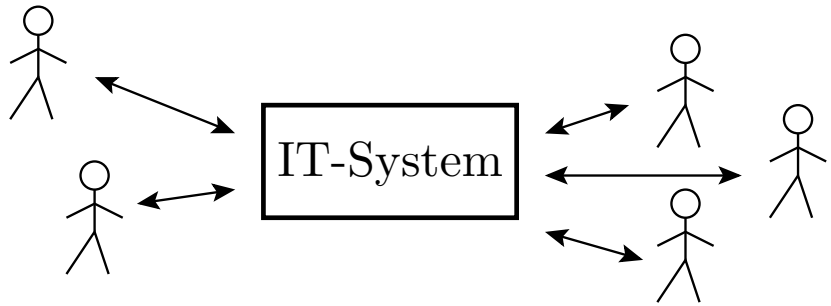
Test und Verlässlichkeit (F1) Kapitel 1: Modellbildung, Wahrscheinlichkeit, Experimente

Prof. G. Kemnitz

Institut für Informatik, Technische Universität Clausthal

13. Juni 2014

Vertrauen und Verlässlichkeit



IT-Systeme automatisierten intellektuelle Aufgaben:

- betriebliche Abläufe
- Steuerung von Prozessen und Maschinen
- Entwurfsaufgaben, ...

Einsatzvoraussetzung ist Vertrauen, dass

- das System, wenn es gebraucht wird, funktioniert,
- seine Service-Leistungen korrekt und pünktlich ausführt,
- keine unkalkulierbaren Schäden und Kosten verursacht.

Fakt 1

Vertrauen setzt Verlässlichkeit voraus.

Fehlfunktionen in IT-Systemen müssen nicht, aber können erheblichen Schaden verursachen:

- Datenverlust,
- Hintertüren für den Datenmissbrauch,
- Unfälle, Selbstzerstörung, Produktionsausfälle, ...

Am 3. Juni 1980 meldete ein Rechner des nordamerikanischen Luftverteidigungszentrums den Anflug sowjetischer Nuklearraketen. Sofort wurden Vergeltungsmaßnahmen vorbereitet. Eine Überprüfung der Daten von Radarstationen und Satelliten konnte den Angriff nicht bestätigen, so dass der Start amerikanischer Raketen mit Nuklearsprengköpfen in letzter Minute gestoppt wurde [2].

Urheber der nahen Katastrophe war ein defekter Schaltkreis in einem Rechner.

In dem Begriff der Verlässlichkeit treffen Wunschvorstellungen und Wirklichkeit zusammen. Das macht eine objektive Bewertung schwierig[1]. Sprichworte mit tiefem Wahrheitsgehalt:

- Allen Leuten recht getan, ist eine Kunst, die keine kann.
- Verlorenen Vertrauen ist schwer wieder herzustellen.
- Whatever can go wrong will go wrong. (Murphys Law[2])
- It ist not a Bug, it is a feature. (Wegreden von Fehlern, statt Beseitigung.)

Der Schlüssel zu objektiv verlässlichen Systemen sind Kontrollen und das Abstellen der dabei erkannten Mängel auf drei Ebenen:

- während Entwurf und Fertigung (Fehlervermeidung),
- vor dem Einsatz und zur Wartung (Fehlerbeseitigung) und
- im laufenden Betrieb (Fehlertoleranz, Schadensvermeidung).

[1] Wie bei der Verlässlichkeit zwischenmenschlichen Beziehungen, in der Politik, Wirtschaft ... ist es auch für IT-Systeme kaum möglich, allgemein akzeptierte mess- oder abschätzbare Kriterien zu definieren.

[2] Viele Menschen denken pessimistisch, d.h. die negativen Erfahrungen bleiben viel stärker im Gedächtnis haften als die positiven.

Inhalt und Lernziel der Vorlesung

- Bewertung der Verlässlichkeit
- Kontrollen, Fehlertoleranz und Schadensvermeidung
- Test und Fehlerbeseitigung
- Fehlervermeidung.

Inhalt Foliensatz F1

Modellbildung

Der Begriff »Modell« in der Informatik

Selbst die einfachsten Sachverhalte in der Informatik wie die Abarbeitung eines Befehls werden sehr schnell kompliziert, wenn alle Details berücksichtigt werden.

Definition 2

Ein Modell ist ein Mittel, um einen Zusammenhang zu veranschaulichen. Es stellt die wesentlichen Sachverhalte dar und verbirgt unwesentliche Details.

In dieser Vorlesung sind die Modelle Zufallsexperimente für

- das Funktionieren und Versagen von IT-Systemen,
- für Kontrollen, Tests, Reviews, Fehler, Ausfälle, ... ,
- Fehlervermeidung, Fehlerbeseitigung und Schadensbegrenzung.

für Systeme aus Hardware und/oder Software [+ Mechanik].

Zufallsexperiment

Zufallsexperiment

Die einzelnen Aspekte der Verlässlichkeit

- ist das System verfügbar,
- sind die gelieferten Ergebnisse richtig und
- entsteht kein unkalkulierbarer Schaden

sollen mit Wahrscheinlichkeiten bewertet werden. Die Basis für die Definition von Wahrscheinlichkeiten sind Zufallsexperimente:

Definition 3

Ein Zufallsexperiment ist ein Experiment mit mehreren möglichen Ergebnissen und zufälligem Ausgang.

Beispiele für Zufallsexperimente:

- Zählen der Fehler in einem System. WB: 0, 1, 2, ...
- Aufdecken eines Fehlers mit einem Test. WB: ja, nein
- Messen der Zeit bis zum Ausfall: WB: $t_A \geq 0\,s$

Bernoulli-Versuche

Das einfachste Zufallsexperiment ist der Bernoulli-Versuch. Er hat zwei mögliche Ergebnisse 0/1 (nein/ja, falsch/wahr, ...) und die Verteilung

$$\begin{aligned} P\{X=0\} &= 1-p \\ P\{X=1\} &= p \end{aligned}$$

(p – Wahrscheinlichkeit, dass das Ergebnis 1, ja oder wahr ist).
Bernoulli-Versuche für Aspekte der Verlässlichkeit:

- Kontrolle, ob ein Service verfügbar ist?
- Kontrolle, ob ein Service korrekt ausgeführt wird?
- Test, ob ein System fehlerhaft ist?
- Test, ob ein Fehler nachweisbar ist?

Aus den Ergebnissen von Bernoulli-Versuchen lassen sich weitere Kenngrößen abschätzen, z.B. die Anzahl der Fehler im System, die Häufigkeit der Fehlfunktionen durch Fehler, ...

Service-Modell

Das Service-Modell

Die Modellierung der Verlässlichkeit mit Bernoulli-Versuchen verlangt, dass die Operationen des Systems sowie die möglichen Fehlfunktionen, Fehler, Ausfälle, ... zählbar sind. Die zu zählenden Einzeloperation eines Systems werden im Weiteren als Service-Leistung oder kurz Service bezeichnet:

Definition 4

Eine Service sei ein Berechnungsablauf, der mit der Entgegennahme der Service-Anfrage beginnt, aus den Daten der Service-Anfrage Ausgaben berechnet und dieser weitergibt.

Die Ein- und Ausgabe sind ganz allgemein bedatete Objekte mit einem auf die Art des Services abgestimmten Format. Das Format hängt von der Art des Systems ab und legt die Struktur und Bedeutung der Daten festlegt. Die Berechnung hat eine Soll-Funktion und optional Vorgaben für die Ausführungszeit.

Programme als Service-Anbieter

Funktionsaufruf:

```
int UP(int a, uint b){
    return 23*(a+b);
};
```

- Eingabeobjekt: Variable a vom Typ int und b vom Typ uint bedatet mit den Aufrufwerten.
- Ausgabeobjekt: Rückgabewert vom Typ int.
- Soll-Funktion[3]: Rückgabewert $\leftarrow$ 23*(a+b)

Programm zur Bilderkonvertierung von JPG nach PNG.

- Eingabeobjekt: JPG-Datei
- Ausgabeobjekt: PNG-Datei
- Soll-Funktion: Bildformatkonvertierung.
- Ausführungszeit: 10 ms (Beispielannahme).

[3] Die Soll-Funktion weicht im Beispiel von der Ist-Funktion ab.

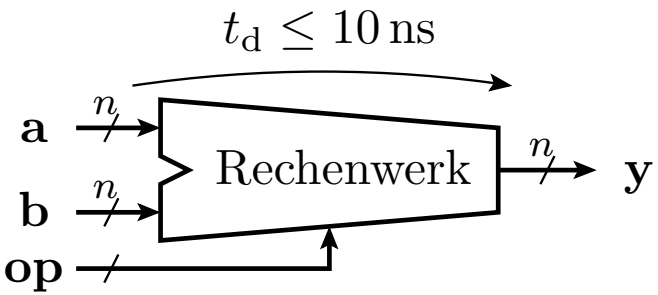
Client-Server-Systeme und Hardware

Anfrage an einen Server (z.B. eine Suchmaschine).

- Eingabeobjekt: Internet-Datenpaket
- Ausgabeobjekt: Internet-Datenpaket
- Soll-Funktion: Rückgabe einer Web-Seite mit den Anfrageergebnissen.

Digitale Verarbeitungsfunktion, z.B. ein Rechenwerk.

- Eingabeobjekt: Operanden und Op.-Code.
- Ausgabeobjekt: Ergebnis.
- Funktion: arithmetische oder logische Operation
- Ausführungszeit: 10 ns (Beispielannahme).



a, **b**	Operanden
op	Operationscode
y	Ergebnis
n	Bitbreite
t_d	Verzögerungszeit

Mechatronische Systeme

Motorsteuergerät.

- Eingabeobjekt: Soll-Position x_{soll}, Ist-Position x_{ist}.
- Ausgabeobjekt: Stellwert s, Anzeigewerte, ...
- Soll-Funktion: PI-Regelung (Beispielannahme)

$$\begin{aligned} s &= I + k_{\text{p}} \cdot (x_{\text{soll}} - x_{\text{ist}}) \\ I &= I + k_{\text{i}} \cdot (x_{\text{soll}} - x_{\text{ist}}) \end{aligned}$$

(I – Integralwert; k_{p}, k_{i} – Reglerparameter);

- Ausführungsperiode: 10 ms (Beispielannahme).

Waschmaschine.

- Eingabeobjekt: Programmauswahl, Waschmitteldosierung, ...
- Ausgabeobjekt: Messbare Parameter des Waschergebnisses
- Soll-Funktion: Abarbeitung des gewählten Waschprogramms.
- Ausführungszeit 1 Stunde (Beispielannahme).

Systeme mit und ohne Gedächtnis

Bei allen Systemtypen (Schaltung, Programm, ...) ist zwischen Systemen ohne und mit Gedächtinis zu unterscheiden. Ohne Gedächtnis ist die Soll-Ausgabe eine Funktion der Eingabe. Mit Gedächtnis werden Zwischenergebnisse gespeichert und bei nachfolgenden Service-Anfragen mit ausgewertet.

	ohne Gedächtnis	mit Gedächtnis
Unterprogramm	Berechnung Logarithms	OOP-Methoden zur Objektbearbeitung.
Programm	Compiler	Textverarbeitung
Serverdienst	Bafög-Rechner	Datenbankanfrage
digitale Schaltung	Rechenwerk	Prozessor
mechatr. System	Waschmaschine	PI-Regler

Systeme ohne Gedächtnis sind viel einfacher zu modellieren:

- Das Ergebnis jeder Service-Anfrage ist unabhängig von den vorherigen Anfragen.
- Die Service-Leistungen versagen unabhängig voneinander, modellierbar durch unabhängige Bernoulli-Versuche.

Bei einem System mit Gedächtnis:

- ist ein Teil der Daten, die in die Berechnung einfließen und die berechnet werden, nicht von außen steuer- und beobachtbar. Erschwert Kontrolle, Test und Fehlersuche.
- Bei einem Versagen sinkt die Verlässlichkeit, weil interne Daten kontaminiert sein können. Extremfall Absturz (keine Systemreaktion mehr). Erfordert in der Regel einen Neustart.
- Die Service-Leistungen sind nur bis zum ersten Versagen nach Initialisierung durch unabhängige Bernoulli-Versuche annäherbar, dann erst wieder nach Neuinitialisierung.

Systeme ohne Gedächtnis können durch Fehler ein Gedächtnis bekommen, z.B. bei vergessener Variableninitialisierung.

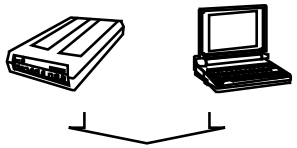
Service-Hierarchie

IT-Systeme sind hierarchisch aufgebaut:

- Client-Server-Systeme bestehen aus Rechnern und Netzwerkkomponenten.
- Rechner, Netzwerkkomponenten, ... bestehen aus Hard- und Software.
- Software besteht aus Programmbausteinen, diese sind aus Programmieranweisungen zusammengesetzt, die ihrerseits aus Maschienenbefehlen nachgebildet werden.
- Maschinenbefehle sind Service-Leistungen der Hardware. Die Hardware bestehen aus Funktionsbausteinen, diese meist aus Gattern und diese wiederum aus Transistoren.

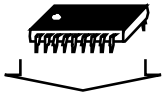
Hierarchie der Hardware

Geräte

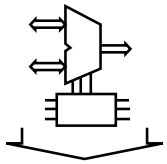


Baugruppen

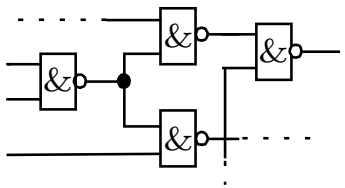
Schaltkreise



Funktionsblöcke



Gatterschaltungen



Im Service-Modell

- stellen die Transistoren die Service-Leistungen Ein- und Ausschalten bereit, die von den Gattern für die Nachbildung logischer Funktionen genutzt werden.
- Gatter stellen logische Funktionen und Speicherelemente Speicherfunktionen bereit, mit denen übergeordnete Funktionseinheiten wie Rechenwerke, Register bis hin zu kompletten Rechnern nachbildet werden.
- Die Software-Bausteine nutzen die Service-Funktionen der Hardware, für die Bereitstellung übergeordneter Funktionen.

Ein IT-System funktioniert korrekt, wenn alle Service-Leistung hierarchisch absteigend korrekt funktionieren.

Fehler

Fehler in Systemen ohne Gedächtnis

Wenn ein Service versagt, enthält er Fehler, gab es Störungen oder war die Eingabe falsch. In einem System ohne Gedächtnis ist ein Fehler daran zu erkennen, dass der Service bei Wiederholung der Anfrage immer oder oft in derselben Weise versagt.

Zur Fehlerlokalisierung wird im Berechnungsfluss nach dem Teil-Service gesucht, der aus korrekten Eingaben falsche Ausgaben produziert. In diesem Teil-Service wird, wenn Zugriff auf die Datenschnittstellen besteht, wiederum der Teil-Service gesucht, der aus einer korrekten Anforderung falsche Ergebnisse bildet.

Ergebnis einer solchen Fehlerlokalisierung sind der unterste mit den verfügbaren Diagnosemitteln lokalisierbare fehlerhafte Service und die Bedatung der Anfrage, mit der er versagt.

Definition 5

Ein permanenter (ständig nachweisbarer) Fehler sei der unterste lokalisierbare Service, der mit mindestens einer Bedatung falsch ausgeführt wird[a].

[a] Es ist möglich, dass ein Service versagt, wenn alle genutzten Teilservice-Leistungen korrekt ausgeführt werden, z.B. wenn Service-Aufrufe vergessen oder falsch sind.

