Zur notwendigen Theorie

2

In diesem Kapitel werden die theoretischen Grundlagen der Fehlerbaumanalyse erläutert. Dazu stellen wir zunächst abstrakt die Modellierung hinsichtlich Notation und Symbolik vor und widmen uns ausführlich der Bedeutung im Hinblick auf die Analyse und Auswertung. Da die Methode sowohl qualitativ als auch quantitativ eingesetzt werden kann, werden neben einer generellen Einführung in das logische Modell eines Fehlerbaums (s. Abschn. 2.1) die Quantifizierungsmöglichkeiten erläutert (s. Abschn. 2.7). Dabei sollen neben der allgemein mathematischen, wahrscheinlichkeitstheoretischen Basis auch weiterführende Aspekte der zeitlichen Auswertung von Fehlzuständen dargelegt werden.

Leser, die mit der Theorie von Fehlerbäumen bereits vertraut sind, können dieses Kapitel beim ersten Lesen des Buches getrost überspringen und bei Bedarf später zurückkehren, um Details nachzuschlagen.

2.1 Aufbau und Notation des Fehlerbaums

Fehlerbäume bilden graphisch den Zusammenhang zwischen *Ereignissen* und deren Verknüpfung mittels logischen *Gattern* ab. Wie der Name vermuten lässt, entspricht die Struktur im mathematischen/informatischen Sinne der eines *Baumes*, d. h. eines verzweigten, gerichteten (zyklenfreien) Graphens aus Knoten und Kanten.[1] Dieser folgt ein paar wenigen syntaktischen Regeln, die den Kern der Modellierung ausmachen. Die grundsätzliche Struktur und der Aufbau ist an einem Beispiel in Abb. 2.1 veranschaulicht, dem wir uns schrittweise nähern wollen.

Ein Ereignis (engl. *event*) beschreibt das Auftreten eines Systemzustands, dass in der Regel einen Fehler oder Fehlzustand charakterisiert.[2] Prinzipiell können aber auch Er-

[1] Die Wurzel wird in der Regel oben notiert, so dass der Baum nach unten „wächst".
[2] Deshalb spricht die DIN EN 61025 auch von Fehl*zustands*baumanalyse.

© Springer-Verlag Berlin Heidelberg 2015

F. Edler, M. Soden, R. Hankammer, *Fehlerbaumanalyse in Theorie und Praxis*,
DOI 10.1007/978-3-662-48166-0_2

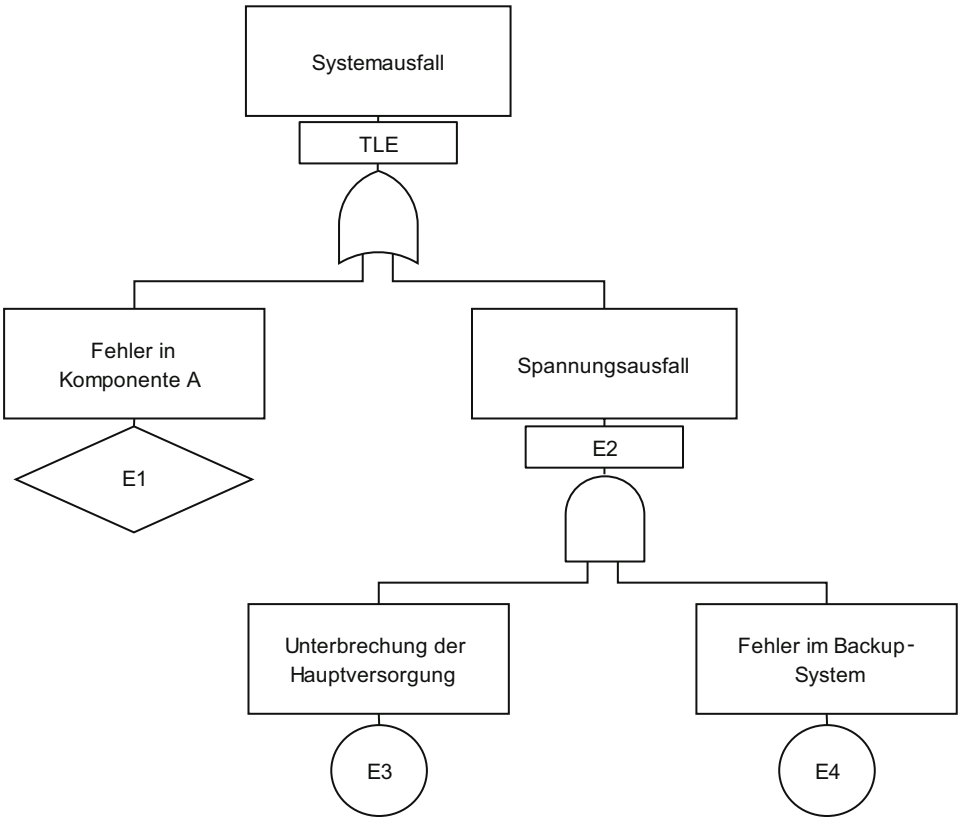


Abb. 2.1 Fehlerbaumbeispiel: Hauptereignis, UND/ODER-Gatter, Primärereignisse

eignisse, die keine Fehler sind, ausgedrückt werden – also z. B. intendierte Zustände, Einflussgrößen oder Bedingungen – falls diese in der Analyse als relevant erachtet werden. Notiert werden alle Ereignisse als Rechtecke, in denen ein möglichst kurzer und präziser Text das Ereignis beschreibt. Aus praktischen Gesichtspunkten erhält jedes Ereignis zusätzlich einen eindeutigen Bezeichner (*identifier*, kurz: *ID*), der die Arbeit an einer FTA erleichtert.

Ereignisse können durch eine Auswahl an verschiedenen Gattern (engl. *gates*) weiter detailliert werden. Dieser Verfeinerungsschritt entspricht einer Präzisierung im Sinne einer Zerlegung in einzelne „Teilereignisse“, die logisch verknüpft werden.[3] Da diese wie Eingänge einer Schaltung mit dem Gatter verbunden sind, nennt man diese auch Ein-

[3] Für weitergehende Betrachtungen zur Beziehung zwischen Ereignissen auf unterschiedlichen Ebenen siehe auch Kap. 3 und 4.

gangsereignisse (engl. *input events*) und das ursprüngliche Ereignis Ausgangsereignis (engl. *output event*). Die Darstellung für Gatter ist ebenfalls an Elektronik-Schaltpläne angelehnt, so dass sich die Symbole überwiegend gleichen. Zur Identifikation von Gattern werden üblicherweise auch hier Bezeichner verwendet.

Das Beispiel in Abb. 2.1 zeigt fünf Ereignisse und zwei Gatter. Die Wurzel des Fehlerbaumes stellt das sog. Hauptereignis (auch: Top-Ereignis) der Analyse dar (engl. *top level event*, hier: „Systemausfall"). Darunter ist ein ODER-Gatter mit zwei Eingangsereignissen E1 und E2 notiert. Dadurch wird ausgedrückt, dass ein Systemausfall entweder durch „Fehler in Komponente A" [E1] oder durch einen „Spannungsausfall" [E2] auftreten kann. Während der Baum auf der linken Seite bei E1 endet, wurde Ereignis E2 mittels eines UND-Gatters weiter untergliedert, so dass das Ereignis „Spannungsausfall" präzisiert wurde zu „Unterbrechung der Hauptversorgung" bei gleichzeitigem Eintreten eines „Fehlers im Backup-System". Im Prinzip können die Gatter mehr als zwei Eingangsereignisse besitzen, so dass eine Verknüpfung von n Ereignissen beschrieben werden kann.

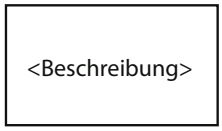
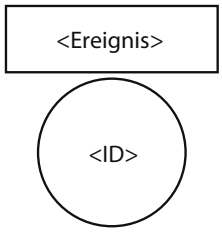
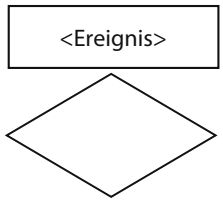
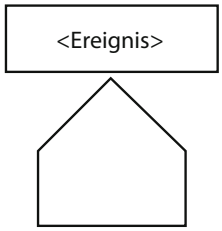
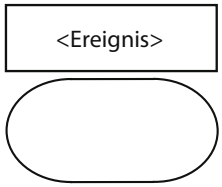
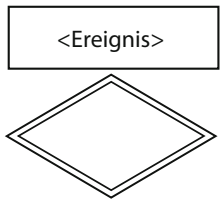
Gatter und Zwischenereignisse werden in der Praxis häufig synonym verwendet. So beschreibt ein Zwischenereignis letztlich eine Ereigniskombination, weshalb man häufig auch von der „Gatterbeschreibung" an Stelle der Beschreibung eines Zwischenereignisses spricht (s. Abschn. 2.3).

Die drei Ereignisse E1, E3 und E4 nennt man Primärereignisse (engl. *primary events*), da sie nicht weiter untergliedert sind. Ereignis E2 bezeichnet man als Zwischenereignis (engl. *intermediate event*). Den Abschluss durch ein Primärereignis markiert zusätzlich ein Symbol unterhalb der Textbox, um die Art des Ereignisses mitzuteilen. Elementare Ereignisse, die sich nicht weiter untergliedern lassen, werden Basisereignisse oder Grundereignisse genannt (engl. *basic events*) und mit einem Kreis gekennzeichnet. Nicht weiter untersuchte Ereignisse (engl. *undeveloped events*) werden mit einer Raute versehen. Durch diese unterschiedlichen Kennzeichnungen kann ein Analyst leicht erkennen, an welcher Stelle der Fehlerbaum vollständig entwickelt ist und wo evtl. noch weiterer Bedarf einer Detaillierung besteht. Als Graph betrachtet sind in einem Fehlerbaum somit alle „Blätter" Primärereignisse, alle anderen neben dem Hauptereignis Zwischenereignisse. Über das Beispiel hinaus existieren noch weitere Typen von Primärereignissen, die Tab. 2.1 zusammenfasst.

Es sei an dieser Stelle bereits angemerkt, dass die Zuordnung, wann ein Ereignis als Basisereignis gekennzeichnet wird und wann nicht, in der Praxis unterschiedlich angewandt wird. Je nach Analysegrad und -zweck, Analyst, aber auch bedingt durch Einschränkungen der FTA-Werkzeuge kann die Typisierung der Primärereignisse variieren. Die gute Nachricht allerdings ist, dass dies für die Auswertung der möglichen Ereigniskombinatorik keine Rolle spielt und alle Primärereignisse gleich behandelt werden (vgl. Abschn. 2.4).

Folgen wir dem Beispiel weiter. Möchte man das Ereignis „Fehler in Komponente A" [E1] weiterentwickeln, so können wir entweder den Fehlerbaum direkt ergänzen (wie bei

Tab. 2.1 FTA-Symbolik für Ereignisse

Symbol	Name	Bedeutung
	Ereignisbeschreibung	Beschreibung des Ereignisses, das einen Fehler/ Fehlzustand, System-/Komponentenzustand, eine Bedingung oder Aktion bezeichnet.
	Basisereignis	Elementares Ereignis, das sich nicht weiter entwickeln lässt oder dessen weitere Entwicklung als nicht relevant erachtet wurde.
	Nicht untersuchtes Ereignis	Ereignis, dessen Ursache nicht weiter analysiert wurde oder werden konnte (engl. *undeveloped event*).
	Hausereignis	Ereignis, das einen erwartbaren Zustand beschreibt (also in der Regel keinen Fehler). Dient oftmals als technisches Hilfsmittel, um im Fehlerbaum durch Ein-/Ausschalten gewisse Zustände zu setzen.
	Bedingungsereignis	Ereignis, das eine Bedingung ausdrückt. Meist als Eingang in ein Bedingungsgatter oder PUND-Gatter benutzt (s. a. Tab. 2.2), kann auch zur Kennzeichnung möglicher systematischer Fehler (z. B. von Software) benutzt werden.
	Schlafender Fehler	Fehler mit besonderen Merkmalen, der meist einen schlafenden Fehler beschreibt (engl. *dormant failure*, auch: latenter Fehler).

