Fehlerbaumanalysen - Fallstudien

Fallstricke bei der Erarbeitung von Fehlerbäumen und deren Auswirkungen auf die Korrektheit der quantitativen Berechnungsergebnisse

Paul Granzin, 16.03.2021

Abstract

Als langjähriger RAMS/LCC-Dienstleister gehört für die IZP Dresden die Erstellung und Berechnung von Fehlerbaumanalysen zum grundlegenden Handwerkszeug. Im Zuge unserer Projektarbeiten konnten wir zahlreiche, bereits bestehende Analysen evaluieren und weiterentwickeln. Dabei sind uns des Öfteren Ungenauigkeiten bei den Modellierungen der Fehlerbäume begegnet. Das vorliegende Paper stellt eine anonymisierte und abstrahierte Zusammenfassung der aus unserer Sicht am häufigsten auftretenden Fehler dar.

Zunächst möchten wir die FTA-Methodik vorstellen. Dies dient einerseits als thematische Einführung, andererseits sollen damit die Grundlagen und das Vokabular für unsere Fallbeispiele gegeben werden. Folgende Fehler werden im Anschluss untersucht:

- Keine bewusste Unterscheidung der möglichen Top-Ereignisse (**Beispiel 1**).
- Fehlerhafte Übertragung von Zuverlässigkeitsblockdiagrammen in Fehlerbäume (**Beispiel 2**).
- Keine Berücksichtigung der Mission time (**Beispiel 3**).
- Anwendung der FTA-Methodik als Ersatz für Markow-Prozesse (**Beispiel 4**).
- Keine Untersuchung auf Unabhängigkeit der Ereignisse (**Beispiel 5**).
- Missinterpretation des Begriffs „k oo n-System“ (**Beispiel 6**).
- Keine Berücksichtigung der Instandsetzung einzelner Komponenten in einem Parallelsystem bei der Ermittlung der System-Ausfallrate (**Beispiel 7**).
- Falsche Quantifizierung aus der Not heraus (**Beispiel 8**).
- Falsche Interpretation von Kenngrößen (**Beispiel 9**).
- Unwissenheit bei der Einbindung von Inspektionszeiten bei verdeckten Fehlern (**Beispiel 10**).

Was ist eine Fehlerbaumanalyse? - Ein kurzer Umriss

Die Fehlerbaumanalyse (Fault Tree Analysis; FTA) ist eine sogenannte Systemanalyse, bei der das Zusammenwirken einzelner Komponenten eines Systems untersucht wird.

Die FTA-Methodik wird angewendet, um

- Die Einhaltung von Sicherheits-, Zuverlässigkeits- oder Verfügbarkeitsanforderungen an das System zu demonstrieren
- Kritische Komponenten/Ereignisse zu identifizieren
- Den Prozess der konstruktiven Auslegung des Systems zu unterstützen (Identifikation geeigneter Redundanzkonzepte, Allokation von Systemanforderungen auf Subsysteme)
- Betreiber- und Instandhaltungskonzepte zu bewerten und ggf. zu optimieren (Interventionszeiten, Inspektionsintervalle)

Vorgehen bei der Analyse: Ausgehend von einem unerwünschten Systemzustand (dem Top-Ereignis) wird analysiert, welche Konstellationen von Komponentendefekten zu diesem Top-Ereignis führen. Dabei wird jeweils von binären Zuständen Z=1 (intakt) und 0 (defekt) ausgegangen. Etwas formalistischer ausgedrückt wird eine sogenannte Strukturfunktion

$$Z_{Sys} = Z_{Sys}(Z_{c1}, Z_{c2}, \dots, Z_{cn})$$

erarbeitet, welche die Abhängigkeit des Systemzustandes von den Komponentenzuständen beschreibt.

Der Fehlerbaum ist dann lediglich eine grafische Repräsentation ebendieser Funktion. Die Blätter des Baumes sind die Elementarereignisse (sprich die Komponentenausfälle) und die Knoten beschreiben die Logik der Fehlerfortpflanzung bis zu dem Topereignis (der Wurzel des Baumes).

Die wichtigsten Knoten sind in der folgenden Übersicht festgehalten.

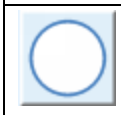
Knoten	Beschreibung
	koon-Gatter („k out of n"): Das Ereignis tritt ein, sobald mind. k der n direkt untergeordneten Ereignisse eingetreten sind
	ODER-Gatter: Das Ereignis tritt ein, sobald eines der direkt untergeordneten Ereignisse eintritt. Dies ist ein Spezialfall des koon-Gatters mit k=1
	UND-Gatter: Das Ereignis tritt ein, wenn alle direkt untergeordneten Ereignisse eingetreten sind. Dies ist ein Spezialfall des koon-Gatters mit k=n
	Elementarereignis: Keine weitere Untergliederung. Die „Blätter" des Fehlerbaumes

Tabelle 1: Gatter und Ereignisse

Darüber hinaus gibt es noch weitere Gatterarten, z.B. das „Priority-AND" (die untergeordneten Ereignisse müssen in einer bestimmten Reihenfolge eintreten) oder das „X-OR" (das Ereignis tritt genau dann ein, wenn exakt 1 untergeordnetes Element ausgefallen ist). Bei Anwendung des Priority-AND-Gatters wird eine komplexere Strukturfunktion erforderlich, da zeitliche Verläufe der Komponentenzustände den Systemzustand bedingen [1].

