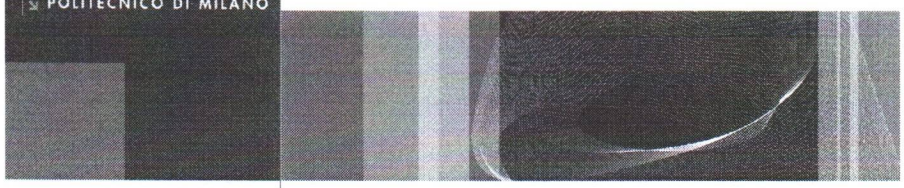
POLITECNICO DI MILANO

CRC

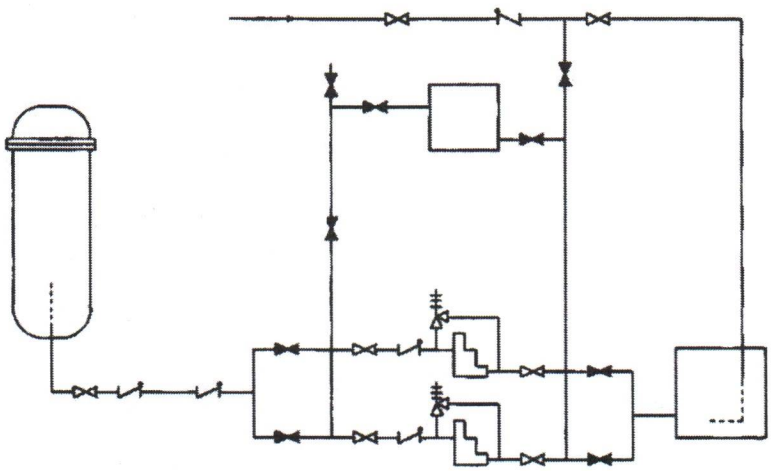


Fault tree analysis

Prof. Enrico Zio

Politecnico di Milano
Dipartimento di Energia

↳ characterize the elements that compose a system, evaluate whatever property we want to evaluate (failure behavior, repair behavior, ...) at the elements level and then recompose/reconstruct the system behavior from its logic and from the components behavior.



Fault Tree Analysis (FTA)

- **Systematic and quantitative**
- **Deductive**

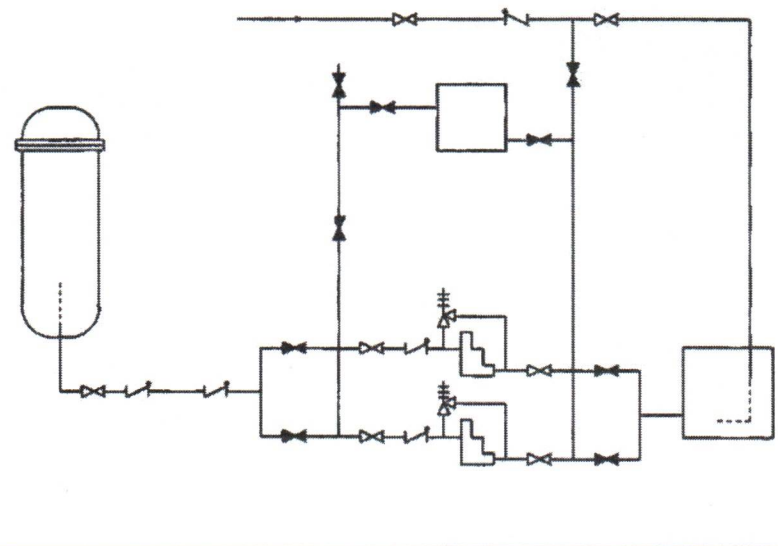
↳ because eventually we calculate the probability of failure of a system given the probability of failure of its components

we define a fault at a system level and by looking at the system logic or function we deduce the combinations of failure of the elements of the system that make the system fail

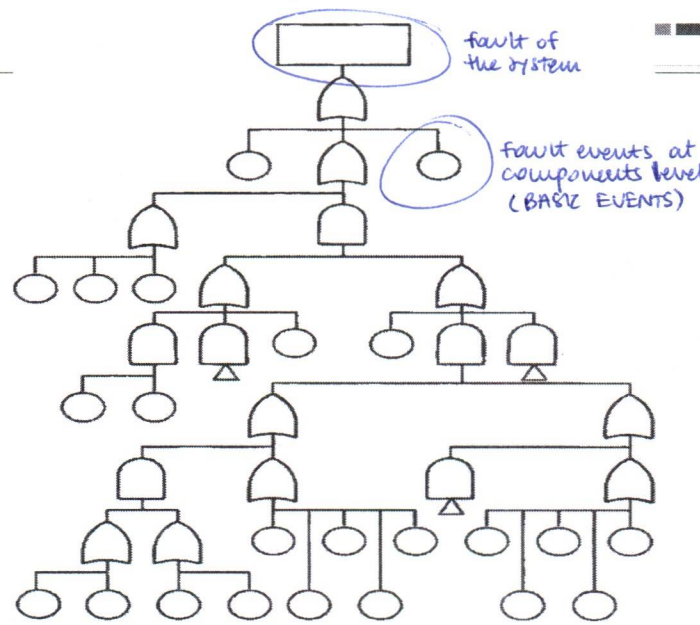
AIM:

1. **Decompose the system failure in elementary failure events of constituent components**
2. **Computation of system failure probability, from component failure probabilities**

from this :



to this :
(this represents the fault logic of the system)

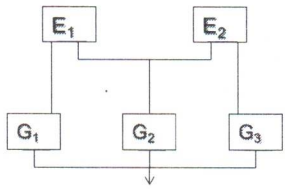


FT construction

FT construction: Procedure steps

1. **Define top event (system failure)**

Electric power generation system



E1, E2 = engines

G1, G2, G3 = generators, each one is rated at 30 KVA

We have to define what is the functioning we want and so what is the failure

T = Failure to supply at least 60 kVA

we define the events in rectangular boxes (and then we decompose them)

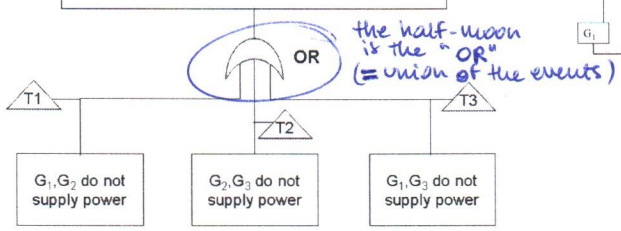
FT construction: Procedure steps

1. **Define top event (system failure)**
2. **Decompose top event by identifying subevents which can cause it.**

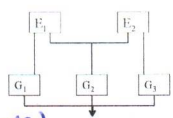
At least two out of the three generators do not work

event

T = Failure to supply at least 60 kVA



the half-moon is the "OR" (= union of the events)

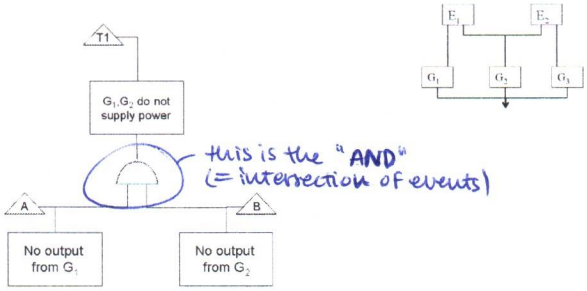


decomposition of the event

we're saying that the failure event is the output of a logic "OR" gate which has as inputs the 3 events: T_1, T_2 and T_3 (they're events so again they're in rectangle boxes → let's decompose them)

FT construction: Procedure steps

1. **Define top event (system failure)**
2. **Decompose top event by identifying subevents which can cause it**
3. **Decompose each subevent in more elementary subevents which can cause it**



this is the "AND" (= intersection of events)