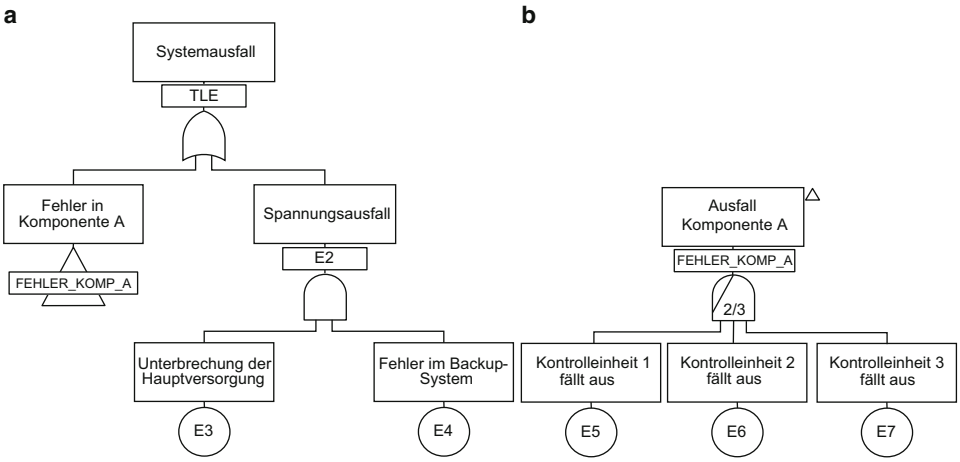


Abb. 2.2 Erweitertes Beispiel „Systemausfall"; **a** Hauptereignis mit Dachstruktur des Fehlerbaums, **b** Teilbaum zum Ausfall der Komponente A

Ereignis E2), oder mittels eines sog. Transfergatters eine Referenz auf ein anderes Ereignis setzen. Durch Transfergatter lässt sich ein Fehlerbaum in übersichtliche Teilbäume strukturieren. Abbildung 2.2a zeigt die entsprechende Dreiecksnotation eines Transfergatters unter E1.[4]

Transfergatter stellen ein rein syntaktisches Hilfsmittel zur Strukturierung komplexer Fehlerbäume dar und haben keine eigene Logik. Das heißt ein Transfergatter hat stets nur ein Eingangsereignis, das referenzierte Ereignis, auch Ziel des Transfergatters genannt (hier mit Bezeichner FEHLER_KOMP_A in Abb. 2.2b). Man könnte also alle Transfergatter eines Fehlerbaumes eliminieren und würde die festgehaltene Ereigniskombinatorik nicht verändern. Abschnitt 4.5.1 gibt Tipps zur sinnvollen Aufteilung und Gruppierung.

Abbildung 2.2b zeigt das neben UND- und ODER-Gattern am dritthäufigsten verwendete Mehrheitsentscheidungsgatter (engl. *voting gate*), kurz: Entscheidungsgatter. Damit kann eine *n-aus-m*-Kombinatorik beschrieben werden, bei der das Ausgangsereignis eintritt, wenn n oder mehr Eingangsereignisse auftreten. Die Zahl m ergibt sich automatisch aus der Anzahl der Eingangsereignisse. Im Beispiel müssen also mindestens zwei der drei angedeuteten Kontrolleinheiten ausfallen, um ein Versagen der Komponente A zu bewirken. Damit ergeben sich vier verschiedene Kombinationen unter der die Komponente A ausfällt: $\{E5, E6\}$, $\{E5, E7\}$, $\{E6, E7\}$ und $\{E5, E6, E7\}$.

[4] Mathematisch betrachtet führt das Transfergatter eigentlich dazu, dass sich ein Fehler*graph* bildet. Dieser lässt sich allerdings durch logisches Duplizieren eines referenzierten Zweiges mit mehrfach verknüpften Ereignissen auflösen, so dass man weiterhin von einer Baumstruktur sprechen kann (s. a. Abschn. 3.4).

Anmerkung zum Entscheidungsgatter: Die Form des Entscheidungsgatters deutet auf einen interessanten Sachverhalt hin: Würde man n gleich m setzen, erhielte man ein einfaches UND-Gatter, da „m-aus-m“ (also *alle*) Ereignisse auftreten müssen. Wenn man allerdings umgekehrt $n = 1$ setzt, erhielte man ein ODER-Gatter („1-aus-m“-Kombination). Deshalb wird das Mehrheitsentscheidungsgatters auch generell als abgewandelte Form eines dieser Gatter notiert und könnte die bisherigen Gatter komplett ersetzen! Wir werden diese Beziehungen zwischen den Gattern in Abschn. 2.4 formaler fassen und sehen, dass sich andersherum auch der Mehrheitsentscheider durch eine Kombination von UND/ODER-Gattern ersetzen lässt.

In einer Analyse wird man häufig nicht umhin kommen, einige Ereignisse an mehreren Stellen im Fehlerbaum einzutragen. Zum Beispiel könnte ein Fehler in der Spannungsversorgung gleichzeitig Fehler in verschiedenen Komponenten verursachen. Neben dem Transfergatter sieht man häufig Ereignisse mit gleichem Namen bzw. gleicher ID in unterschiedlichen Teilbäumen, die denselben Sachverhalt ausdrücken. Trägt man ein Ereignis mehrfach ein, so spricht man von einem *mehrfach verknüpften* oder mehrfach auftretenden Ereignis (engl. *multiple occuring event*, MOE), was faktisch eine Abhängigkeit zwischen verschiedenen Teilbäumen erzeugt. Strenggenommen kann man bei diesen Ereignissen zwischen einem „wiederkehrenden Fehler im System“ (quasi als Folge des Designs) und einer „gemeinsamen Fehlerursache“ unterscheiden. Letztere nennt man in der Praxis *Common-Cause*-Fehler (engl. *common cause failures*, CCF).[5] Diese werden häufig durch Stress an Bauteilen verursacht, wie zum Beispiel durch Vibrationen, Temperaturschwankungen, Verunreinigungen, Schwächen bei Wartungsarbeiten etc. und in späteren Analyseschritten eingearbeitet. In puncto Quantifizierung werden wir diese Unterscheidung aufgreifen, da es hier alternative Modellierungsmöglichkeiten gibt (s. Abschn. 2.8).

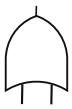
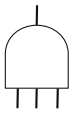
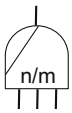
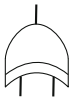
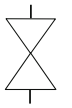
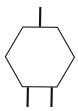
Die bisher vorgestellten Gatter stellen die in der Praxis am Häufigsten verwendeten dar. Darüber hinaus existieren noch weitere Typen von „Spezialgattern“, die teils auf andere Gatter zurückzuführen sind, teils die Basistheorie der FTA erweitern. Der Vollständigkeit halber besprechen wir diese im Folgenden und deuten eventuelle Schwierigkeiten bei der Verwendung an. Tabelle 2.2 bietet vorab eine Übersicht.

2.2 Unabhängigkeit der Ereignisse

Für die Analyse – und insbesondere die spätere Quantifizierung – ist die *Unabhängigkeit* der Primärereignisse von entscheidender Bedeutung, da sonst die Auswertung des Fehlerbaums fehlerhaft sein kann. Ereignisse werden als unabhängig betrachtet, wenn ihre *Auftretenswahrscheinlichkeit* von keinem anderen Ereignis beeinflusst wird. Diesem

[5] In der sprachlichen Praxis verschwimmt diese Unterscheidung oftmals und man spricht gerne von *common cause*, sobald ein Ereignis gemeinsame Ursache für andere Ereignisse ist.

Tab. 2.2 FTA-Symbolik für Gatter

Symbol	Name	Bedeutung
	ODER-Gatter	Logische VerODERung der Eingangsereignisse
	UND-Gatter	Logische VerUNDung der Eingangsereignisse
	Mehrheitsentscheidungsgatter	Beschreibt eine n-aus-m-mögliche Kombination (engl. *voting gate*)
	XOR-Gatter	Exklusives ODER (d. h. entweder A oder B, aber nicht beide Ereignisse)
	Nicht-Gatter	Verneinung (Inversion) des Eingangsereignisses
	Bedingungsgatter	Gatter mit zusätzlicher Bedingung, die logisch verUNDet wird
	PUND-Gatter	Prioritäts-UND-Gatter, bei dem die Reihenfolge der Eingangsereignisse entscheidend ist
	Transfergatter	Stellt eine Referenz („Link“) zu einem Teilbaum her

Aspekt der stochastischen Unabhängigkeit und einer entsprechenden Formalisierung widmen wir uns im Abschn. 2.4.

Die FTA-Notation verhindert per se keine inhaltlichen Überlappungen der Ereignisse, bietet aber einige Konstrukte mit deren Hilfe man Abhängigkeiten festhalten kann:

1. Mehrfach verknüpfte Ereignisse und Common-Cause-Fehler (vgl. Abschn. 4.2.5),
2. Bedingungen und bedingte Ereignisse,
3. Gegenseitiger Ausschluss von Ereignissen und Inversion eines Ereignisses,
4. Sequentielle Abhängigkeiten von Ereignissen und Fehlern.

An dieser Stelle möchten wir bereits eine allgemeine Warnung aussprechen, dass die Ausdrucksmöglichkeit im Fehlerbaum für komplexere Fälle von Abhängigkeiten sehr eingeschränkt ist. Ein Fehlerbaum stellt an sich ein statisches Modell dar, in das sich nur direkte Abhängigkeiten zwischen Ereignissen ausdrücken lassen. Sobald komplexe dynamische Aspekte eine Rolle spielen, wie zum Beispiel die Schrittweise Degradierung bei redundanten Systemen durch Abschalten einzelner Teilsysteme, stößt das Modell schnell an seine Grenzen. Einige Alternativen und Erweiterungen der FTA zeigt Abschn. 2.9.

Die einfachste Form von Abhängigkeiten zwischen Teilbäumen einer FTA haben wir mit mehrfach verknüpften Ereignissen bereits im vorigen Abschnitt kennengelernt. Ein weiterer häufig anzutreffender Fall sind Fehler, die nur unter bestimmten Voraussetzungen eintreten. Sollte ein Ereignis nur unter einer gegebenen Bedingung eintreten, so spricht man von einem *bedingten Ereignis* (engl. *conditional event*). Beispiele wären chemische Reaktionen, die nur ab einer Mindesttemperatur ablaufen oder die Präsenz eines geeigneten Katalysators voraussetzen. Ein häufiger Fall in elektronischen Systemen ist die Überwachung einer Komponente, bei der ein kritischer Fehler nur dann eintritt, wenn die überwachende Einheit den Fehler auf Grund einer Diagnoselücke nicht feststellen kann (s. a. Abschn. 10.2). Solche Ereignis-Bedingungs-Kombinationen können mit Hilfe eines *Bedingungsgatters* (engl. *inhibit gate*) beschrieben werden. Abbildung 2.3 zeigt die Wabenform des Gatters (linke Seite), in das neben dem Ereignis A die Bedingung C (engl. *conditioning event*) eingeht. Zu interpretieren wäre dies als: Ereignis A ist notwendig, alleine aber nicht hinreichend. Erst wenn Bedingung C erfüllt ist, tritt Ereignis B ein. Anders formuliert könnte man auch sagen, dass, nur wenn Ereignis A *und* Bedingung C eintreten, Ereignis B folgt. Aus diesem Grund lässt sich das Bedingungsgatter rein logisch durch ein einfaches UND-Gatter ersetzen (vgl. Abb. 2.3), wobei die Bedingung durch ein sog. Bedingungsereignis kenntlich gemacht wird.

Eine weitere Möglichkeit zwei Ereignisse zu verbinden ist der gegenseitige Ausschluss durch ein *exlusives Oder* (XOR, s. Abb. 2.4). Ein XOR-Gatter, in das zwei Ereignisse eingehen, drückt aus, dass nur ein Ereignis das Ausgangsereignis eintreten lässt – und zwar *in Abwesenheit* des jeweils anderen! Dieser Punkte kann nicht genug betont werden, da in weiten Teilen der Literatur eine verkürzte Darstellung suggeriert, beim XOR-Gatter reiche „genau ein“ Ereignis aus. Tatsächlich führt das Gatter implizit *invertierte* Ereignisse ein: das jeweils andere Ereignis des Gatters muss gerade *nicht* eingetreten sein, damit das Ausgabeereignis eintritt! Abbildung 2.4 zeigt deshalb die äquivalente Beschreibung mittels UND/ODER-Gattern.