Ein Fehlerbaum für die Bremsfunktion eines Fahrrades könnte sich wie folgt gestalten:

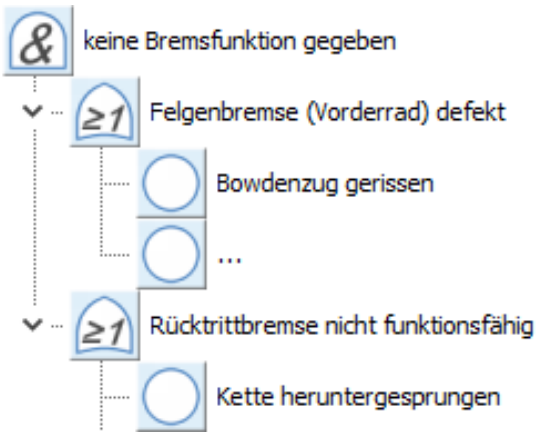


Abbildung 1: Beispielfehlerbaum

Die Ausfallraten von koon-Gattern können mithilfe der nachfolgenden Formeln approximiert werden [3], sofern die folgenden Bedingungen gegeben sind:

- Alle intakten Komponenten – auch die Reservekomponenten - erfahren die gleiche Belastung („heiße Redundanz").
- Alle Komponenten weisen die gleiche Ausfallrate λ_c und Instandsetzungsrate μ_c auf.
- Die Lebensdauern der Komponenten sind jeweils unabhängig zueinander, ebenso die Instandsetzungsdauern.

Formel	Verwendung
$\lambda_{Gatter} \approx \frac{n! * (\lambda_c)^k}{(n-k)! * (\mu_c)^{k-1}}$ (1)	Bei Instandsetzung defekter Komponenten
$\lambda_{Gatter} = \frac{\lambda_c}{\sum_{i=n-k+1}^{n} 1/i}$ (2)	Keine Instandsetzung defekter Komponenten

Tabelle 2: Redundanzformeln für Ausfallraten

Hieraus ergibt sich unmittelbar ein direkter Nutzen der FTA-Methodik.

Für die meisten Anwendungsfälle sind die oben genannten Bedingungen jedoch zu restriktiv, womit komplexere Berechnungsansätze notwendig werden.

Streng genommen folgt aus dem Sachverhalt, dass die Ausfallraten der Komponenten des koon-Systems konstant sind, im Allgemeinen nicht, dass das Gleiche für das System gilt. Die berechneten Werte λ_{Gatter} stellen somit eine Vereinfachung dar und können insbesondere als Kehrwert des System-MTBF (Mean Time Between Failure) interpretiert werden.

Kommen wir nun zu dem eigentlichen Anliegen des Papers, der Benennung von Fehlermöglichkeiten bei der Erstellung von Fehlerbäumen.

Beispiel 1 - Keine bewusste Unterscheidung der möglichen Top-Ereignisse

Bei der Modellierung von Fehlerbäumen ist die unzureichende Spezifizierung des Top-Ereignisses eine mögliche Fehlerquelle. Ein einfaches „System defekt" stellt in manchen Fällen eine unzureichende Vereinfachung dar. Stellen Sie sich dazu beispielsweise ein System vor, das den Zugang von einem geschützten Bereich in einen Gefahrenbereich kontrolliert (Intrusion detection System (ID)). Aufgabe des ID-Systems ist es, nicht-autorisierte Zugänge an die Leitstelle zu melden. Für solch ein System sind zwei Top-Ereignisse zu analysieren: „Meldung eines nicht-autorisierten Zugangs, obwohl keiner vorliegt" und „keine Meldung eines nicht-autorisierten Zugangs, obwohl einer vorliegt". Während das erste Top-Ereignis lediglich eine betriebliche Beeinträchtigung nach sich ziehen wird, stellt das zweite Top-Ereignis ein akutes Sicherheitsrisiko dar. Eine Trennung der Top-Ereignisse (und damit eine separate Berechnung der jeweiligen Ausfallraten) ist in diesem Fall also äußerst ratsam.

