitzt einen Identifizierer(vgl. Abbildung 2: A, B, C) und ein dazugehöriges Ausfallereignis(vgl. Abbildung 2 „A Fails“, ...). Realitätsnahe Beispiele wären: „Pumpe A versagt“, oder „Ventil B schließt nicht vollständig “. Jedes dieser Ereignisse besitzt eine bestimmte Ausfallwahrscheinlichkeit, angegeben in der „Meantime To Failure “. Diese Ereignisse können mit logischen Verknüpfungen verbunden werden, dazu stehen UND- und ODER-Verknüpfungen zur Verfügung. Jede Verknüpfung besitzt einen eindeutigen Identifizierer (vgl. Abbildung 2 G2 - ODER-Verknüpung) und ein zusammen gesetztes Ereignis (B OR C Fails). Ist der Baum aufgestellt so kann, mit Hilfe der Ausfallwahrscheinlichkeiten der einzelnen Teilkomponenten und Rechenregeln für die Verknüpfungen, die Gesamtausfallwahrscheinlichkeit der Komponente bestimmt werden. Die folgenden Gleichungen zeigen die Rechenvorschriften für die UND- und ODER-Verknüpfungen:

UND - Verknüpfungen:
$P(A \cap B) = P(A)xP(B)$

ODER - Verknüpfungen:
$P(A \cup B) = P(A) + P(B) - P(A)xP(B)$

Der folgende Abschnitt beschreibt ein weiteres Analyseverfahren.

B. Failure Modes And Effects Analyse

Die Failure Modes And Effects Analyse (FMEA) [8] ist eine textuelle Methode, um mögliche Fehler in der Entwicklung bzw. des Designs frühzeitig zu erkennen. Außerdem dient sie der Fehlervermeidung und der Erhöhung der Zuverlässigkeit. Das Ziel der FMEA ist es anhand eines initiierten Fehlverhalten einer Teilkomponente das Verhalten der beteiligten Komponenten zu analysieren. Beispiele für Fehlverhalten einer Komponente sind:

- Fehlerhafte Daten
- Antwort zu früh am Ziel eingetroffen
- verzögertes Antwortverhalten
- keine Antwort

Das Fehlverhalten, sowie die Reaktion der anderen Komponenten darauf, wird in einer Tabellenstruktur festgehalten. Die folgende Grafik zeigt ein Beispiel einer FMEA.

Data Label	Data Item	Data Fault Type	Description	Effect
999	MODE WD 1	Absent Data or Timing of Data Wrong	No Data (Flight Control Failure) or Obsolete Data (Flight Control Failure should be declared)	ACTIVE ALTITUDE TYPE ^{X,Y}, ALTITUDE ARM/ALTITUDE ABORT: VERTICAL ARM MODE (Altitude Arm, Vertical Approach); ACTIVE ALTITUDE TYPE, MANY OTHER ELEMENTS: VERTICAL CAPTURE MODE/Vertical Degradation removed 5 sec after Flight Control Failure
999	MODE WD 1	Incorrect Data	Wrong Data (or Flight Control Failure incorrectly declared)	ACTIVE ALTITUDE TYPE ^{X,Y}, ALTITUDE ARM/ALTITUDE ABORT: VERTICAL ARM MODE (Altitude Arm, Vertical Approach); ACTIVE ALTITUDE TYPE, MANY OTHER ELEMENTS: VERTICAL CAPTURE MODE/Vertical Degradation may be missing or inadvertently displayed

Abbildung 3. Beispiel Failure Modes and Effects Analyse
Quelle: [8]