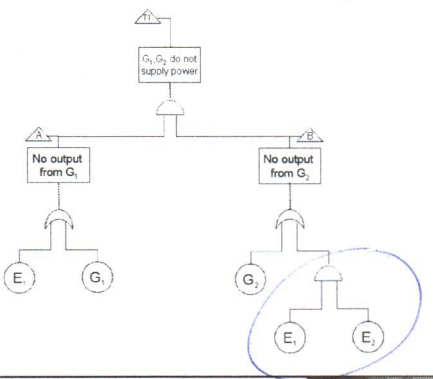
FT construction: Procedure steps

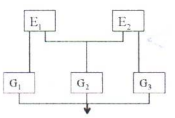
1. Define top event (system failure)
2. Decompose top event by identifying subevents which can cause it
3. Decompose each subevent in more elementary subevents which can cause it
4. Stop decomposition when subevent probability data are available (resolution limit): subevent = basic or primary event

we stop at those events which are independent among each other and for which we can characterize the failure and the repair properties/behaviors

FT construction: Procedure steps

1. Define top event (system failure)
2. Decompose top event by identifying subevents which can cause it
3. Decompose each subevent in more elementary subevents which can cause it
4. Stop decomposition when subevent probability data are available (resolution limit): subevent = basic or primary event

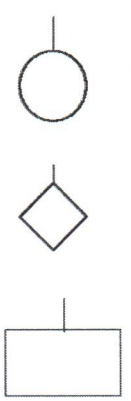




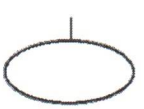
since G_2 is alimented by both E_1 and E_2 then both must not be working

($\cap$)

FT event symbols



Basic event with sufficient data



Undeveloped event

elements that need other analysis (e.g. events based on humans, ...)

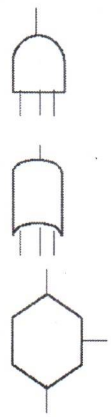
Event represented by a gate

Condition event used with inhibit gate

(we have an event that is realized depending on a combination of AND/OR of events in inputs and a condition)

Transfer symbol

FT gate symbols



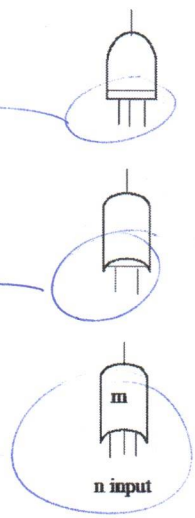
AND gate (intersection)	Output event occurs if all input events occur simultaneously.
OR gate (union)	Output event occurs if any one of the input events occurs.
Inhibit gate	Input produces output when conditional event occurs.

FT gate symbols

Sequential order (from left to right)

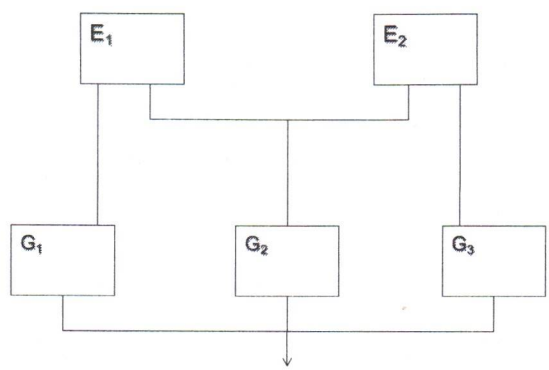
if only one occurs, not both

the output event occurs when m out of n events occur

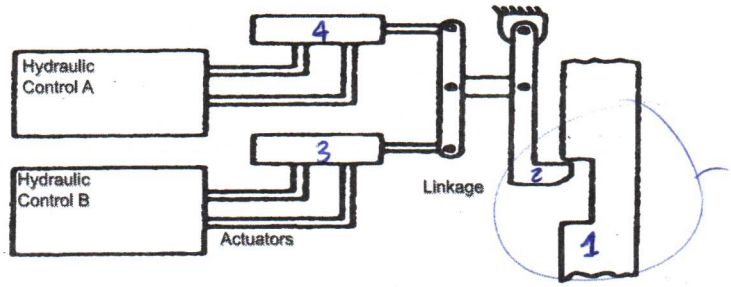


Priority AND Gate	Output event occurs if all input events occur in the order from left to right
Exclusive OR Gate	Output event occurs if one, but not both, of the input events occur.
m out of n gate (volume or sample gate)	Output event occurs if m out of n input events occur. (at least)

FT Example 1: Electric Power Generation System



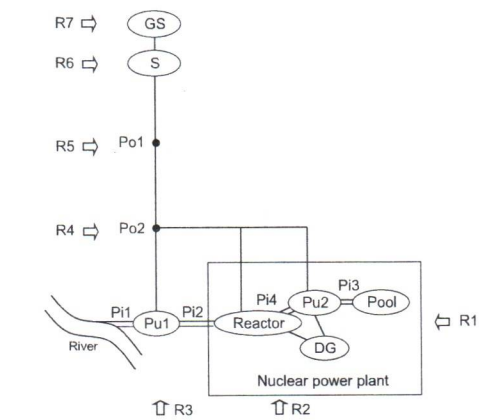
FT Example 2: The Shutdown System



the goal is to release 1 (let with fall down) by moving 2

The components are:
2, 3, 4, A, B
in parallel

FT Example 3: The System of Systems



Pipe
Power line
Road access

Internal emergency devices:

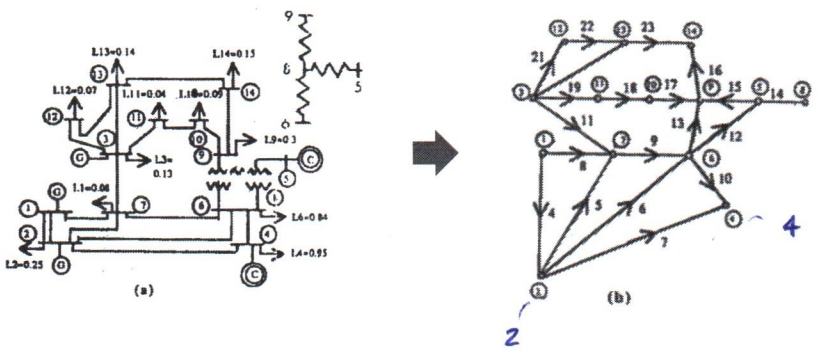
- Power system
  - Diesel Generator (DG)
- Water system
  - Pipe (Pi)
  - Pump (Pu)
  - Pool

Interdependent CIs:

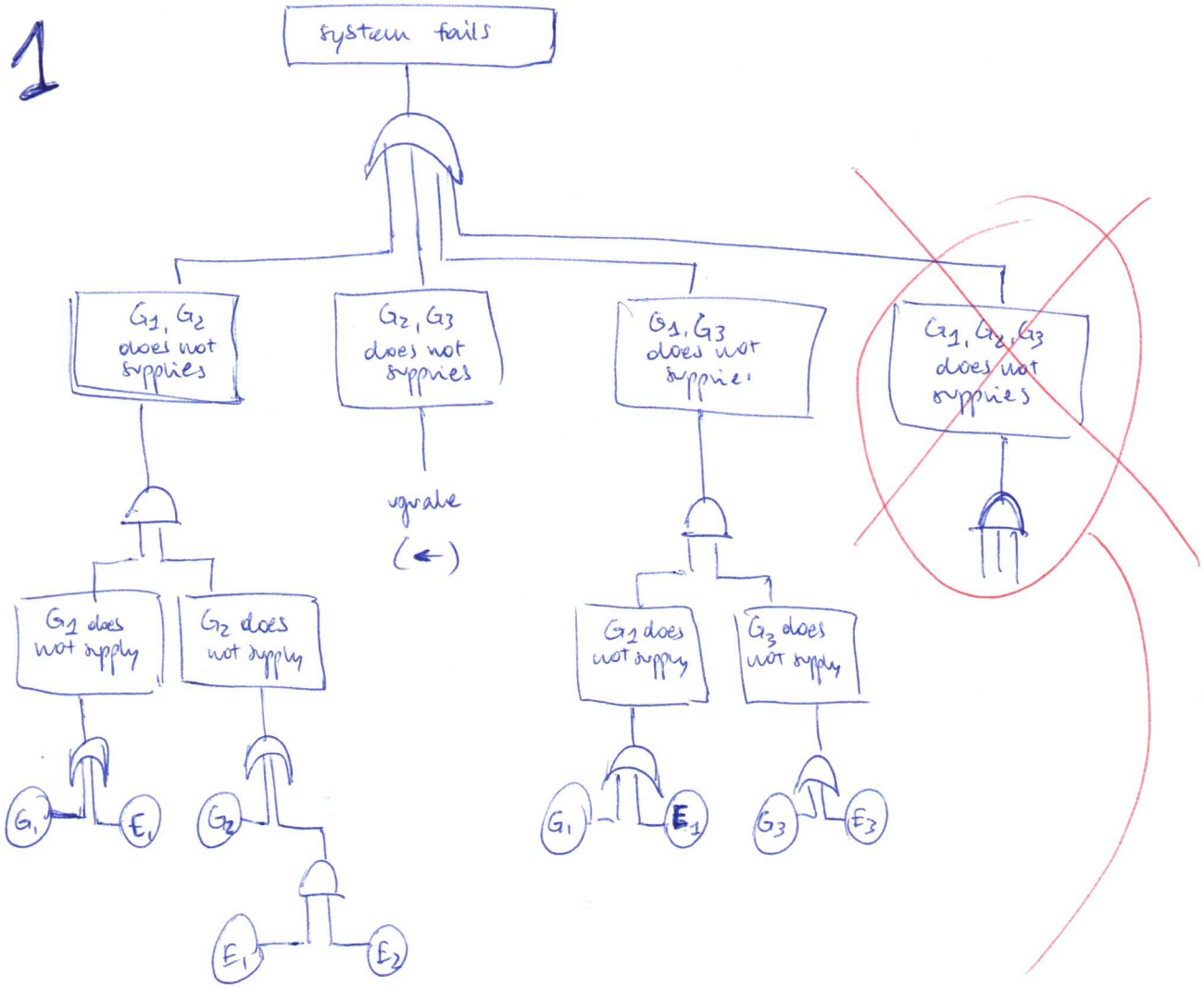
- Power system
  - Generation Station (GS)
  - Substation (S)
  - Pole (Po)
- Water system
  - Pipe (Pi)
  - Pump (Pu)
  - River
- Road transportation system
  - Road access (R)

FT Example 4: IEEE14 Bus Power Distribution System

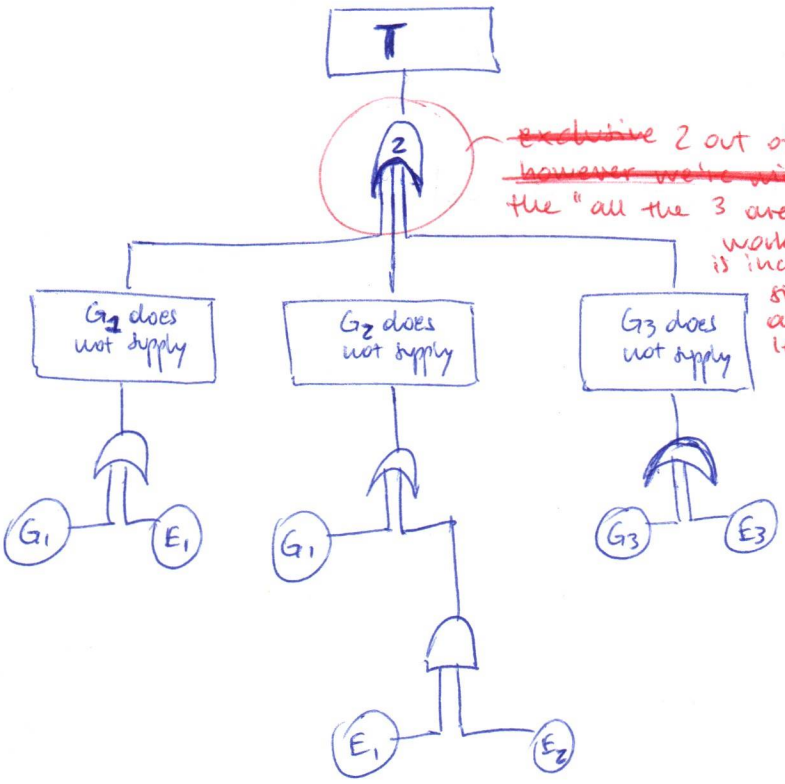
Generators (G1, G2 , G3)
Loads (2, 3, 4, 6, 7, 9, 10, 11, 12, 13, 14)
Power delivery paths: lines (L) and buses (B).



EX 1



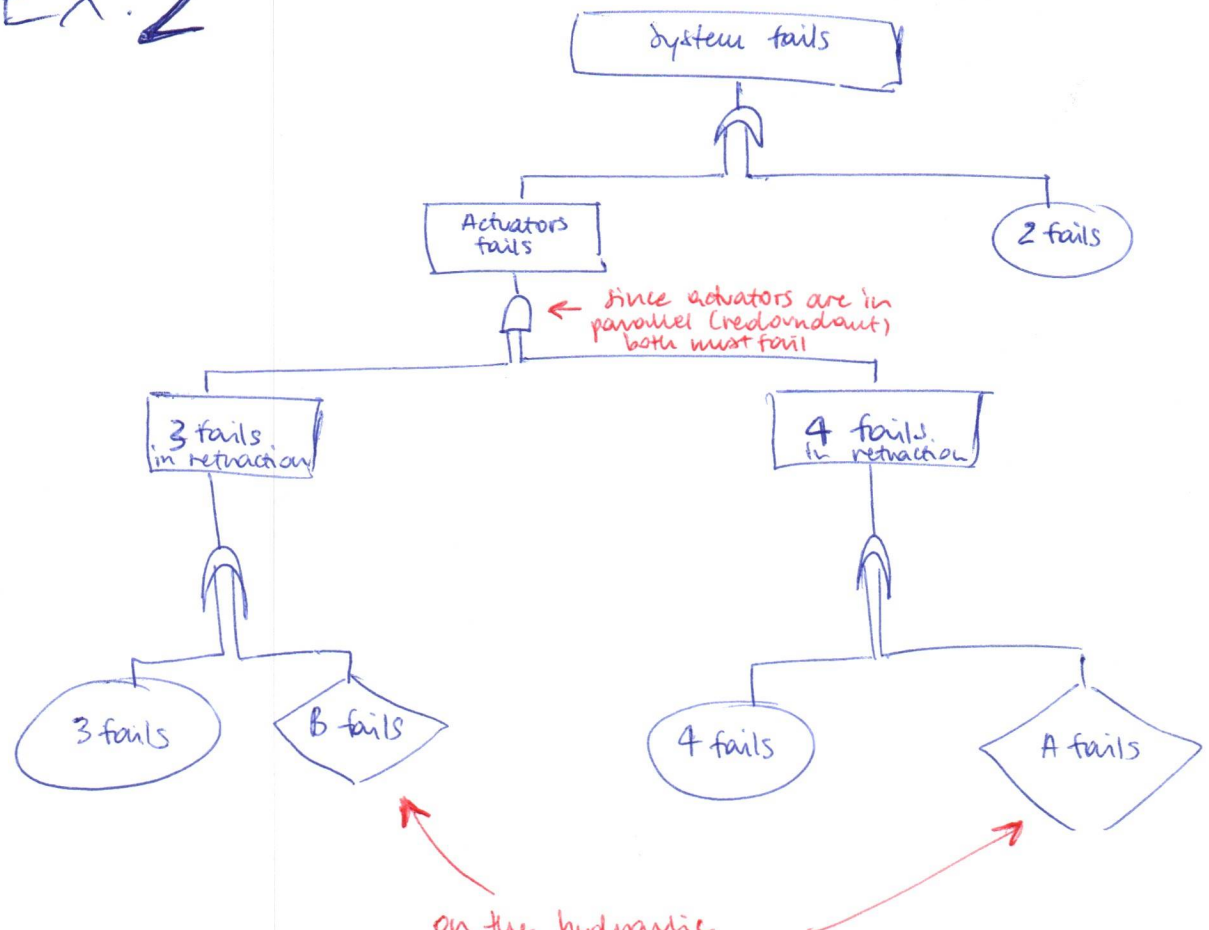
ALTERNATIVE



This is not to be included because ^ represents the union and the event "G_1, G_2, G_3 not supplying" is characterized by the union of any other two events listed (^ it's not exclusive)

+ disegno del prof.