Im Allgemeinen müssen Sie davon ausgehen, dass unterschiedliche Top-Ereignisse mit unterschiedlichen Häufigkeiten einhergehen werden, da die jeweiligen Fehlerbäume unterschiedlich aufgebaut sein werden. Dies gilt insbesondere für unser Beispiel, dem ID-System, da hohe Zuverlässigkeitsanforderungen sicherheitsgerichteter Funktionen typischerweise mit Redundanzen realisiert werden.

Die folgende Übersicht illustriert diesen Ansatz. „Na" stelle dabei den Zustand, resp. die Meldung „nicht-autorisierter Zugang" dar und „A" entsprechend „autorisierter Zugang oder geschlossen". Für das redundant ausgelegte Sensorpaar sind nun 4 Paare von Meldungen möglich:

Sensor 1 meldet:	Na	Na	A	A
Sensor 2 meldet:	Na	A	Na	A

Tabelle 3: Meldekombinationen bei einem Sensorpaar

Wie sollen die Auswertepaare interpretiert werden, sodass der sicherheits-relevante Fehler möglichst minimiert wird? Die Auswertelogik wird in den unklaren Fällen (Na, A), (A, Na) sicherheitshalber da-von ausgehen, dass tatsächlich der Zustand Na anliegt. Nur in dem Fall, dass (A, A) gemeldet wird, wird die Logik den Zustand „A" melden.

Sensor 1 meldet:	Na	Na	A	A
Sensor 2 meldet:	Na	A	Na	A
Auswertelogik	Na	Na	Na	A

Tabelle 4: Meldekombinationen und Auswertelogik

Im Umkehrschluss bedeutet dies, dass wir das Top-Ereignis „Keine Meldung eines nicht-autorisierten Zuganges, obwohl einer vorliegt" exakt dann vorliegen haben, wenn beide Sensoren fälschlicherweise A melden, obwohl Na anliegt.

Das andere Top-Ereignis tritt ein, wenn tatsächlich A vorliegt, aber mind. einer der beiden Sensoren ein falsches Na-Signal liefert:

Die entsprechenden Fehlerbäume ergeben sich wie folgt:

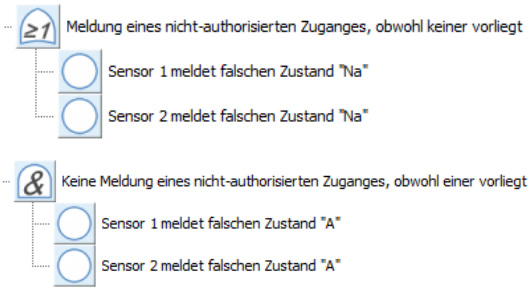


Abbildung 2: Fehlerbäume Sensorik

Intuitiv merken wir bereits, dass die Einbindung eines zweiten Sensors in Kombination mit der Auswertelogik einen signifikanten Zuverlässigkeitsgewinn für das erste Top-Ereignis bedeutet, dies aber durch häufigere Fehlalarme erkauft werden muss. Die Anwendung der Formel (1) verdeutlicht dies.

$$\lambda_{keine\ Meldung\ Na,obwohl\ Na\ vorliegt} = \frac{2 * (\lambda_{Sensor})^2}{\mu_{Sensor}}$$

$$\lambda_{Meldung\ Na,obwohl\ Na\ nicht\ vorliegt} = 2 * \lambda_{Sensor}$$

Bemerkung: Der Einfachheit halber sollen beide Fehlerarten der Sensoren die gleiche Ausfallrate λ_{Sensor} aufweisen.

Fazit: Vor Beginn der Analyse sollte in jedem Fall eine Rücksprache mit dem Initiator bzw. Auftraggeber der Analyse erfolgen, um die Motivation/ den Grund für die Untersuchung besser verstehen und somit passende Top-Ereignisse definieren zu können.

Beispiel 2 – Fehlerhafte Übertragung von Zuverlässigkeitsblockdiagrammen in Fehlerbäume

Das Zuverlässigkeitsblockdiagramm (Reliability Block Diagram – RBD) dient analog zur Fehlerbaumanalyse zur logischen und graphischen Darstellung, wie die Zustände von Systemkomponenten und deren logische Verknüpfung den Erfolgs- bzw. Fehlerzustand des Gesamtsystems beeinflussen. Während ein RBD auf den Systemerfolg fokussiert, zielt die FTA auf den Systemausfall. RBD und FTA nutzen dieselbe Mathematik auf Basis logischer Gleichungen mit Booleschen Variablen.

Abhängig von den nutzbaren graphischen Elementen ist es möglich RBD und FTA in einander zu überführen.