Die Fehlerhierarchie folgt der Service-Hierarchie.

Digitale Schaltung:

- Transistorfehler: Transistor schaltet nicht ein oder aus.
- Gatterfehler: falsche logische Ausgabe, ...
- Rechner: falsche/keine Operationsausführung

Software:

- falsche Ausführung von Maschinenbefehlen,
- falsche Ausführung von Befehlsfolgen, Unterprogrammen,
- fehlerhafte Programmbausteine, ...

Anzahl der Fehler

Nach der Fehlerdefinition sind Fehler die kleinsten lokalisierbaren defekten (Teil-) Service-Leistungen und in der Praxis meist die kleinsten reparier- oder austauschbaren Einheiten.
Größenordnung ist die Anzahl der kleinsten isoliert betrachtbaren Service-Systeme. Je nach Betrachtungsebene kann das ein kompletter Server, ein Serverdienst, ein Programmbaustein, eine Anweisung, ein Gatter oder ein Transistor sein.

Bei Reparatur durch Ersatz ist die potentielle Fehleranzahl eins. System ist fehlerhaft und muss ersetzt werden.

In einem modular aufgebauten System wie einem Laptop werden Software- und Hardware-Fehler getrennt beseitigt, Hardware-Fehler durch Komponententausch, Software-Fehler durch Änderung von Konfigurationseinstellungen (Anpassungsprogrammierung) oder Ersatz (andere Version, Alternativprogramme).

- Fehler nach der Fertigung:
  Nach der Fertigung z.B. einer Baugruppe kann es Fehler geben, die bei einer Baugruppe im Einsatz, die schon mal funktioniert hat, ausgeschlossen sind: fehlende und falsch bestückte Bauteile, Zinnbrücken zwischen Schaltkreisanschlüssen, ...

- Fehler nach dem Entwurf:
  Entwürfe haben andere Fehlermöglichkeiten als Systeme im Einsatz, die schon mal (überwiegend) funktioniert haben, insbesondere auch grobe Fehler wie falscher Algorithmen, vergessene Dienste, ... Bei fehlerhaften Hardware-Entwürfen haben Bauteile vom selben Typ den selben Entwurfsfehler. Die Anzahl der potentiellen Fehler orientiert sich an der Typenanzahl, statt der Bausteinanzahl.

- Nur potentielle Fehler können tatsächliche Fehler sein.

Unbeständige Fehler und Systeme mit Gedächtnis

Zwischen der Eingabe und dem Versagen eines Services kann auch nur eine Korrelation bestehen. Der Service versagt bei bestimmten Eingaben nicht immer, sondern nur gewisser Häufigkeit, z.B. wenn das Ergebnis zusätzlich zur Eingabe abhängt

- von nicht steuerbaren gespeicherten Zuständen oder
- von Fremdeinflüssen (Temperatur, Versorgungsspannung).

Die Fehlerbeseitigung erfordert Tests für die Erfolgskontrolle nach jedem Reparaturversuch. Für unbeständige Fehler, die bei einer bestimmten Service-Anfrage mit einer Wahrscheinlichkeit kleiner eins versagen, ist die Kontrolle mehrfach zu wiederholen.

Unbeständige Fehler in Systemen mit Gedächtnis werden oft beständig, wenn außer der Bedatung auch der gespeicherte Zustand und die Fremdeinflüsse, bei denen der Service versagt, in die Testfallbeschreibung einbezogen werden.

Die Alternative zur mehrfachen Wiederholung, um zu prüfen, ob ein unbeständiges Fehlverhalten vorliegt oder ein unbeständiger Fehler erfolgreich beseitigt wurde, ist bei einem sequentiellen System, die nicht direkt steuerbaren Zustände über die vorherigen Anfragen reproduzierbar einzustellen:

- Neuinitialisierung und Wiederholung aller Service-Anfragen bis zu dem Service, der versagt, oder
- regelmäßige Sicherung des Systemszustands. Bei Versagen Initialisierung mit dem letzten gesicherten Zustand und Wiederholung aller Service-Anfragen ab Sicherungszeitpunkt.

Die Ursache unbeschändiger Fehler können auch beständige Fehler in Teilsystem sein, z.B.

- Fehler in der Adressrechnung, die bewirken, dass Variablen anderer Programme verändert werden und
- fehlende Initialisierung lokaler Variablen.

Ausfall

Die meisten Fehler in IT-Systemen sind solche, die von den Tests vor dem Einsatz nicht erkannt wurden. Da die Iteration aus Test und Fehlerbeseitigung vor dem Einsatz vorrangig die gut nachweisbaren Fehler beseitigt, versagen Systeme im Einsatz nur selten aufgrund nicht erkannter Entwurfs- und Fertigungsfehler.

Die Hardware und Mechanik eines IT-Systems unterliegt einem Verschleiß. Der kann bewirken, dass Fehler im Einsatz entstehen, gut und schlecht nachweisbare. Ein neuer gut nachweisbaren Fehler erhöhen die Häufigkeit des Versagens von Service-Leistungen sprunghaft, im Extremfall bis zur Nichtverfügbarkeit.

Definition 6

Ein Ausfall ist ein Ereignis, bei dem ein neuer Fehler entsteht, der die Häufigkeit des Versagens von Service- Leistungen sprunghaft dauerhaft (Nicht nur bis zur Neuinitialisierung) erhöht.

Aufgaben

Aufgabe 1.1: Fehler und Fehlerbehandlung

1. Was ist der Unterschied zwischen einem unbeständigen und einem beständigen (permanenten) Fehler?
2. Was sind die üblichen Arten der Problembehandlung
   - auf das Versagen einer Service-Leistung eines Systems ohne Gedächtnis,
   - auf das Versagen einer Service-Leistung eines Systems mit Gedächtnis,
   - auf einen Ausfall?

Aufgabe 1.2: Wertebereichsproblem

Eine Service-Leistung sei definiert durch:

- Eingabeformat: a, b: 16-Bit Zweierkomplement
- Ausgabeformat: Rückgabewert 16-Bit Zweirkomplement
- Soll-Funktion: Rückgabe des Wertes des Ausdrucks a-b+25
- Implementierung als C-Funktion:

```
int16_t fkt(int16_t a, int16_t b){
    return a-b+25;
}
```

1. Wie groß sind die kleinsten und größten darstellbaren Ein- und Ausgabewerte?
2. Für welche Bedatungen von a und b unterscheidet sich der Ausgabe-Ist- vom Ausgabe-Soll-Wert?
3. Wie sind Ist- und die Soll-Funktion zu verändern, so dass bei einem Bereichsüber- bzw. -unterlauf des Ergebnisses der größte bzw. kleinste darstellbare Wert zurückgegeben wird?

Aufgabe 1.3: C-typischer Multiplikationsfehler

Eine Service-Leistung sei definiert durch:

- Eingabeformat: zwei Variablen a und b, 8-Bit vorzeichenfrei
- Ausgabeformat: Rückgabewert 16-Bit vorzeichenfrei
- Sollfunktion: Rückgabe des Produkts a*b
- Implementierung als C-Funktion:

```
uint16_t umult16(uint8_t a, uint8_t b){
 return a*b;
}
```

1. Kleinster und größter darstellbarer Ein- und Ausgabewerte?
2. Für welche Bedatungen von a und b unterscheidet sich der Ist- vom Soll-Wert der Ausgabe[4]?
3. Wie ist die Ist-Funktion zu verändern, dass für alle Eingabewerte das korrekte Ergebnis berechnet wird?

[4] In C hat ein Produkt den Typ des Operanden mit dem größten Wertebereichs. Typenumwandlung der Zuweisung erst nach Produktbildung.

Aufgabe 1.4: Typ. Fehler einer Gleitkommadivision

Eine Service-Leistung sei definiert durch:

- Ein- und Ausgabegabeformat: 32-Bit Gleitkommaformat IEEE 754 »single«
- Soll-Funktion: Rückgabe von y=sin(x)/x mit maximaler Soll-/Ist-Abweichung:

$$\frac{|y_{\text{Soll}} - y_{\text{Ist}}|}{y_{\text{Ist}}} < 0.01\%$$

- Implementierung als C-Funktion:

```c
#include <math.h>
float sinc(float x){
  return sin(x)/x;
}
```

1. Beschreiben Sie den Aufbau das Gleitkommaformats IEEE 754 »single«[5].
2. Wie wird der Eingabewert -5.0 dargestellt?
3. Für welchen Eingabebereich weicht das Ist-Ergebnis vom Soll-Ergebnis ab.
4. Verbessern Sie die Implementierung, so dass sie auch für den im Aufgabenteil zuvor bestimmten Wertebereich der Eingabe korrekte Ergebnisse liefert.

[5] Die benötigten Informationen finden unter dem Suchbegriff »IEEE Gleitkommaformat« im Internet.

Aufgabe 1.5: Initialisierungsfehler

Das nachfolgende Unterprogramm soll für das mit einem Zeiger auf den Anfang und der Länge übergebene Feld den kleinsten Wert zurückgeben und hat ein unbeständiges Fehlverhalten.

```
int16_t Feld[]= {231, -13, ...}; // Beispiel für ein Feld
...
int16_t kleinsterWert(int16_t *Feld, uint16_t len){
  int16_t tmp, *ptr;
  for (ptr=Feld; ptr <Feld+len; ptr++){
    if (*ptr<tmp) tmp = *ptr;
  }
 }
```

1. Mit welchen Eingaben und Zusatzbedingungen ist der Fehler nachweisbar?
2. Ändern Sie das Programm so, dass es korrekt funktioniert.

Aufgabe 1.6: Fehler in kombinatorischer Schaltung

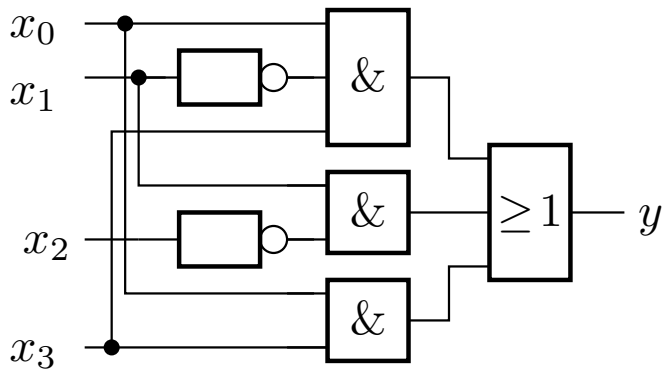
Eine kombinatorische Schaltung mit der Soll-Funktion entsprechend der nachfolgenden Wertetabelle ist durch die Schaltung daneben realisiert.

Soll-Funktion

x_3	x_2	x_1	x_0	y
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0

x_3	x_2	x_1	x_0	y
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	0

Realisierung

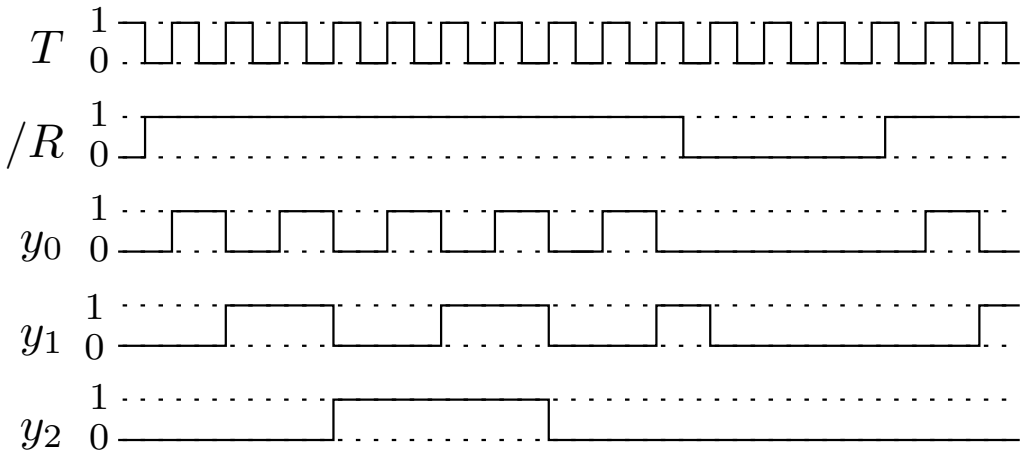


1. Stellen Sie die Wertetabelle der Realisierung auf. Für welche Eingaben weicht die Ausgabe vom Soll-Wert ab.
2. Verbessern Sie die Realisierung so, dass Sie für alle Eingaben richtige Ergebnisse liefert.

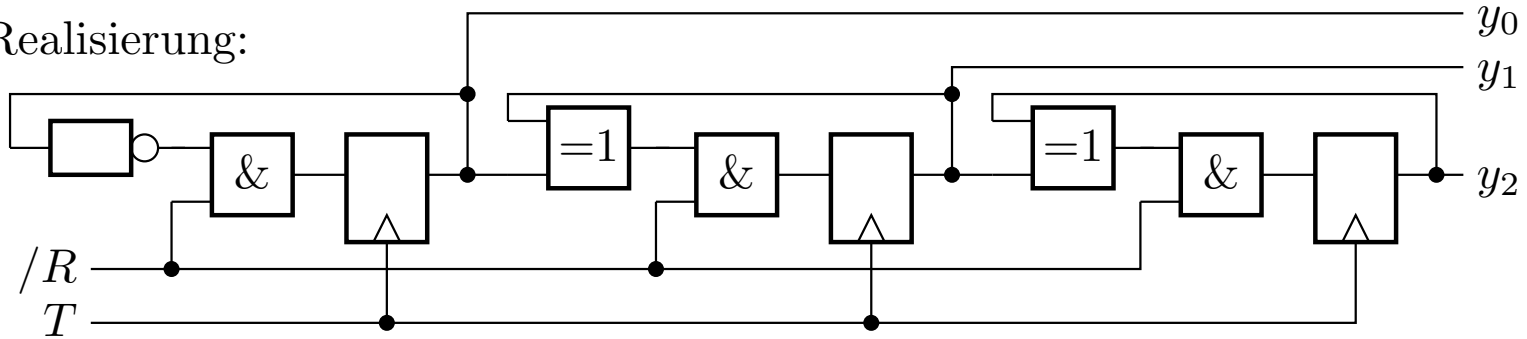
Aufgabe 1.7: Seqentielle Schaltung mit Fehler

Eine Schaltung soll bei $x = 0$ in den Zustand $\mathbf{y} = y_2 y_1 y_0 = 000$ übergehen und sonst bei jeder aktiven Taktflank seinen Wert um eins erhöhen (Binärzähler).

Testbeispiel mit Soll-Werten für die Ausgabe



Realisierung:



Gezeigt sind ein Testbeispiel mit Soll-Signalverläufen und eine fehlerhafte Realisierung.

1. Bestimmen Sie die tatsächlichen Ausgabesignalverläufe y_i für das Testbeispiel.
2. Korrigieren Sie die Schaltungsrealisierung so, dass sie das Testbeispiel richtig abarbeitet.