Genauer gesagt spricht man beim Vorkommen von „Nicht-Ereignissen“ wie in diesem Fall von sogenannten *nicht kohärenten* Fehlerbäumen. Invertierte Ereignisse können auch durch Nicht-Gatter eingeführt werden, die in der Regel nur einen Eingang haben. Damit lässt sich in Kombination mit UND/ODER noch gezielter eine Verbindung zwischen Auf-

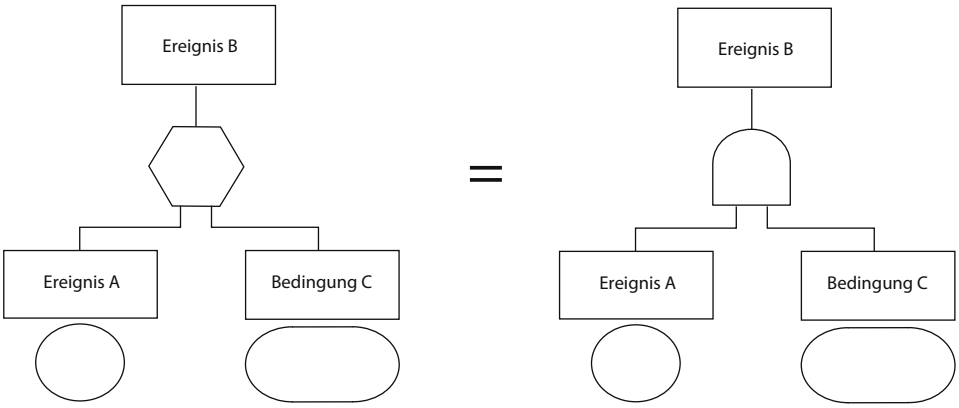


Abb. 2.3 Bedingungsgatter (links) und alternative Beschreibung mittels UND-Gatter (rechts)

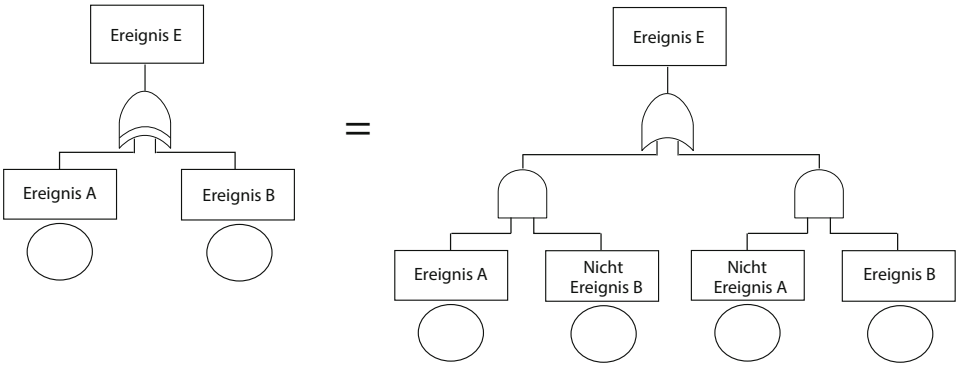


Abb. 2.4 XOR-Gatter (links) und äquivalente Umschreibung mittels ODER/UND-Gattern und invertierten Ereignissen (rechts)

treten und Ausbleiben von Ereignissen modellieren. Diese Verschränkung mag in einer Analyse gewünscht sein, führt aber im Allgemeinen zu schwer verständlichen und interpretierbaren Auswertungen.[6]

Eine ganz andere Kategorie sind Abhängigkeiten, die sich aus der Auftretens*reihenfolge* von Fehlern ergeben (engl. *sequence dependent failures*). Beispiele hierfür sind Standby-Schaltungen mit Komponenten, deren Ausfallen erst zum Systemfehler werden, wenn der Ausfall *vor* der Aktivierung der Standby-Schaltung auftritt. Solche sequentiellen Abhängigkeiten können mittels eines Prioritäts-UND-Gatters (PUND-Gatters) festgehalten werden, bei dem alle Eingangsereignisse in einer spezifizierten Reihenfolge auftreten müssen. Reihenfolgen sind dabei typischerweise am Gatter annotiert bzw. werden aus der

[6] Insbesondere wenn unter dem XOR-Gatter noch komplexe Teilbäume aufgebaut werden, muss deren Logik quasi invertiert werden und manch einer ist überrascht, was dann als Minimalschnitt erscheint (s. Abschn. 2.4).

Eingangsreihenfolge (von links nach rechts) entnommen. Der Vorstellung nach könnte man ersatzweise ein normales UND-Gatter nehmen, bei dem man zu den Eingangsereignissen ein separates Bedingungsereignis einfügt. Je nachdem wie mit Reihenfolgen bei der Auswertung umgegangen wird, fallen Fehlerbäume mit PUND-Gattern allerdings in die Kategorie der *dynamischen Fehlerbäume* (engl. *dynamic fault trees*, DFT). DFTs verbinden die FTA mit weiteren Konzepten von Temporallogik oder Markov-Modellen, um zeitlich-kausale Abhängigkeiten zwischen Ereignissen zu beschreiben. In diesem Buch werden wir diese Inhalte nur anreißen und verweisen auf einschlägige Fachliteratur (s. a. Abschn. 2.9).

2.3 Notationsvielfalt

Die Syntax von Fehlerbäumen ist durch keinen internationalen Standard im Detail festgelegt. Die in Abschn. 2.1 gezeigten Symbole sind die wohl am weitesten verbreiteten, da sowohl die ersten Handbücher zur FTA (wie NUREG-0492 [4] und das der NASA [11]) diese Notation verwenden und eine Vielzahl von Werkzeugen eine entsprechende Implementierung bereitstellen. Spätere Normen wie die IEC 61025 (s. [5]) zur FTA als auch Sicherheitsnormen wie die IEC 61508 (s. [8]) nutzen dementsprechend selbige.

Die Gemeinsamkeiten in diesen de facto Standards sollten (spätestens nach der Lektüre dieses Kapitels) leicht nachvollziehbar sein. Es gibt allerdings einige Abweichungen und Besonderheiten, auf die wir aufmerksam machen möchten:

1. In vielen wissenschaftlichen Veröffentlichungen werden Fehlerbäume skizziert, bei denen Zwischenereignisse weggelassen und direkte Gatter-zu-Gatter Verbindungen gezeichnet werden. Man sollte sich bewusst sein, dass diese Verkürzung lediglich zur besseren Platznutzung in Artikeln dienen, um die logischen Zusammenhänge darzustellen (s. Abschn. 2.4.1). In der Praxis sollte *immer* eine Beschreibung des Zwischenereignisses angegeben werden, da sonst unklar bleibt, zu welchen Fehlern/Zuständen Eingänge in ein Gatter eigentlich führen (s. a. Abschn. 3.2.3).[7]
2. Einige Werkzeuge kombinieren (aus diesem Grund) Zwischenereignisse und logische Gatter zu einer Einheit, so dass man als Nutzer erst gar nicht in die Lage kommt, eine Beschreibung des Zwischenereignisses zu vergessen. Dadurch fallen meist die Bezeichner von Zwischenereignissen weg bzw. werden durch Bezeichner des Gatters ersetzt (s. Abb. 2.5).
3. Anders herum werden in einigen Fällen Ereignis-zu-Ereignis-Kombinationen modelliert, bei denen ein Zwischenereignis direkt durch ein anderes Zwischenereignis verfeinert wird (vgl. Abschn. 4.5.2). Solche Verbindungen eines Ereignisses als direkten Eingang in ein Zwischenereignis bezeichnet man auch als „NULL-Gatter“.

[7] Somit spricht man auch von „Beschreibung eines Gatters“ als Synonym für das Zwischenereignis.

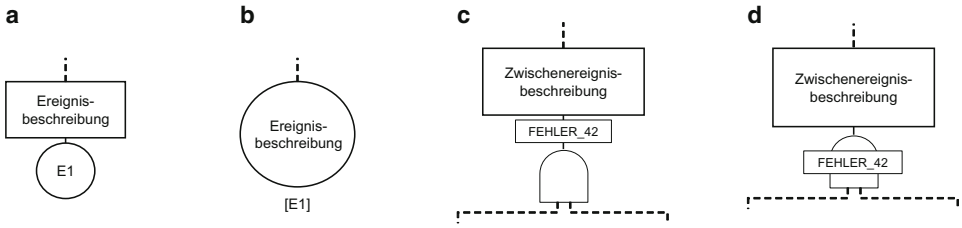


Abb. 2.5 Alternative Darstellungen von Primärerereignissen, Gattern und Zwischenereignissen: **a** Primärereignis mit Beschreibung, **b** Primärereignis mit externem Bezeichner, **c** Zwischenereignis und Gatter getrennt, **d** Zwischenereignis und Gatter kombiniert

4. Einige Werkzeuge zeichnen den Fehlerbaum nicht von oben nach unten (mit dem Hauptereignis als Wurzel an der Spitze), sondern von links nach rechts. Diese Notation ist zum Beispiel im VDA-Band zur FTA zu finden und ist angelehnt an eine Notation zu Fehlernetzen (s. a. Kap. 13).

2.4 Semantik von Fehlerbäumen

Bislang haben wir Fehlerbäume und deren Symbolik recht informell behandelt. In diesem Kapitel wollen wir der Notation eine mathematisch präzise Bedeutung geben und die intuitiv verständlichen Inhalte aus Abschn. 2.1 und 2.2 formal fassen. Dadurch können wir Fehlerbäume nach wohldefinierten Regeln umformen und Lösungen berechnen, unter welchen Kombinationen das Hauptereignis eintritt.

2.4.1 Fehlerbäume als boolesche Formel

Fehlerbäume können mittels der *Booleschen Algebra* bzw. booleschen Formeln präzise beschrieben werden. Boolsche Formeln bestehen aus einer Menge von Variablen und den logischen Operatoren $\wedge$ (UND, Konjunktion), $\vee$ (ODER, Disjunktion) und $\neg$ (NICHT, Negation). Eine Variable ist dabei binär und kann lediglich die Werte *wahr* oder *falsch* annehmen. Bezogen auf einen Fehlerbaum können die Ereignisse auf Variablen und die Gatter auf logische Operatoren abgebildet werden. Seien die Ereignisse eines Fehlerbaums mit $E_1, E_2, E_3, \ldots$ gegeben, dann ergibt sich für die Gatter:

i. ODER-Gatter mit Eingängen E_1, E_2 entspricht: $E_1 \vee E_2$.
ii. UND-Gatter mit Eingängen E_1, E_2 entspricht: $E_1 \wedge E_2$
iii. Entscheidungsgatter mit Eingängen $E_1, \ldots, E_n$ und m als Schwellwert:
$(E_1 \wedge \cdots \wedge E_m) \vee (E_2 \wedge \cdots \wedge E_{m+1}) \vee \cdots \vee (E_{n-m} \wedge \cdots \wedge E_n)$

Tab. 2.3 Wahrheitstabellen der booleschen Operatoren UND ($\wedge$), ODER ($\vee$) und NICHT($\neg$)

$E1$	$E2$	$E1 \wedge E2$
0	0	0
0	1	0
1	0	0
1	1	1

$E1$	$E2$	$E1 \vee E2$
0	0	0
0	1	1
1	0	1
1	1	1

$E1$	$\neg E1$
0	1
1	0

iv. Bedingungsgatter mit Ereignis E und Bedingung C: $E \wedge C$
v. XOR-Gatter mit E_1, E_2 entspricht: $(E_1 \wedge \neg E_2) \vee (\neg E_1 \wedge E_2)$
vi. Nicht-Gatter eines Eingangsereignisses E_1 entspricht: $\neg E_1$
vii. PUND-Gatter mit E_1, E_2 als Eingangsereignissen wird zu:
$(E_1 \wedge E_2) \wedge P$ mit $P = \{E_1 \text{ vor } E_2\}$[8]

Die logischen Operatoren haben die in Tab. 2.3 dargestellte Semantik.

Zwischenereignisse und Transfergatter lassen sich sukzessive durch entsprechende Formeln für Teilbäume ersetzen, so dass am Ende eine Gesamtformel nur mit Primärereignissen übrig bleibt. Der Fehlerbaum lässt sich also mit allen Gattern und Verknüpfungen als eine einzige boolesche Funktion darstellen, die man auch als *Strukturfunktion* bezeichnet.