Für die Umwandlung von RBD in FTA gelten folgende Grundregeln:

- Funktionsblöcke → Fehlerereignisse
- parallele Struktur → UND-Gatter
- serielle Struktur → ODER-Gatter

Bei komplexeren RBD-Darstellungen ist es jedoch häufig nicht möglich, einen RBD-Block allein mit einem FTA-Gatter abzubilden. Es ist daher detailliert zu untersuchen, welche Fehlerkombinationen zum Fehler des Gesamtsystems führen. Ein typisches Beispiel dafür ist die „Brückenschaltung":

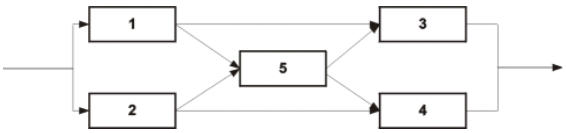


Abbildung 3: RBD einer Brückenschaltung

Für dieses System existieren vier verschiedene Fehlerkombinationen (minimale Schnitte):

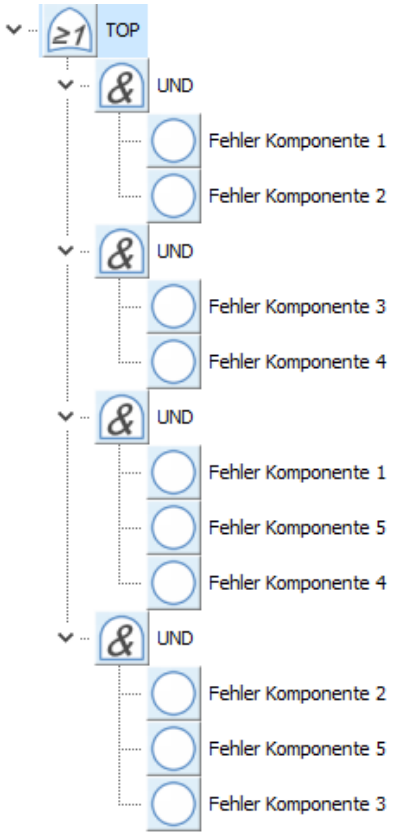


Abbildung 4: Fehlerbaum einer Brückenschaltung

Bei der Berechnung dieses Fehlerbaums ist zu beachten, dass die einzelnen Ereignisse mehrfach vorkommen bzw. jeweils gleichzeitig mehrere Gates beeinflussen und somit nicht als unabhängig betrachtet werden dürfen. Es sind sogenannte Mehrfachereignisse (siehe auch Kapitel 5).

Beispiel 3 – Keine Berücksichtigung der Mission time

Manche Betreiberkonzepte verhindern gezielt das unmittelbare Instandsetzen bei Defekten; so führen kleinere Fehler an Eisenbahnzügen zum Beispiel nicht zu einem sofortigen Austausch des betroffenen Zuges, sondern es muss damit bis zur Beendigung der Mission (z.B. bis zum Ende des Tages) gewartet werden. Ein ähnliches Szenario kann sich für schwer zugängliche Anlagen wie zum Beispiel Offshore-Windanlagen ergeben, wo Reparaturen kleinerer Defekte erst zu vorab festgesetzten Fristen (und somit mit zeitlichem Verzug) abgearbeitet werden. Insbesondere kann die eigentliche Instandsetzungszeit (der MTTR-Wert) für die Berechnung der Unverfügbarkeit unwesentlich werden, wie es das folgende Beispiel illustriert:

Gegeben sei ein redundantes HVAC-System (Heating, ventilation and air conditioning) für einen Zug. Dieser sei 18h am Tag im Einsatz (=Mission time). Mit einer intakten HVAC-Komponente kann der Zug weiterhin betrieben werden, erst der Ausfall der 2. Komponente führt zu einem Funktionsverlust und erfordert die Außerbetriebnahme des Zuges (2oo2-Gatter). Die Instandsetzung der defekten Komponente(n) erfolgt bis zum Beginn des nächsten Tages.

Während zur Berechnung der Ausfallrate des HVAC-Systems die Formel (1) durchaus angewendet werden kann, muss jedoch darauf geachtet werden, dass die Instandsetzungsrate $\mu_c = 1/9$ gesetzt wird (im Mittel ist eine Komponente 9 Stunden im Defektzustand). Bei der Berechnung der mittleren Dauer bis zu Herstellung der Systemfunktion sollte darauf geachtet werden, nicht die standardmäßige Formel für Systeme mit Instandsetzung zu wählen: Diese würde einen Wert von 4,5h=9h*0,5 liefern (im Mittel tritt der zweite Fehler nach Verstreichen der halben Downtime des ersten Komponentenfehlers ein). Im Modell „Mission time" verhält es sich aber so, dass der erste Fehler im Schnitt nach 33% Tagesverlaufs (hier: nach der 6. Betriebsstunde) eintritt und der zweite Fehler nach 66% Tagesverlauf (hier: nach der 12. Betriebsstunde) eintritt. Somit ergibt sich im eine mittlere Dauer von 18-12=6h bis zur Wiederherstellung der Systemfunktion.