Aufgabe 1.8: Wurzelberechnung

Eine Service-Leistung sei definiert durch:

- Eingabeformat: uint16_t (16 Bit, vorzeichenfrei)
- Ausgabegabeformat: unint8_t (8 Bit, vorzeichenfrei)
- Soll-Funktion: Rückgabe der ganzzahligen Anteils der Wurzel
- Implementierung als C-Funktion:

```
uint8_t wurzel(uint16_t x){
 uint8_t w=0;
 uint16_t sum=0;
 while (sum<x){sum += (w<<1)+1;
 w++;}
 return w;
} <ausprobieren>
```

1. Mit welchen Eingaben ist der Fehler nachweisbar?
2. Ändern Sie das Programm so, dass es korrekt funktioniert.

Wahrscheinlichkeit

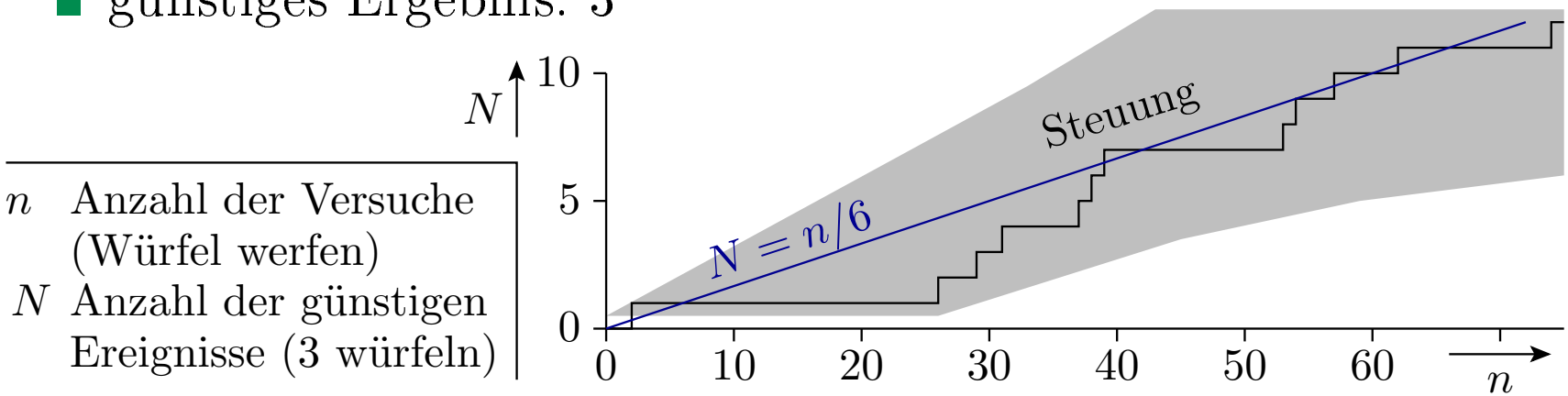
Die Wahrscheinlichkeit von Zufallsexperimenten

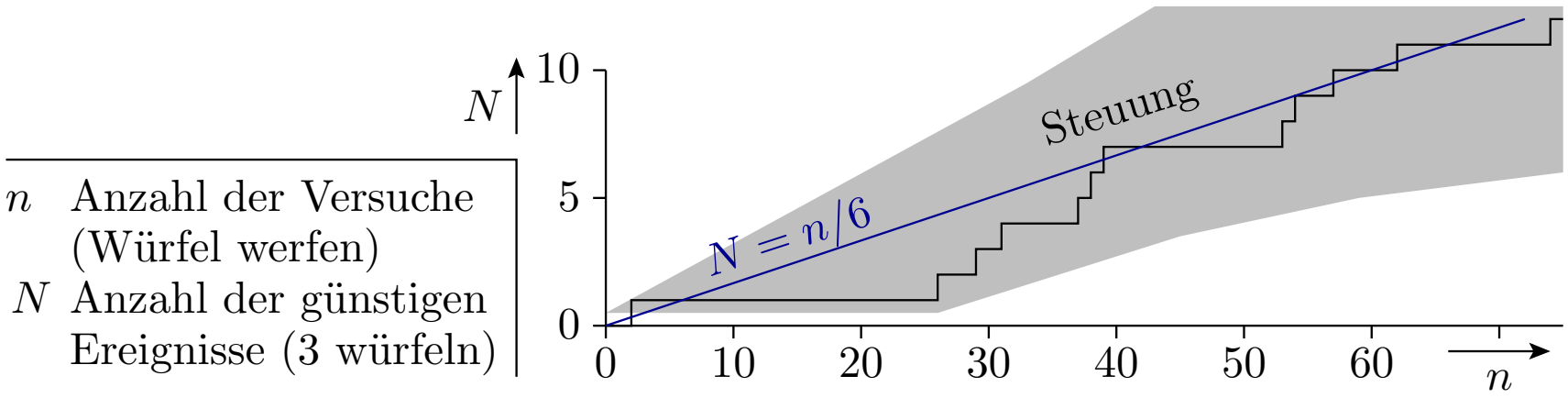
Definition 7

Wahrscheinlichkeit ist das Verhältnis, gegen das bei einem Zufallsexperiment die Anzahl der »günstigen« zur Anzahl aller möglichen Ereignisse mit zunehmender Versuchsanzahl strebt.

Wahrscheinlichkeit, dass eine 3 gewürfelt wird.

- Zufallsexperiment: Würfeln
- Mögliche Ergebnisse: 1, 2, ..., 6
- günstiges Ergebnis: 3





Beim Würfeln wird davon ausgegangen, das alle 6 Möglichkeiten gleichwahrscheinlich sind. Mit Versuchsanzahl $n \to \infty$ strebt das Verhältnis aus günstigen Ergebnisse N zur Versuchsanzahl gegen das Verhältnis aus möglichen günstigen und möglichen Ereignissen:

$$p = \lim_{n\to\infty} \left(\frac{N}{n} \right) = \frac{1}{6}$$

Das bedeutet aber keineswegs, dass bei jedem sechsten Versuch eine 3 gewürfelt wird. Es ist durchaus zu beobachten, dass hintereinander mehrere Dreien und auch mal lange Zeit keine Drei gewürfelt werden.

Aufteilen und verketteten von Experimenten

Zufallsexperimente lassen sich u.U. in mehrere Experimente aufteilen oder mehrere unabhängige Experimente zu einem zusammenfassen. Im nachfolgenden wird bei jedem Experiment zweimal gewürfelt (Ereignisse A und B, Wertebereich jeweils $\{1, 2, \ldots, 6\}$. Daraus werden mit Vergleichsoperatoren die zweiwertigen Ereignisse C und D gebildet und diese einmal UND- und einmal ODER verknüpft und gezählt.

n	1	2	3	4	5	6	7	8	9	10	11	12	13	...	20	...	40
A	6	1	5	4	1	1	2	2	4	6	4	3	1		6		5
B	6	5	6	2	1	3	3	6	4	5	1	3	1		4		3
$C = (A > 3)$	1	0	1	1	0	0	0	0	1	1	1	0	0		1		1
$D = (B < 3)$	0	0	0	1	1	0	0	0	0	0	1	0	1		0		0
$E = (C \wedge D)$	0	0	0	1	0	0	0	0	0	0	1	0	0		0		0
$F = (C \vee D)$	1	0	1	1	1	0	0	0	1	1	1	0	1		1		1
$\sum C$	1	1	2	3	3	3	3	3	4	5	6	6	6		11		21
$\sum D$	0	0	0	1	2	2	2	2	2	2	3	3	4		6		9
$\sum E$	0	0	0	1	1	1	1	1	1	1	2	2	2		5		6
$\sum F$	1	1	2	3	4	4	4	4	5	6	7	7	8		13		24

Nach 40 Versuchen betragen die Schätzwerte der Wahrscheinlichkeiten als Verhältnis der günstigen Ergebnisse, dass die Bedingungen C bis F erfüllt sind, zur Versuchsanzahl:

Ereignis	Schätzwert	Wahrscheinlichkeit
$C = (A > 3)$	$21/40 = 53\%$	$3/6 = 50\%$
$D = (B < 3)$	$9/40 = 23\%$	$2/6 = 33\%$
$E = (C \wedge D)$	$6/40 = 15\%$	$6/36 = 17\%$
$F = (C \vee D)$	$24/40 = 60\%$	$24/36 = 67\%$

Die Wahrscheinlichkeit als Grenzwerte für $n \to \infty$ ergibt sich für jeden Versuch aus dem Verhältnis der günstigen zur Anzahl der möglichen Ergebnisse. Die Würfelexperimente haben 6 mögliche Ergebnisse. Davon sind für die Ereignisse C und D 3 bzw. 2 günstig. Die verketten Ereignisse E und F haben $6^2 = 36$ mögliche Ergebnisse, von denen 6 bzw. 24 günstig sind.

Die Schätzung einer Wahrscheinlichkeit mit weniger als 100 günstigen Ereignissen ist recht ungenau.

Bedingte Wahrscheinlichkeiten

Bei einer bedingten Wahrscheinlichkeit werden nur die Versuche und Ereignisse gezählt, die die Bedingung erfüllen. Beispiel sei die ODER-Verknüpfung sich ausschließender Ereignisse:

$$E = C \vee D \text{ unter der Bedingung } C \wedge D = 0.$$

																				$\sum$	$\sum$
n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	20	16
C	1	0	1	1	0	0	0	0	1	1	1	0	0	1	1	0	1	0	1	1	11	7
D	0	0	0	1	1	0	0	0	0	0	1	0	1	0	1	0	1	0	0	0	6	2
$C \vee D$	1	0	1	1	1	0	0	0	1	1	1	0	1	1	1	0	1	0	1	1	13	9

nicht mitgezählte Ereignisse bzw. Summe ohne diese Ereignisse

Sowohl die Anzahl der gezählten Versuche als auch die günstigen Ergebnisse verringern sich um die vier nicht mitzuzählenden Ergebnisse mit $C \wedge D = 1$. Das undokumentierte Aussortieren ungewollter Ergebnisse ist eine unauffällige und beliebte Technik, Statistiken zu fälschen[6].

[6] Traue nie einer Statistik, die du nicht nicht selbst gefälscht hast.

Verkettete Ereignisse

Wahrscheinlichkeit verketteter Ereignisse

- Wahrscheinlichkeit, dass ein Ereignis A nicht eintritt:

$$P\left(\bar{A}\right) = 1 - P\left(A\right) \tag{1}$$

- Wahrscheinlichkeit, dass von zwei unabhängigen Ereignissen A und B alle eintreten:

$$P\left(A \wedge B\right) = P\left(A\right) \cdot P\left(B\right) \tag{2}$$

- Wahrscheinlichkeit, dass von mehreren unabhängigen Ereignissen mindestens eines eintritt:

$$\begin{aligned} P\left(A \vee B\right) = P\left(\overline{\bar{A} \wedge \bar{B}}\right) &= 1 - \left(1 - P\left(A\right)\right) \cdot \left(1 - P\left(B\right)\right) \\ &= P\left(A\right) + P\left(B\right) - P\left(A\right) \cdot P\left(B\right) \end{aligned} \tag{3}$$

Beispiel: Nachweis unabhängiger Fehler

In einem System mit drei Fehlern seien diese unabhängig voneinander mit den Nachweiswahrscheinlichkeiten $p_1 = 10\%$, $p_2 = 5\%$ und $p_3 = 20\%$ nachweisbar. Wie groß sind die Wahrscheinlichkeiten der verketten Ereignisse, dass

E_1 : alle Fehler,
E_2 : kein Fehler,
E_3 : mindestens ein Fehler und
E_4 : genau zwei Fehler nachgewiesen werden?

Lösung: Definition von Ereignissen F_i für Fehler i nachweisbar und Beschreibung von E_i durch logische Verknüpfungen:

- Alle Fehler nachweisbar:

$$\begin{aligned} E_1 &= F_1 \wedge F_2 \wedge F_3 \\ P(E_1) &= P(F_1) \cdot P(F_2) \cdot P(F_3) \\ &= p_1 \cdot p_2 \cdot p_3 = 10\% \cdot 5\% \cdot 20\% = 0{,}1\% \end{aligned}$$

- Kein Fehler nachweisbar:

$$\begin{aligned} E_2 &= \overline{F_1 \vee F_2 \vee F_3} \\ P(E_2) &= 1 - (1 - (1 - P(F_1)) \cdot (1 - P(F_2)) \cdot (1 - P(F_2))) \\ &= (1 - p_1) \cdot (1 - p_2) \cdot (1 - p_3) = 90\% \cdot 95\% \cdot 80\% = 68{,}4\% \end{aligned}$$

- Mindestens ein (nicht kein) Fehler nachweisbar:

$$\begin{aligned} E_3 &= \bar{E}_2 \\ P(E_3) &= 1 - P(E_2) = 1 - 68{,}4\% = 31{,}6\% \end{aligned}$$

- Genau 2 Fehlern werden nachgewiesen, wenn
  - die ersten beiden und der dritte nicht,
  - die zweiten beiden und der erste nicht oder
  - der erste und der dritte, aber nicht der zweite

  nachgewiesen werden (ausschließendes ODER, nächste Folie):

$$\begin{aligned} E_4 &= \left(F_1 \wedge F_2 \wedge \bar{F}_3\right) \vee \left(\bar{F}_1 \wedge F_2 \wedge F_3\right) \vee \left(F_1 \wedge \bar{F}_2 \wedge F_3\right) \\ P(E_4) &= p_1 \cdot p_2 \cdot (1 - p_3) + (1 - p_1) \cdot p_2 \cdot p_3 + p_1 \cdot (1 - p_2) \cdot p_3 \\ &= 90\% \cdot 5\% \cdot 20\% + 10\% \cdot 95\% \cdot 20\% + 10\% \cdot 5\% \cdot 80\% = 6{,}8\% \end{aligned}$$

Abhängige Ereignisse

Die Wahrscheinlichkeiten der UND- oder ODER-Verknüfung von abhängigen Ereignissen $A \wedge B$ und $A \vee B$ lassen sich nur aus den Wahrscheinlichkeiten $P(A)$ und $P(B)$ der Einzelereignisse bestimmen, wenn sich die Ereignisse ausschließen:

$$P(A \wedge B) = 0$$

Die Wahrscheinlichkeit, dass eines von beiden eintritt ist dann die Summe der Einzelwahrscheinlichkeiten:

$$P(A \vee B)|_{P(A \wedge B)=0} = P(A) + P(B) \tag{4}$$

Ist diese Voraussetzung nicht erfüllt, ist das Experiment so umformulieren, dass sich danach alle zu verkettenden Ereignisse gegenseitig ausschließen oder voneinander unabhängig sind.

Beispiel: voneinander abhängiger Fehlernachweis

Wie groß sind die Wahrscheinlichkeiten, dass von zwei Fehlern in einem System 0, 1 oder 2 Fehler nachweisbar sind, wenn die Nachweiswahrscheinlichkeit für Fehler 1 unabhängig vom Nachweis von Fehler 2 $p_1 = 10\%$ beträgt und für Fehler 2 bei Nachweis von Fehler 1 $p_2 = 20\%$ und sonst 0 beträgt. (Der Nachweis des zweiten Fehler hängt vom Nachweis des ersten ab.)

Lösung: Definition von Ereignissen F_i für Fehler i nachweisbar und E_i für i Fehler nachweisbar.

- Kein Fehler ist nachweisbar, wenn der erste Fehler nicht nachweisbar ist[7]:

$$\begin{aligned} E_0 &= \bar{F}_1 \\ P(E_0) &= 1 - P(F_1) = 1 - p_1 = 1 - 10\% = 90\% \end{aligned}$$

[7] Der Fall, Nachweis des zweiten ohne den ersten Fehler ist ausgeschlossen.