Die Abbildung 5 zeigt einen Auszug aus einer FMEA für ein Flugsteuerungsmodul. Neben einer Daten Kennzeichnung und einer Elementbeschreibung zeigt die Abbildung das initiierte Fehlverhalten, sowie eine Beschreibung dieser und den Effekt, den dieses Fehlverhalten auslöst. Bleiben Daten aus, oder haben sie ein falsches Zeitverhalten, so wird ein „Flight Control Failure“ ausgelöst. Der Effekt beschreibt in diesem Fall, neben diversen anderen Elementen, dass der Verlust fünf Sekunden nach dem Auftreten des „Flight Control Failure“ behoben wurde. Anders in der zweiten Zeile des Beispiels. Hierbei werden inkorrekte Daten gesendet. Effekt zeigt hier, dass der Fehler möglicherweise nicht erkannt wurde, oder irrtümlich angezeigt wird. Solche Analysen sind die Grundlage für weitere Techniken, wie zum Beispiel die zuvor besprochene FTA. Wichtig dabei ist, dass die FMEA bereits in der Entwurfsphase eines Systems durchgeführt werden kann. Somit kann sie dazu beitragen Fehler oder Mängel in einem Design frühzeitig zu erkennen und zu korrigieren. Folglich können die Projektkosten aufgrund einer guten Vorsorge gesenkt werden,

statt durch eine aufwendige Nachsorge.
Eine Erweiterung der FMEA ist die Failure Modes Effects and Criticality Analysis(FMECA). Hierbei werden zusätzlich zu einer FMEA auch noch eine Bestimmung der Auswirkungen der Fehlerzustände vorgenommen. Dies Erfolgt in vier Kategorien, die in dem Standard MIL–STD–882 [9] festgelegt sind. Die Tabelle I zeigt die Festlegung dieser Kategorien.

Tabelle I. Härtegrade eines Fehlers

Kategorie	Beschreibung	Kriterien
1	katastrophal	Kann zu Toten führen, dauerhafter kompletter Verlust der Funktionalität, Verluste übersteigen mehr als 1 Million Dollar, irreversible Schäden an der Umgebung, die Gesetze und Regularien verletzen
2	kritisch	dauerhafter teilweiser Verlust der Funktionalität, Krankheiten oder Verletzungen, die in einem Krankenhaus Aufenthalt resultieren, Verluste zwischen 200 Tausend und 1 Million Dollar oder besteigbarer Schaden an der Umwelt
3	marginal	Verletzungen oder Erkrankungen, die in einem oder mehreren Werktagen kuriert sind, Verluste zwischen zehn und 200 Tausend Dollar, hinnehmbarer Schaden an der Umwelt, der gegen keine Gesetze oder Regularien verstößt
4	unerheblich	Verletzungen oder Erkrankungen, die keinen Werktag zum Kurieren benötigen, weniger als zwei Tausend Dollar Schaden oder minimaler Schaden an der Umwelt

C. Hazard and Operability Analyse

Die Hazard and Operability Analyse (HAZOP Analyse) [10] ist ein Verfahren zum Identifizieren von kritischen Situationen in technischen Systemen. Zu diesem Zweck wird ein Expertenteam aus unterschiedlichen Fachrichtungen zusammen gestellt und unter der Leitung eines Moderators. Das Team analysiert nun systematisch alle Teilkomponenten eines Systems auf mögliche Probleme. Dieses Verfahren erfordert einen großen Ressourcenaufwand, da das Expertenteam aus Mitgliedern mit langjährigen Erfahrungen erfordert, um auch im Vorfeld nicht typische oder sogar sehr seltene Szenarien zu entwerfen, in dem das System Fehler ausweisen könnte. Darüber hin aus müssen nicht nur die Fehlerzustände identifiziert werden, sondern auch alle Konsequenzen in vollem Umfang erfasst werden.
Eine HAZOP Analyse wird wie folgt durchgeführt [11]:
Anfangs wird festgelegt, welches Teilsystem oder Komponente untersucht werden soll. Des Weiteren wird durch befragen des Expertenteams ermittelt, welches Soll-Verhalten diese Komponenten haben soll. Die HAZOP Analyse betrachtet nun alle Verhaltenszustände, die von diesem Soll-Verhalten abweichen. Besonders im Fokus stehen dabei Zustände, die den Normalbetrieb beeinflussen können, oder sogar eine Gefahr für Personal, die Umgebung oder Dritte darstellen. Dabei stellt der Moderator durch so genannte „Guide Words “ sicher, dass bestimmte Aspekte ausreichend beleuchtet werden und die Analyse jedes Teilsystems systematisch abläuft. Der IEC Standard 61882 [12] legt dabei die Rolle dieser „Guide Words“ fest:

The identification of deviations from the design intent is achieved by a questioning process using predetermined “guide words”. The role of the guide word is to stimulate imaginative thinking, to focus the study and elicit ideas and discussion.

