Freie Universität Berlin

Proseminar Technische Informatik WS08/09
bei Georg Wittenburg, M.Sc.

Zuverlässigkeit und Fehlertoleranz in der Technik

Ein Vortrag von
Sebastian Oliver Kalwa

Berlin, 30.01.2009

Was ist Fehlertoleranz?

Fehlertoleranz:

Die Eigenschaft eines Systems auch bei begrenzter Anzahl ausgefallener Subsysteme weiter zu laufen.

Fehler:

Abweichung von korrekter Arbeitsweise, im englischen drei Begriffe:

Fault, *Error* und *Failure*

Fault: Fehlerursache, Fehlergrund

Error: Fehlerzustand des Systems

Failure: Systemversagen

Fehlertoleranz versucht diese kausale Kette zu unterbrechen.

Beispiel

Beispiel:

Fault: Fehler im Algorithmus, z.B. Verwechslung von m/s und km/h

Error: Falscher Variablenwert, z.B. Geschwindigkeit einer Rakete nach Start um Faktor 3,6 kleiner als angenommen

Failure: Rakete nimmt kürzere Flugbahn und schlägt in eigenem Gebiet ein

Mögliche Gegenmaßnahme:

- Mehrfachimplementierung des Algorithmus durch mehrere Leute
- Diese wären zu unterschiedlichen Ergebnissen gekommen: Error
- Dem Error folgt kein Failure, da Fehler erkannt und Start abgebrochen

Anwendungsbereiche

Fehlertoleranz notwendig in vielen Fällen:

Hochzuverlässige Systeme: Telefonnetze dürfen nur wenige Minuten im Jahr ausfallen

Wartungsfreie Systeme: Satelliten müssen lange durchhalten, da sie nicht gewartet werden können

Festgelegte Wartungsintervalle: Ein Transportschiff wird nur im Heimathafen gewartet, sonst nicht

Hochleistungsrechner: Haben hohe Zahl parallel arbeitender Komponenten, Ausfall einzelner Teile normal

Sicherheitsrelevante Anwendungen: Moderne Kampfjets sind bei Ausfall des Bordcomputers für einen Menschen unmaneuvrierbar.

Was ist Zuverlässigkeit?

Zuverlässigkeit:

Eigenschaft, die geforderte Funktion in einem Zeitraum zu erfüllen.

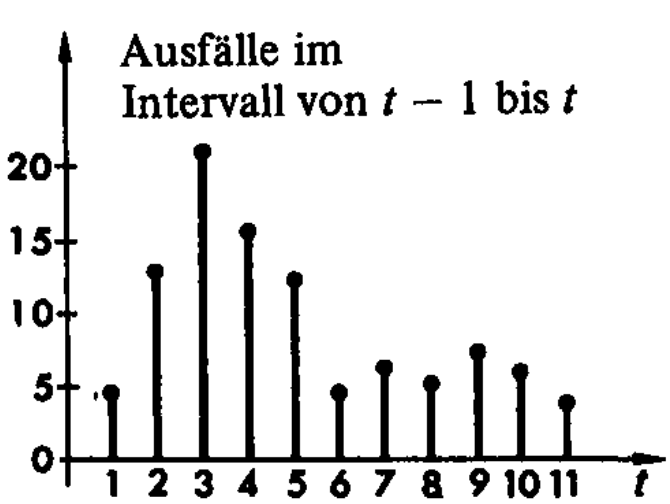
Beispiel:

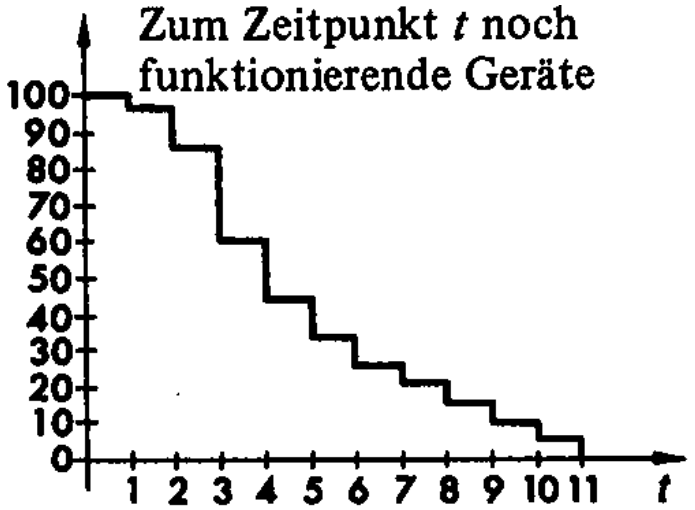
Eine Testreihe mit 100 Autos liefert nach einem Jahr folgende Ergebnisse zur Zuverlässigkeit:

Monat	0	1	2	3	4	5	6	7	8	9	10	11
Ausfälle im Monat	0	0,04	0,13	0,22	0,16	0,12	0,04	0,06	0,05	0,07	0,06	0,05
Noch funk-tionierend	1	0,96	0,83	0,61	0,45	0,33	0,29	0,23	0,18	0,11	0,05	0

Beispiel

Die Werte auf einen Graphen übertragen:





Die Angabe wie viele Geräte nach einer Zeit t noch funktionieren nennt man *Überlebenswahrscheinlichkeit* v(t).

Hier z.B.: v(10)=0,05

Berechnung von Überlebenswahrscheinlichkeit

Beispielsystem besteht aus zwei Komponenten e_1 und e_2.

a) Fällt aus, wenn bereits eine Komponente ausfällt.

Entspricht folgendem Schaltbild:

Verknüpfung entspricht dem Schnitt zweier Mengen

P(„System überlebt“)

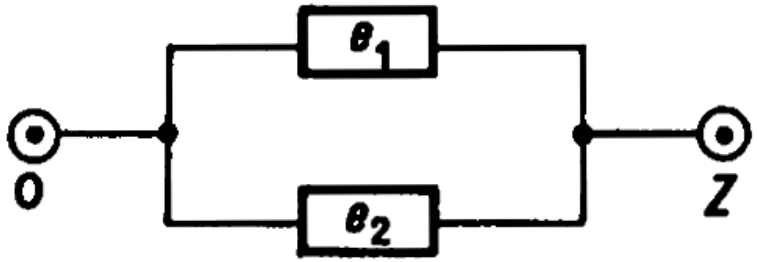
=P(„e_1 überlebt“ und „e_2 überlebt“)

= P(„e_1 überlebt“) * P(„e_2 überlebt“)

Berechnung von Überlebenswahrscheinlichkeit

b) Fällt aus, wenn beide Komponenten ausfallen.

Entspricht folgendem Schaltbild:



Verknüpfung entspricht der Vereinigung zweier Mengen.

P(„System überlebt“)

=P(„e_1 überlebt“ oder „e_2 überlebt“)

=P(„e_1 überlebt“) + P(„e_2 überlebt“)

– P(„e_1 überlebt“) * P(„e_2 überlebt“) Inklusion und Exklusion

Wie erhöht man Überlebenswahrscheinlichkeiten?

Seien nun an 80% dieser Ausfälle ein Bauteil schuld, das mit hoher Wahrscheinlichkeit in den ersten zwei Jahren kaputt geht.

Möglichkeiten zur Erhöhung der Ausfallwahrscheinlichkeit:

Verzicht auf dieses Teil und Wahl eines anderen Designs
- Nicht immer Wahl von anderen Konstruktionen möglich

Einbau eines Bauteils eines anderen Herstellers
- Eventuell nicht möglich, da nur ein Produzent vorhanden oder andere zu teuer

Mehrfacher Einbau dieses Bauteils
- Wenn eins ausfällt, sind noch immer weitere vorhanden, die die Funktion gewährleisten
- Kann auch teuer sein, wenn dieses Teil selbst teuer ist

Wie erhöht man Überlebenswahrscheinlichkeiten?

Beispiel:

Bauteil fällt mit 50% Wahrscheinlichkeit aus, kann nicht ersetzt werden.