- Ein Fehler ist nachweisbar, wenn der erste Fehler nachweisbar ist und der zweite nicht:

$$\begin{aligned} E_1 &= F_1 \vee \bar{F}_2 \\ P(E_1) &= p_1 \cdot (1 - p_2) = 10\% \cdot 80\% = 8\% \end{aligned}$$

- Zwei Fehler sind nachweisbar, wenn beide Fehler nachweisbar sind:

$$\begin{aligned} E_2 &= F_1 \vee F_2 \\ P(E_2) &= p_1 \cdot p_2 = 10\% \cdot 20\% = 2\% \end{aligned}$$

- Probe: Summe der Wahrscheinlichkeiten aller möglichen Ergebnisse muss immer 100% sein:

$$90\% + 2\% + 8\% = 100\%\surd$$

Beispiel: Wahrscheinlichkeit einer Bedatung

Wie groß ist die Wahrscheinlichkeit, dass ein 8-Bit-Vektor für eine Service-Anfrage an eine Schaltung mit dem Wert $\mathbf{x} =$ "11111110" bedatet wird, wenn

1. unabhängig voneinander für jedes Bit mit einer Wahrscheinlichkeit[8] von $g = 50\%$ zufällig eine Eins und sonst eine Null gewählt wird.
2. Dasselbe wie im Aufgabenteil zuvor, nur mit $g = 60\%$.
3. Dasselbe wie in den Aufgabenteilen zuvor, nur dass für die höchtwertigen vier Bits immer derselben Zufallswert ausgewählt wird.

[8] Die Wahscheinlichkeit g wird auch als Wichtung der Bitstelle bezeichnet. Wichtung wird beim Test eingesetzt, um die Nachweiswahrscheinlichkeiten sehr schlecht nachweisbarer Fehler zu erhöhen.

Lösung: Definieren von Ereignissen G_i, dass für Bit i eine Eins ausgewählt wird. Für die beiden ersten Teilaufgaben gilt:

$$\begin{aligned} \mathbf{x} = \text{"11111110"} &= G_7 \wedge G_6 \wedge G_5 \wedge G_4 \wedge G_3 \wedge G_2 \wedge G_1 \wedge \bar{G}_0 \\ P(\mathbf{x} = \text{"11111110"}) &= g^7 \cdot (1-g) \end{aligned}$$

Für die letzte Teilaufgabe folgt aus $G_7 = G_6 = G_5 = G_4$:

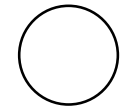
$$\begin{aligned} \mathbf{x} = \text{"11111110"} &= G_4 \wedge G_3 \wedge G_2 \wedge G_1 \wedge \bar{G}_0 \\ P(\mathbf{x} = \text{"11111110"}) &= g^4 \cdot (1-g) \end{aligned}$$

g	50%	60%
G_4 bis G_7 unabhängig	$2^{-8} \approx 0{,}004$	$0{,}6^7 \cdot 0{,}4 = 0{,}01$
$G_7 = G_6 = G_5 = G_4$	$2^{-5} \approx 0{,}03$	$0{,}6^4 \cdot 0{,}4 = 0{,}05$

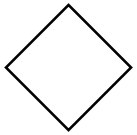
Fehlerbaumanalyse

Fehlerbaumanalyse (FTA – fault tree analysis)

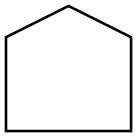
Die FTA dient zur Beschreibung von Fehlersituationen und zur Abschätzung deren Wahrscheinlichkeiten. Sie unterscheidet:



Basisereignisse, die nicht weiter untersucht werden, weil sie gut bekannt sind, oder Wahrscheinlichkeiten dafür existieren.



Nicht untersuchte Ereignisse.

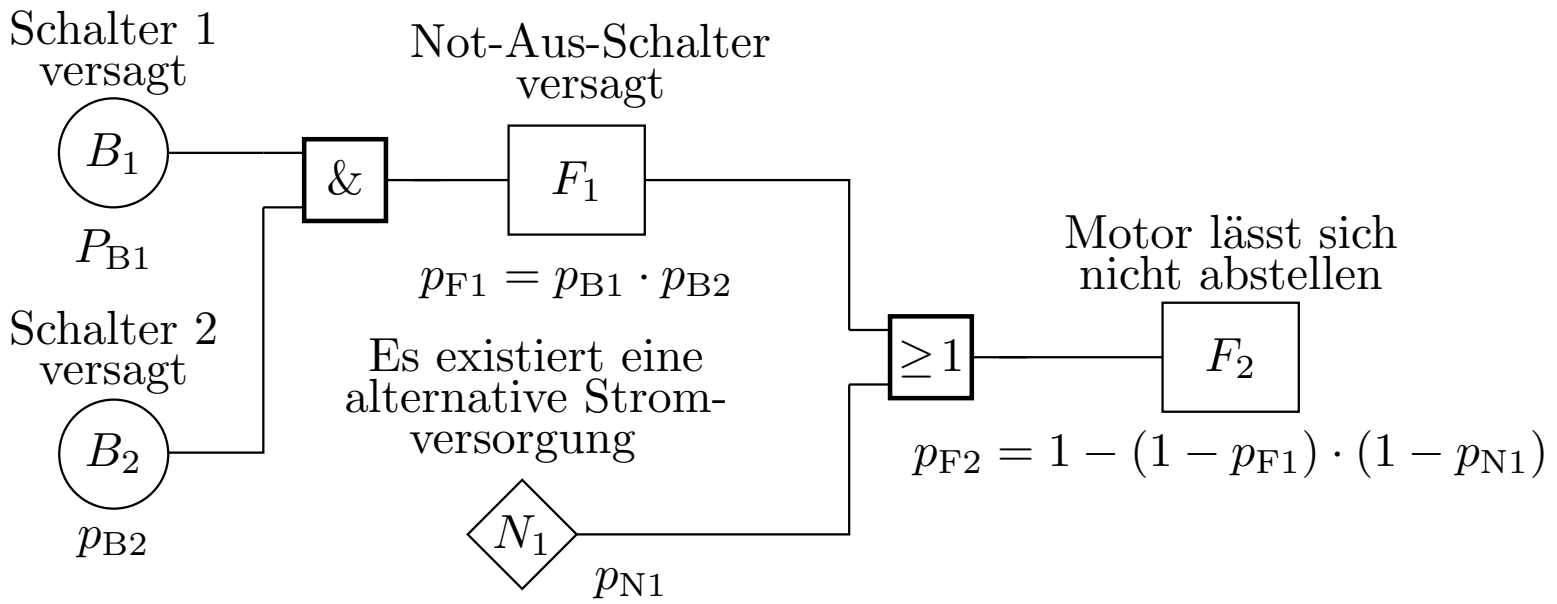


Ereignisse, die im gewöhnlichen Betrieb auftreten, aber in Kombination mit anderen Ereignissen Fehlerquelle sein können.

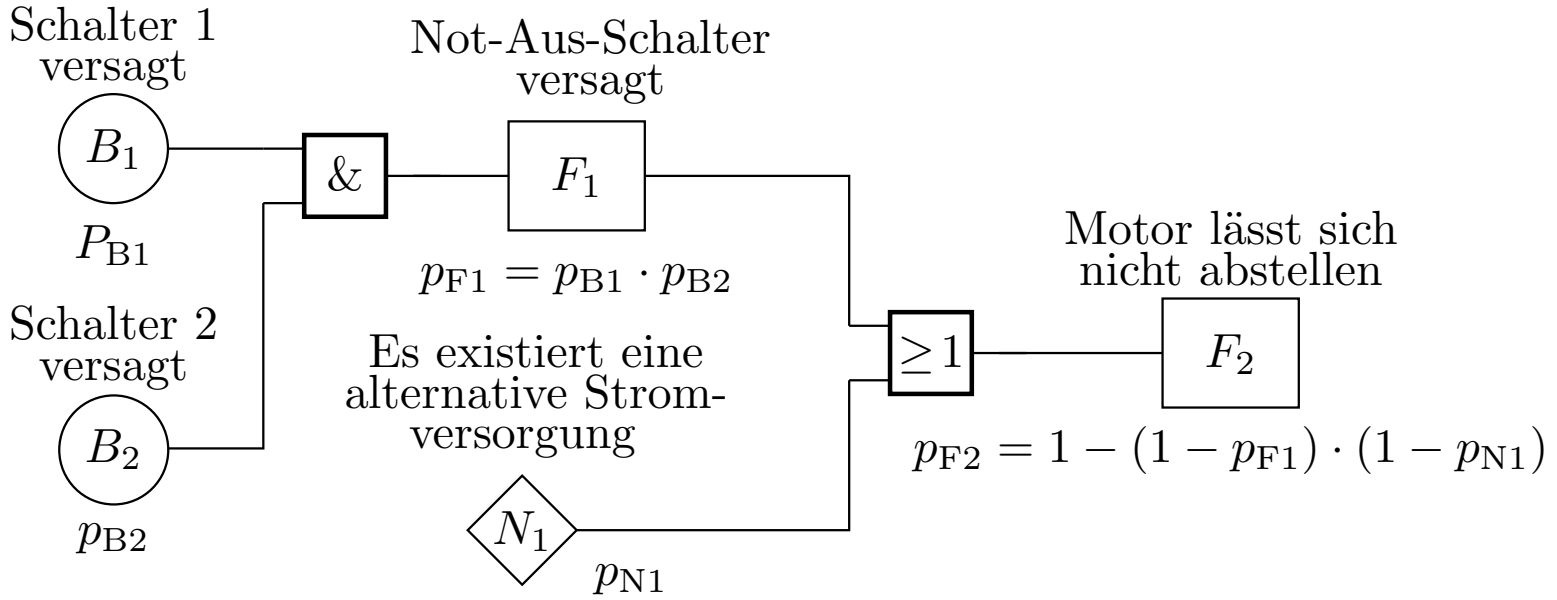
Fehlerereignisse, die sich nicht weiter aufteilen lassen.

Die Ereignisse können direkt oder als Komplementereignis UND und ODER verknüpft sein.

Beispiel: Motor lässt sich nicht abstellen



- Zusammenstellen und Klassifizierung der zu berücksichtigenden Ereignisse.
- Verknüpfung mit UND-, ODER-, NICHT
- Abschätzung der Wahrscheinlichkeit der verketteten Ereignisse nach den Gl. 1 bis 4:



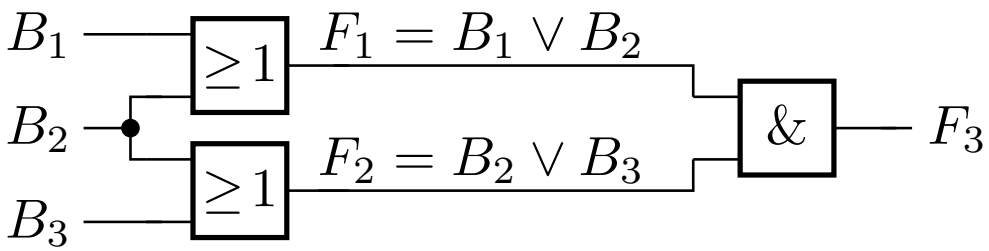
Angenommen, die Schalter versagen mit einer Wahrscheinlichkeit von 10^{-3} und die Wahrscheinlichkeit, dass eine alternative Stromversorgung existiert, ist nicht größer als 10^{-8}, ergibt sich für den betrachteten Fehlerfall eine Wahrscheinlichkeit:

$$p_{F2} < 10^{-3} \cdot 10^{-3} + 10^{-8} - 10^{-3} \cdot 10^{-3} \cdot 10^{-8} \approx 10^{-6}$$

Am Überschlags kann jetzt diskutiert werden, ob das Risiko von 10^{-6}, dass der Motor nicht abschaltet, akzeptiert wird oder weitere Maßnahmen ergriffen werden, z.B. dritter Schalter.

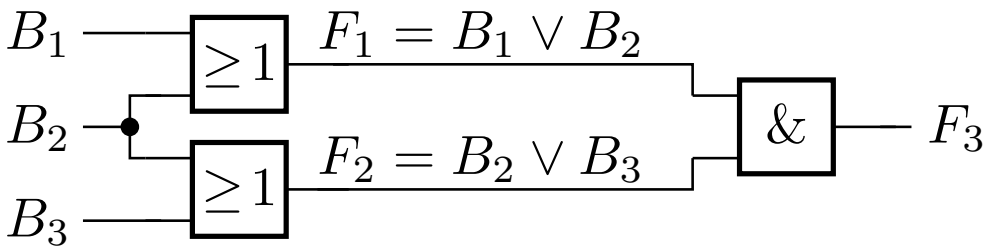
Fehlerbäume mit rekonvergenten Auffächerungen

- Rekonvergente Auffächerungen sind Schlaufen in gerichteten Graphen.
- An den Verzweigungen am Schlaufenbeginn werden abhängige Ereignisse gebildet und am Schlaufenende miteinander verknüpft.



- Im Beispiel sind die beiden UND-verknüpften Ereignisse beide von B_2 abhängig, schließen sich aber auch nicht aus, so das für die ODER-Verknüpfung weder Gl. 3 noch 4 gilt.

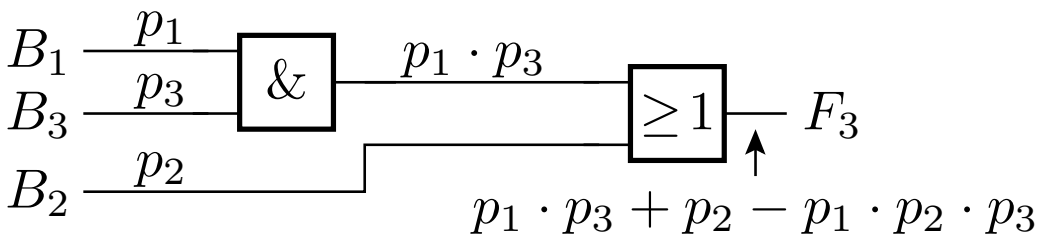
Beseitigung der Rekonvergenz durch Umformung



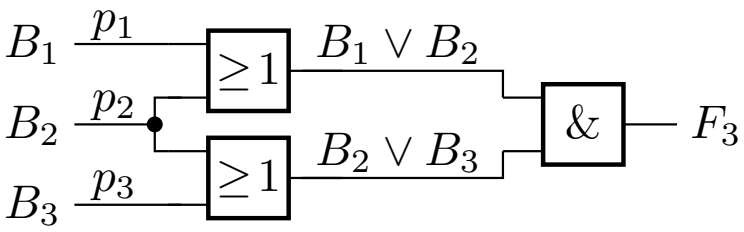
- Extraktion und Umformung/Vereinfachung des nachgebildeten logischen Ausdrucks:

$$(B_1 \vee B_2) \wedge (B_2 \vee B_3) = B_2 \vee (B_1 \wedge B_3)$$

- Funktionsgleicher rekonvergenzfreier Fehlerbaum:



Umformung über Wertetabelle



1 Ereignis eingetreten
0 Ereignis nicht eingetreten

B_1	B_2	B_3	F_3	Wahrscheinlichkeit
0	0	0	0	
0	0	1	0	
0	1	0	1	$(1-p_3)\cdot p_2\cdot(1-p_1)$
0	1	1	1	$+\quad(1-p_3)\cdot p_2\cdot p_1$
1	0	0	0	
1	0	1	1	$+\,(1-p_3)\cdot p_2\cdot(1-p_1)$
1	1	0	1	$+\quad p_3\cdot p_2\cdot(1-p_1)$
1	1	1	1	$+\quad p_3\cdot p_2\cdot p_1$

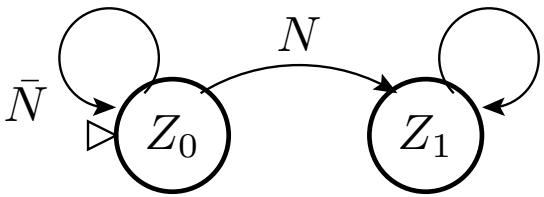
Jede logische Funktion ist durch eine Wertetabelle beschreibbar. Die Auswahlwahrscheinlichkeiten der Zeilen sind Wahrscheinlichkeitprodukte. Die gleichzeitige Auswahl mehrerer Zeilen ist ausgeschlossen, so dass die Gesamtwahrscheinlichkeit die Summe der Wahrscheinlichkeitsprodukte der »günstigen Eingabezeilen« ist.