Wichtig ist jedoch dabei, dass diese „Guide Words “ je nach Anwendungsfall angepasst werden können bzw. müssen. Die folgende Abbildung zeigt ein Beispiel für typische Guide Words.

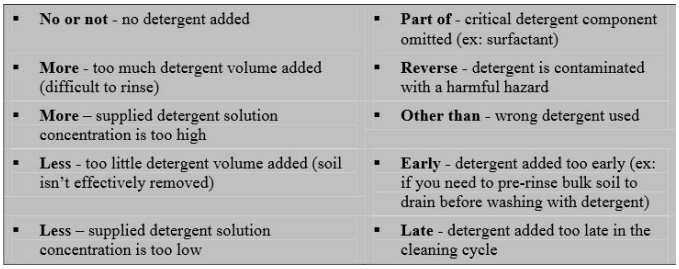

Abbildung 4. Beispiel Guide Words für eine HAZOP Analyse
Quelle: [13]

Nach diesem Verfahren werden nun alle Teilkomponenten untersucht, solange bis alle Teile des Systems und ihre übergeordneten Systeme analysiert wurden.

Die HAZOP Analyse ist ein sehr Zeit und Ressourcen aufwendiges Verfahren, welches sich aber in der Industrie weitgehend eingesetzt wird. Wichtig ist hierbei, dass ‚egal wie gründlich diese Analyse durchgeführt wird, es keine Garantie gibt, alle Verhaltenszustände des Systems beleuchtet zu haben. Dennoch eignet sich die HAZOP Analyse gut als eine Ausgangspunkt zu dem zusätzlich noch weitere Analysen des Systems durchgeführt werden müssen.

Der nun folgende Abschnitt befasst sich mit anderen Arbeiten und Ansätzen, die verwandt mit dem Thema dieser Arbeit sind.

VI. Verwandte Arbeiten & aktueller Stand der Forschung

Im Bereich der Safety-Analysen mit FTA/FMEA werden in der aktuellen Forschung insbesondere die modellbasierten Analysen (model-based Safety Analysis, MBSA) betrachtet.

Beispielsweise ist es möglich mit Hilfe von Methoden und einer Tool Unterstützung modellbasierte Safety-Analysen und modellbasiertes System Engineering(model-based System Engineering, MBSE) zusammen an einem SysML-Modell zu betrachten. In einem zweiten Beispiel wird außerdem noch eine Variante für MBSE auf Matlab - Simulink Modellen erläutert. Wichtig ist hierbei zu erwähnen, dass Tools, die für solche Analysen genutzt werden, nach dem gleichem SIL eingestuft. Dies bedeutet, das Tool, muss die gleichen Anforderungen im Bezug auf die Sicherheit einhalten, wie die zu untersuchende Funktion. Der Systemingenieur ist dadurch in der Lage automatisch generierte Analysen durchzuführen und somit eine schnelle Verifikation des Systems, bereits in der Design Phase, zu erhalten [14]. Um diese Automatisierung durchführen zu können müssen zunächst die als Text vorliegenden Safety-Requirements in so genannte „Failure Cases“ überführt werden. Attribute an diesen Failure Cases definieren

dabei die maximale Fehlerrate, die für diese Requirements vorliegen darf. Diese Cases beschreiben dabei eine oder mehrere Funktionen. Damit der Failure Case eintritt muss also eine oder mehrere dieser Funktionen versagen. Wichtig dabei ist, dass diese Funktionen als Ausgangspunkt genutzt werden können, um die Ausbreitung des Fehlers im Design nachzuverfolgen. Die genannten Funktionen werden auf der Ebene der funktionalen Architektur beschrieben und schlussendlich in, späteren Schritten, tatsächlichen Komponenten zugeordnet. Die folgende Abbildung beschreibt den Prozess, der von einen Systemingenieur vorgenommen werden muss.