Dies lässt sich am Besten an dem Fehlerbaum aus Abb. 2.1 veranschaulichen. Für das Hauptereignis TLE ergibt sich schrittweise:

$$TLE = E1 \vee E2 \tag{2.1}$$
$$E2 = E3 \wedge E4 \tag{2.2}$$
$$\Rightarrow TLE = E1 \vee (E3 \wedge E4) \tag{2.3}$$

Die Strukturfunktion stellt eine statische Abstraktion der Zusammenhänge aller Fehler von Systemkomponenten bzw. Ereignisse dar und wird wie gezeigt durch schrittweise Ersetzung der Gatter gebildet. Dadurch ist die Strukturfunktion immer eindeutig definiert.

2.4.2 Regeln und Normalformen

Die boolesche Algebra bringt einige hilfreiche Gesetzmäßigkeiten mit, die für die weiteren Betrachtungen von Fehlerbäumen unerlässlich sind und mit denen man die Strukturfunktion weiter umformen kann. Damit wir einen Nutzen daraus ziehen können, betrachten wir zunächst abstrakt die Gesetze und übertragen sie anschließend auf den Aussagen

[8] *Anmerkung*: Das PUND-Gatter ist hier mittels der Hilfskonstruktion über eine explizite Bedingung beschrieben. Eigentlich kann es nur mit einer erweiterten Logik präzise definiert werden in der Prädikate der Form „*A vor B*“ ausgedrückt werden können.

über den Fehlerbaum. Für Variablen A, B, C und den Symbolen $\top$ (verum, wahr) und $\bot$ (falsum, falsch) gilt:

1. Assoziativgesetz: $(A \wedge B) \wedge C = A \wedge (B \wedge C)$ sowie $(A \vee B) \vee C = A \vee (B \vee C)$
2. Kommutativgesetz: $A \wedge B = B \wedge A$ sowie $A \vee B = B \vee A$
3. Idempotenzgesetz: $A \wedge A = A$ sowie $A \vee A = A$
4. Absorptionsgesetz: $A \vee (A \wedge B) = A$ sowie $A \wedge (A \vee B) = A$
5. Neutralitätsgesetz: $A \wedge \top = A$ sowie $A \vee \bot = A$
6. Extremalgesetz: $A \wedge \bot = \bot$, $A \vee \top = \top$
7. Distributivgesetz: $A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$ sowie $A \vee (B \wedge C) = (A \vee B) \wedge (A \vee C)$
8. Doppelnegationsgesetz (Involution): $\neg(\neg A) = A$
9. Komplementärgesetz: $A \wedge \neg A = \bot$ sowie $A \vee \neg A = \top$
10. De Morgansche Gesetze: $\neg(A \wedge B) = \neg A \vee \neg B$ sowie $\neg(A \vee B) = \neg A \wedge \neg B$

Betrachten wir die Gesetze im Lichte des Fehlerbaums, dann ergeben sich aus diesen – zunächst sehr theoretisch anmutenden Äquivalenzen – direkte praktische Bezüge. Zum Beispiel folgt aus der Assoziativität der logischen Operatoren $\wedge$ und $\vee$, die eigentlich jeweils nur zwei Operanden haben, dass ein UND/ODER-Gatter mit mehr als zwei Eingangsereignissen einfach durch mehrere Operatoren in einer Formel beschrieben werden kann. Zusätzlich sind durch das Kommutativgesetz Reihenfolgen der Eingangsereignisse eines Gatters irrelevant. Dabei kann also z. B. ein UND-Gatter mit drei Eingangsereignissen leicht in $E_1 \wedge E_2 \wedge E_3$ übersetzt werden, ohne dass man sich weiter mit der Klammerung oder Reihenfolge auseinandersetzen müsste.

Die Idempotenz und das Absorptionsgesetz sind in zweierlei Hinsicht von Bedeutung: zum einen verändert ein mehrfaches Hinzufügen eines Ereignisses zu einem Gatter nicht die Strukturfunktion (diese Ereignisse können „herausgekürzt“ werden). Zum anderen reduzieren sich UND-Gatter sehr schnell, wenn über mehrere Teilbäume dasselbe Ereignis eingeht. Man mache sich klar, dass dies im Allgemeinen bedeutet, dass sich eine (Fehler-) VerUNDung schnell auf einen einzigen Eingang reduziert, welcher dann in der Regel einen Einfachfehler charakterisiert (vgl. auch Abschn. 2.5 und Kap. 4).

Das Neutralitätsgesetz und Extremalgesetz werden relevant, wenn Hausereignisse in einen Fehlerbaum eingefügt werden. Diese dienen oftmals als Schalter und werden entweder auf $\top$ oder $\bot$ geschaltet. Anhand der Gesetze lässt sich erahnen, dass damit ganze Teilbäume wegfallen können, je nach Schaltung und Verknüpfung.

Für nicht kohärente Fehlerbäume bilden die de Morganschen Gesetze mit Involution und doppelter Negation die Basis der Umformung und Auswertung.

Mit dem Distributivgesetz nähern wir uns dem Thema Auswertung, denn es erlaubt die Umformung von komplexen Gatterkombinationen durch Faktorisierung und Bildung von *Normalformen*. Für unsere Betrachtungen ist die Disjunktive Normalform (DNF) von besonderer Bedeutung, da sie die Lösungsmenge eines Fehlerbaums repräsentiert (s. Minimalschnitte in Abschn. 2.5). Die DNF ist eine „Disjunktion von Konjunktionstermen“,

d. h. eine VerODERung von UND-Termen und wird uns näher in Abschn. 2.6 beschäftigen.

2.5 Auswertung: Minimalschnitt

Die Basis der qualitativen und quantitativen Auswertung bilden die Kombinationen, unter welchen das Hauptereignis *wahr* wird. Dadurch wird die Auswertung mathematisch auf das Problem der Findung der Menge aller Lösungen der Strukturfunktion reduziert. In Abschn. 2.4 haben wir die booleschen Operatoren kennengelernt und gesehen, unter welchen Bedingungen sie wahr werden. Theoretisch könnte man also einfach die Strukturformel nehmen und z. B. über eine Wahrheitstabelle alle Kombinationen auflisten. Dabei stellen sich aber zwei Probleme:

1. Bei einem hinreichend großen FTA-Baum würde man selbst mit Hochleistungscomputern auf Grund der Komplexität mit der Auflistung nicht fertig werden! Für eine boolesche Funktion mit n Variablen ergibt sich generell eine Anzahl von 2^n möglichen Lösungen. Ein hinreichend komplexer Fehlerbaum mit 200 Basisereignissen, birgt bereits eine Anzahl von zu prüfenden Ereigniskombinationen, die größer ist als die Zahl der Atome unserer Sonne.[9]
2. Die Lösungsmenge der Strukturformel umfasst alle Möglichkeiten von Variablen, die zur Erfüllung der Formel führen. Für die Fehlerbetrachtung reicht uns allerdings eine reduzierte, *minimale* Lösungsmenge, die die wesentlichen Kombinationen von Fehlern enthält. Dazu muss man sich klarmachen, dass es zum Beispiel für ein ODER-Gatter ausreichend ist, wenn *ein* Ereignis auftritt, damit das Gatter aktiv wird. Ob das zweite Ereignis dazu kommt oder nicht, spielt für den Ausgang keine Rolle mehr, da das Gatter ohnehin schaltet. Damit reduziert sich die gesamte Lösungsmenge auf eine – für die Auswertung wesentliche – Teilmenge, die *Minimalschnitt* genannt wird.

Beispiel
Das Hauptereignis in Abb. 2.6 kann prinzipiell durch die folgenden Kombinationen erfüllt werden:

1. $\{A, B\}$
2. $\{A, C\}$
3. $\{A, B, C\}$

[9] Unsere Sonne hat geschätzte 10^{57} Atome. Erhöht sich die Zahl der Ereignisse auf „nur“ 300, ergeben sich schon mehr Möglichkeiten als die Anzahl der Atome im Universum, die auf ca. 10^{80} geschätzt werden.

Für den Eintritt des Hauptereignisses ist allerdings die letzte Kombination insofern redundant, als dass die Teilmengen $\{A, B\}$ oder $\{A, C\}$ ausreichen, damit T auftritt. Deshalb berechnet sich der Minimalschnitt dieses Fehlerbaums zu:

1. $\{A, B\}$
2. $\{A, C\}$

Ein weiteres anschauliches Beispiel ist das besprochene Mehrheitsentscheidungsgatter aus Abb. 2.2. Bei der abgebildeten 2-aus-3-Logik spielt die Kombination, dass alle drei Kontrolleinheiten ausfallen im Hinblick auf das Ereignis „Ausfall Komponente A“ keine Rolle: zwei Ausfälle genügen bereits, um das Ereignis zu verursachen, denn genau das wurde ja modelliert! Damit wird plausibel, dass auch dort die Kombination $\{E5, E6, E7\}$ entfällt.

Allgemein berechnet sich der Minimalschnitt, indem man aus der Lösungsmenge alle Ereignismengen eliminiert, für die eine Teilmenge ebenfalls Lösung der Formel ist. Mengentheoretisch sind diese zu eliminierenden Mengen alle Übermengen. Deshalb bleiben genau die (minimalen) Ereignismengen übrig, bei denen bereits die Wegnahme eines einzelnen Ereignisses dazu führt, dass die Menge keine Lösung mehr ist. Jede einzelne der verbleibenden Lösungen wird als ein Minimalschnitt bezeichnet (engl. *minimal cut set*). Zwei Algorithmen zur Bestimmung der Minimalschnitte werden in Kap. 14 vorgestellt.

Die Anzahl der Ereignisse im Minimalschnitt stellt seine *Ordnung* dar. Bei der Auswertung ist man in der Regel besonders an Minimalschnitten mit niedriger Ordnung interessiert, oft nur die der Ordnungen eins und zwei, da sie Einfach- bzw. Zweifachfehler repräsentieren. Die meisten FTA-Werkzeuge bieten deshalb Optionen an, nur Minimalschnitte bis zu einer gewissen Ordnung zu erzeugen (s. a. Abschn. 2.9).

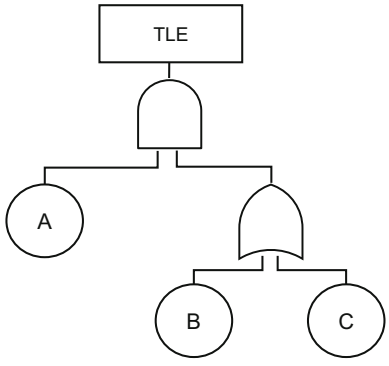


Abb. 2.6 Beispiel eines Fehlerbaums zur Minimalschnittberechnung

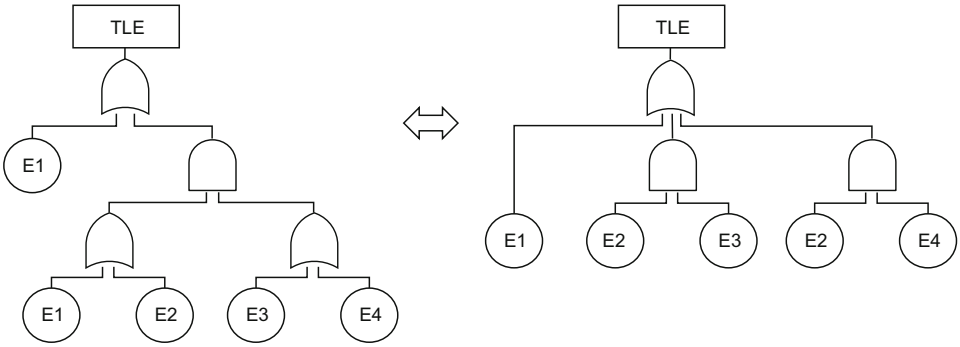


Abb. 2.7 Äquivalenter Minimalschnittbaum an einem Beispiel

2.6 Äquivalente Fehlerbäume

Aus dem Vorangegangenen sollte klar geworden sein, dass die Minimalschnitte die wesentlichen Informationen enthalten, um den Fehlerbaum zu repräsentieren. Sie sind also einerseits die „Lösungsmenge“ eines Fehlerbaums, andererseits können sie zu einem neuen Baum zusammengesetzt werden, der zum Ausgangsbaum *äquivalent* ist. Wir sprechen von zwei äquivalenten (Fehler-)Bäumen, wenn sie die gleichen Minimalschnitte haben. Die Grundidee ist die folgende:

1. Ein Minimalschnitt entspricht einer VerUNDung der enthaltenen Ereignisse. (Es müssen ja alle Ereignisse zusammen eintreten, damit das Hauptereignis auftritt.)
2. Alle Minimalschnitte repräsentieren alternative Lösungen, so dass diese zusammengenommen mit einer VerODERung das Hauptereignis exakt beschreiben.