Markov-Ketten

Markow-Ketten[9]

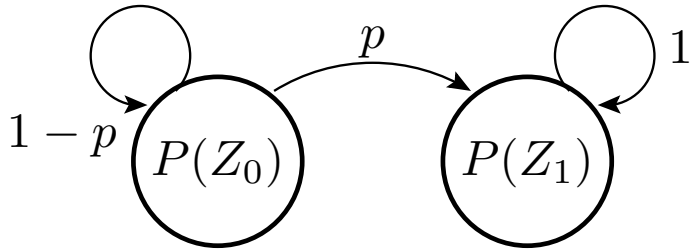
Modellierung eines stochastischen Prozesses durch einen Zustandsautomaten, dessen Kanten mit Übergangswahrscheinlichkeiten beschriftet sind. Ein einfaches Beispiel ist die Beschreibung des Nachweises eines Fehlers mit n Service-Aufrufen. Das System habe die Zustände Z_0 (Fehler nicht nachgewiesen) und F_1 (Fehler nachgewiesen).

Automat zur Beschreibung des Fehlernachweise



Z_i	Zustände
N	Nachweisereignis
$\triangleright$	Anfangszustand

Markov-Kette zum Automaten



$P(Z_i)$	Zustandswahrscheinlichkeiten
p	Nachweiswahrscheinlichkeit
	Anfangswert: $P(Z_0) = 1$

[9] Nach Andrej Andreevič Markov, russischer Mathematiker, 1856-1922.

Eine Markov-Kette ist ein lineares System, das sich auch als Matrixgleichung beschreiben lässt. Rekursive Form:

$$\begin{pmatrix} P(Z_0) \\ P(Z_1) \end{pmatrix}_{i+1} = \begin{pmatrix} 1-p & 0 \\ p & 1 \end{pmatrix} \cdot \begin{pmatrix} P(Z_0) \\ P(Z_1) \end{pmatrix}_i$$

Die Summe der Zustandswahrscheinlichkeiten und die Summe der Übergangswahrscheinlichkeiten aus einem Zustand (in einer Spalte) muss immer eins sein. Die Zustandswahrscheinlichkeiten nach n Service-Aufrufen betragen im Beispiel:

$$\begin{pmatrix} P(Z_0) \\ P(Z_1) \end{pmatrix}_n = \begin{pmatrix} 1-p & 0 \\ p & 1 \end{pmatrix}^n \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Die Wahrscheinlichkeit, dass ein Fehler mit n Service-Aufrufen nachgewiesen wird, ist die, dass er nicht mit keinem nachgewiesen wird:

$$p(n) = 1 - (1-p)^n$$

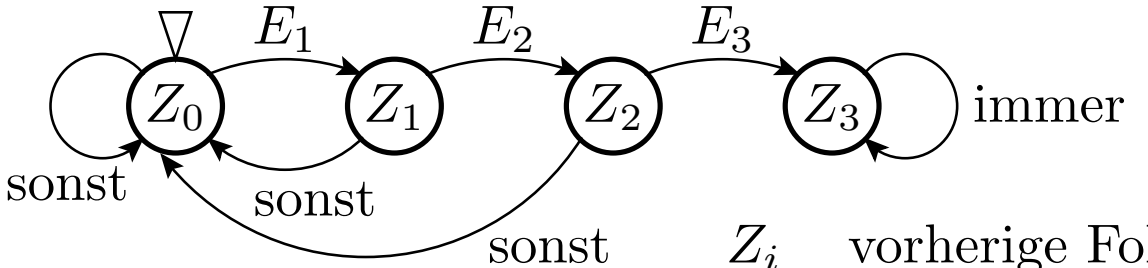
Unsere spezielle Markov-Kette geht in n-Schritten mit dieser Wahrscheinlichkeit von Zustand Z_0 (Fehler nicht nachgewiesen) nach Z_1 (Fehler nachgewiesen) und verbleibt dort:

$$\begin{pmatrix} Z_0 \\ Z_1 \end{pmatrix}_n = \begin{pmatrix} (1-p)^n & 0 \\ 1-(1-p)^n & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Weiteres Beispiel: Schätzen der Wahrscheinlichkeit, dass zufällig eine bestimmte Datenfolge erzeugt wird

Ein Zufallsgenerator erzeugt die Testeingabewerte X_i jeweils mit einer Wahrscheinlichkeit p_i. Wie groß ist die Wahrscheinlichkeit, dass in einer Folge der Länge n in drei aufeinanderfolgenden Schritten die Teilfolge $X_1X_2X_3$ enthalten ist?

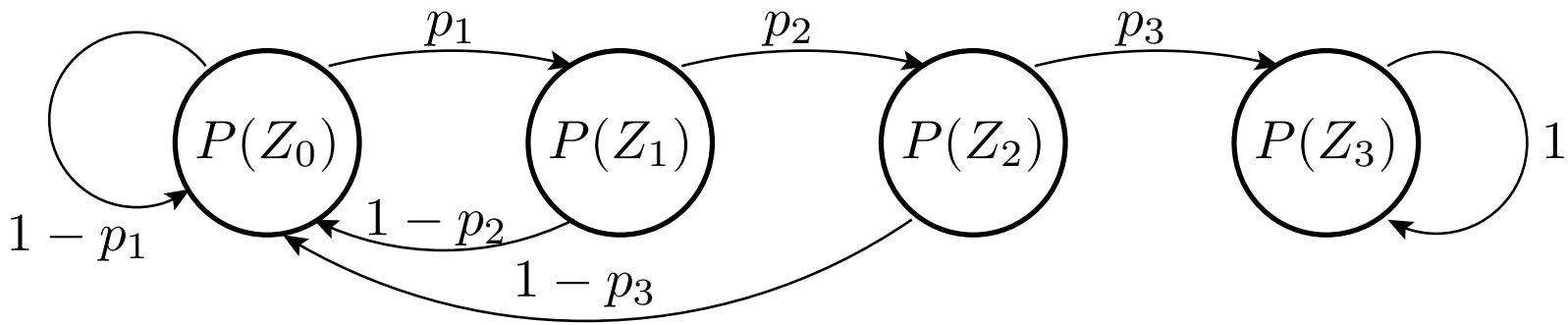
- Entwurf eines Akzeptorautomaten, der in einem Anfangszustand startet und bei Erkennen der Folge $X_1X_2X_3$ in einen Endzustand übergeht.



Z_i vorherige Folge bestand aus den ersten i richtigen Werten

E_i Wert ist X_i

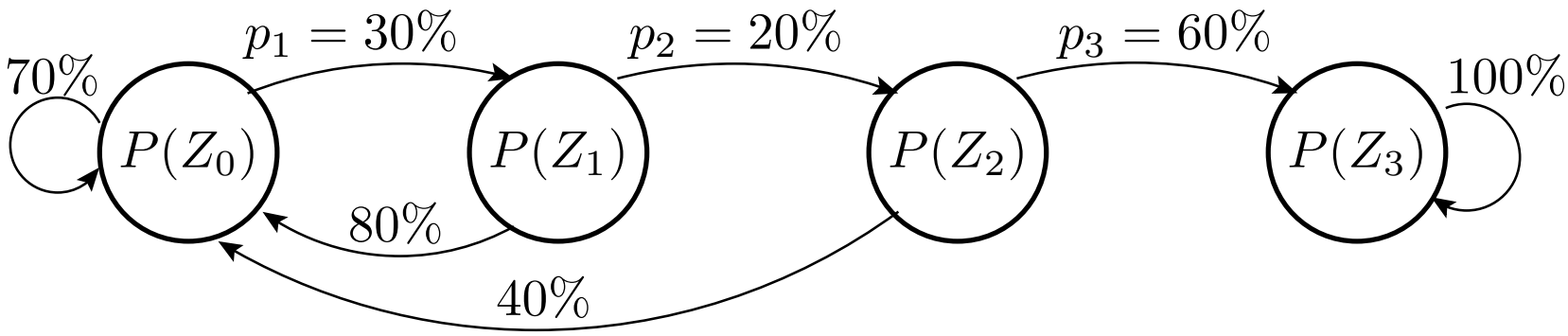
- Ersatz der Übergangsbedingungen durch die Übergangswahrscheinlichkeiten $p_i = P(E_i)$ und der Zustände durch Zustandswahrscheinlichkeiten $P(Z_i)$.



- Simulation mit den Startwert Z_0 ($P(Z_0) = 1$ und $P(Z_{i\neq 0}) = 0$) für n Schritte:

$$\begin{pmatrix} P(Z_0) \\ P(Z_1) \\ P(Z_2) \\ P(Z_3) \end{pmatrix}_n = \begin{pmatrix} 1-p_1 & 1-p_2 & 1-p_3 & 0 \\ p_1 & 0 & 0 & 0 \\ 0 & p_2 & 0 & 0 \\ 0 & 0 & p_3 & 1 \end{pmatrix}^n \cdot \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Mit Beispielwerten für p_1 bis p_3:



Schritt	$P(Z_0)$	$P(Z_1)$	$P(Z_2)$	$P(Z_3)$	Summe
0	100,00	0,00	0,00	0,00	100,00
1	70,00	30,00	0,00	0,00	100,00
2	73,00	21,00	6,00	0,00	100,00
3	70,30	21,90	4,20	3,60	100,00
4	68,41	21,09	4,38	6,12	100,00
...	...	...	...	...	...
10	59,43	18,34	3,77	18,46	100,00
...	...	...	...	...	...
50	19,27	5,95	1,22	73,56	100,00
...	...	...	...	...	...
100	4,73	1,46	0,30	93,53	100,00

Redundantes Master-Checker System

Ein Rechnersystem für höchte Sicherheitsanforderungen soll aus vier Rechnern bestehen, einem Master, der rechnet, einem Checker, der die Ergebnisse kontrolliert, und zwei Reserverechner, die bei Ausfall die Service-Anforderungen des Masters oder des Checkers übernehmen. Das System startet mit Rechner 1 als Master und Rechner 2 als Checker und gilt solange als funktionsfähig, wie noch ein Master und ein Checker funktionieren. Die Ausfallwahrscheinlichkeit p der Rechner je Service-Abforderung sei 10%.

1. Beschreiben Sie dem Sachverhalts durch eine Markov-Kette in Matrixform.
2. Erweiterung der Fehlerreaktion um einen Reparaturprozess, in dem jeder defekte Rechner während, Master und Checker eine Service-Anfage abarbeitet, mit einer Wahrscheinlichkeit von 20% repariert wird.

Die relevanten Systemzustände sind 0, 1, 2 und mehr als 2 Rechner ausgefallen (Z_0 bis Z_3). Ohne Reparatur erhöht sich die Anzahl der ausgefallenen Rechner

- um zwei, wenn Master und Checker gleichzeitig ausfallen $p_2 = p^2 = 1\%$,
- um eins, wenn entweder der Master oder der Checker ausfällt, $p_1 = 18\%$ oder
- um keinen, wenn weder Master noch Checker ausfällt, $p_0 = (1-p)^2 = 81\%$:

$$\begin{pmatrix} P(Z_0) \\ P(Z_1) \\ P(Z_2) \\ P(Z_3) \end{pmatrix}_n = \begin{pmatrix} 81\% & 0\% & 0 & 0 \\ 18\% & 81\% & 0 & 0 \\ 1\% & 18\% & 81\% & 0 \\ 5\% & 1\% & 19\% & 100\% \end{pmatrix}^n \cdot \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Mit Reparatur können bis zu zwei fehlerhafte Rechner hinzukommen und max. alle zuvor fehlerhaften Rechner repariert werden. Das wird recht kompliziert ...(Fortsetzung als Aufgabe 3.8).

Aufgaben

Aufgabe 2.1: Wahrscheinlichkeiten von Würfelexperimenten

X und Y seien die zufälligen Augenzahlen bei der Durchführung des Versuchs »Würfeln mit zwei Würfeln«. Berechnen Sie die Wahrscheinlichkeiten folgender Ereignisse:

1. $X + Y > 8$
2. $X > Y$
3. $(X = 5) \wedge (Y < 5)$
4. $X \cdot Y$ ist durch drei teilbar.

Geben Sie jeweils die Anzahl der möglichen Ereignisse an und zählen Sie die günstigen Ereignisse auf.

Aufgabe 2.2: Verkettete Würfelereignisse

- Welche möglichen Ergebnisse hat das Zufallsexperiment »auswürfeln einer Zahl, bei einer Sechs darf ein zweites Mal gewürfelt werden«?
- Mit welcher Wahrscheinlichkeit tritt jedes der möglichen Ergebnisse ein?

Aufgabe 2.3: Fehlfunktionen und Fehlernachweis

Ein System habe vier unabhängig voneinander nachweisbare Fehler mit den Nachweiswahrscheinlichkeiten je Service-Aufruf von $p_1 = 10\%$, $p_2 = 20\%$, $p_3 = 5\%$ und $p_4 = 1\%$.

1. Mit welcher Wahrscheinlichkeit versagt eine einzelne Service-Anforderung.
2. Wie hoch ist die Wahrscheinlichkeit, dass zehn aufeinanderfolgende Service-Anforderungen korrekt ausgeführt werden.
3. Wie groß ist die Wahrscheinlichkeit für jeden der vier Fehler, dass er bei einem der zehn aufeinanderfolgenden Service-Aufrufe nachgewiesen wird (mindestens ein Versagen verursacht).

Aufgabe 2.4: Erstellen eines Fehlerbaums

Herr M. möchte um Mitternacht in seinem Büro einen Bericht lesen. Er muss dazu in sein Büro, braucht Licht und eine Brille. Ereignisse (B_i Basisereignisse ; N_i nicht untersuchte Ereignisse; F_i Fehlerereignisse):

- B_1 Tür klemmt, $p_{\mathrm{B1}} = 0,1\%$
- B_2 Deckenlampe defekt, $p_{\mathrm{B1}} = 0,2\%$
- B_3 Tischlampe defekt, $p_{\mathrm{B1}} = 0,2\%$
- B_4 Lesebrille defekt, $p_{\mathrm{B1}} = 0,3\%$
- B_5 Ersatzbrille defekt, $p_{\mathrm{B1}} = 0,5\%$
- N_1 Schlüssel vergessen, p_{N1} unbekannt
- N_2 Lesebrille vergessen, p_{N2} unbekannt
- N_3 Ersatzbrille im Schreibtisch eingeschlossen, p_{N3} unbekannt
- F_1 Büro verschlossen

- F_2 Büro unbeleuchtet
- F_3 Keine Brille
- F_4 Bericht ungelesen

1. Stellen Sie den Fehlerbaum auf.
2. Schätzen Sie die Wahrscheinlichkeiten der Fehlerereignisse F_1 bis F_4 unter der Annahme, dass die Wahrscheinlichkeiten der unberücksichtigten Ereignisse nicht größere als 1% sind.

Fehlerbaumanalyse 2

1. Entwickeln Sie den Fehlerbaum für folgenden Zusammenhang:
   - Ereignis F_1 tritt ein, wenn entweder B_1 und nicht B_2 oder nicht B_1 und B_2 eintritt.
   - Das Ereignis F_2 tritt nur ein, wenn F_1 und B_3 eintreten.