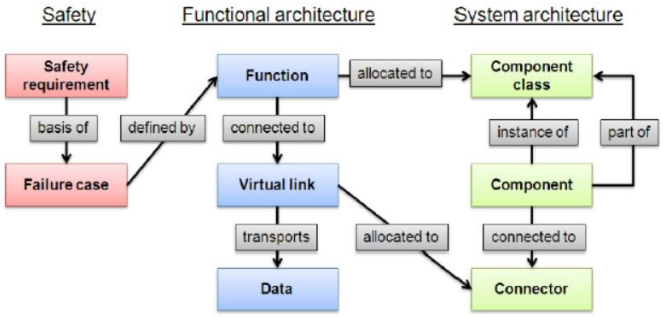

Abbildung 5. Safety Meta Modell
Quelle: [14]

Werden alle diese Schritte durchgeführt, so kann, mit Hilfe von Tools, wie zum Beispiel FaultTree+ [15] oder QuantUM [16], folgende Aussagen im Bezug auf Fehlerbäume automatisch generiert werden:

- Ein „Minimal Cut Set“, wie in Abschnitt V-A beschrieben, erzeugt werden. Dies kann für jeden Failure Case und jede System Alternative durchgeführt werden.
- Zusätzlich kann zu jedem „Minimal Cut Set“ zu Veranschaulichung ein Blockdiagramm generiert werden.
- Ein Sicherheitsbericht in dem festgehalten wird, in wie weit jeder die Failure Case und jede System Alternative die vorher definierten maximal erlaubten Fehlerraten einhält.
- Ein Bericht, der aufzeigt, ob eine System Alternative in der Lage ist alle Einschränkungen, die durch die Failure Cases definiert wurden, einzuhalten.

Abschließend ist zu erwähnen, dass die in diesem Abschnitt vorgestellte Methode und die damit verbunden Tools keinen Anspruch auf eine Zertifizierung erheben. Fernen soll dieses System nur als Unterstützung in der Design Phase dienen, um mit verringertem Aufwand alternative System Konzepte auf Basis von funktionalen Systemanforderungen zu überprüfen.

Ein weiter Ansatz in der Forschung ist die modellbasierte Erstellung von Fault Trees aus Matlab - Simulink Modellen [17]. Hier bei ist hervorzuheben, dass dieser Ansatz in allen Phasen der Entwicklung benutzt werden kann. Angefangen in der Planungs- bzw. Designphase in der die Modelle noch abstrakt sind. Bis hin zu detaillierten Komponenten mit ihren Subkomponenten, die im Laufe der Entwicklungsphasen modelliert werden. Hierbei kann es sich um Schaltungselemente oder programmierbare Identitäten handeln. Dieser Ansatz beruht auf der Annahme, dass mit voranschreitender Entwicklung das Modell immer detaillierter modelliert werden kann. Zusätzlich zu der Modellierung des Systems müssen Bereiche der Safety-Analyse berücksichtigt werden. Aus diesem Grund wird eine HAZOP Analyse (vlg. Abschnitt V-C) durchgeführt, die das fehlerhafte Verhalten einer Entität des Modells in drei verschiedene Kategorien unterteilt. Diese sind jedoch ähnlich der FMEA (vgl. V-B). Die Kategorien sind:

- Service versagt und erzeugt keinen oder fehlerhaften Output
- Output wird verspätet oder zu früh erzeugt
- Werte des Outputs unbrauchbar: verzerrt, außerhalb des Rahmens oder haben eine lineare / nicht lineare Abweichung