Beispiel 4 – Anwendung der FTA-Methodik als Ersatz für Markow-Prozesse

Nicht immer lassen sich Zustände sinnvoll nach „intakt" und „defekt" unterteilen. Dies kann zum Beispiel der Fall sein, wenn die zu untersuchende Funktion mit einem Leistungsvermögen verknüpft ist. Offen-

sichtlich wird das am Beispiel einer Fahrzeugflotte: Diese werden im Allgemeinen ausreichend groß dimensioniert, sodass voraussichtlich anfallende präventive und korrektive Instandhaltungsmaßnahmen an den Fahrzeugen durchgeführt werden können, ohne die geforderte Systemleistung zu unterschreiten. Solange instand zu haltende Fahrzeuge durch Reservefahrzeuge ersetzt werden können, wird die geforderte Leistung hundertprozentig erfüllt. Dieses Szenario allein könnte noch mit der FTA-Methodik, genauer mit dem koon-Gatter, gerechnet werden. Nun verhält es sich aber so, dass mit jedem zusätzlichen Verlust an Fahrzeugen die Leistungsfähigkeit des Systems sinkt, man aber zu keinem Zeitpunkt von einem defekten System sprechen sollte. Der Zustand sollte vielmehr als Grad der Leistungsfähigkeit, sprich als das Verhältnis „Anzahl einsatzbereite Fahrzeuge" zu „Anzahl geforderte Fahrzeuge" interpretiert werden. (Der Zustand kann also Werte zwischen 0 und 1 annehmen, und nicht nur 1 und 0, wie es für die FTA-Methodik vorausgesetzt wird. Der Fall, dass mehr Fahrzeuge einsatzbereit als gefordert sind, wird ebenso mit 1=100% Leistungsfähigkeit gewertet).

Anzahl intakte Fahrzeuge	„Naiver" Systemzustand gemäß koon-Gatter (k=2, n=8) $= Z_{naiv,i}$	Tatsächlicher Systemzustand bzw. Leistungsfähigkeit $= Z_{Leistung,i}$
8	1	1
7	1	1
6	0	6/7
…	…	…
2	0	2/7
1	0	1/7
0	0	0/7

Tabelle 5: Vergleich Konzepte zur Zustandsbewertung

Die resultierende Frage die sich stellt, ist, wieviel Zeit in den jeweiligen Zuständen verbracht werden wird. Genauer gesagt sind die prozentualen Zeitanteile p_i in den Zuständen Z_i von Interesse. Sobald die p_i berechnet sind, kann die zu erwartende Verfügbarkeit wie folgt geschätzt werden:

$$A = \sum_{i=0}^{8} Z_i * p_i$$

Die p_i lassen sich mit Hilfe von Markow-Prozessen berechnen. Als Eingangsdaten werden dafür lediglich die Ausfall- und Instandsetzungsraten für die einzelnen Fahrzeuge benötigt.

Um obiges Beispiel weiterzuführen, wollen wir hierbei einmal annehmen, dass ein einzelnes Fahrzeug die Ausfallrate $\lambda = 1/1000\,[1/\mathrm{h}]$ und die Instandsetzungsrate $\mu = 1/24\,[1/\mathrm{h}]$ aufweist. Reserveelemente sind in heißer Redundanz, gleichzeitig anliegende Defekte können simultan instandgesetzt werden.

Angenommen, unsere Flotte würde aus 8 Fahrzeugen bestehen, wobei 7 Fahrzeuge für den nominalen Betrieb benötigt würden und 1 Fahrzeug das Ersatzfahrzeug darstellt, dann ließe sich obiger Ansatz wie folgt zusammenfassen:

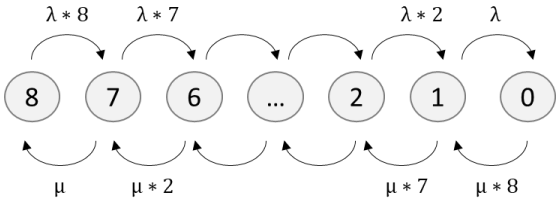


Abbildung 5: Markowprozess "Anzahl intakte Fahrzeuge"

Mithilfe des Kalküls der Markow-Prozesse berechnen sich die p_i wie folgt:

Anzahl intakte Fahrzeuge	p_i
8	0,827
7	0,159
6	0,013
…	…
2	4,43E-09
1	3,04E-11
0	9,11E-14

Tabelle 6: Prozentuale Aufenthaltenszeiten in den versch. Zuständen

Damit berechnet sich A wie folgt:

$$A = \sum_{i=0}^{8} Z_{Leistung,i} * p_i$$
$$= 1 * 0{,}827 + 1 * 0{,}159 + \frac{6}{7} * 0{,}013 + \cdots + \frac{1}{7} * 3{,}04E^{-11} = 99{,}790\%$$

Die Anwendung der Formel für das koon-Gatter hingegen würde eine wesentlich geringere Verfügbarkeit von 98,6% berechnen. Was darin begründet liegt, dass die reduzierten Leistungen mit 6 oder weniger intakten Fahrzeugen nicht honoriert werden. Man beachte, dass gilt:

$$98{,}6\% = \sum_{i=0}^{8} Z_{naiv,i} * p_i$$

Die Rechnung mittels Markow-Prozessen führt also zu gleichen Ergebnissen wie das koon-Kalkül, wenn anstatt der $Z_{leistung}$ weiterhin die Z_{naiv} genutzt werden. Das Markowkalkül kann in diesem Sinne als konsistente Erweiterung des FTA-Ansatzes betrachtet werden.