2. Berechnen Sie die Wahrscheinlichkeit für F_1 und F_2 für den Fall, dass die Wahrscheinlichkeiten der Basisereignisse $p_{\mathrm{B1}} = 2\%$, $p_{\mathrm{B2}} = 10\%$ und $p_{\mathrm{B3}} = 5\%$ betragen.

Aufgabe 2.5: Übertragungsfehler

Bei der Übertragung von vier möglichen Zeichen A, B, C und D betrage die Wahrscheinlichkeit, das ein Zeichen in eines der drei anderen verfälscht wird, je $p_F = 5\%$. Die Wahrscheinlichkeit, dass es unverfälscht übertragen wird, ist $p_\mathrm{U} = 1 - 3 \cdot p_\mathrm{F} = 85\%$:

$$\begin{pmatrix} P(A) \\ P(B) \\ P(C) \\ P(D) \end{pmatrix}_{i+1} = \begin{pmatrix} 85\% & 5\% & 5\% & 5\% \\ 5\% & 85\% & 5\% & 5\% \\ 5\% & 5\% & 85\% & 5\% \\ 5\% & 5\% & 5\% & 85\% \end{pmatrix} \cdot \begin{pmatrix} P(A) \\ P(B) \\ P(C) \\ P(D) \end{pmatrix}_{i}$$

1. Stellen Sie den Zusammenhang als Markov-Kette dar.
2. Bestimmen Sie die Wahrscheinlichkeit, dass ein »A« nach der 5. Übertragung immer noch ein »A« ist.

Aufgabe 2.6: Risikoanalyse

Eine schwerwiege Fehlfunktion bei einer Maschine kann nur auftreten, wenn sie vom Nomalzustand Z_0 nacheinander in höhere Riskozustände Z_1 bis Z_4 übergeht. Das Bedienpersonal erkennt erhöhte Risikozustände mit einer Wahrscheinlichkeit $p_1 = 80\%$ und initialisiert das System dann neu (Rückkehr in den Grundzustand Z_0). Die Wahrscheinlichkeit für den Übergang von einem in den nächsten Risikozustand betrage in jedem Zeitschritt, wenn nicht neuinitialisiert wird, $p_2 = 10\%$. In Risikozustand Z_4 tritt ohne rechtzeitige Neuinitialisierung mit $p_3 = 5\%$ die schwerwiegende Fehlersituation ein.

1. Beschreiben Sie den Sachverhalt mit einer Markov-Kette und einer Matrixgleichung zur Simulation mit Matlab.
2. Bestimmen Sie für 100 Schritte die Zustandswahrscheinlichkeiten $P(Z_0)$ bis $P(Z_4)$ und die Wahrscheinlichkeit, dass die schwerwiegende Fehlersituation eintritt.

Aufgabe 2.7: Verfügbarkeit

Für eine stark ausfallgefährdete Rechnerbaugruppe, die eine hohe Verfügbarkeit haben muss, hat der Hersteller drei Ersatzkomponenten dazugestellt, von denen jede bei Ausfall der aktuell genutzten Komponente die Aufgabe übernehmen kann. Bei jeder Service-Anfrage betrage die Ausfallwahrscheinlichkeit der genutzten Komponente $p_A = 10\%$ und die der Ersatzkomponenten null. Die defekten Ersatzpomponenten werden während der Dauer jeder Service-Anfrage mit einer Wahrscheinlichkeit von $p_R = 8\%$ repariert. Wenn mehrere kaputt sind, wird jede mit einer Wahrscheinlichkeit p_R repariert.

1. Beschreiben Sie den Sachverhalt mit einer Markov-Kette mit der Anzahl der defekten Komponenten als Zustandsnummer.
2. Stellen Sie die Übergangsmatrix auf.
3. Bestimmen Sie durch Simulation die Wahrscheinlichkeit, dass der Service der Rechnerbaugruppe verfügbar ist.

Aufgabe 2.8: Master-Checker-System mit Reparatur

Stellen Sie für das Beispiel auf Folie 69 die Übergangsmatrix für den Fall auf, dass nicht nur Master und Checker mit einer Wahrscheinlichkeit von 10% ausfallen, sondern dass auch jeder defekte Rechner mit 20% Wahrscheinlichkeit repariert wird.

1. Erweiterung um einen Zustand Z_4 (4 Rechner ausgefallen).
2. Bestimmen Sie für die Zustände Z_0 bis Z_4, wie viele Rechner in Summe ausfallen und repariert werden können, und die zugehörigen Übergangswahrscheinlichkeiten dafür.
3. Schreiben Sie ein Matlab-Programm zur Simulation dieser Markov-Kette.
4. Bestimmen Sie die Zustandswahrscheinlichkeiten nach 100, 200 und 1000 Service-Anforderungen. Gegen welche Wahrscheinlichkeit strebt die Wahrscheinlichkeit, dass der Service verfügbar, d.h. die Markov-Kette in einem der Zustände Z_0, Z_1 oder Z_2 ist.

Spezielle Experimente

Experimente zur Bewertung der Verlässlichkeit

Der Schlüssel zu objektiv verlässlichen Systemen sind Kontrollen und das Abstellen der dabei erkannten Mängel auf drei Ebenen:

- während Entwurf und Fertigung (Fehlervermeidung),
- vor dem Einsatz und zur Wartung (Fehlerbeseitigung) und
- im laufenden Betrieb (Fehlertoleranz, Schadensvermeidung).

In diesem Abschnitt werden spezielle Experiment für

- das Versagen von Service-Leistungen,
- Kontrollen der Service-Ergebnisse,
- Tests zur Fehlererkennung und -klassifizierung und
- die Prozesse der Fehlerentstehung

beschrieben. Diese Modelle und die mit ihnen abschätzbaren Wahrscheinlichkeiten dienen im Weiteren für Abschätzungen der Risiken, dass IT-Systeme im Einsatz versagen, Schaden verursachen, zur Planung der erforderlichen Tests und Kontrollen, ...

Service-Versagen

Erfolg und Versagen einer Service-Leistung

Ein IT-System kann auf eine Service-Anfrage reagieren mit

- einem richtigen Ergebnis,
- einem fehlerhaften Ergebnis oder
- keinem Ergebnis.

Modellierung durch ein Zufallsexperiment:

Anfragenr.	1	2	3	4	5	6	7	8	9	10	...
Ergebnis	K	K	K	N	K	K	F	K	F	K	...

K Ergebnis korrekt
N kein Ergebnis
F Ergebnis fehlerhaft
n Anzahl der Service-Anfragen

Fehlerbehandlung nach jedem Versagen

Nach einem Versagen ist im allgemeinen eine Fehlerbehandlung erforderlich (Neuinitialisierung, Wiederholung, ...). Die Schritte und Service-Anforderungen zur Fehlerbehandlung, das können sehr viele sein, zählen im betrachteten Experiment nicht mit.

Wahrscheinlichkeiten für Erfolg und Versagen

Anfragenr.	1	2	3	4	5	6	7	8	9	10	...
Ergebnis	K	K	K	N	K	K	F	K	F	K	...

Fehlerbehandlung nach jedem Versagen

K Ergebnis korrekt
N kein Ergebnis
F Ergebnis fehlerhaft
n Anzahl der Service-Anfragen

Wahrscheinlichkeit für

- eine korrekte Service-Leistung

$$p_{\mathrm{SK}} = \lim_{n\to\infty}\left(\frac{\mathrm{Anz}\,(K)}{n}\right)$$

- Service nicht verfügbar

$$p_{\mathrm{SN}} = \lim_{n\to\infty}\left(\frac{\mathrm{Anz}\,(N)}{n}\right)$$

- Service-Leistung fehlerhaft (aber verfügbar).

$$p_{\mathrm{SF}} = \lim_{n\to\infty}\left(\frac{\mathrm{Anz}\,(F)}{n}\right)$$

Ein Simulationsexperiment dazu

Die nachfolgende Beispielsimulation berechnet für jedes Versagen die Anzahl der Service-Anforderungen bis dahin ($p_{\mathrm{SF}} = 1\%$).

n	N	Service-Leistungen bis zum Versagen
75	1	75
258	2	183
359	3	101
391	4	32
469	5	78
562	6	93
687	7	125
738	8	51
774	9	36
797	10	23

Schätzwert $p = \frac{10}{797} \approx 1,25\%$

Simulationsprogramm

```
import random;
rand=random.random;
for i in range(10):
 ct=0;
 while rand()>0.01:
  ct = ct+1;
 print i, ct;
```

n Service-Anforderungen
N davon haben versagt

Wegen $p_{\mathrm{SF}} = 1\%$ strebt $\frac{N}{n}$ für $n \to \infty$ Service-Anforderungen gegen 1%.

Art des Versagens und Fehlerbehandlung

Ein Versagen (kein oder ein falsches Ergebnis) kann unterschiedliche Ursachen haben, die unterschiedliche Arten der Fehlerbehandlung erfordern.

Ursachen für das Versagen	Fehlerbehandlung
Störung ohne Zustandskontaminierung	Wiederholung
Störung mit Zustandskontaminierung	Neuinitialisierung und Wiederholung
Umgehbarer Fehler	Anfrage-Umformulierung[10]
zu beseitigender Fehler, Ausfall	Reparatur, Ersatz

[10] Anfrage-Umformulierung (input work-around): Geänderte Anfrage, die im fehlefreien Fall dasselbe Ergebnis liefert (anderen Bedatung, andere Schnittstellen, andere Ausführungsreihenfolge, ...)

Klassifikation der Ursache für das Versagen

Die Tabelle auf Folie zuvor ist so ausgelegt, dass sich jedem Versagen über den Fehlerbehandlungsablauf eine der aufgelisteten Ursachen zuordnen lässt.

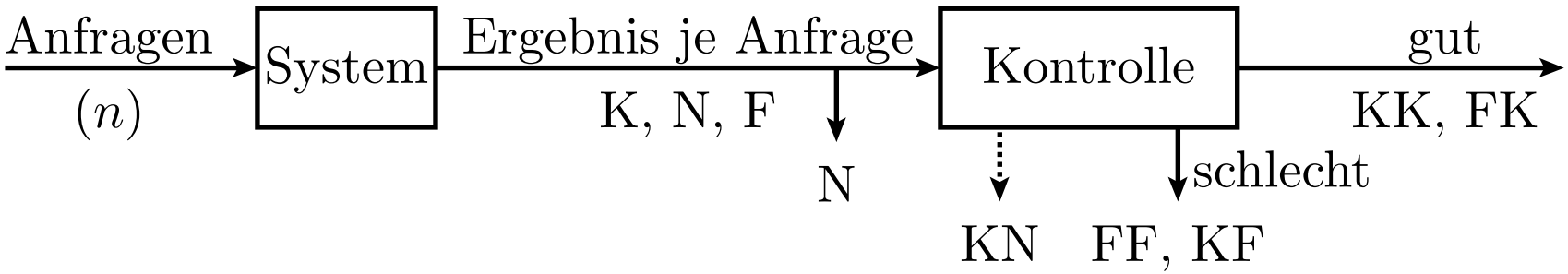
- Mehrfache Wiederholung ohne Neuinitialisierung:
  Wenn sich das Fehlverhalten nicht wiederholt, war die Ursache wahrscheinlich eine Störung, die keine Zustände kontaminiert hat.
  Wenn in einem System ohne Gedächtnis immer dieselbe Fehlfunktion beobachtete wird, ist die Ursache vermutlich ein permanenter Fehler ohne Speicherwirkung.
- Mehrfache Wiederholung mit Neuinitialisierung:
  Wenn sich das Fehlverhalten jetzt nicht mehr wiederholt, war die Ursache wahrscheinlich eine Störung, die Zustände kontaminiert hat.

- ...
  Bei wiederholt gleicher Fehlfunktion ist die Ursache vermutlich ein beständiger Fehler und bei wechselnder Fehlfunktion ein unbeständiger Fehler.
- Anfrageumformulierung:
  Bei größeren Software-Systemen gibt es viele Möglichkeiten, die Service-Anforderung für ein gewünschtes Ergebnis zu stellen (über unterschiedliche Menüs, Shortcuts, mehre Bedatungsmöglichkeiten). Wenn das hilft, ist es ein umgehbarer Fehler, d.h. ein Fehler, der die Systemnutzung beeinträchtigt, aber nicht in Frage stellt.
- Reparatur, Ersatz:
  Wenn nur das hilft und dieselben Service-Anfragen bis dahin nicht versagt haben, kann auf ein Ausfall geschlussfolgert werden.

Kontrolle als Filter

Kontrolle der Service-Ergebnisse

Wenn ein Service kein Ergebnis liefert, wird das immer erkannt.
Ein falsches Ergebnis erfordert eine zusätzliche Kontrolle.



Eine Kontrolle liefert ein zweiwertiges Ergebnis (gut oder schlecht), hat zwei Möglichkeiten richtig zu reagieren und drei Möglichkeiten zu versagen. Jeder der fünf Möglichkeiten lässt sich eine Wahrscheinlichkeit zuordnen:

- p_{KK} Klassifizierung korrekter Ergebnisse als korrekt,
- p_{FF} Klassifizierung fehlerhafter Ergebnisse als fehlerhaft.
- p_{KN} Kontrolle nicht verfügbar. Kein Klassifizierung.
- p_{FK} Klassifizierung fehlerhafter Ergebnisse als korrekt.

- p_{KF} Klassifizierung korrekter Ergebnisse als fehlerhaft.[11]

Das Zufallsexperiment »Klassifizierung des Service-Ergebnisses anhand des Kontrollergebnisses« hat vier mögliche Ergebnisse:

- Service-Leistung als korrekt klassifiziert (Fall K),
- Service-Leistung als fehlerhaft klassifiziert (Fall F),
- kein Kontrollergebnis (Fall KN) und
- und kein Service-Ergebnis (Fall N).

Das Kontrollergebnis »Service korrekt« entsteht bei korrekter Ausführung und korrekter Klassifikation oder fehlerhafter Ausführung und fehlerhafter Klassifikation:

$$p_{\mathrm{K}} = p_{\mathrm{SK}} \cdot p_{\mathrm{KK}} + p_{\mathrm{SF}} \cdot p_{\mathrm{FK}} \tag{5}$$

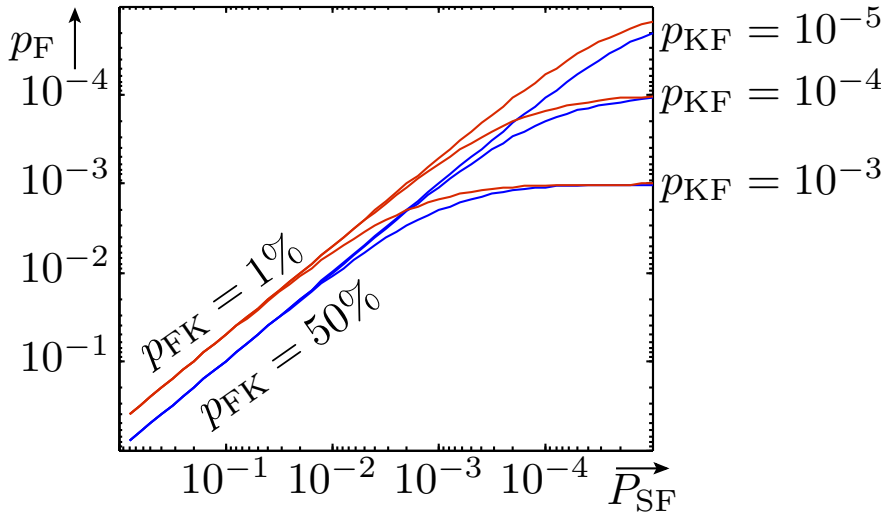
Das Kontrollergebnis »Service fehlerhaft« entsteht bei fehlerhafter Ausführung und korrekter Klassifikation oder korrekter Ausführung und fehlerhafter Klassifikation:

$$p_{\mathrm{F}} = p_{\mathrm{SF}} \cdot p_{\mathrm{FF}} + p_{\mathrm{SK}} \cdot p_{\mathrm{KF}} \tag{6}$$

[11] In der Messtechnik wird der Klassifikationsfehler FK als Messfehler 1. Art der Klassifikationsfehler KF als Messfehler 2. Art bezeichnet.

