

Safety and Reliability of Embedded Systems (Sicherheit und Zuverlässigkeit eingebetteter Systeme)

Fault Tree Analysis Mathematical Background and Algorithms

Safety and Reliability of Embedded Systems

ENGINEERING
SOFTWARE
DEPENDABILITY

• Prof. Dr. Lüggenmeyer, 0

Content

- ☐ Definitions of Terms
- ☐ Introduction to Combinatorics
- ☐ General Formulas for AND-, OR-, NOT-, XOR-Gates
- ☐ Calculation of Top-Event Probability
- ☐ Results beyond Top-Event Probability
- ☐ Importance Measures
- ☐ Other Issues in Quantitative Analysis

Safety and Reliability of Embedded Systems

ENGINEERING
SOFTWARE
DEPENDABILITY

• Prof. Dr. Lüggenmeyer, 1

Definitions of Terms

- ☐ **Failure** is any behavior of a component or system that deviates from the specification
- ☐ **Fault** is an abnormal state or condition within a component that can lead to a failure
- ☐ **Accident** is an undesired event that causes death or injury of persons or harm to goods or to the environment
- ☐ **Hazard** is a state of a system *and* its environment where the occurrence of an accident depends only on influences that are not controllable by the system
- ☐ **Risk** is the combination of hazard probability and severity of the resulting accident
- ☐ **Acceptable Risk** is a level of risk that authorities or other bodies have defined as acceptable according to acceptance criteria

Safety and Reliability of Embedded Systems

ENGINEERING
SOFTWARE
DEPENDABILITY

• Prof. Dr. Lüggenmeyer, 2



Other definitions exist,
but many of them are **unpractical**

Definitions of Terms

- ☐ **Safety** is freedom from unacceptable risks
 ⚠ **Safety analysis** aims at proving that the actual risk is below the acceptable risk
- ☐ **Availability** is the property of a system to fulfill its purpose at a given point in time / is the probability that the system fulfills its purpose at a given point in time
 ⚠ The focus is on **uninterrupted service**
- ☐ **Reliability** is the property of an entity to fulfill its reliability requirements during or after a given time span under given application conditions
 ⚠ **Reliability** is related to the probability of a failure event over the mission time

Safety and Reliability of Embedded Systems

ENGINEERING
SOFTWARE
DEPENDABILITY

• Prof. Dr. Lüggenmeyer, 3

Modeling of Reliability

- **Reliability Function R(t):**
 - F(t) gives the probability that at time t the (non-repairable) system has failed
 - Thus $R(t) = 1 - F(t)$ is the probability that at time t no failure has occurred yet
- **Probability Density f(t):**
 - The probability density f(t) describes the modification of the probability that a system fails over time:

$$f(t) = \frac{dF(t)}{dt}$$
- **Failure Rate:**
 - The failure rate is the relative boundary value of failed entities at time t in a time interval that approximates zero, referring to the entities still functional at the beginning of the time interval:

$$\lambda(t) = \frac{f(t)}{R(t)} = \frac{dF(t)/dt}{R(t)} = -\frac{dR(t)/dt}{R(t)}$$



States have a probability.
Events have a probability density
and an (occurrence) rate

Cut Sets, Minimal Cut Sets, Path Sets

- A **Cut Set** is a set of basic events, which in conjunction cause the top event
- A **Minimal Cut Set (MCS)** is a cut set that no longer is a cut set if any of its basic events is removed
- A **Path Set** is a set of basic events that, if they are false, inhibit the top event from occurring
- A **Minimal Path Set (MPS)** is a path set that no longer is a path set if any of its basic events is removed

Introduction to Combinatorics: Truth Values

AND: $A \wedge B$ OR: $A \vee B$

Proposition A →	False	True
Proposition B ↓	False	True
False	False	True
True	False	True
	True	True

- ⚡ True and False are often represented by 0 and 1
- ⚡ The propositions are usually of the type "Component X is in a failed state"

Introduction to Combinatorics: Probabilities

A is true with probability P1, B with probability P2

Proposition A →	1-P1	P1
Proposition B ↓	(1-P1) * (1-P2)	P1 * (1-P2)
1-P2	(1-P1) * (1-P2)	P1 * (1-P2)
P2	(1-P1) * P2	P1 * P2