Beispiel 5 – Keine Untersuchung auf Unabhängigkeit der Ereignisse

- Common-Cause-Fehler (CCF) beschreiben das gleichzeitige Auftreten mehrerer Elementarereignisse aufgrund technischer oder externer Ursachen. Mit Redundanzsystemen angestrebte Zuverlässigkeitssteigerungen werden durch CCF teilweise stark abgeschwächt. Eine Nichtberücksichtigung der CCF zeichnet daher ein zu optimistisches Bild ebendieser Redundanzen. Als Beispiele für Ursachen können Hitze, Feuchtigkeit, Vibration, Strahlung (externe Ursachen) sowie Konstruktions- und Produktionsfehler, Softwarefehler (technischer Ursachen) genannt werden. Allgemein lässt sich festhalten, dass mehr Nähe der Komponenten zueinander (im räumlichen sowie im technischen Sinne) einen größeren CCF-Einfluss bedeutet.

Im Kontext der Fehlerbaumanalysen kann der Einfluss als Beta-Faktor quantifiziert werden. Dieser beschreibt den prozentualen Anteil der Komponentenausfälle mit gemeinsamer Ursache. Die Formel für die Systemausfallrate (1) lässt sich somit wie folgt erweitern:

Formel	Verwendung
$\lambda_{Gatter} = \beta * \lambda_c + \frac{n! * (\lambda_c)^k}{(n-k)! * (\mu_c)^{k-1}}$ (3)	Instandsetzung defekter Komponenten mit Instandsetzungsrate μ_c und beta-Faktor 0<β<1

Tabelle 7: Redundanzformel mit CCF-Einfluss

Aufgrund der Komplexität des Schätzverfahrens und um subjektiven, nicht reproduzierbaren Rechnungen vorzubeugen, wurde in der DIN EN 61508 [2] ein standardisiertes Verfahren zur Schätzung von Beta-Faktoren etabliert.

Die nachfolgende Tabelle illustriert den Einfluss des Beta-Faktors auf das Gesamt-Lambda am Beispiel eines 2oo2-Gatters mit $\lambda_c = 0{,}0001\ [1/\mathrm{h}]$ und $\mu_c = 0{,}1\ [1/\mathrm{h}]$:

β	0	0,01	0,02	0,06	0,1
λ_{2oo2}	2,00 E-07	1,20 E-06	2,20 E-06	6,20 E-06	1,02 E-05

Tabelle 8: : Einfluss des CCF auf ein Parallelsystem

- Bei der „kalten" Redundanz erfahren die Reserveelemente keine Belastung und haben somit eine Ausfallrate = 0. Die relative Vorteilhaftigkeit im Vergleich zur „heißen" Redundanz hängt insbesondere von der Zuverlässigkeit des Umschalt-

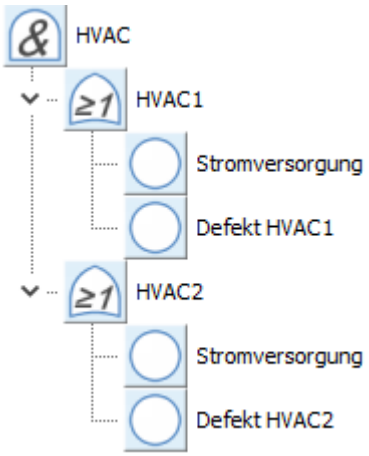
mechanismus (Reserve->aktiv) ab. Eine Modellierung mittels Markow-Prozessen bietet sich in diesem Fall an. Für den Fall mit einem (1) Reserveelement kann auch die folgende Formel verwendet werden:

$$\lambda_{Gatter} \approx \frac{n * (n * \lambda_c + (1 - P) * \mu_c) * \lambda_c}{\mu_c + n * (P + 1) * \lambda_c}$$

Hierbei stellt P die Erfolgswahrscheinlichkeit des Umschaltvorganges dar und n die Anzahl der aktiven Komponenten.

- Bei der leistungsgeteilten Reserve teilen sich sämtliche intakte Komponenten die zu erbringende Leistung zu gleichen Teilen. Bei Wegfall einer oder mehrerer Komponenten erhöhen sich die zu erbringenden Leistungen der verbleibenden Komponenten und somit deren Ausfallraten über den entsprechenden Zeitraum. Eine Modellierung mittels Markow-Prozessen bietet sich in diesem Fall an.