Ein Service im Einsatz liefert fast immer richtige Ergebnisse $p_{\mathrm{SK}} \approx 1$ und die meisten fehlerhaften Ergebnisse werden erkannt ($p_{\mathrm{FF}} \approx 1$). Die Wahrscheinlichkeit, dass eine Fehlfunktion signalisiert wird, ist etwa die Summe der Wahrscheinlichkeiten, dass der Service versagt und dass die Kontrollfunktion richtige Ergebnisse als falsch klassifiziert:

$$p_{\mathrm{F}} \approx p_{\mathrm{SF}} + p_{\mathrm{KF}}$$



Bei sehr seltenem Versagen sind fast alle ausgewiesen Fehlfunktionen Fehlklassifikationen.

Phantomfehler

Fehlklassifikationen korrekter Ergebnisse als fehlerhaft führen dazu, dass für das System Phantomfehler diagnostiziert werden. Das sind »erkannte« Fehler, die Wirklichkeit keine sind. Phantomfehler lösen Fehlerbehandlungsmaßnahmen aus. Außer unützen Aufwand kann es passieren, dass das System dabei kaputt repariert wird (siehe später Foliensatz #).

Wenn nur sehr selten wirkliche Fehler oder Gefahrensituationen auftreten, sind die meisten diagnostizierten Fehler Phantomfehler.

Jeder im Institut für Mathematik bisher ausgelöste Feuerarlarm war bisher ein blinder Alarm.

Es gab und gibt in der Informatik Tendenzen, wirkliche Fehler zu Phantomfehlern zu erklären: »It is not a bug, it is a feature.«

Testexperimente

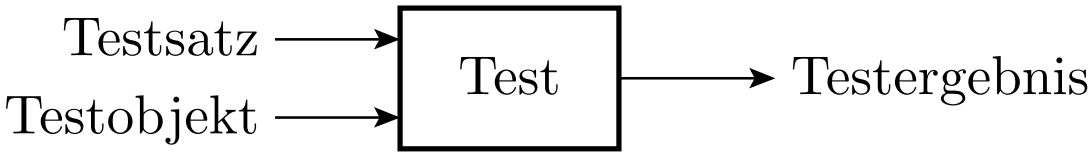
Test

Eine Test dient zur Kontrolle, ob ein Service fähig ist, korrekte Leitungen zu liefern. Er besteht aus einer Folge von Anforderungen mit Ergebnisüberwachung. Das Mindestergebnis ist eine gut/schlecht-Aussage. Bei der Testaussage »schlecht« (nicht zur korrekten Leistungen fähig) kann ein Test weitere Zusatzinformationen liefern:

- Bedatungen der Service-Anforderungen, die versagen,
- die Art des Fehlverhaltens (kein oder falsches Ergebnis, einmaliges, beständiges oder unbeständiges Fehlverhalten) und
- bei falschen Ergebnis die Verfälschungen.

In Systemen mit isoliert testbaren Teilsystemen ist weiterhin eine Fehlerlokalisierung und darüber einer Angabe über die Art und Anzahl der enthalten Fehler möglich.

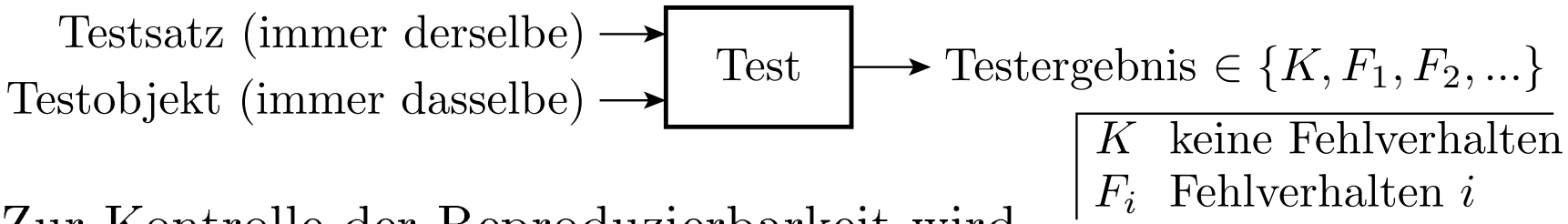
Test als Zufallsexperiment



Über die Auswahlregeln für den Testsatz (immer gleich, zufällig) und das Testobjekt (immer dasselbe, unterschiedliche) sowie über die zu bestimmenden Testergebnisse (nur gut/schlecht, mit Zusatzangaben, welche Service-Leistungen wie versagen oder mit lokalisierten Fehlern) lassen sich unterschiedliche Zufallsexperimente mit unterschiedlichen Zielstellungen definieren:

- Kontrolle der Reproduzierbarkeit des Fehlverhaltens,
- Experimente zum Schätzen des Fehleranteils und
- Experimente zum Schätzen der Fehlernachweiswahrscheinlichkeiten und der Fehlernachweisdichte.

Experiment zur Kontrolle der Reproduzierbarkeit



Zur Kontrolle der Reproduzierbarkeit wird derselbe Test mit demselben Testobjekt mehrfach wiederholt und als Ereignisse das korrekte und die unterschiedlichen Fehlverhalten gezählt. Mögliche Ergebnisse und Diagnosen:

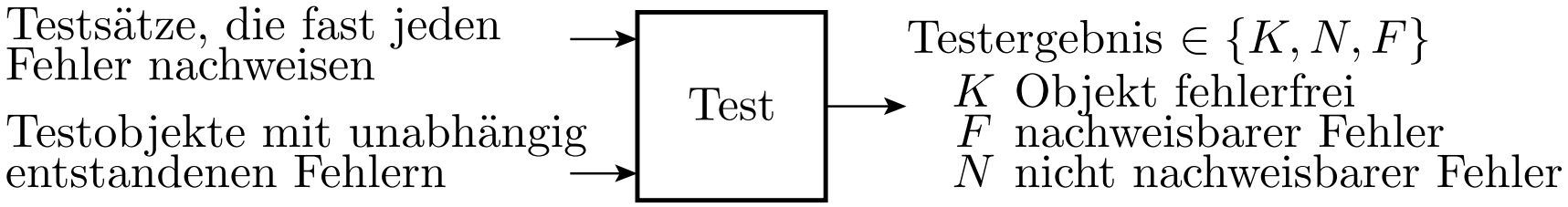
- einmaliges Fehlverhalten ⇒ Störung,
- seltene Fehlverhalten ⇒ Schwachstelle im System, die Störungen begünstigt.
- häufige, aber unterschiedliche Fehlverhalten ⇒ unbeständige Fehler.
- immer dasselbe Fehlverhalten ⇒ beständiger Fehler.

Die so gewonnen Diagnoseergebnisse sind wichtig für die Suche und Behebung der Fehler (siehe später Foliensatz F3).

Schätzen des Fehleranteils

Der Fehleranteil DL (Defekt Level) ist der Anteil der defekten Systeme in einer Menge entworfener oder gefertigter gleichartiger Systeme. Der Richtwerte für ungtestete Software-Codezeilen ist 3% bis 10%. Für getestete elektronische Bauteile wird der Fehleranteil in dpm (Defects per Million) angegeben. Richtwerte:

Typ	DL
Leiterplatten	10 dpm
Schaltkreise	200 dpm
diskrete Bauteile	10 dpm
Lötstellen	1 dpm



Das Experiment zur Abschätzung des Fehleranteils ist der Test vieler vergleichbarer Objekte mit unabhängig entstandenen Fehlern mit Testsätzen mit einer Fehlerüberdeckung[12]:

$$FC = \frac{\text{Anz}(F)}{\text{Anz}(F) + \text{Anz}(N)}$$

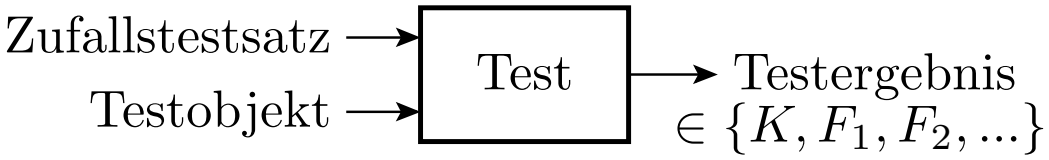
Der Fehleranteil der Objektstichprobe

$$DL = \frac{\text{Anz}(F) + \text{Anz}(N)}{n} = \frac{\text{Anz}(F)}{FC \cdot n} > \frac{\text{Anz}(F)}{n}$$

(n – Anzahl der getesteten Objekte) ist um den Kehrwert der Fehlerüberdeckung größer als der Anteil der nachweisbaren Fehler. Da die Fehlerüberdeckung selbst durch das Experiment nicht bestimmbar ist, sollte sie nahe 100% sein.

[12] Die Fehlerüberdeckung ist der Anteil der nachweisbaren Fehler.

Schätzen von Fehlernachweiswahrscheinlichkeiten



K	keine Fehlverhalten
F_i	Fehlverhalten i

Bei einer zufälligen Bedatung der Testeingaben oder beim Test im laufenden Betrieb ist der Nachweis eines vorhandenen Fehlers Zufall. Das Experiment »Schätzen der Fehlernachweiswahrscheinlichkeiten« besteht in einer mehrfachen Testwiederholung mit unterschiedlicher zufälliger Bedatung. Das Ergebnis eines Einzeltests j ist die Menge M_j der nachweisbaren Fehler. Ohne Fehlerlokalisierung kann es die leere Menge (keine Fehler nachweisbar) oder $\{F\}$ (F – Fehler nachweisbar) sein. Mit Lokalisierung ist es allgemein ein Menge nachweisbarer Fehler F_i (F_i – fehlerhaftes Teilsystem[13] i).

[13] Teilsystem, das auf eine korrekte Service-Anfrage keine oder korrekte Ergebnisse liefert.

Abschätzbare Wahrscheinlichkeiten:

- Nachweiswahrscheinlichkeit Fehler i:

$$p_{\mathrm{F}.i} = \frac{\mathrm{Anz}\,(F_i)}{n}$$

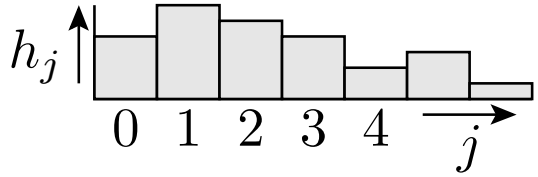
($\mathrm{Anz}\,(F_i)$ – Anzahl der Zufallstest, die F_i nachgewiesen haben; n – Anzahl der durchgeführten Zufallstests).

- Nachweiswahrscheinlichkeit System fehlerhaft:

$$p_{\mathrm{F}} = \frac{\mathrm{Anz}\,(F)}{n}$$

($\mathrm{Anz}\,(F)$ – Anzahl der Zufallstest, die mindestens einen der Fehler F_i nachgewiesen haben.)

Fehlernachweisdichte



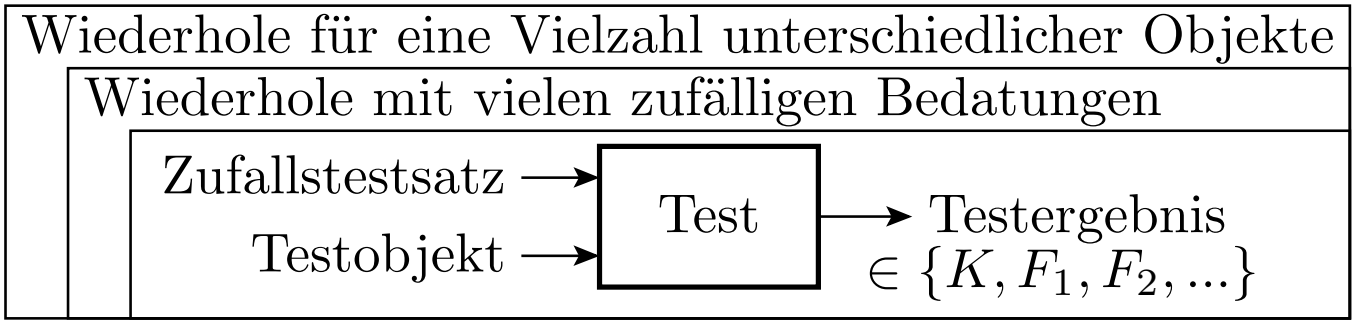
Die Fehlernachweisdichte beschreibt die relative Auftrittshäufigkeit von Fehlern in Abhängigkeit von deren Nachweiswahrscheinlichkeit für eine Klasse von Systemtypen. Eine typische Beobachtung ist, dass, wenn man die Nachweiswahrscheinlichkeiten in Intervalle

$$I_j = \left[v^{-j}, v^{-(j+1)}\right)$$

($j \in \{0, 1, 2, \ldots\}$ – Intervallnummer; v – Parameter für die Intervallgröße, z.B. 2 für $\left[1, \frac{1}{2}\right)$, $\left[\frac{1}{2}, \frac{1}{4}\right)$, ...) nimmt die Häufigkeit

$$h_j = \frac{\text{Anz}\left(F_i|_{p_i \in I_j}\right)}{\text{Anz}\left(F_i\right)}$$

der den Intervallen zuzurechnenden Fehler mit der Intervallnummer, d.h. abnehmender Nachweiswahrscheinlichkeit ab.



Das Experiment zur Abschätzung der Fehlernachweisdichte besteht aus den Experimenten zum Schätzen der Nachweiswahrscheinlichkeit vorhandener Fehler für eine Vielzahl unterschiedlicher Objekte einer betrachteten Klasse von Systemen. Jedes Einzelexperiment schätzt die Nachweiswahrscheinlichkeiten der Fehler im getesteten Objekt und liefert eine Menge von Wahrscheinlichkeitswerten. Jeder Wahrscheinlichkeitswert wird ihrem Wahrscheinlichkeitsbereich j zugeordnet, für den die Anzahl der aufgetretenen Fehler um eins erhöht wird. Die geschätzte Häufigkeiten h_i ist die akumulierte Anzahl geteilt durch die Anzahl aller Fehler, für die in allen Experimenten zusammen Wahrscheinlichkeiten bestimmt wurden.

Die Fehlernachweisdichte dient später zur Abschätzung der Anzahl der nicht gefundenen Fehler in Systemen und der Häufigkeit, mit der diese Fehler im Einsatz Service-Leistung versagen lassen.

Das skizzierte Experiment ist extrem aufwändig und vermutlich noch nie mit einer aussagekräftigen Versuchsanzahl durchgeführt wurden. Im Abschnitt Fehlernachweis (ab Folie ??) wird gezeigt, dass sich die Fehlernachweisdichte auch aus der Systemstruktur abschätzen lässt.