Dadurch entstehen logische Verknüpfungen zwischen Komponenten, die bei Versagen, Auswirkungen auf über geordnete Komponenten haben. Des Weiteren muss für jede Komponente eine Fehlerrate definiert werden. So kann zum Beispiel der Ausfall eines Prozessors zur Folge haben, dass alle daran gekoppelten Softwaresysteme mit Ausfallen und dadurch andere Hardwarekomponenten nicht steuerbar sind. Dieser Ansatz legt großen Wert auf Modularität. In Folge dessen können Ergebnisse aus Teilsystemen an anderer Stelle im Projekt oder sogar in anderen Projekten wieder verwendet werden. Hier durch kann eine Zeitersparnis geschaffen werden. Sind alle diese Vorarbeiten erledigt, so kann durch einen Algorithmus automatisch ein Fehlerbaum synthetisiert werden. Dieser Algorithmus folgt der Verbreitung eines Fehlers rückwärts durch das System angefangen bei dem Versagen eines Aktors bis hin zu den Inputs des Systems. Dabei werden alle Abhängigkeiten, die zu weiteren Fehlerzuständen in anderen Komponenten des Modells führen können. Des Weiteren werden auch alle Fehler in Betracht gezogen, die durch die Verbindung von Komponenten entstehen können, hierzu gehören verbindendes Material, Energie oder Datenfluss / Austausch. Die gewonnenen Daten werden dazu genutzt einen Fehlerbaum zu generieren. Dieser kann zur unterstützend für die Überprüfung des System Design eingesetzt, wie in Abschnitt V-A bereits beschrieben wurde. Die Abbildung 6 zeigt den Aufbau der in diesem Ansatz vorgestellten Erweiterung für Matlab Simulink.

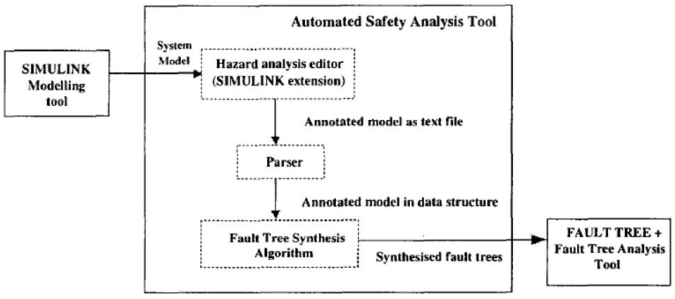

Abbildung 6. Architektur der Matlab Simulink Erweiterung
Quelle: [17]

Als Input in dieses Tool wird das System als Matlab Simulink Modell voraus gesetzt. Es folgt eine Beschreibung der einzelnen Fehlerfälle durch eine HAZOP Analyse. Diese kann dann mittels eines Parsers an den Fehlerbaum

Algorithmus übergeben werden und ein Fehlerbaum synthetisiert werden. Das Ergebnis ist ein Fehlerbaum, der nun in dafür vorgesehenen Programmen, wie zum Beispiel dem gängigen Analysetool für Fehlerbäume „Fault Tree +“ weiter untersucht werden.

Abschließend ist zu erwähnen, dass dieser Ansatz zwar in der Lage ist automatisch Fehlerbäume aus Matlab Simulink Modellen zu erzeugen, jedoch beschränkt sich dies nur auf Systeme moderater Größe. Werden die Systeme zu komplex so skaliert die Anzahl der zu untersuchenden Fehlerzustände exponentiell. Somit ist das untersuchen dieser Zustände ein NP-Vollständiges Problem und kann nicht ohne weiteres gelöst werden.

Der folgende Abschnitt dieser Arbeit befasst sich mit der Abgrenzung der eigenen Arbeit bzw. der darauf aufbauenden folgenden Arbeiten zu den verwandten bzw. ähnlichen Arbeiten.

VII. Abgrenzung zur eigenen Arbeit

Aufbauend auf diesen und ähnlichen Arbeiten werden im weiteren Verlauf Verfahren entwickelt, mit denen die Generierung von Fehlerbäumen aus unterschiedlichen Modellen ermöglicht werden kann. Des Weiteren werden die Vorzüge der verwandten Arbeiten als Grundlage für eine Fortführung in Betracht gezogen. Als Basis sollen dafür SysML Modelle genutzt werden. Es bleiben aber noch offene Fragen:

- Wie vollständig muss ein Modell sein, damit Fehlerbäume generiert werden können?
- Welche zusätzlichen Informationen müssen in einem SysML Modell enthalten sein, beziehungsweise werden überhaupt zusätzliche Informationen benötigt?
- Gibt es Voraussetzungen, die ein SysML Diagramm erfüllen muss, damit es für die Generierung eines Fehlerbaums geeignet ist?