5-facher Einbau:

Überlebenswahrscheinlichkeit = W-keit, dass nicht alle Teile ausfallen

$= 1 - 0{,}5^5 = 0{,}96875$

Vorher: *Single Point of Failure*

- Ausfall eines Teils reicht zum Ausfall
- Wahrscheinlichkeit zum Ausfall 50%

Jetzt: *Redundanz* vorhanden

- Mehrere Teile müssen zum Komplettausfall versagen
- Wahrscheinlichkeit für Ausfall bei 5 Teilen unter 5%

Redundanz Beispiel

Viele Systeme heutzutage ohne Redundanz nicht alltagstauglich:

Schlechte Idee:
Alle Triebwerke eines Flugzeugs mit einem gemeinsamen Stromkreis
- Eine kaputte Leitung reicht zum Ausfall aller Triebwerke

Gute Idee:
Jedes Triebwerk mit mehreren eigenen Stromkreisen
- Ausfall einzelner Leitungen undramatisch
- Selbst bei Ausfall eines Triebwerks die anderen noch lauffähig
- Ausfall aller Leitungen extrem unwahrscheinlich

Redundanzarten

Strukturelle Redundanz:

Komponenten sind mehrfach vorhanden und übernehmen untereinander bei Ausfall die Arbeit

Beispiel: mehrere Stromleitungen für Flugzeugtriebwerke

Zeitredundanz:

Im Fehlerfall wird mehr Zeit für das Ergebnis benötigt

Beispiel: Zahlensortieren nach Quicksort, wenn Ergebnis nicht korrekt, dann sortieren mit Heapsort

Informationsredundanz:

Daten erhalten zusätzliche Informationen um Korrektheit zu überprüfen

Beispiel: Letzte Ziffer in Personalausweisnummer, CRC, FEC

Redundanzarten

Funktionsredundanz:
System erhält weitere Funktionen, ohne die es auch lauffähig wäre
Beispiel: Testfunktionen

Nicht immer ist eine klare Abgrenzung möglich.

Beispiel:
Zwei Komponenten berechnen Ergebnis, dritte prüft Übereinstimmung
- Funktionsredundanz, auch bei einmaligem Berechnen lauffähig
- Strukturredundanz, Komponente mehrfach vorhanden
- Zeitredundanz, Überprüfung auf Gleichheit braucht Zeit

Weitere Redundanzeinteilung

Statische Redundanz:

- Hat nur Nutzen im Fehlerfall
- Kann nur die gleiche Arbeit machen wie die zu ersetzenden Komponenten und keine eigene Aufgabe übernehmen

Beispiel: Ersatzstromkreise in Flugzeugen bringen keinen Nutzen solange es noch andere Stromkreise gibt

Dynamische Redundanz:

- Auch bei keinem Fehler nützlich
- Kann neben den anderen redundanten Komponenten eigene Aufgaben übernehmen

Beispiel: Mehrprozessorsystem in dem CPUs ausfallen dürfen, CPU kann unabhängig von anderen arbeiten, aber auch bei Ausfall der anderen für alle einspringen

Fazit

- Fehlertoleranz, insbesondere mit Redundanz, ist sinnvoll für Zuverlässigkeit von Systemen.

- Hersteller von Geräten oder von Software ist auch für dessen Fehler verantwortlich.

- Lieber etwas mehr Geld investieren als unzuverlässige Produkte zu machen

- Kann Menschenleben und vor finanziellen Schäden schützen
  Beispiel: Explodierende Handyakkus wegen fehlendem Kurzschluss-Schutz, hat sogar schon Menschenleben gekostet

- Zuverlässigkeit von Produkten wichtig für Image, Sparen kann im Endeffekt sogar zu Umsatzeinbußen führen

Quellen

- Deutscher Personalausweis
  http://www.pruefziffernberechnung.de/P/Personalausweis-DE.shtml

- Fehlertoleranz, Jürgen Ruf und Thomas Kropf
  http://www-ti.informatik.uni-tuebingen.de/~ruf/seminar0102/Fehlertoleranz.pdf

- Zuverlässigkeit in der Technik, Arnold Kaufmann
  R. Oldenburg Verlag München Wien 1970

- Fehlertolerante Rechensysteme, Prof. Dr. Winfried Görke
  R. Oldenburg Verlag München Wien 1989

- Tod durch Handy-Explosion, Markus C. Schulte von Drach
  http://www.sueddeutsche.de/wissen/194/425951/text/