- Mehrfachfehler: Bei der Erstellung von Fehlerbäumen kann es durchaus erforderlich werden, ein und dasselbe Elementarereignis an mehreren Teilbäumen anzubringen. Dadurch entsteht eine Abhängigkeit zwischen den Teilbäumen. Die Formel (1) ist in diesem Fall nicht mehr anwendbar. Dieser Fall tritt insbesondere dann ein, wenn im Rahmen der CCF-Analyse konkrete Ursachen benannt und quantifiziert werden können. Als Beispiel könnte hierbei eine gemeinsame Stromversorgung dienen.



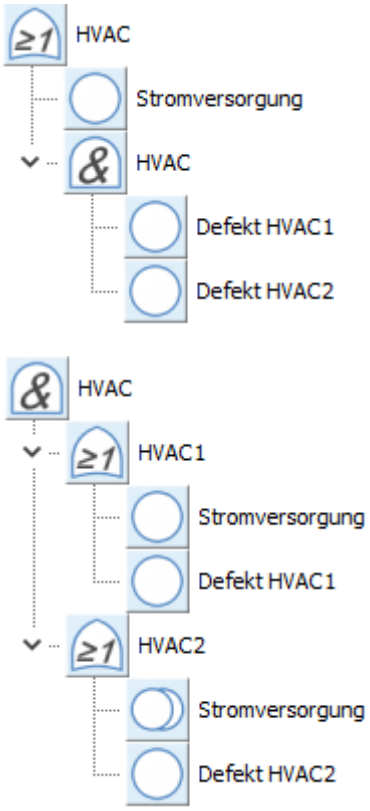


Abbildung 6: Verschiedene Konzepte zur Berücksichtigung des CCF in einem Fehlerbaum

Eine Nichtberücksichtigung der Mehrfachfehler (Baum Oben) resultiert insbesondere in zu optimistischen Berechnungsergebnissen wie das Beispiel CCF weiter oben illustriert. In manchen Fällen ist eine grafische Umordung des Baumes möglich, um so die Abhängigkeiten aufzulösen, so auch in unserem Beispiel (Baum Mitte). Diese Möglichkeit ist aber nicht immer gegeben. Einige FTA-Softwaretools sind in der Lage diese Form der Abhängigkeit direkt in den Ereignissen abzubilden und mittels Mehrfachereignissen zu modellieren (Baum unten).

Beispiel 6 – Missinterpretation des Begriffs „k oo n- System"

Fehlerbäume sind negierte Zuverlässigkeitsblockdiagramme. Erstere liefern die Fehlersicht auf ein System, letztere die dementsprechend zugehörige Funktionssicht. FTA und RBD benutzen die Begriffe koon–Gatter bzw. koon-System. Hierbei gilt zu beachten, dass die Begriffe je nach Kontext unterschiedlich zu deuten sind: Wird der Ausdruck koon im Kontext eines

RBD genutzt, so sollten Sie in der Regel davon ausgehen dürfen, dass gemeint ist „mindestens k der n Elemente müssen intakt sein, damit das System intakt ist". Wird der Begriff jedoch im Fehlerbaumkontext verwendet, sollten Sie in der Regel davon ausgehen dürfen, dass damit gemeint ist „mindestens k der n Elemente müssen defekt sein, damit das System defekt ist". Wenn Ihnen der Begriff begegnet, so sollten Sie sich zuerst fragen, welche Bedeutung er im konkreten Fall hat.

Diese Art von Verwechslungen gibt es insbesondere bei der Interpretation der Begriffe „UND-Gatter" und „ODER-Gatter" („UND" und „ODER" sind Spezialfälle des koon-Gatters).

Wir empfehlen im Rahmen von FTA stets von Gattern zu sprechen anstelle von Systemen, wenn es um die logische Verknüpfung von Unterereignissen geht. Damit würde eine Verwechslung vermieden. Wenn von einem koon-Gatter gesprochen wird, ist die fehlerorientierte Sicht anzuwenden.