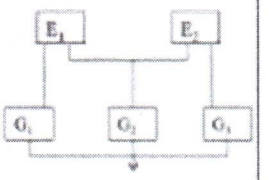
EX. 2

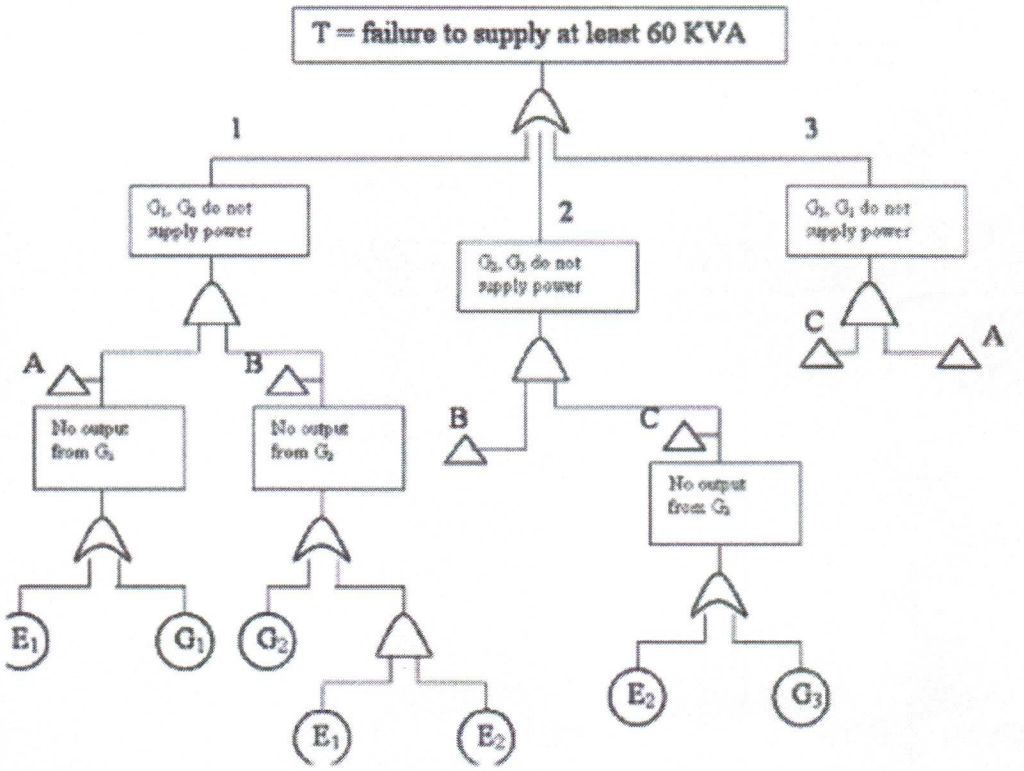


on the hydraulic controls we have to do more analysis (different ways of failing etc.)

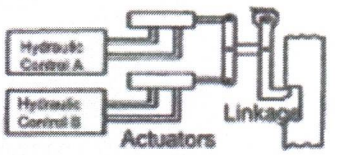
We assume the hydraulic control UNDEVELOPED events.

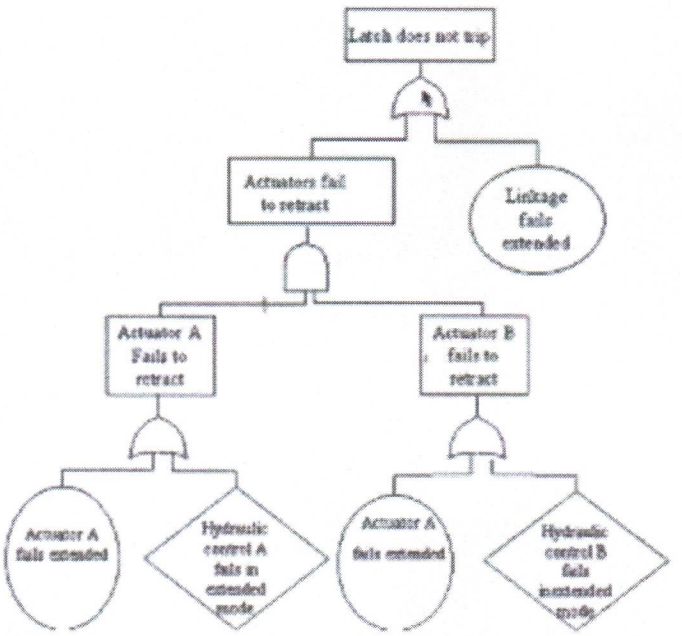
FT Example 1





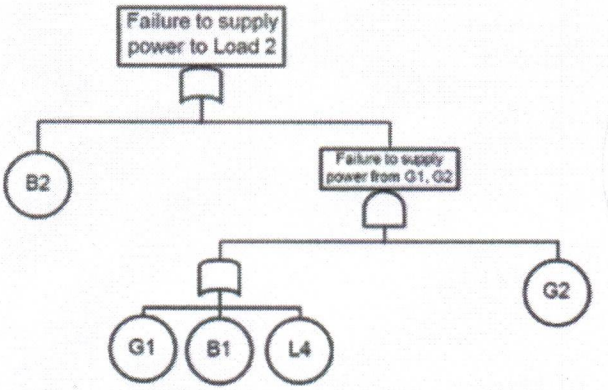
FT Example 2: The Fault Tree





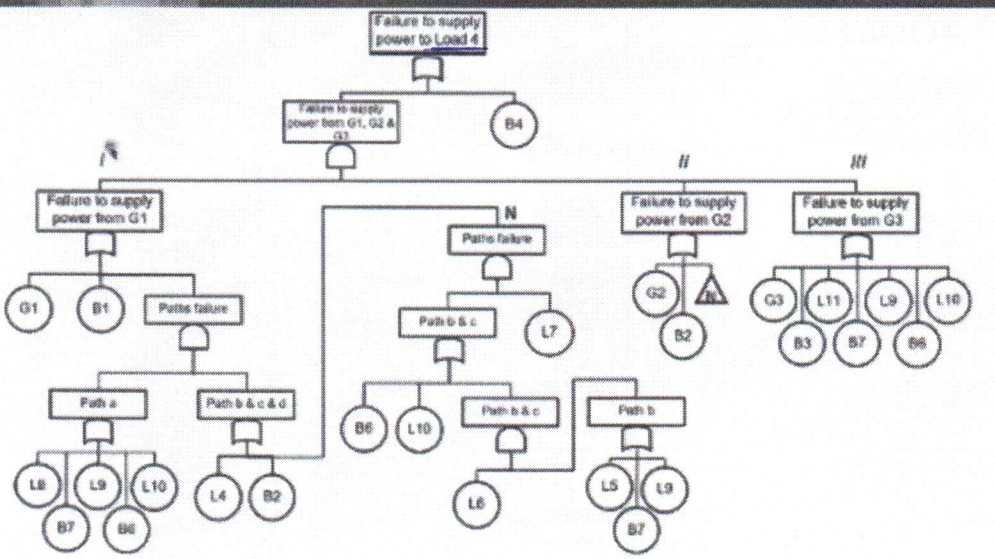
FT Example 4: IEEE14 Bus Power Distribution System

Find the Mcs for the top event "failure to supply power to bus 2" (Load2)