Fehlerentstehung

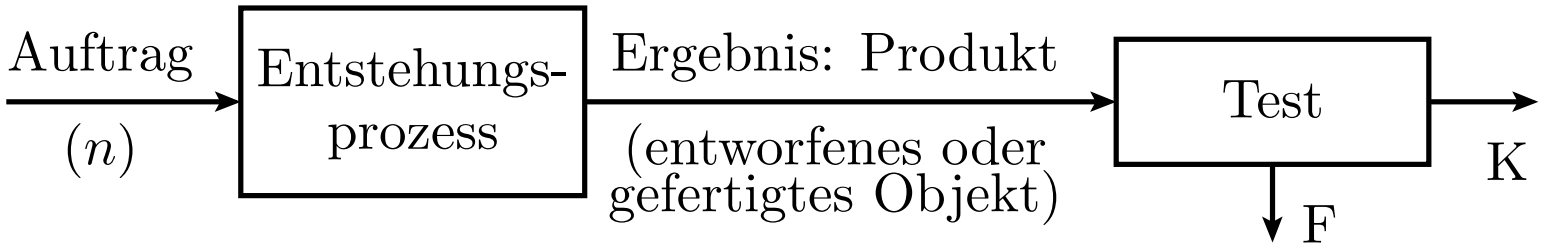
Fehlerentstehung

Die Fehler in einem IT-System entstehen während des Entwurfs, bei der Fertigung und durch Ausfälle während des Betriebs[14]. Entstehungsprozesse lassen sich auch mit dem Service-Modell beschreiben. Dazu werden sie gedanklich so in Einzelaktivitäten (Service-Leistungen) aufgeteilt, dass ein Versagen einer Entsehungs-Service-Leistung einen Fehler im entstehenden System verursacht. Beispiele für Entstehungs-Service-Leistungen:

- Entwurf einer Teilschaltung, eines Bauteils, ...
- Bestückung einer Baugruppe mit einem Bauteil,
- Fertigen eines Bauteils, einer Baugruppe, ...
- Schreiben einer Programmanweisung, eines Programms, ...

[14] Fertigungsfehler und Ausfälle spielen nur für Hardware eine Rolle. Software als rein intellektuelle Ware unterliegt keinem Verschleiß und hat außer dem Entwurf keinen fehlerträchtigen Entstehungsprozess.

Das Zufallsexperiment zur Untersuchung der Fehlerentstehung besteht in Analogie zur Untersuchung von Service-Anfragen für IT-Systeme, in einem Entwurfs- oder Fertigungsauftrag und dem Test der Produkte. Mögliche Ergebnisse einer einzelnen Entstehungs-Service-Leistung sind kein oder ein Fehler.



K Es ist ein als korrekt klassifiziertes Objekt entstanden
F Es ist ein als fehlerhaft klassifiziertes Objekt entstanden

Wahrscheinlichkeit der Fehlerentstehung:

$$p \approx \frac{Anz\,(K)}{n}$$

Modellerweiterungen um »kein Ergebnis« und mögliche Klassifizierungsfehler des Tests (nicht erkannte Fehler, Phantomfehler) sind möglich, aber für das Weitere nicht erforderlich.

Fehleranteil der entstehenden Produkte

Ein ungetestetes (Teil-) System ist fehlerfrei, wenn in ihm kein Fehler entsteht. Bei einem Entstehungsprozess aus n Schritten mit einer unabhängigen Entstehungswahrscheinlichkeit p ist der Fehleranteil als die Wahrscheinlichkeit, dass kein Fehler entsteht:

$$DL = 1 - (1 - p)^n$$

In einem hierachischen System aus (getesteten) Komponenten und Verbinungselementen ist die Fehlerwahrscheinlichkeiten der Komponenten deren Fehleranteil. Der Fehleranteil des übergeordneten Systems ist mindestens die Wahrscheinlichkeit, dass mindestens eine Kompnente fehlerhaft ist:

$$DL_{\text{Sys}} > 1 - \prod_{i=1}^{Anz} (1 - DL_i)$$

(Anz – Anzahl der Komponeneten).

Fehleranteil einer Baugruppe

Eine Baugruppe soll aus nachfolgenden Komponenten mit gegebenen Fehleranteilen bestehen:

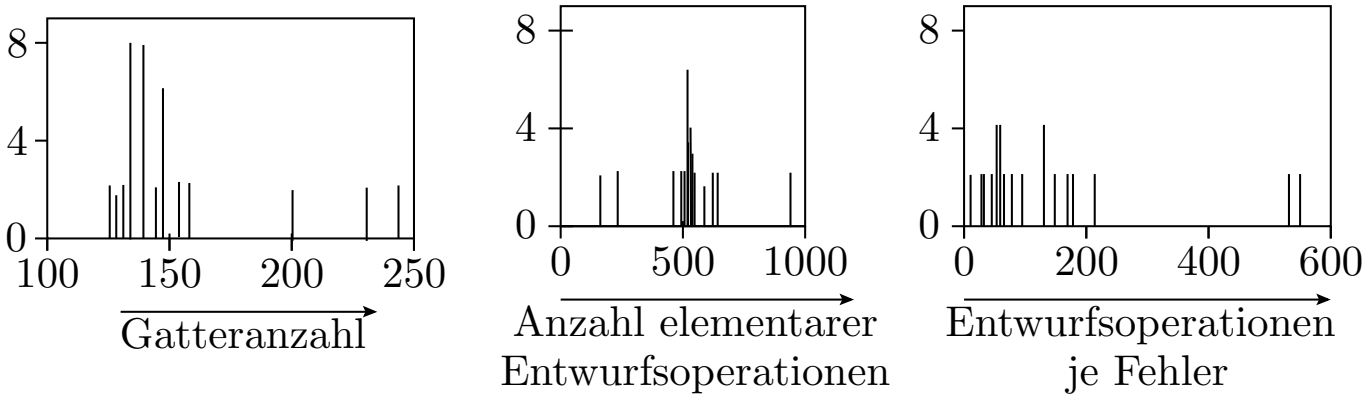
Typ	Anzahl	DL_{BT}
Leiterplatte	1	10 dpm
Schaltkreise	20	200 dpm
diskrete Bauteile	35	10 dpm
Lötstellen	560	1 dpm

Welcher Fehleranteil ist für die Baugruppe zu erwarten, wenn die bei der Baugruppenfertigung zusätzlich entstehenden Fehler zahlenmäßig vernachlässigbar sind oder alle beseitigt werden:

$$\begin{aligned} DL_{\mathrm{Sys}} &= 1 - \left(1 - 10^{-5}\right) \cdot \left(1 - 2 \cdot 10^{-4}\right)^{20} \cdot \left(1 - 10^{-5}\right)^{35} \cdot \left(1 - 10^{-6}\right)^{560} \\ &\approx 10^{-5} + 20 \cdot 2 \cdot 10^{-4} + 35 \cdot 10^{-5} + 560 \cdot 10^{-6} \\ &\approx 5000\,\mathrm{dpm} = 0,5\% \end{aligned}$$

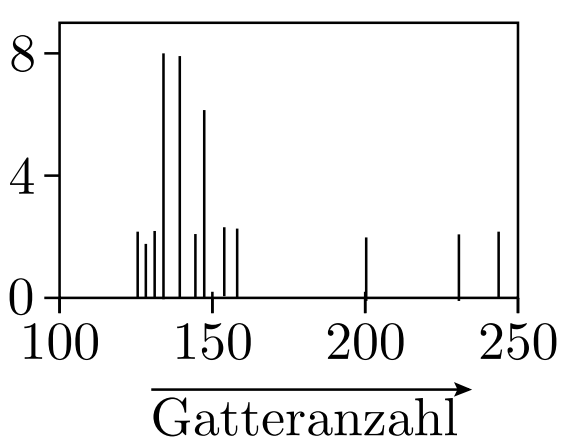
Experiment aus [1]

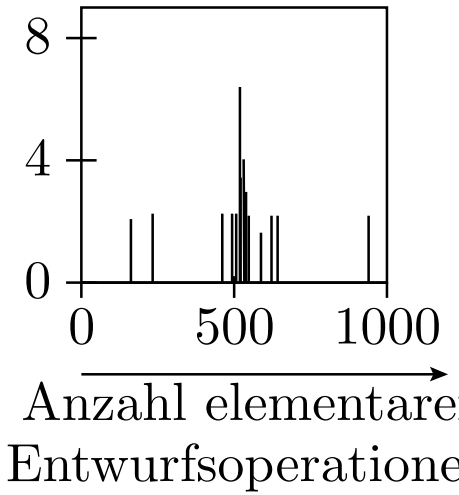
Eine Gruppe von 72 Studenten hatte die Aufgabe, aus der Beschreibung eines PLAs[15] eine Gatterschaltung zu entwickeln und diese über die grafische Benutzeroberfläche eines CAD-Systems in den Rechner einzugeben. Für jeden Entwurf wurden die elementaren Entwurfsoperationen[16], die Gatteranzahl und die Entwurfsfehler gezählt.

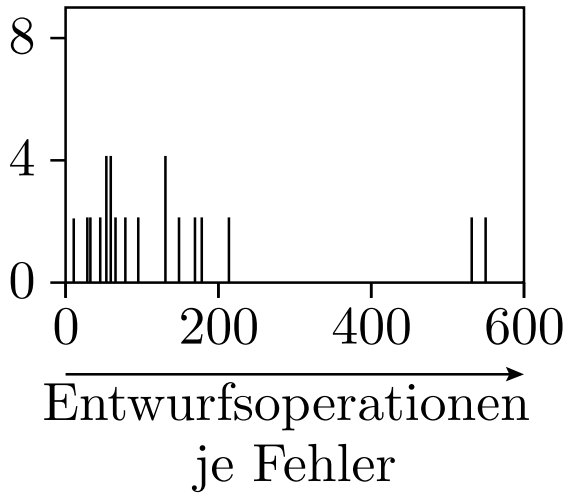


[15] PLA: programmable logic array

[16] Als elementare Entwurfsoperationen galten das Anordnen eines Gatters auf dem Bildschirm und das Zeichnen einer Verbindung.







Die Gatteranzahl der Entwürfe bewegte sich in einem Bereich von 131 bis 245, der gemessene Entwurfsaufwand zwischen 160 und 940 elementaren Entwurfsoperationen je Fehler in einem Bereich von 29 bis 550. Das Experiment zeigt, dass aus der Größe eines Entwurfs (hier Gatteranzahl) oder dem Entwurfsauwand (hier in elementaren Entwurfsopeationen) Rückschlüsse auf den Fehleranteil der entstehenden Produkte möglich sind.

Aufgaben

Aufgabe 3.1: Wahrscheinlichkeit von Kontrollergebnissen

Begründen Sie, warum die beiden Wahrscheinlichkeitsprodukte in Gleichung 5

$$p_{\mathrm{K}} = p_{\mathrm{SK}} \cdot p_{\mathrm{KK}} + p_{\mathrm{SF}} \cdot p_{\mathrm{FK}}$$

und Gleichung 6

$$p_{\mathrm{F}} = p_{\mathrm{SF}} \cdot p_{\mathrm{FF}} + p_{\mathrm{SK}} \cdot p_{\mathrm{KF}}$$

addiert, statt nach Gleichung 3 über die Regel

$$P(A \vee B) = 1 - (1 - P(A)) \cdot (1 - P(A))$$

zusammengefasst werden.

Aufgabe 3.2: Service-Versagen und Kontrolle 1

In einem System, in dem auf die Ereignisse »keine Service-Leistung« oder »kein Kontrollergebnis« solange mit einer wiederholten Service- oder Kontrollanforderung reagiert wird ($p_N = p_{SN} = 0$), seien die Wahrscheinlichkeiten für Kontrollergebnis »Service korrekt«: $p_K = 95\%$ und die beiden korrekten Klassifikationen $p_{FF} = 90\%$ und $p_{KK} = 97\%$ bekannt. Wie groß sind die Wahrscheinlichkeiten

1. p_{SF} dass ein Service-Ergebnis korrekt ist,
2. p_{KF} dass ein als korrekt klassifiziertes Service-Ergebnis falsch ist und
3. p_{FK} dass ein als falsch klassifiziertes Service-Ergebnis korrekt ist?

Aufgabe 3.3: Service-Versagen und Kontrolle 2

Ein Service sei mit 99%-iger und seine Kontrolle mit 100%-iger Wahrscheinlichkeit verfügbar. Wie groß ist die Wahrscheinlichkeit, dass der Service fehlerhaft ausgeführt wird, wenn 99% der Kontrollen keinen Fehler erkennen und die Kontrollerkennungswahrscheinlichkeit 80% beträgt?

Aufgabe 3.4: Aufgaben zum Test

1. In einem Experiment zur Abschätzung des Fehleranteils wurden 31 Objekte als fehlerhaft und 712.981 Objekte als fehlerfrei klassifiziert. Für den Testsatz sei anzunehmen, dass er mindesten 80% der Fehler nachweisen kann. Schätzen Sie ab, in welchem Bereich der Fehleranteil liegt.
2. Bestimmen Sie die Fehlernachweisdichte für folgende Menge von geschätzen Nachweiswahrscheinlichkeiten ... und den Intervallunterteilungsparameter $v = 2$.

Aufgabe 3.5: Fehlerentstehung

- Wie viele Fehler sind in einem großen Software-System mit 10^5 Programmzeilen zu erwarten, wenn beim Entwurf 3% der Programmzeilen falsch sind und der Test 60% der Fehler erkennt?
- Durch eine Störung in einem Fertigungsprozess verdoppelt sich die Anzahl der fehlerhaft gefertigten Bauteile. Wie wirkt sich das auf die Häufigkeit der Fehlfunktionen eines Systems aus, bei dem dieser Bauteiltyp bisher 10% der Fehlfunktionen verursacht hat?

Aufgabe 3.6: Fehleranteil

Ein Rechner besteht aus Leiterplatten, Schaltkreisen, diskreten Bauteilen (Widerstände, Kondensatoren, ...) und Lötstellen. Die nachfolgende Tabelle zeigt für einen Beispielrechner für alle eingesetzten Bauteiltypen deren Anzahl und deren zu erwartenden Fehleranteil DL_{BT}.

Typ	Anzahl	$E\left(DL_{\mathrm{BT}}\right)$
Leiterplatten	10	10 dpm
Schaltkreise	100	200 dpm
diskrete Bauteile	200	10 dpm
Lötstellen	10000	1 dpm

Was für einen Fehleranteil hat ein solcher Rechner, wenn alle anderen Arten von Fehlern anzahlmäßig vernachlässigt werden können?

Aufgabe 3.7: Chipgröße, Fehleranteil und Herstellungskosten

Der Fehleranteil der Transistoren eines Fertigungsprozesses für integrierte Schaltkreise sei bekannt und betrage:

$$DL_{\mathrm{Tr}} \approx 10^{-6}$$

Andere Fehlerarten seien zu vernachlässigen. Wie hoch ist der Fehleranteil für Chips mit

1. 10^5
2. 10^6 und
3. 10^7 Transistoren.

Schätzen Sie die Herstellungskosten der Halbleiter-Chips ab unter der Annahme, dass ein Chip mit 10^6 Transistoren 1\$ beträgt, die Kosten sich proportional zur Chipfläche verhalten und die Kosten für die auszusortierenden defekten Schaltkreise zu denen der fehlerfrei gefertigten hinzugefügt werden müssen.

Literatur

[1] J. E. Aas and I. Sundsbo.
Harnessing the human factor for design quality.
IEEE Circuits and Devices Magazine, 11(3):24–28, 1995.

[2] J. Hartmann.
Analyse und Verbesserung der probabilistischen Testbarkeit kombinatorischer Schaltungen.
PhD thesis, Diss. UnversitÃ¤t des Saarlandes, 1992.