Beispiel 7 – Keine Berücksichtigung der Instandsetzung einzelner Komponenten in einem Parallelsystem bei der Ermittlung der System-Ausfallrate

Ein relativ trivialer, aber durchaus folgenreicher Fehler ist die Nichtberücksichtigung der Instandsetzung in instandsetzbaren Systemen, sprich die Anwendung der Formel (2) anstatt der Formel (1), wie die nachfolgende Tabelle für verschiedene koon-Gatter mit λ_c =0,0001 [1/h] und μ_c =0,1 [1/h] illustriert:

koon-Gatter	λ_{koon} [1/h] mit Instandsetzung (Formel 1)	λ_{koon} [1/h] ohne Instandsetzung (Formel 2)
1oo2	2,00E-04	2,00E-04
2oo2	2,00E-07	6,67E-05
1oo3	3,00E-04	3,00E-04
2oo3	6,00E-07	1,20E-04
3oo3	6,00E-10	5,45E-05

Tabelle 9: Vergleichsrechnung zwischen den Formel mit/ohne Instandhaltung

Beispiel 8 - Falsche Quantifizierungen

Folgende Missstände sind uns bisher begegnet:

- Übernahme von default-Werten in einer bereits befüllten Tabelle ohne diese kritisch zu hinterfragen bzw. entsprechend anzupassen (Beispiel: Instandsetzungsdauer = 1 Stunde).
- vorschnelles Festlegen auf einen Wert, ohne eine tiefergehende Überlegung anzustellen (z.B.: müssen ein oder zwei Anreisen zum Fehlerort eingeplant werden? Muss neben der eigentlichen Reparaturzeit noch eine Testzeit eingeplant werden?).

Gerade in einem Redundanzsystem hat die Instandsetzungszeit signifikanten Einfluss auf die Zuverlässigkeit bzw. Verfügbarkeit; die Ausfallrate eines 2oo2-Systems beispielsweise weist einen linearen Zusammenhang mit der Instandsetzungsdauer der einzelnen Komponente auf (siehe Formel 1).

Beispiel 9 – Falsche Interpretation von Kenngrößen

Folgende Missstände sind uns bisher begegnet:

- Anwender von Programmen versuchen diese mechanisch auszufüllen, ohne die genauen Bedeutungen der einzelnen Felder zu kennen. Beispiel: Eingabefeld Q, w (Eingabefelder für Unverfügbarkeit und Fehlerhäufigkeit im FTA-Programm „Fault Tree plus")
- Angabe von Fehlerhäufigkeiten ohne Benennung der zugrundeliegenden Einheiten.

Beispiel 10 – Unwissenheit bei der Einbindung von Inspektionszeiten bei verdeckten Fehlern

Verdeckte Fehler haben definitionsgemäß keinen unmittelbaren Einfluss auf den normalen Betriebsauflauf, können aber, in Kombination mit weiteren Fehlern oder äußeren Umständen, zu ungewünschten Er-

eignissen führen. Beispielhaft seien defekte Reserveelemente, Elemente einer Brandmeldeanlage oder die Gegensprechanlage eines Fahrstuhles genannt. Intuitiv ist klar, dass mit der Streckung der Inspektionsintervalle resp. der Abstände zwischen den Fehlerfindungsmaßnahmen auch die Wahrscheinlichkeit eines Systemversagens im Anforderungsfall steigt. Die Formel (1) ist hier nur noch bedingt anwendbar;

die Instandsetzungsrate für den verdeckten Fehler könnte zwar mit dem halben Inspektionsintervall (plus der eigentlichen Instandsetzungszeit) approximiert werden, um so die tatsächliche Unverfügbarkeitszeit zu berücksichtigen. Es müsste dann aber irgendwie festgehalten werden, wie dieser Wert gemeint ist, um so späteren Missverständnissen vorzubeugen. An dieser Stelle sei daher darauf verwiesen, dass einige Fehlerbaum-Softwarelösungen (so auch das **Fehlerbaum-Softwaretool** der IZP Dresden) die separate Angabe von Inspektionsintervallen zulassen und so eine transparente, nachvollziehbare und vorallem korrekte Rechnung erlauben.

Gerne beantworten wir Ihre Fragen und freuen uns auf Ihre Hinweise.

Für weitere Informationen wenden Sie sich bitte an:

Rückfragen

Paul Granzin
(Systemanalyst)
+49 173 6094502
p.granzin@izp.de

Dr. Harald Jung
(Geschäftsführender Gesellschafter)
+49 351 80403-23
h.jung@izp.de

Quellenangaben

[1] DIN EN 61025: Fehlzustandsbaumanalyse

[2] DIN EN 61508 -Teil 6 – Anhang D: Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer elektronischer Systeme – Teil 6: Anwendungsrichtlinie für IEC 61508-2 und IEC 61508-3 Anhang D: Eine Methode zur Quantifizierung der Auswirkungen von hardwarebedingten Ausfällen infolge gemeinsamer Ursache in E/E/PE Systemen

[3] ROME LABORATORY RELIABILITY ENGINEER'S TOOLKIT, April 1993

Weiterführende Information

Fehlerbaum-Softwaretool der IZP Dresden http://www.izp.de/fta oder unter **info@izp.de**.

Impressum

IZP Dresden mbH
Am Waldschlösschen 4
D-01099 Dresden

Geschäftsführender Gesellschafter Dr. Harald Jung
Prokurist und Gesellschafter Klaus Kühnert

Amtsgericht Dresden HRB 32118