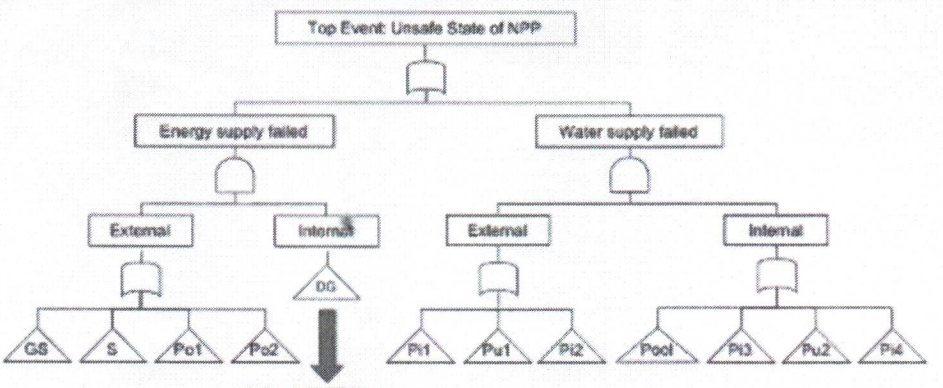


$M_1 = \{B_2\}$
$M_2 = \{G_1, G_2\}$
$M_3 = \{B_1, G_2\}$
$M_4 = \{L_4, G_2\}$

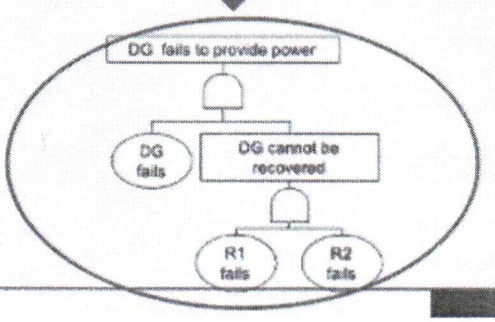
FT Example 4: IEEE14 Bus Power Distribution System



FT Example 3: Fault Tree



Example:



Hp: elements that fail can be immediately repaired/replaced if the access through the road system does not fail → roads considered as "reserve components".

FT Example 4: IEEE14 Bus Power Distribution System

Draw a Fault Tree (FT) for the top event "failure to supply power to bus 2" (Load2)

FT Example 4: IEEE14 Bus Power Distribution System

Draw a Fault Tree (FT) for the top event "failure to supply power to bus 4" (Load4)

FT qualitative analysis

We transformed the functional logic in a fault logic (fault trees) → what is this useful to?
We try to identify combinations of failures which make the fault event of the system occurs.
→ This is called QUALITATIVE ANALYSIS because we used no probability (yet).
(We just want to see in which configurations of its components the system is in a failed state)

We look for the minimal combinations that make the system fail. If A and B make the system fail we don't care to consider A and B and C.

How can we find minimal combinations in "automatic" ways?

FT qualitative analysis

Introducing:

X_i : binary indicator variable of i-th component state (basic event)

$$X_i = \begin{cases} 1 \text{ failure event } \underline{\text{true}} \\ 0 \text{ failure event } \underline{\text{false}} \end{cases}$$

= state of a component (component i)

for each event in the system we can have an indicator saying true or false.

"Generator 1 fails": $= \begin{cases} 1 \text{ true} \\ 0 \text{ false} \end{cases}$

"Generator 1 and 2 fails" $= \begin{cases} 1 \text{ true} \\ 0 \text{ false} \end{cases}$

(we have an indicator for **every** event)

- **FT = set of boolean algebraic equations (one for each gate) => structure (switching) function Φ:**

$$X_T = \Phi(X_1, X_2, \ldots, X_n)$$

The indicator variable of the top event X_T is a function of the states of all the events in the tree. This function is called: STRUCTURE FUNCTION because it represents the logic structure of the fault logic of the system

State of the top event; given all the states $(X_1, \ldots, X_n)$ and the logic in the fault tree (represented with $\Phi(\bullet)$) the indicator

$X_T = \begin{cases} 1 & \text{the top event occurred} \\ 0 & \text{the top event did } \underline{\text{not}} \text{ occur} \end{cases}$

For sure:

$X_1 = 1, \ldots, X_n = 1$: $\Rightarrow X_T = 1$

and:

$X_1 = 0, \ldots, X_n = 0$: $\Rightarrow X_T = 0$

Fundamental Products

- **FT = set of boolean algebraic equations (one for each gate) => structure (switching) function Φ:**

$$X_T = \Phi(X_1, X_2, \ldots, X_n)$$

An important theorem states that a structure function can be written uniquely as the union of the fundamental products which correspond to the combinations of the variables which render the function true. This is called the canonical expansion or disjunctive normal form of Φ.

fundamental product = product of a variable as it is or negated

Fundamental Laws

1) Commutative Law
   (a) $XY = YX$
   (b) $X + Y = Y + X$

2) Associative Law
   (a) $X(YZ) = (XY)Z$
   (b) $X + (Y + Z) = (X + Y) + Z$

3) Idempotent Law
   (a) $XX = X$
   (b) $X + X = X$

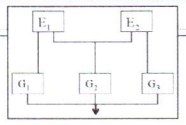
4) Absorption Law
   (a) $X(X + Y) = X$
   (b) $X + XY = X$

5) Distributive Law
   (a) $X(Y + Z) = XY + XZ$
   (b) $(X + Y)(X + Z) = X + YZ$

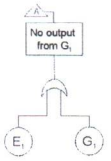
6) Complementation*
   (a) $X\bar{X} = \varnothing$
   (b) $X + \bar{X} = \Omega$
   (c) $\bar{\bar{X}} = X$

7) Unnamed relationships but frequently useful
   (a) $X + \bar{X}Y = X + Y$
   (b) $\bar{X}(X + Y) = \bar{X}\bar{Y}$

Structure function: Example 1

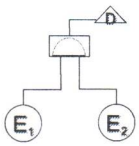


OR gate

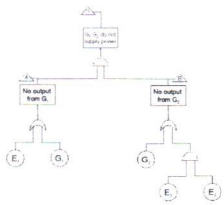


$$X_A = X_{E_1} + X_{G_1} - X_{E_1} X_{G_1} =$$
$$= 1 - (1 - X_{E_1})(1 - X_{G_1})$$

AND gate



$$X_D = X_{E_1} X_{E_2}$$



$$X_{T_1} = \Phi(X_{E_1}, X_{E_2}, X_{G_1}, X_{G2})$$

Coherent structure functions

A physical system would be quite unusual (or perhaps poorly designed) if improving the performance of a component (that is, replacing a failed component by a functioning one) caused the system to change from the success to the failed state.

Thus, we restrict consideration to structure functions that are monotonically increasing in each input variable. These structure functions do not contain complemented variables; they are called *coherent* and can always be expressed as the union of fundamental products.

The main properties of a coherent structure function are:

(Notice: we're changing notation)

1. $\Phi(1) = 1$ if all the components are in their success state, the system is successful;
2. $\Phi(0) = 0$ if all the components are failed, the system is failed;
3. $\Phi(\underline{X}) \geq \Phi(\underline{Y})$ for $\underline{X} \geq \underline{Y}$.

COHERENCE:
If a system works even if it has a component, say A, that then it MUST work also if we repair A. The repairing of a component CANNOT make a system not work.

FT qualitative analysis

Coherent structure functions can be expressed in reduced expressions in terms of minimal path or cut sets. A path set is a set $\underline{X}$ such that $\Phi(\underline{X}) = 1$; a cut set is a set $\underline{X}$ such that $\Phi(\underline{X}) = 0$. Physically, a path (cut) set is a set of components whose functioning (failure) ensures the functioning (failure) of the system.