Somit kann der sogenannte *Minimalschnittbaum* (engl. *(minimal) cut set tree*) konstruiert werden, indem man unterhalb des Hauptereignisses ein ODER-Gatter platziert und unterhalb dessen pro Minimalschnitt ein UND-Gatter mit den jeweiligen Primärereignissen (s. Abb. 2.7 für ein Beispiel und auch Abschn. 4.1.2). Dabei sind natürlich mehrfach verknüpfte Ereignisse (MOEs) in der Regel sehr häufig. Zudem hat der Fehlerbaum immer eine „Tiefe“ von zwei und ist tendenziell (auf Grund der großen Anzahl von Minimalschnitten) sehr breit.

2.7 Quantifizierung von Fehlerbäumen

Die Quantifizierung eines Fehlerbaums bildet eine mächtige Erweiterung dadurch, dass Eintretenswahrscheinlichkeiten für Ereignisse bestimmt und Verfügbarkeitsanalysen durchgeführt werden können. Mathematisch basieren die Konzepte auf der Wahrscheinlichkeitstheorie und stochastischen Modellen, so dass die Variablen der Strukturfunktion

als Zufallsvariablen interpretiert werden. Das Vorgehen besteht in der Bezifferung der (Eintretens-)Wahrscheinlichkeit der Basisereignisse, aus denen sich dann die Wahrscheinlichkeiten von übergeordneten Ereignissen und dem Hauptereignis ableiten lassen. Dabei kann man zwei grundlegende Kategorien von Modellen unterscheiden:

1. Feste Wahrscheinlichkeiten: Jedem Ereignis E wird eine Wahrscheinlichkeit zugeordnet, die zeitlich konstant ist.
2. Zeitabhängige Wahrscheinlichkeiten: Jedes Ereignis E tritt zeitabhängig mit verschiedenen Wahrscheinlichkeiten ein. Unter Rückgriff auf stochastische Prozesse und einer Wahrscheinlichkeitsverteilung kann eine Aussage über das zeitabhängige Ausfallverhalten getroffen werden.

Aus der Wahrscheinlichkeitstheorie ergeben sich einige Regeln und Zusammenhänge, die wir an dieser Stelle knapp vorstellen möchten, so dass man die quantitativen Auswertungen nachvollziehen kann. Eine umfassende Einführung in Wahrscheinlichkeitstheorie und Stochastik würde den Rahmen dieses Buches sprengen und wir verweisen auf die einschlägige Fachliteratur zum Thema (z. B. [3] für eine Einführung im Bereich der Zuverlässigkeitsanalysen).

2.7.1 Wahrscheinlichkeitstheorie

In Zufallsexperimenten wird einem Elementarereignis E eine Wahrscheinlichkeit $P(E)$ mit Werten im Intervall $[0, 1]$ zugeordnet, wobei 0 das unmögliche Ereignis („tritt niemals auf“) und 1 das sichere Ereignis („tritt immer auf“) ist. Man spricht in diesem Fall von *Zufallsvariablen*. Die Gegenwahrscheinlichkeit für *nicht* E folgt dann als $P(\neg E) = 1 - P(E)$. Verknüpft man mehrere Ereignisse mittels logischer Operatoren, lassen sich kombinierte Wahrscheinlichkeiten für Ereignisse $E_1, \cdots, E_n$ wie folgt berechnen:

1. UND-Verknüpfung zweier unabhängiger Ereignisse:

$$P(E_1 \cap E_2) = P(E_1) \cdot P(E_2)$$

2. ODER-Verknüpfung für sich ausschließende (inkompatible) Ereignisse:

$$P(E_1 \cup E_2) = P(E_1) + P(E_2)\,.$$

3. ODER-Verknüpfung für unabhängige Ereignisse:

$$P(E_1 \cup E_2) = P(E_1) + P(E_2) - P(E_1) \cdot P(E_2)\,.$$

4. XOR-Verknüpfung für unabhängige Ereignisse:

$$P(E_1 \cup E_2) = P(E_1) + P(E_2) - 2 \cdot P(E_1) \cdot P(E_2)\,.$$

Eine Grundvoraussetzung für die Anwendung dieser Gesetze ist die Unabhängigkeit der Ereignisse. Zwei Ereignisse $E1$, $E2$ sind *unabhängig* genau dann, wenn das Eintreten des einen Ereignisses $E1$ nicht die Wahrscheinlichkeit von $E2$ beeinflusst (und umgekehrt). Für Ereignisse im Fehlerbaum wird deshalb generell die Unabhängigkeit vorausgesetzt.[10]

Eine UND-Verknüpfung wird intuitiv als Produkt der Einzelwahrscheinlichkeiten berechnet. Nimmt man das Beispiel eines Würfels, dann ist nachvollziehbar, dass sich z. B. die Wahrscheinlichkeit zwei 6en zu würfeln aus der Wahrscheinlichkeit $P\{6 \text{ fällt}\} = \frac{1}{6}$ für eine Sechs wie folgt berechnet:

$$P\{\text{Zwei 6en}\} = P\{6 \text{ fällt}\} \cdot P\{6 \text{ fällt}\} = \frac{1}{6} \cdot \frac{1}{6} = \frac{1}{36}$$

Etwas komplizierter wird es für die ODER-Verknüpfung von Ereignissen. Wollen wir beispielsweise berechnen, wie groß die Wahrscheinlichkeit von einer Eins ODER einer Sechs bei einem Wurf mit zwei Würfeln ist, so folgt die Berechnung der Intuition mit:

$$P\{1 \text{ oder } 6\} = P\{1 \text{ fällt}\} + P\{6 \text{ fällt}\} = \frac{1}{6} + \frac{1}{6} = \frac{1}{3}$$

Dies lässt sich plausibel darstellen, wenn man die zwölf Würfel-Kombinationen explizit auflistet (die Würfel sind ja nicht benannt, d. h. es spielt keine Rolle, auf welchem Würfel die Zahl fällt):

$$\{1,1\},\{1,2\},\{1,3\},\{1,4\},\{1,5\},\{1,6\},\{6,1\},\{6,6\},\{6,5\},\{6,4\},\{6,3\},\{6,2\}$$

Diese einfache Summenregel gilt allerdings nur, da die elementaren Ereignisse sich gegenseitig ausschließen. Genauer gesagt ergibt die Summe aller möglichen Einzelwahrscheinlichkeiten des Würfels 1: kein Ereignis überlappt mit anderen. Wahrscheinlichkeitstheoretisch wird die gesamte Ergebnismenge als Ω bezeichnet:

$$P(\Omega) = \sum_{i=1}^{n} P_i = 1 \tag{2.4}$$

Nehmen wir hingegen ein anderes Beispiel: Die Firma Damage Inc. produziert Widerstände. Die letzte Qualitätsprüfung hat ergeben, dass (A) 70 % der produzierten Bauteile einen erheblich zu kleinen Widerstandswert aufzeigen und (B) 50 % zu kurze Anschlussdrähte besitzen. Wie groß ist die Wahrscheinlichkeit für einen Kunden, dass ein geliefertes Bauteil fehlerhaft ist? Zur Lösung können die Einzelwahrscheinlichkeiten nicht einfach addiert werden, da man sonst die Bauteile mit beiden Fehlerarten doppelt zählen würde (und man hätte unplausible 120 % Fehleranteil). Deshalb gilt die allgemeinere Summenregel:

$$\begin{aligned} P\{\text{A oder B}\} &= P\{\text{A}\} + P\{\text{B}\} - P\{\text{A und B}\} \\ &= 0{,}70 + 0{,}50 - (0{,}70 \cdot 0{,}50) = 0{,}85 \end{aligned}$$

[10] Ausnahme bildet hier die implizite Common-Cause Modellierung, s. Abschn. 2.7.5.

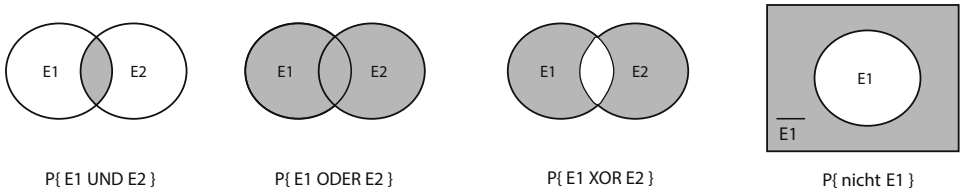


Abb. 2.8 Venn-Diagramme zur Veranschaulichung von kombinierten Wahrscheinlichkeiten

Um diesen Fall anschaulich darzustellen helfen Venn-Diagramme, die die Ereignismengen und deren Überlappung visualisieren (s. Abb. 2.8). Im besprochenen Fall der ODER-Verknüpfung würde man also die Schnittfläche doppelt zählen und entsprechend einmal subtrahieren. In Konsequenz für den Fehlerbaum ergibt sich aus dieser Darstellung Folgendes:

1. Primärereignisse werden als unabhängig behandelt und haben generell sich nicht ausschließende Eintretenswahrscheinlichkeiten. Diese Annahme ist nachvollziehbar, da man in der Regel nicht davon ausgehen kann, dass ein Fehlzustand einen anderen ausschließt.
2. Ein ODER-Gatter kann den Fall von sich ausschließenden Ereignissen nicht modellieren. D. h. zum Beispiel die verschiedenen Ergebnisse eines Würfelexperiments können mit den logischen Verknüpfungen eines Fehlerbaums nicht modelliert werden.
3. Ein XOR-Gatter bezeichnet *nicht* sich ausschließende Ereignisse, sondern das (exklusive) Eintreten von einem unabhängigen Ereignis in Abwesenheit des anderen.

2.7.2 Kombinierte Wahrscheinlichkeiten

Auf der Basis der Wahrscheinlichkeiten von Primärereignisse und den Regeln für deren logische Kombination aus Abschn. 2.7.1 können wir nun Wahrscheinlichkeiten für Zwischenereignisse, Minimalschnitte und dem Hauptereignis berechnen. Der Intuition folgend könnte man versuchen, von unten nach oben mit den Regeln aus Abschn. 2.7.1 schrittweise jedes Gatter auszurechnen. Doch hier gibt es einen Fallstrick: dieses Verfahren funktioniert im Allgemeinen *nicht*, da mehrfach verknüpfte Ereignisse nicht korrekt in die Rechnung eingehen würden! „Nicht korrekt“ meint hier auch nicht einfach nur „mehrfach zählen“ (und dadurch evtl. zu einer konservativen Schätzung kommend): ein Einzelfehler könnten u.U. bei diesem Vorgehen verschluckt werden, wenn er über verschiedene Zweige mit anderen Ereignissen UND verknüpft wird und deshalb im Beitrag zur Wahrscheinlichkeit des Hauptereignisses falsch eingerechnet wird.

Ein allgemeingültiges Verfahren basiert deshalb auf den Minimalschnitten, die alle Ereigniskombinationen eindeutig beschreiben. Da ein Minimalschnitt (MS) der VerUNDung

der Primärereignisse entspricht, ergibt sich seine Wahrscheinlichkeit deshalb aus dem Produkt der Einzelereignisse:

$$P(MS) = \prod_{i=1}^{k} P(E_i)\,, \quad \text{mit } k \text{ Ereignissen} \tag{2.5}$$

Wenn wir uns nun nochmal den Minimalschnittbaum aus Abschn. 2.6 vor Augen führen, der eine äquivalente Beschreibung verkörpert, kann die exakte Wahrscheinlichkeit des Hauptereignisses durch die VerODERung der Minimalschnitte berechnet werden:

$$\begin{aligned} P(TLE) = &\sum_{i=1}^{n} P(MS_i) - \sum_{i=2}^{n}\sum_{j=1}^{i-1} P(MS_i \cap MS_j) \\ &+ \cdots + (-1)^{r-1} \sum_{1 \le j_1 < j_2 < \cdots < j_r \le n} P(MS_{j_1} \cap \cdots \cap MS_{j_r}) \\ &+ \cdots + (-1)^{n-1} P(MS_1 \cap \cdots \cap MS_n) \end{aligned} \tag{2.6}$$

Dieser komplexe Ausdruck kommt dadurch zustande, da für die exakte Wahrscheinlichkeit beim ODER-Gatter alle Schnittmengen der Minimalschnitte „abgezogen“ werden müssen (vgl. Abschn. 2.7.2). Bei mehreren Ereignissen sind diese Überlappungen etwas komplexer (man stelle sich sehr viele überlappende Kreise vor), so dass das Vorzeichen wie in (2.6) ersichtlich alterniert. Für Details siehe zum Beispiel [9].