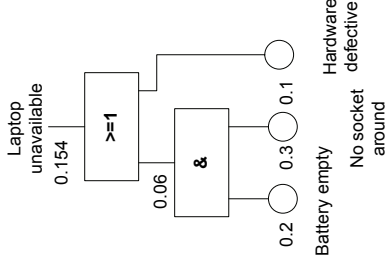
AND: $P(A \wedge B) = P1 * P2$
OR: $P(A \vee B) = P1 * (1-P2) + (1-P1) * P2 + P1 * P2$
 $= 1 - [(1-P1)*(1-P2)]$
 $= P1 + P2 - P1*P2$

General Formulas for AND / OR with n Inputs, NOT, XOR

- ☐ AND-Gate: $P_{out} = \prod_{i=1}^n P_i$
 - ☐ OR-Gate: $P_{out} = 1 - \prod_{i=1}^n (1 - P_i)$
 - ☐ NOT-Gate: $P_{out} = 1 - P_{in}$ (only one input)
 - ☐ XOR-Gate: $P_{out} = \sum_{i=1}^n P_i \cdot \prod_{j=1}^n (1 - P_j)$
- ⚠️ XOR is normally defined for 2 inputs only
- ☐ The n-out-of-m Voter can be replaced by a combination of AND / OR gates
 - ☐ The Inhibit-Gate can be replaced by an AND and a NOT gate
- ⚠️ The Priority-AND has no static combinatorial formula

Calculation of Top-Event Probability

- ☐ Apply gate formulas in a bottom-up fashion
 - ☐ Stop if top-event is reached
- ⚠️ Bottom-up calculation is not efficient for large FTs
- There are two practical algorithms...



Calculation Method 1: Minimal Cut Sets

- ☐ The top-event is the union of all intersection-free minimal cut sets
- ☐ If cut-set probabilities are small (below 0.1), then intersection probabilities are even smaller
- ☐ The top-event probability is the sum of all MCS probabilities
- ☐ The probability of each MCS is the product of the probabilities of the included basic events

$$P_{top} = \sum_{all \text{ MCS } i} P_{MCS \ i}$$

$$P_{MCS} = \prod_{all \text{ events } i \in MCS} P_i$$

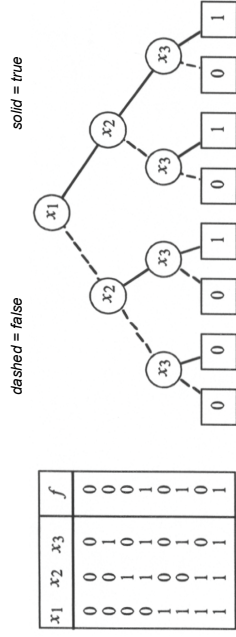
⚠️ This algorithm leads to an approximation. It does not work for NOT gates

Finding Minimal Cut Sets

- ☐ Decompose the tree recursively
- ☐ For each OR gate
 - Generate as many entries as there are inputs {(i1), (i2), (i3),...}
- ☐ For each AND gate
 - Generate one entry containing all inputs {(i1, i2, i3,...)}
- ☐ Repeat until all gates are resolved
- ☐ Cancel cut sets that are not minimal (redundant)

Calculation Method 2: Using BDDs

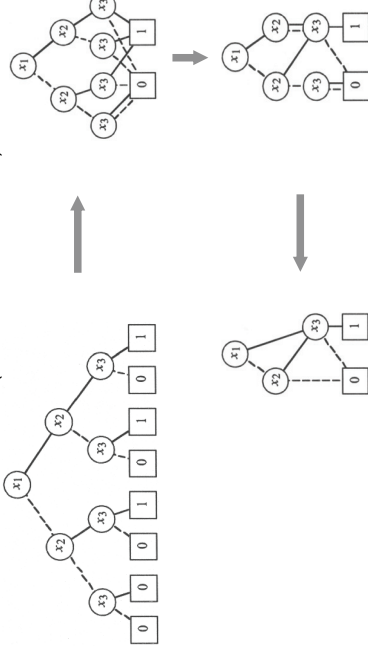
- ☐ BDD = Binary Decision Diagram
- ☐ OBDD = Ordered BDD (defined variable order)
- ☐ ROBDD = Reduced Ordered BDD (after elimination of redundancies)
- ☐ ROBDDs are an efficient representation of Boolean formulas



$$f = (x1 \vee x2) \wedge x3$$

BDD Reduction

- ☐ Removal of redundant nodes (terminal and non-terminal) and tests:



Impact of Variable Order

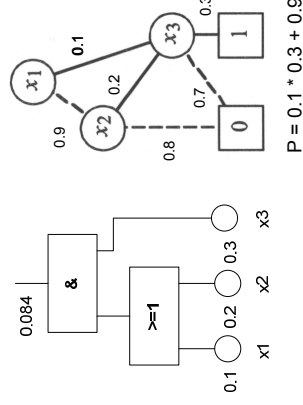
- ☐ The variable order may have considerable influence on the size of the OBDDs
- ☐ Same function, different variable order:



Finding the best variable order is NP-complete (= unachievable for large FTs)

Calculation of Top-Event Probability via BDDs

- ☐ Annotate probabilities from FT to true branches
- ☐ Annotate 1-P to false branches
- ☐ For each path multiply branch probabilities
- ☐ Sum up all paths that lead to terminal 1



Results beyond Top-Event Probability

- ☐ Probability of cut sets with order 1 (consisting of only one event)
 - Single points of failure should have extremely low probability
- ☐ Failure probabilities of (technical) sub-systems
 - Here, redundancy can reduce failure probability of the system
- ☐ Equivalent failure rates
 - Specify, which percentage of the intact systems are expected to fail within a given time span

Importance Measures

- ☐ Importance measures quantify the significance of FT events in terms of their contribution to the top-event probability
- ☐ To know the importance of part of the FT is important for
 - **Robustness Estimation:** How much will my result change if input values are roughly estimated or change during operation?
 - **Work planning:** You should rather spend your time on system changes that have significant impact on overall failure probability

Some Importance Measures

- ☐ Fussell-Vesely Importance
 - Absolute or relative (= percentage) contribution to the top-event probability
- ☐ Risk Reduction Worth or Top Decrease Sensitivity
 - Decrease of top-event probability if a given event is assured not to occur
- ☐ Risk Achievement Worth
 - Increase of top-event probability if a given event occurs
- ☐ Binbaum's Importance Measure
 - Rate of change of top-event probability in relation to rate of change of a given event

Other Issues in Quantitative Analysis

- ☐ Uncertainty Quantification
 - Event data is taken from samples or from other environment
 - Sensitivity analysis or formal uncertainty analysis (assigning a probability distribution)
- ☐ Coverage Factors
 - Take into account that some failures do not lead to catastrophic results
- ☐ Time or Phase Dependent Analysis
 - Use different models or rates for different time intervals according to mission phases

Literature

☐ FTA and Probabilistic Risk Assessment

- Brolin A., Reliability engineering: theory and practice (third ed.), New York, Springer, 1999
- Kececioglu, D.: Reliability Engineering Handbook, Vol. 1 & 2, PTR Prentice Hall, 1991.

☐ Minimal Cut Sets

- Fussel JB, Vesely WE, A new methodology for obtaining cut sets for fault trees. in Trans. American Nuclear Society, 15, 262, 1972
- T. Kohda & EJ Henley, On Digraphs, Fault Trees and Cut Sets, Reliability Engineering and System Safety, 20 (1), 1988, p. 35-61

☐ BDD Algorithm

- Bryant, R.E.: Graph-based algorithms for boolean function manipulation. IEEE Transactions on Computers, C-35(8):677–691, Aug. 1986.
- O. Coudert and J. C. Madre. Fault tree analysis: 10'20 prime implicants and beyond. In Proceedings of the Annual Reliability and Maintainability Symposium, pages 240-5, Atlanta, GA, 26-28 January 1993
- A. Rauzy, An Brief Introduction to Binary Decision Diagrams. RAIRO-API-JESA, 30(8):1033-1051, 1996