Diese Fragen gilt es in den folge Arbeiten zu klären. Der letzte Teil dieser Arbeit gibt eine Zusammenfassung über die betrachteten Aspekte, sowie einen Ausblick in dem ein Überblick über weiterführende Arbeiten aufgezeigt wird.

VIII. Ausblick

Zusammenfassend wurde in dieser Arbeit ein Überblick über grundlegende Begrifflichkeiten im Bereich des „Safety-Analysen“ erläutert. Kernpunkte waren dabei Themen, wie relevante Begriffe:

- Safety
- Safety-Requirements
- Safety-Critial Systems
- Funktionale Sicherheit

Aber auch die gängigen Verfahren und Techniken, die zum Nachweis für funktionale Sicherheit genutzt werden. Die genannten Verfahren sind: HAZOP, FMEA und FTA. Anschließend wurden Auszüge aus dem aktuellen Stand der Forschung und Entwicklung vorgestellt. Daran anknüpfend wurde eine Abgrenzung dieser und folgender Arbeiten von dem genannten Forschungsstand aufgeführt.

Mit Blick auf weiterführende Arbeiten in folgenden Projekten ist jedoch noch unklar, welche Modelle als Ausgangspunkt betrachtet werden. Aus diesem Grund ist müssen zunächst die unterschiedlichen Modelle erzeugt und auf ihre Verwendbarkeit für dieses Vorhaben analysiert werden. Darauf aufbauend folgt eine Ausw ertung und Bewertung der einzelnen Verfahren. Als weitere Aufgabe muss betrachtet werden, in wie weit sich diese Verfahren in automatisierte Methoden umsetzen lassen. Eine Machbarkeitsanalyse wird dann aufzeigen, welche Verfahren und Modelle in einen Prototyp einfließen können.

Literatur

[1] A. V. Marco Bozzano, “Design and safety assessment of critical systems,” CRC Press Taylor & Francis Group, 2010.

[2] I. E. Commission, “Normen im bereich der elektrotechnik und elektronik,” http://www.iec.ch/ Abgerufen: Februar 2015.

[3] TÜVSüd, “Funktionale sicherheit im automobilbereich,” http://www.tuev-sued.de/automotive/elektrik_/_elektronik Abgerufen: Februar 2015.

[4] N. R. Storey, Safety Critical Computer Systems. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1996.

[5] N. Leveson, “Engineering a safer world,” IEEE, 2012.

[6] W. E. Vesely, FaultTreeHandbook. Washington, D.C. 20555: U.S. Nuclear Regulatory Commission, 1981.

[7] D. Stamatelatos, Fault Tree Handbook with Aearospace Applications. NASA Headquarters Office of Safety and Mission Assurance, 2002.

[8] R. R. Lutz, “Safety analysis of requirements for a product family,” ICRÉ98, 1998.

[9] USDeparmentofDefence, “Mil–std–882,” 2012, http://www.system-safety.org/Documents/MIL-STD-882E.pdf Abgerufen Februar 2015.

[10] T. Kletz, Hazop And Hazan. Rugby, Warwickshire CV213HQ, UK: Institution Of Chimical Engineers, 1999.

[11] J. A. McDermid, “A development of hazard analysis to aid software design,” COMPASŚ94, 1994.

[12] InternationalElectrotechnicalCommission, “Hazard and operability studies(hazop studies)–application guide,” 2001.

[13] ManufacturingTechnologyCommittee–RiskManagementWorkingGroup, “Risk management training guides,” http://www.pqri.org/pdfs/mtc/hazop_training_guide.pdf Abgerufen: Februar 2015.

[14] P. Helle, “Automatic sysml-based safety analysis,” 2012.

[15] isograph, “Faulttree+,” http://www.isograph.com Abgerufen: Februar 2015.

[16] P. D. S. L. Adrian Beer, Florian Leitner-Fischer, “Quantum,” http://quantum-tool.com Abgerufen: Februar 2015.

[17] M. M. Yiannis Papadopoulos, “Model-based synthesis of fault trees from matlab - simulink models,” 2001.