Der erste Term aus Gl. 2.6 entspricht als Summe der Gesamtmenge aller Minimalschnitte, bei dem die „Korrekturen“ von zweiter-, dritter- bis n-ter Ordnung nicht berücksichtigt sind. Bei kleinen Einzelwahrscheinlichkeiten und geringen Mehrfachverknüpfungen kann deshalb die Gesamtwahrscheinlichkeit des Hauptereignisses in der Näherung mit abgeschätzt werden (*Rare Event Approximation*):

$$P(TLE) \approx P_{\text{rare}}(TLE) = \sum_{i=1}^{n} P(MS_i) \tag{2.7}$$

Eine weitere Näherung erhält man, wenn man für die Wahrscheinlichkeit des ODER-Gatters mittels boolescher Regeln die folgende äquivalente Formel ansetzt:

$$P(A \cup B) = P(\neg(A \cap B)) = 1 - ((1 - P(A)) \cdot (1 - P(B))) \tag{2.8}$$

Übertragen auf die Menge der Minimalschnitte erhält man so die sog. *Minimal Cut Set Upper Bound* Näherung:

$$P_{\text{MCSUB}}(TLE) = 1 - \prod_{i=1}^{n} (1 - P(MS_i)) \tag{2.9}$$

Abbildung 2.9 zeigt einen quantifizierten Fehlerbaum, bei dem allen Basisereignissen feste Wahrscheinlichkeiten zugeordnet wurden.

Für die Gatter ergeben sich mit den Regeln aus Abschn. 2.7.1 also die folgenden Werte:

- $P(G3) = P(E1 \cup E2) = P(E1) + P(E2) - P(E1)P(E2) = 0{,}050095$
- $P(G4) = P(E3 \cup E4) = P(E3) + P(E4) - P(E3)P(E4) = 0{,}02098$
- $P(G2) = P(G3 \cap G4) = P(G3)P(G4) = 0.0010509931$

Für das Hauptereignis TLE ergibt sich:

$$\begin{aligned} P(TLE) &= P[E1 \cup ((E1 \cup E2) \cap (E3 \cup E4))] \\ &= P[E1 \cup (E2 \cap E3) \cup (E2 \cap E4))] \\ &= P(E1) + P(E2)P(E3) + P(E2)P(E4) \\ &\quad - [P(E1)P(E2)P(E3) + P(E1)P(E2)P(E4) \\ &\quad + P(E2)P(E3)P(E4)] + [P(E1)P(E2)P(E3)P(E4)] \\ &= 0{,}00148895 \end{aligned}$$

2.7.3 Zeitabhängige Wahrscheinlichkeiten

In der Realität kann den wenigsten Ereignissen eine feste Wahrscheinlichkeit zugeordnet werden, da sich die Auftretenswahrscheinlichkeit der meisten Fehler über die Zeit ändert. Dies liegt z. B. an Alterungsprozessen, so dass ein Ausfall umso wahrscheinlicher wird, je länger eine Komponente beansprucht wurde. Trägt man die Anzahl der Ausfälle gegen die Zeit in einem Diagramm ab, so erhält man eine Wahrscheinlichkeitsverteilung, die charakteristisch für eine Komponente ist.

Abbildung 2.10 zeigt zwei Verteilungen: links die Todesfälle beim Menschen nach Alter, rechts die typische Verteilung für ein elektronisches Bauteil. Wie angedeutet kann die Häufigkeitsverteilung bei diskreten Zeitintervallen noch in einem Histogramm dargestellt werden. Aus diesem Diagramm lässt sich die Ausfallhäufigkeit über die Zeit ablesen und daraus eine *Fehlerrate* ermitteln. Die Fehlerrate (engl. *failure rate*, auch Ausfallrate oder *hazard rate*) einer Komponente ist definiert als die (relative) Änderungsrate zu einem Zeitpunkt t:

$$r(t) \stackrel{\text{def}}{=} \lim_{dt \to 0} \frac{P\{\text{Ausfall in } [t + dt]\}}{P\{\text{Kein Fehler bis zum Zeitpunkt } t\}} \tag{2.10}$$

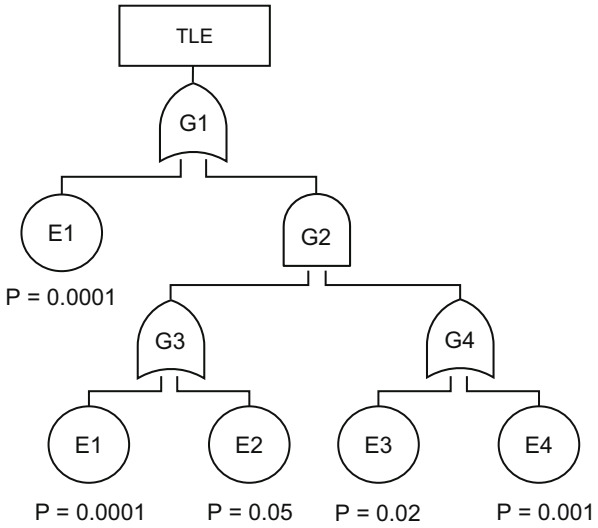


Abb. 2.9 Beispiel eines quantifizierten Fehlerbaums

Abhängig von der Fehlerrate lässt sich eine (kumulative) Verteilungsfunktion bestimmen, die angibt, mit welcher Wahrscheinlichkeit die Komponente *höchstens* (bis zu einem Zeitpunkt t) ausfällt[11] :

$$F(t) \stackrel{\text{def}}{=} P\{\text{Fehlerzeitpunkt} \leq t\} \tag{2.11}$$

$F(t)$ bezeichnet man als die *Unzuverlässigkeit* (engl. *unreliability*) und bezieht sich auf das Zeitintervall $[0, t]$. Dazu wird angenommen, dass die Komponente zum Zeitpunkt $t = 0$ funktionierte und (so gut wie) neu ist.

Beim Übergang zu Wahrscheinlichkeitsexperimenten mit kontinuierlich zeitabhängigen Zufallsvariablen wird F(t) in der Regel über eine Wahrscheinlichkeitsdichte $f(t)$

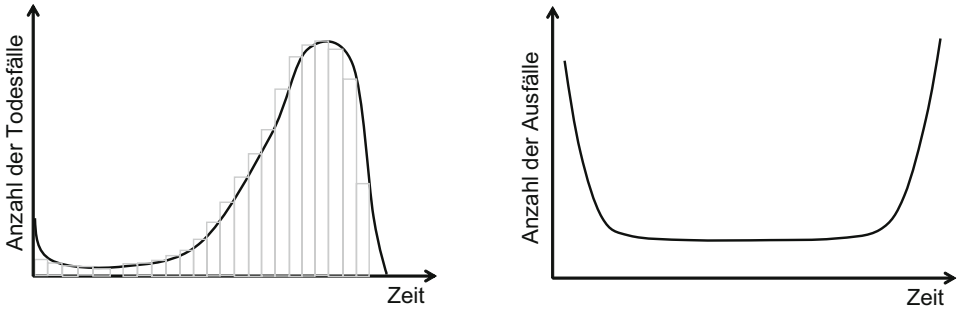


Abb. 2.10 Beispiele für Wahrscheinlichkeitsverteilungen

[11] Wir verwenden hier die englische(n) Abkürzung(en), die sich weitestgehend im Bereich der Zuverlässigkeitsanalysen und in FTA-Werkzeugen durchgesetzt haben.

charakterisiert:

$$f(t) \stackrel{\text{def}}{=} \lim_{dt \to 0} \frac{P\{\text{Komponente fällt aus in } t + dt\}}{dt} \tag{2.12}$$

Von der Dichtefunktion fordern wir die Eigenschaft, dass das Integral von 0 bis unendlich 1 ergibt (da sonst die Eintretenswahrscheinlichkeit größer als 1 werden könnte). Damit ergibt sich die Eintretenswahrscheinlichkeit über das Zeitintervall $[0, t]$ aus dem Integral:

$$F(t) = \int_0^t f(u)du \tag{2.13}$$

Umgekehrt lässt sich die *Zuverlässigkeit* (engl. *reliability*) definieren:

$$R(t) \stackrel{\text{def}}{=} P\{\text{Fehlerzeitpunkt} > t\} = 1 - F(t) \tag{2.14}$$

Generell gilt also zu jedem Zeitpunkt, dass eine Komponente funktioniert oder ausgefallen ist, so dass $F(t) + R(t) = 1$ plausibel ist. Es kann gezeigt werden, dass unabhängig von der Verteilungsfunktion $F(t)$ der Zusammenhang zwischen Fehlerrate, Fehlerdichte und Eintretenswahrscheinlichkeit wie folgt gilt (vgl. [9]):

$$r(t) = \frac{f(t)}{R(t)} = \frac{f(t)}{1 - F(t)} \tag{2.15}$$

Somit bilden $F(t)$, $f(t)$ und $r(t)$ einen mathematischen Zusammenhang, bei dem der Ausfall einer Komponente durch eine der drei Funktionen eindeutig beschrieben ist.

Für die meisten praktischen Anwendungen trifft man die vereinfachte Annahme, dass die Fehlerrate konstant ist. Mit Blick auf Abb. 2.10 gilt das z. B. für Elektronikbauteile für eine sehr lange Zeit, blendet man die frühen Ausfälle (*infant mortality*) und die Spätausfälle (*wear out*) aus. Für diesen sehr häufigen Spezialfall einer konstanten Fehlerrate, in der Regel mit λ bezeichnet, ergeben sich die folgenden Beziehungen:

$$F(t) = 1 - \mathrm{e}^{-\lambda t} \tag{2.16}$$

$$f(t) = \frac{d}{dt}(1 - \mathrm{e}^{-\lambda t}) = \lambda \mathrm{e}^{-\lambda t} \tag{2.17}$$

$$r(t) = \frac{f(t)}{1 - F(t)} = \frac{\lambda \mathrm{e}^{-\lambda t}}{1 - [1 - \mathrm{e}^{-\lambda t}]} = \lambda \tag{2.18}$$

Somit kann die Wahrscheinlichkeit eines Komponentenfehlers durch die Angabe der Fehlerrate bis zu einem Zeitpunkt t durch $1 - \mathrm{e}^{-\lambda t}$ bestimmt werden. Für weiterführende Betrachtungen zur Bestimmung einer Fehlerrate s. a. Kap. 10.

2.7.4 Unzuverlässigkeit vs. Verfügbarkeit

Neben der Unzuverlässigkeit kann für Systeme die (Nicht-)Verfügbarkeit bestimmt werden. Nichtverfügbarkeit (engl. *unavailability*) ist definiert als:

$$Q_S(t) \stackrel{\text{def}}{=} P\{S \text{ funktioniert nicht zum Zeitpunkt } t\} \tag{2.19}$$

Im Gegensatz zur Unzuverlässigkeit $F(t)$ bezeichnet $Q(t)$ also die Eintretenswahrscheinlichkeit zu einem *Zeitpunkt*. Umgekehrt definiert sich Verfügbarkeit (engl. *Availability*) als:

$$A_S(t) \stackrel{\text{def}}{=} P\{S \text{ funktioniert zum Zeitpunkt } t\} \tag{2.20}$$

Bislang haben wir mit der Unzuverlässigkeit von der Wahrscheinlichkeit *eines* Fehlers im Zeitintervall $[0, t]$ gesprochen, d. h. implizit wurde davon ausgegangen, dass ein Fehler (bzw. die Komponente) nicht reparierbar ist. Das heißt, es kann im Prinzip auch nur *einen* Fehler im Intervall $[0, t]$ auftreten. Dementsprechend gilt:

$$Q(t) = F(t)\,, \quad \text{für nicht reparierbare Komponenten} \tag{2.21}$$

$$Q(t) \leq F(t)\,, \quad \text{im Allgemeinen} \tag{2.22}$$

Berücksichtigt man reparierbare Komponenten, kommt bei der Betrachtung der Eintretenswahrscheinlichkeit der Aspekt der *Ausfallhäufigkeit* (engl. *failure frequency*) hinzu. Bei der Definition der Ausfallhäufigkeit ω als zeitkontinuierliche Größe verhält es sich ähnlich wie bei der Eintretenswahrscheinlichkeit, sie ist als Integral über eine Häufigkeitsdichte definiert:

$$\omega(t) = \int_0^t w(u)du \tag{2.23}$$

Die Häufigkeitsdichte $w(t)$ ist für nicht reparierbare Komponenten dieselbe wie $f(t)$, für reparierbare Komponenten mit konstanter Fehlerrate (und vernachlässigbarer Reparierdauer) kann sie mit $w(t) \approx \lambda$ angenähert werden.