- **Reduce Φ in terms of minimal cut sets (mcs)**

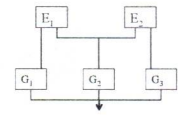
- **cut sets** = logic combinations of primary events which render true the top event (top event = failure)
- **minimal cut sets** = cut sets such that if one of the events is not verified, the top event is not verified

FT qualitative analysis

- FT = set of boolean algebraic equations (one for each gate) => structure (switching) function Φ:

$$X_T = \Phi(X_1, X_2, \ldots, X_n)$$

- Boolean algebra to solve FT equations



$$X_{T_1} = X_A X_B =$$

$$= (X_{E_1} + X_{G_1} - X_{E_1} X_{G_1})(X_{G_2} + X_{E_1} X_{E_2} - X_{E_1} X_{E_2} X_{G_2}) =$$

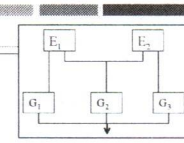
$$= X_{E_1} X_{G_2} + X_{E_1} X_{E_2} - X_{E_1} X_{E_2} X_{G_2} + X_{G_1} X_{G_2} + X_{E_1} X_{E_2} X_{G_1} +$$

$$- X_{E_1} X_{E_2} X_{G_1} X_{G_2} - X_{E_1} X_{G_1} X_{G_2} - X_{E_1} X_{E_2} X_{G_1} + X_{E_1} X_{E_2} X_{G_1} X_{G_2} =$$

$$= X_{E_1} X_{G_2} + X_{E_1} X_{E_2} + X_{G_1} X_{G_2} - X_{E_1} X_{E_2} X_{G_2} - X_{E_1} X_{G_1} X_{G_2}$$

we have to change the structure to get minimal cut sets

FT Example 1: mcs



$$X_{T_1} = X_{E_1} X_{G_2} + X_{E_1} X_{E_2} + X_{G_1} X_{G_2} - X_{E_1} X_{E_2} X_{G_2} - X_{E_1} X_{G_1} X_{G_2}$$

$$= 1 - [1 - X_{E_1} X_{G_2} - X_{E_1} X_{E_2} - X_{G_1} X_{G_2} + X_{E_1} X_{E_2} X_{G_2} + X_{E_1} X_{G_1} X_{G_2}] =$$

$$= 1 - [1 - X_{E_1} X_{G_2} - X_{E_1} X_{E_2} - X_{G_1} X_{G_2} + X_{E_1} X_{E_2} X_{G_2} + X_{E_1} X_{G_1} X_{G_2} + X_{E_1} X_{E_2} X_{G_1} X_{G_2} - X_{E_1} X_{E_2} X_{G_1} X_{G_2}] =$$

$$= 1 - [1 - X_{E_1} X_{E_2} - X_{G_1} X_{G_2} + X_{E_1} X_{E_2} X_{G_1} X_{G_2} - X_{E_1} X_{G_2} + X_{E_1} X_{E_2} X_{G_2} + X_{E_1} X_{G_1} X_{G_2} - X_{E_1} X_{E_2} X_{G_1} X_{G_2}] =$$

$$= 1 - [1 - X_{E_1} X_{E_2} - X_{G_1} X_{G_2} + X_{E_1} X_{E_2} X_{G_1} X_{G_2} - X_{E_1} X_{G_2} (1 - X_{E_1} X_{E_2} - X_{G_1} X_{G_2} + X_{E_1} X_{E_2} X_{G_1} X_{G_2})] =$$

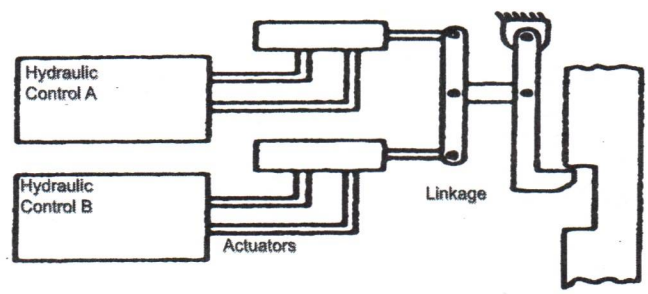
$$= 1 - [(1 - X_{E_1} X_{G_2})(1 - X_{E_1} X_{E_2} - X_{G_1} X_{G_2} + X_{E_1} X_{E_2} X_{G_1} X_{G_2})] =$$

$$= 1 - [(1 - X_{E_1} X_{G_2})(1 - X_{E_1} X_{E_2})(1 - X_{G_1} X_{G_2})]$$

3 minimal cut sets:

$$M_1 = \{E_1 G_2\}$$
$$M_2 = \{E_1 E_2\}$$
$$M_3 = \{G_1 G_2\}$$

FT Example 2: The Shutdown System



EX 2. MCS

$$X_T = 1 - (1 - X_{actuator})(1 - X_{linkage})$$

$$= 1 - (1 - X_{actuator\,A}\, X_{actuator\,B})(1 - X_{linkage})$$

$$= 1 - \left(1 - \left[1 - (1 - X_{AA})(1 - X_{HA})\right]\left[1 - (1 - X_{AB})(1 - X_{HB})\right]\right)(1 - X_{link})$$

$$= 1 - \left[1 - \left(1 - \left(1 - X_{AA} - X_{HA} + X_{AA} X_{HA}\right)\right)\left(1 - \left(1 - X_{AB} - X_{HB} + X_{AB} X_{HB}\right)\right)\right] \cdot \left(1 - X_{link}\right)$$

$$\geq 1 - \left[1 - \left(X_{AA} + X_{AH} - X_{AA} X_{HA}\right)\left(X_{AB} + X_{HB} + X_{AB} X_{HB}\right)\right](1 - X_{link})$$

5 minimal cut sets:

$M_1 = X_L$

$M_2 = X_{AA}\, X_{AB}$

$M_3 = X_{AH}\, X_{HB}$

$M_4 = X_{AA}\, X_{AH}$

$M_5 = X_{AB}\, X_{HB}$

check slides after

replicate. On the other hand, a 4th order MCS contains four basic events. The lower the order of a MCS the higher the importance of that MCS is. There are many algorithms available to perform the qualitative analysis of fault trees. A comprehensive list of these algorithms is available in (Ruijters & Stoelinga, 2015). The detail descriptions of these algorithms are out of scope of this paper. However, in this paper I briefly describe a popular algorithm — MOCUS (Fussel & Vesely, 1972).

MOCUS — Method of Obtaining Cut Sets

MOCUS (Fussel & Vesely, 1972) is a top-down approach and it is one of the primary standard fault tree analysis algorithms. Many other algorithms are developed based on this algorithm. This algorithm starts its operation with the top event of the fault tree and recursively explores the fault tree by expanding the intermediate events into their contributing basic events. This process continues until all the intermediate events are expanded and no more basic events are left in the tree.

Following are the steps of the algorithm (Walker, 2009):

1. Create a table where each row of the table represents a cut set and each column represents a basic event in the cut set.
2. Insert the top event of the fault tree in the first column of the first row.
3. Scan through the table, and for each fault tree gate:
    a. If the gate in an AND gate, then insert each of its input in a new column.
    b. If the gate is an OR gate, then insert each of its input in a new row.
4. Repeat step 3 until all the gates in the fault tree is explored and the table only contains the basic events.
5. Use Boolean laws to remove all redundancies within the table.

After performing all the above steps, each row of the resulting table will contain a minimal cut set. In terms of accuracy and speed of execution, MOCUS performs very well for smaller trees. However, the table size could grow much larger for large fault trees, hence this approach faces difficulties to analyze large fault trees. To illustrate the use of MOCUS algorithm for the analysis of fault trees, consider the fault tree of Fig. 5 used by Walker (2009).

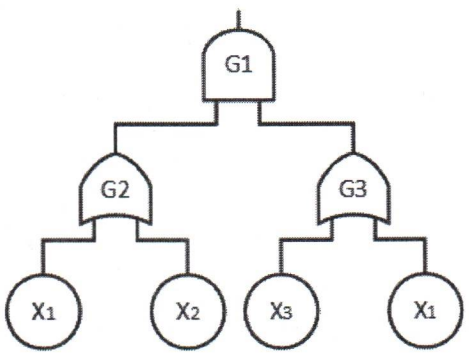


Fig 5. Fault tree to illustrate MOCUS algorithm

In the first step, a table is created and in the second step, the top event, G1 is put into the table.