2.7.5 Quantifizierung von Common-Cause-Ereignissen

Common-Cause-Fehler (CCF) bilden eine Form von abhängigen (Fehler-)Ereignissen (engl. *dependent events*). Zum Beispiel könnte eine Temperaturerhöhung gleichzeitig die Eintretenswahrscheinlichkeit von Bauteilen eines redundanten Kanals erhöhen. Solche komplexen Wirkungszusammenhänge lassen sich mittels einer Standard-FTA nicht mehr beschreiben, da wir die Unabhängigkeit von Ereignissen fordern.

Die einfachste Modellierung von CCF ist durch ein *explizites* Ereignis, welches den Fehler beschreibt und im Fehlerbaum (an den auftretenden Stellen u.U. mehrfach) verknüpft wird. Zum Beispiel könnten im Fehlerbaum aus Abb. 2.2b die Ereignisse für die

Kontrolleinheiten 1 bis 3 durch ein übergeordnetes ODER-Gatter mit einem CCF-Ereignis verknüpft werden (s. Abschn. 4.2.5 für ein modelliertes Beispiel).

Darüber hinaus existiert die *implizite Common-Cause*-Modellierung, die direkten Einfluss auf die quantitativen Berechnungen nimmt. Voraussetzung ist eine Deklaration einer Gruppe von Ereignissen, die unter Einfluss eines CCF stehen. Werden diese Ereignisse ausgewertet (z. B. im Kontext eines Minimalschnitts), dann muss der CCF berücksichtigt werden.

Die einfachste CCF-Quantifizierung für Fehler in redundanten Strukturen (d. h. mit gleicher Ausfallwahrscheinlichkeit) kann durch einen sog. *Beta-Faktor* erreicht werden. Dieser trifft die vereinfachende Annahme, dass ein bestimmter Prozentsatz der zunächst unabhängigen Fehler auf Grund des CCF zu einem gleichzeitigen Ausfall aller redundanten Strukturen führen. Ein Beta-Faktor von z. B. $\beta = 0{,}05$ besagt also, dass 5 % der Fehler durch gemeinsame CCF verursacht werden. Dieser fließt dann wie folgt in die Berechnung ein:

$$P(A \wedge B, \beta) = \underbrace{(P(A)(1-\beta) \cdot P(B)(1-\beta))}_{\text{unabhängiger Anteil}} + \underbrace{P(A)\beta}_{\text{CCF}} \tag{2.24}$$

Wohlgemerkt gilt in der Regel $P(A) \approx P(B)$, da das Modell für ähnliche oder baugleiche redundanten Strukturen entwickelt wurde (vgl. [9]). Der Term $P(A)(1-\beta)$ beschreibt dabei den verbleibenden Anteil der unabhängigen Eintretenswahrscheinlichkeit einer Komponente.[12]

Für die Bestimmung des Beta-Faktors existieren mehrere Vorgehen, z. B. über Checklisten wie in der IEC 61508 (vgl. [8], Teil 6). In der Praxis liegen die Werte für β in der Regel zwischen 0,01 bis 0,3 – je nach gemeinsam geteilten Ressourcen, Diversität der Bauteile und weiteren Überlegungen zu systematischen CCF-Ursachen.

2.8 Quantitative Auswertung

Die einfachste quantitative Auswertung ist eine zeitunabhängige Berechnung der Wahrscheinlichkeit des Hauptereignisses wie in Abschn. 2.7.2 gezeigt. Aus der Tatsache heraus, dass die Auftretenswahrscheinlichkeit eines Ereignisses zeitabhängig ist, muss die Fragestellung nach der Eintretenswahrscheinlichkeit des Systems (bzw. Eintretenswahrscheinlichkeit des Hauptereignisses) in Abhängigkeit von einer Betriebsdauer des Systems gestellt werden, der sogenannten Gesamtbetriebsdauer (engl. *mission time*). Abhängig davon können mehrere Systemgrößen berechnet werden:

1. Nichtverfügbarkeit bzw. Verfügbarkeit
2. Unzuverlässigkeit bzw. Zuverlässigkeit

[12] Teilweise wird das Modell auch für $P(A) \neq P(B)$ angewandt. In diesem Fall kann der CCF-Anteil z. B. durch Mittelwertbildung mit $(P(A)P(B)/2)\beta$ berechnet werden. Hierbei gibt es allerdings keine einheitlichen Regeln, da die Vorgehensweise über die zu Grunde liegenden Annahmen des Beta-Faktormodells hinausgehen.

3. Ausfallhäufigkeit im Zeitraum
4. Importanzen der Basisereignisse

Importanzen sind dabei Metriken der Primärereignisse, die deren Beitrag zum Hauptereignis errechnen. Sie stellen eine Möglichkeit dar, mit deren Hilfe man als Analyst die quantifizierten Ereignisse besser einordnen kann, um das Systemdesign zu optimieren.

Für alle diese Kennzahlen wird die Menge der Minimalschnitte als Berechnungsbasis genutzt, da sie alle Kombinationen von Ereignissen beschreiben, die das Hauptereignis eintreten lassen. Die detaillierten Berechnungen erläutern die folgenden Abschnitte.

2.8.1 Nichtverfügbarkeit des Systems

Die Nichtverfügbarkeit des Systems Q_{sys} kann für einen Fehlerbaum schrittweise über die Minimalschnitte berechnet werden. Dabei wird zunächst für jedes Ereignis eines Minimalschnitts die Nichtverfügbarkeit q bestimmt. Generell ist q für einen nicht reparierbaren Fehler gleich der Wahrscheinlichkeit, also für eine Komponente mit konstanter Fehlerrate λ zum Beispiel:

$$q(t) = P(t) = 1 - \mathrm{e}^{-\lambda t} \tag{2.25}$$

Bei reparierbaren Komponenten können Fehler prinzipiell durch Wartung behoben oder durch Testintervalle erkannt und behandelt werden. Dadurch ergibt sich zeitlich das periodisches Muster für die Nichtverfügbarkeit, das Abb. 2.11 veranschaulicht („Sägezahnkurve“).

Für reparierbare Komponenten existieren abhängig von der Wartungsdauer (*Mean Time to Repair*, MTTR) unterschiedliche Näherungen. Generell gilt für die Nichtverfügbarkeit eines reparierbaren Fehlers einer überwachten Komponente:

$$q_M(t) = \frac{\lambda}{\lambda + \mu}(1 - \mathrm{e}^{-(\lambda+\mu)t}) \approx \frac{\lambda}{\lambda + \mu} \tag{2.26}$$

Die Variable μ[Einheit: 1/h] bezeichnet dabei die Reparatur-/Wartungsfrequenz (also $\mu = 1/\text{MTTR}$). Für eine Standby-Komponente mit Testintervall τ ergibt sich in der Näherung für die Nichtverfügbarkeit eines reparierbaren Fehlers (Mittelwert der Sägezahnkurve):

$$\overline{q}(t) \approx \frac{\lambda\tau}{2} \tag{2.27}$$

Hier ist allerdings Vorsicht geboten, da für die Minimalschnitte faktisch Mittelwerte der Sägezahnkurve herangezogen werden. Bei mehreren solcher periodischen Verläufe ergeben sich dadurch allerdings zu optimistische Nichtverfügbarkeiten (vgl. [7] oder IEC 61508, Teil 6 [8]).

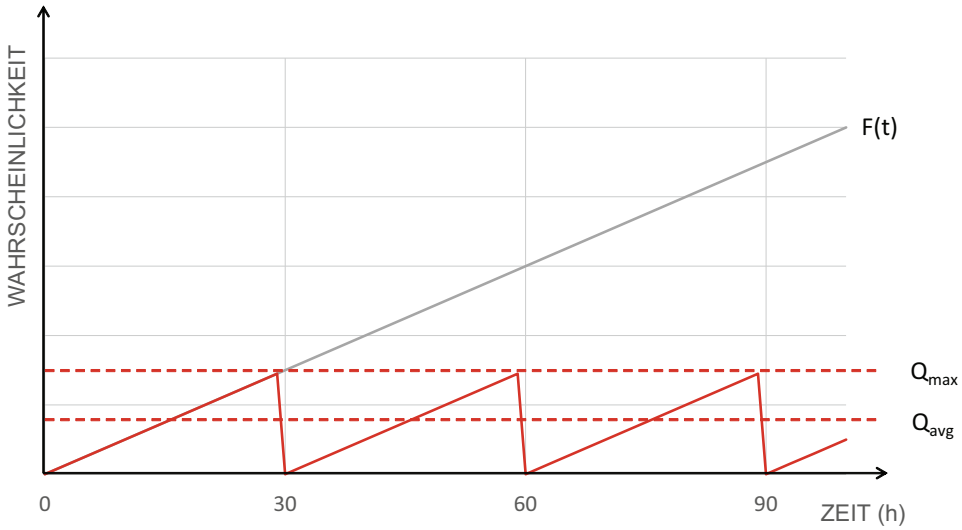


Abb. 2.11 Periodizität der Nichtverfügbarkeit durch Wartung- und Testintervalle

Das Maximum Nichtverfügbarkeit eines reparierbaren Fehlers (Peak der Sägezahnkurve) kann hingegen als obere Schranke angesehen werden und stattdessen genutzt werden, um auf der sicheren Seite zu sein:

$$q_{\max}(t) \approx \lambda\tau \tag{2.28}$$

Hat man die einzelnen Wahrscheinlichkeiten bestimmt, folgt die Nichtverfügbarkeit des Minimalschnitts Q_{MS} als Produkt der Einzelwahrscheinlichkeiten:

$$Q_{MS}(t) = \prod_{i=1}^{k} q_i(t)\,, \quad \text{mit } k \text{ Ereignissen} \tag{2.29}$$

Daraus ergibt sich für das System (d. h. Hauptereignis) die Nichtverfügbarkeit gemäß der Gl. 2.7 oder der Näherungen aus Abschn. 2.7.2.

2.8.2 Unzuverlässigkeit des Systems

Enthält der Fehlerbaum ausschließlich nicht reparierbare Komponenten ist die Unzuverlässigkeit für das Hauptereignis gleich der Nichtverfügbarkeit. Im Falle von reparierbaren Komponenten kann die Unzuverlässigkeit anhand eines Fehlerbaums nicht exakt berechnet werden[13]. Es lassen sich aber gute Näherungen mittels der Ausfallhäufigkeit errechnen.

[13] Jedenfalls nicht im allgemeinen Fall mit nicht konstanten Fehlerraten.