G1

In the third step, this top event is expanded. As G1 is an AND gate, its inputs (G2 and G3) are put in a new column.

G2	G3

After that, G2 and G3 are expanded as well. As they are OR gates, their inputs are inserted in a new row.
G2 is expanded first:

X_1	G3
X_2	G3

G3 is expanded next:

X_1	X_3
X_1	X_1
X_2	X_3
X_2	X_1

Now no gates in the fault tree are left to be expanded. Boolean laws can be used to check if there exists any redundancies. Using the Idempotent law, two identical events in the same row can be reduced to just one, i.e. X_1 AND $X_1 \Leftrightarrow X_1$, thus:

X_1	X_3
X_1	
X_2	X_3
X_2	X_1

Using the Absorption law, a row can be eliminated if it contains all the elements of another row, so:

X_1	
X_2	X_3

This gives two minimal cut sets, one containing just X_1 and the other containing X_2 AND X_3.

Despite its limitations, MOCUS is one of the simplest and the most popular of FTA techniques, as evidenced by its 43 year life-span. As it is accurate and easy to understand, it is a good approach to analyse smaller fault trees; moreover, it makes an excellent foundation for further expansion – or extension – with new techniques. However, it is not the most efficient technique and algorithms like MICSUP (Pande, Spector, & Chatterjee, 1975), ELRAFT (Semanderes, 1971) tend to be faster.

Quantitative analysis of a fault tree can estimate the top event occurrence probability from the given failure rates/probabilities of basic failure events of the system (Vesely et al., 1981). In the quantification process, the basic events are usually assumed to be statistically independent. However, methodologies like Bobbio, Portinale, Minichino, & Ciancamerla (2001) can quantify fault trees with statistically dependent events. Usually, in FTA, as the top event is represented as the disjoint sum of the MCS, an approximate value of the probability of the top event can be determined by calculating the probability of each MCS and then adding them together, given that the probability of MCSs are small. In the Fault Tree Handbook (Vesely et al., 2002), this approximation is termed as "rare event approximation" and it is also stated that if the basic events probabilities are below 0.1 then this approximation are typically sufficiently accurate. In addition to this approximation, depending on the applications, different kinds of probabilities like time-dependent probabilities could be calculated provided that the proper failure distributions of the components/events are available.

To be able to perform quantitative analysis to get top event probability, the basic events are usually given one of the following types of data (Vesely et al., 2002):

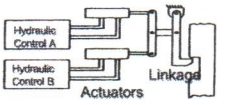
1. an event occurrence probability or a component failure probability in some time interval,
2. unavailability of a component, and
3. a pure event probability.

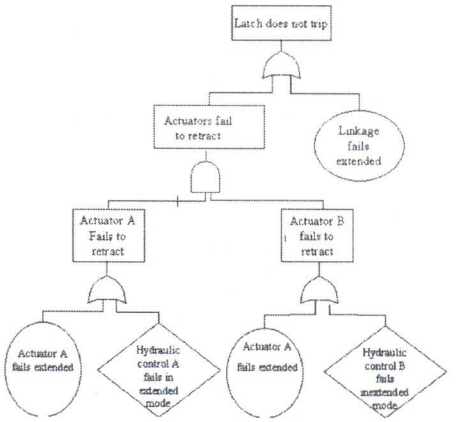
A detail description of the quantitative analysis of the fault trees is available in (Ruijters & Stoelinga, 2015). Although the primary focus of the quantitative analysis of a fault tree is to estimate the top event probability, it is possible to estimate the probability of any intermediate events as well as the basic events. Dominance of the minimal cut sets could be determined based on the significance of their contribution to the top event. The cut set which contributes the most to the top event is considered as the most dominant. In addition to determining dominant MCS, importance of basic events could also be obtained in the similar way.

2.3 Limitations of Standard Fault Tree

Modern large and complex systems are becoming increasingly dynamic in nature. Such systems often have the capability to response to failure by partial self-repair. For example, they may equipped with backup components and

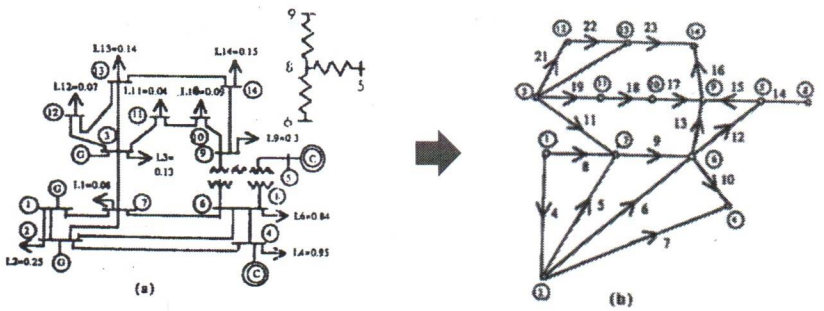
FT Example 2: The Fault Tree





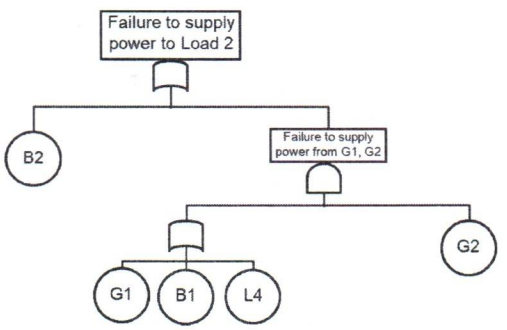
FT Example 4: IEEE14 Bus Power Distribution System

Generators (G1, G2 , G3)
Loads (2, 3, 4, 6, 7, 9, 10, 11, 12, 13, 14)
Power delivery paths: lines (L) and buses (B).



FT Example 4: IEEE14 Bus Power Distribution System

Find the Mcs for the top event “failure to supply power to bus 2” (Load2)



FT qualitative analysis: results

1. mcs identify the component basic failure events which contribute to system failure
2. qualitative component criticality: those components appearing in low order mcs or in many mcs are most critical

cut-set of order k: it contains k components
cut set of order 1: it takes the failure of 1 component to fail the system

Notice: probability has to be introduced. If a minimal cut set of order 2 has a higher probability to happen (to fail) than the minimal cut set of order 2 may be more critical of a minimal cut set of order 1)

Without probabilities the lower the order of the cut set the more critical the involved components are (if a cut set has only the element A and another cut set has the elements B,C,D,E then A is most critical as the failure of A fails the system. If B or C or D or E fail alone then the system does not fail)
However, probabilities play a fondamental role!

FT quantitative analysis

System diagram → fault tree → boolean algebra equation

→ most reduced form & minimal cut sets → probabilistic terms

How?
Two ways

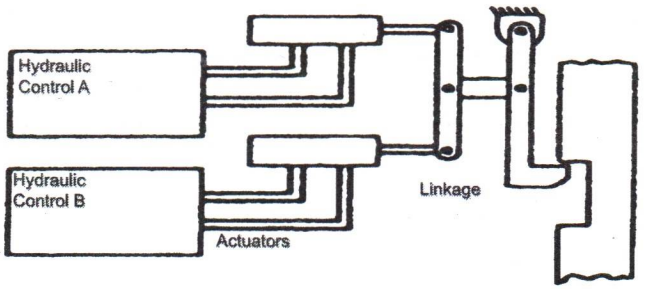
FT quantitative analysis

Compute system failure probability from primary events probabilities by:

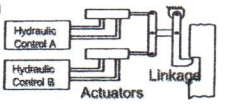
1. using the laws of probability theory at the fault tree gates

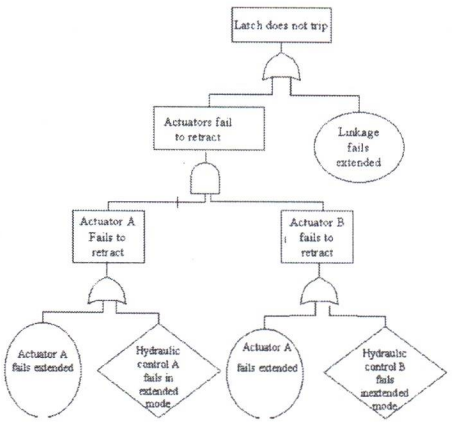
Instead of substituting for each gate its boolean algebra equation, we substitute with the probability law for that gate. The probability of the union/intersection → ..