Zunächst muss dazu die Häufigkeit für jeden Fehler bestimmt werden. Für nicht reparierbare Komponenten ist die Ausfallhäufigkeit gleich der Fehlerdichte:

$$\omega_E(t) = f(t) = \lambda e^{-\lambda t} \tag{2.30}$$

Für eine reparierbare, überwachte Komponente ergibt sich analog zur Nichtverfügbarkeit mit μ als mittlere Wartungsfrequenz (unter Annahme von konstanten Wartungsintervallen):

$$\omega_E(t) = \frac{\lambda\mu}{\lambda+\mu}(1 - e^{-(\lambda+\mu)t}) \approx \frac{\lambda\mu}{\lambda+\mu} \tag{2.31}$$

Ferner ist die Unzuverlässigkeit eines reparierbaren Fehlers für regelmäßig getestete Komponenten mit Testintervall τ die Ausfallhäufigkeit:

$$\omega_E(t) = \lambda e^{-(t-\lambda\tau)} \tag{2.32}$$

Mit diesen Werten lässt sich die Ausfallhäufigkeit eines Minimalschnitts bestimmen:

$$\omega_{MS}(t) = \sum_{i=1}^{k} \omega_i(t) \cdot \prod_{\substack{j\neq i \\ j=1}}^{k} q_j(t) \tag{2.33}$$

Die Unzuverlässigkeit des Hauptereignisses kann nun über die Ausfallhäufigkeit des Systems angenähert werden (*system failure frequency*):

$$F_{\text{sys}} \approx \omega_{\text{sys}}(t) = \sum_{i=1}^{n} \omega_{MS,i}(t) \cdot \prod_{\substack{j\neq i \\ j=1}}^{n} 1 - Q_{MS,j}(t) \tag{2.34}$$

Hierbei muss betont werden, dass $\omega_{\text{sys}}(t)$ strenggenommen keine Wahrscheinlichkeit darstellt, da die Anzahl der Ausfälle beliebig groß werden kann (generell gilt also $\lim_{t\to\infty} = \infty$). Dennoch ergeben sich für typische Betrachtungszeiträume und Bauteilausfallraten sinnvolle Werte.

Alternativ erhält man die Unzuverlässigkeit des Hauptereignisses über die Fehlerrate des Systems. Es gilt der folgende Zusammenhang (vgl. [9], S.415):

$$F_{\text{sys}}(t) = 1 - e^{-\int_0^t r_{\text{sys}}(u)du} \tag{2.35}$$

Da die Bestimmung der Fehlerrate durch den Einfluss von reparierbaren Komponenten zu komplexen mathematischen Formeln führt, beschränkt man sich in der Regel auf Näherungen. Neben Gl. 2.34 existiert noch eine weitere, die sogenannte „Vesely-Näherung“.

Diese setzt statt der Systemfehlerrate die bedingte Fehlerdichte (engl. *conditional failure intensity*) des Systems, die auch mit $\lambda_{\text{sys}}(t)$ abgekürzt wird:

$$F_{\text{sys}}(t) = 1 - \mathrm{e}^{-\int_0^t \lambda_{\text{sys}}(u)du} \tag{2.36}$$

$$\lambda_{\text{sys}}(t) = \frac{\omega_S(t)}{1 - Q_S(t)} \tag{2.37}$$

Für eine konstante Fehlerrate sind bedingte Fehlerdichte $\lambda(t)$ und Fehlerrate $r(t)$ gleich. Im Allgemeinen stimmt diese Gleichheit zwar nicht, aber es ergeben sich dennoch brauchbare Werte.

2.8.3 Birnbaum-Importanz

Die Birnbaum-Importanz bestimmt die maximale Erhöhung der Eintretenswahrscheinlichkeit für den Ausfall einer Komponente im Vergleich dazu, dass die Komponente nicht ausfällt. Umgekehrt ausgedrückt bezeichnet die Importanz, welche Verbesserung sich einstellen würde, wenn die Komponente nicht mehr ausfällt (z. B. durch Reparatur). Formal wird dies über die partielle Ableitung für ein Ereignis ausgedrückt (vgl. [2]):

$$BI(E) = \frac{\partial P_{\text{Sys}}}{\partial P(E)} \tag{2.38}$$

Berechnen kann man die Ableitung z. B. über die Gesamtwahrscheinchlichkeit aller Minimalschnitte (MS), die E enthalten in Relation zu der Eintretenswahrscheinlichkeit des Ereignisses selbst. Dazu kann in Näherung die Summe der Minimalschnitte verwendet werden (s. a. Abschn. 2.7.2). Ein Berechnungsbeispiel dazu geben wir in Kap. 5 an.

Die dort verwendete Näherung ergibt sich, wenn Gl. 2.38 auf die *Rare Event Approximation* (Gl. 2.7) angewandt wird:

$$BI(E) \approx \frac{\sum P(\text{MS, die } E \text{ beinhalten})}{P(E)} \tag{2.39}$$

Mathematisch ergibt sich darüber hinaus der folgende Zusammenhang unter Anwendung der Shannon-Dekomposition (vgl. z. B.[6]):

$$\frac{\partial P_{\text{sys}}}{\partial P(E)} = P(\text{Sys}|E) - P(\text{Sys}|\neg E) \tag{2.40}$$

Dabei bedeutet $P(\text{Sys}|E)$ die Wahrscheinlichkeit des Hauptereignisses, die sich ergibt wenn $E = \top$ gesetzt wird. Umgekehrt bedeutet $P(\text{Sys}|\neg E)$ die Gesamtwahrscheinlichkeit mit $E = \bot$. D. h. die Birnbaum-Importanz kann als Differenz dieser Gesamtwahrscheinlichkeiten exakt berechnet werden.

2.8.4 Fussell-Vesely-Importanz

Die Fussell-Vesely-Importanz gibt an, welchen relativen Anteil die Minimalschnitte (MS), an denen der Ausfall einer bestimmten Komponente beteiligt ist, an der Gesamtwahrscheinlichkeit des Hauptereignisses haben:

$$FVI(E) = \frac{P(\text{MS, die } E \text{ beinhalten})}{P_{\text{Sys}}} \approx \frac{\sum P(\text{MS, die } E \text{ beinhalten})}{\sum P(\text{aller MS})} \tag{2.41}$$

Der Näherungsterm ergibt sich bei Anwendung der *Rare Event Approximation* (Gl. 2.7).

Ein Berechnungsbeispiel in Relation zu Diagnosendeckungsgraden geben wir dazu in Kap. 5 an.

2.8.5 Kritikalitätsimportanz

Die Kritikalitätsimportanz gibt ein Maß an, dass das Auftreten des Hauptereignisses eine Folge des Eintretens eines bestimmten Ereignisses ist. Da die Birnbaum-Importanz die Wahrscheinlichkeit des Ereignisses selbst nicht berücksichtigt, bezieht die Kritikalitätsimportanz diese mit ein:

$$CI(E) = BI(E) \cdot \frac{P(E)}{P_{\text{sys}}} \tag{2.42}$$

Eine Näherung kann auch hier über die Minimalschnitte berechnet werden, indem man die Summe der Minimalschnitte mit E durch die Summe aller Minimalschnitte dividiert. Für weitere Details siehe z. B. [10].

2.8.6 Barlow-Proschan-Importanz

Die Barlow-Proschan-Importanz (auch: engl. *initiator importance*) bezieht die Ausfallhäufigkeitsdichte in die Gewichtung mit ein. Somit ergibt sich für die Berechnung (vgl. [1]), bzw. als Näherung mittels der *Rare Event Approximation* (Gl. 2.7):

$$BPI(E) = \frac{f(E,t) \cdot \partial P_{\text{sys}}(t)}{f_{\text{Sys}}(t) \cdot \partial P(E,t)} \approx \frac{\omega(E,t) \cdot \sum P(\text{MS, die } E \text{ beinhalten}, t)}{\omega_{\text{Sys}}(t) \cdot P(E,t)} \tag{2.43}$$

Ein mathematisches Merkmal der Importanz ist, dass die Summe der *BPI* aller Ereignisse 1 ergibt. Eine Beispielrechnung zur Bestimmung der Importanz werden wir in Kap. 5 betrachten.

2.9 Weiterführende Betrachtungen

Im Zusammenhang mit der Theorie zur Fehlerbaumanalyse haben wir in den vorangegangenen Abschnitten das beschrieben, was wir aus der Praxis heraus als den *common sense* ansehen. Darüber hinaus existieren zahlreiche nicht standardisierte Erweiterungen und Implementierungen, denen man als Analyst mit notwendiger Vorsicht begegnen sollte.

Ein nicht einheitlicher Punkt sind Vereinfachungen, die bei der Auswertung vorgenommen werden. Zum Beispiel werden in Werkzeugen gerne Näherungen zu quantitativen Kennzahlen und Minimalschnitten angewendet, die einen empfindlichen Einfluss auf das Ergebnis haben können. Gerade in Zusammenhang mit sogenannten „*cut offs*“, bei denen Minimalschnitte z. B. ab einem bestimmten unteren Schwellwert für die Eintretenswahrscheinlichkeit nicht weiter berücksichtigt werden, entstehen unbekannt große Fehler in der Berechnung. Hier sollte man prüfen, unter welchen Kriterien die Algorithmen die Auswertungen optimieren. Hilfreiches Basiswissen zur Algorithmik haben wir dazu in Kap. 14 zusammengefasst.

Ein neueres Gebiet sind die *dynamischen Fehlerbäume* mit Prioritäts-UND-Gattern oder Gattern zu Ereignissequenzen. Hier sind in den letzten Jahren unterschiedliche Gatter vorgeschlagen worden, z. B. *cold spare* (CSP), *functional dependency* (FDEP) oder *sequence enforcing*. Diese Gatter haben teilweise unterschiedliche Interpretationen zur Gleichzeitigkeit von Ereignissen bzw. deren Reihenfolge.[14] Prinzipiell werden durch die Berücksichtigung der Reihenfolge von Ereignissen Minimalschnitte zu minimalen „Sequenzschnitten“ (engl. *cut sequences*), was eine völlig neue Auswertungsmethode verlangt. Man sollte bei diesen Neuerungen die zu Grunde liegenden Annahmen verstehen, um sie korrekt einzusetzen. Das gilt insbesondere für quantitative Berechnungen, auf die die Gatter eine Auswirkung haben (können).

2.10 Fazit

Die Beherrschung der theoretischen Grundlagen der FTA bildet das Fundament für eine korrekte Analyse. Die boolesche Logik und die graphische Symbolik versetzen den Analysten in die Lage, auch komplizierte Zusammenhänge und Sachverhalte korrekt im Fehlerbaum abzubilden. Die Bestimmung der Minimalschnitte ermöglicht eine gezielte Auswertung der FTA-Logik. Das Einbeziehen der Wahrscheinlichkeitstheorie und die darauf aufbauenden stochastischen Modelle bilden die Brücke zur Zuverlässigkeitstheorie, so dass Systeme hinsichtlich der relevanten Größen wie Fehlerwahrscheinlichkeiten, Fehlerraten usw. beurteilt werden können. Importanzkenngrößen ermöglichen spezifische Auswerteverfahren, um deren Beitrag zu diesen Größen zu identifizieren. Die in diesem Kapitel vermittelten Grundlagen sollen Analysten in die Lage versetzen, die quantitativen Auswertungen in Form von Unzuverlässigkeitsbetrachtungen oder Nichtverfügbarkeiten zu verstehen und ggf. zu überprüfen.

[14] Z. B. werden einige Gatter auf Markov-Modelle abgebildet, andere auf temporallogische Strukturformeln.

Literatur

1. Barlow RE, Proschan F (1974) Importance of system components and fault tree events. Stochastic Processes and their Applications 3(2):153–173
2. Birnbaum Z (1968) On the importance of different components in a multicomponent system. Technical Report 54
3. Birolini A (2010) Reliability Engineering: Theory and Practice. Sixth Edition, Springer
4. Commission UNR (1981) Fault Tree Handbook. Systems and Reliability Research Office of Nuclear Regulatory Research U.S. Nuclear Regulatory Commission Washington, D.C. 20555
5. DIN Deutsches Institut für Normung eV (2007) DIN EN 61025. Beuth Verlag GmbH
6. Dutuit Y, Rauzy AB (2005) Approximate estimation of system reliability via fault trees. Reliability Engineering & System Safety 87(2):163–172
7. Edler F, Soden M, Hankammer R (2015) An improved estimation of multiple-point fault probabilities if the faults have different periodic latencies. Journal of System Safety
8. International Electrotechnical Commission (2010) IEC 61508 – Functional safety of electrical/electronic/programmable electronic safety-related systems (parts 1–7)
9. Kumamoto H, Henley EJ (2000) Probablistic Risk Assessment and Management for Engineers and Scientists, Second edn. IEEE Press, New York
10. Rauzy A (1993) New algorithms for fault tree analysis. Reliability Engieering and System Safety 40:203–211
11. Stamatelatos M, Vesely W, Dugan J, Fragola J, Minarick J, Railsback J (2002) Fault Tree Handbook with Aerospace Applications. NASA Office of Safety and Mission Assurance, NASA Headquarters, Washington, D.C. 20546