FT Example 2: The Shutdown System



FT Example 2: The Fault Tree

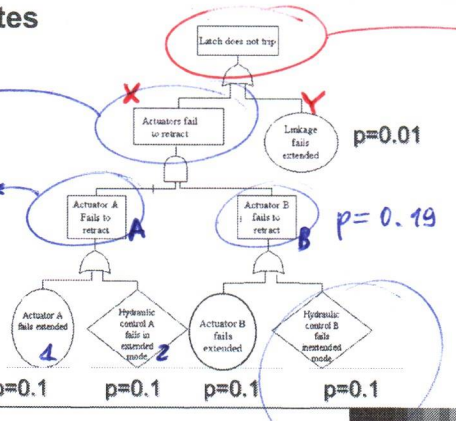




FT quantitative analysis-Example 2

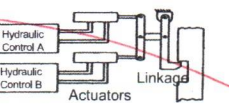
Compute system failure probability from primary events probabilities by:

1. **using the laws of probability theory at the fault tree gates**



$IP(A \cap B) = IP(A,B) \overset{!}{=} IP(A)\,IP(B) \overset{!}{=} (0.19)^2 = 0,0361$

$IP(1 \cup 2) = IP(1) + IP(2) - IP(1,2)$
since $1 \perp\!\!\!\perp 2$ (they're basic events) we have:
$IP(1 \cup 2) = IP(1) + IP(2) - IP(1)\,IP(2) \overset{!}{=} 0.1 + 0.1 - (0.1)^2 \overset{!}{=} 0.19$



$IP(X \cup Y) = IP(X) + IP(Y) - IP(X,Y) \overset{!}{=} IP(X) + IP(Y) - IP(X)\,IP(Y) \overset{!}{=} 0,0361 + 0.01 - 0.0361 \cdot 0.01 \overset{!}{=} 0.0457$

The reliability of the system is $\sim 1 - 0.0457$

we overestimate usually (since doing the analysis would be expensive)

FT quantitative analysis

Compute system failure probability from primary events probabilities by:

1. **using the laws of probability theory at the fault tree gate**
2. **using the mcs found from the qualitative analysis**

$$P[\Phi(\underline{X})=1] = \sum_{j=1}^{mcs} P[M_j] - \sum_{i=1}^{mcs-1}\sum_{j=i+1}^{mcs} P[M_i M_j] + \cdots + (-1)^{mcs+1} P[\prod_{j=1}^{mcs} M_j]$$

probability of the failure of the system

we recompose it from the union of the minimal cut sets (minimal cut set $j \rightarrow M_j$)

! NOTE: we wrote $\mathbb{P}(M_i M_j)$ NOT $\mathbb{P}(M_i)\mathbb{P}(M_j)$

THE MINIMAL CUT SETS MAY NOT BE INDEPENDENT.

It can be shown that:

$$\sum_{j=1}^{mcs} P[M_j] - \sum_{i=1}^{mcs-1}\sum_{j=i+1}^{mcs} P[M_i M_j] \le P[\Phi(\underline{X})=1] \le \sum_{j=1}^{mcs} P[M_j]$$

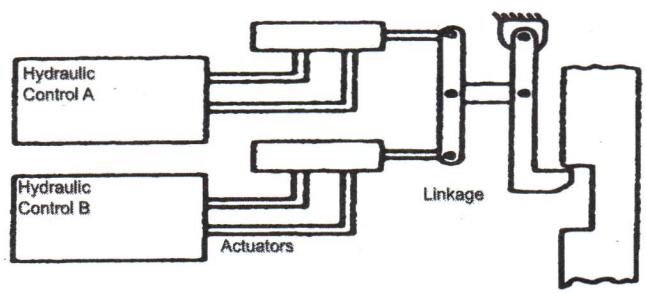
GOOD APPROXIMATION of $\mathbb{P}(\Phi(\underline{X})=1)$

with the simple sum we OVERESTIMATE → VERY GOOD!!

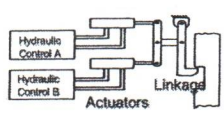
RARE EVENTS APPROXIMATION

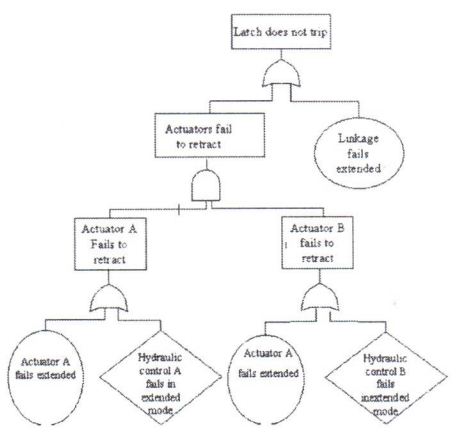
Rare because the components in the minimal cut sets rarely fail and it is even rarer that the whole cut set fails (moreover it's even rarer that multiple cut sets fail, that's why we neglect them)

FT Example 2: The Shutdown System

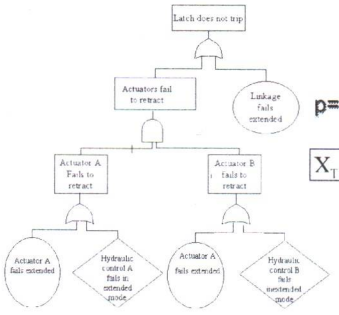


FT Example 2: The Fault Tree

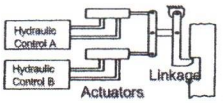




FT Example 2: mcs



p=0.01



$$X_T = 1-(1-X_L)(1-(X_A+X_{HA}-X_AX_{HA})(X_B+X_{HB}-X_BX_{HB}))$$

p=0.1 p=0.1 p=0.1 p=0.1

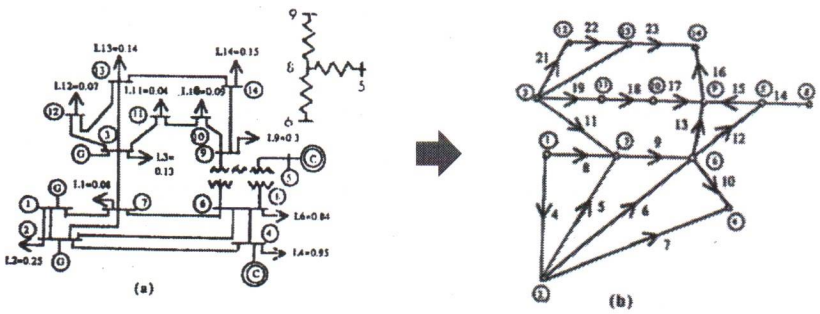
5 minimal cut sets:

$M_1=X_L$
$M_2=X_AX_B$
$M_3=X_AX_{HB}$
$M_4=X_{HA}X_B$
$M_5=X_{HA}X_{HB}$

Quantitative analysis?

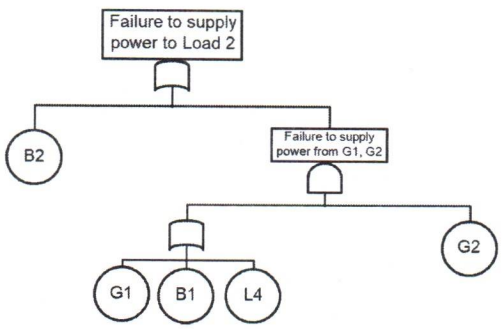
FT Example 4: IEEE14 Bus Power Distribution System

Generators (G1, G2 , G3)
Loads (2, 3, 4, 6, 7, 9, 10, 11, 12, 13, 14)
Power delivery paths: lines (L) and buses (B).



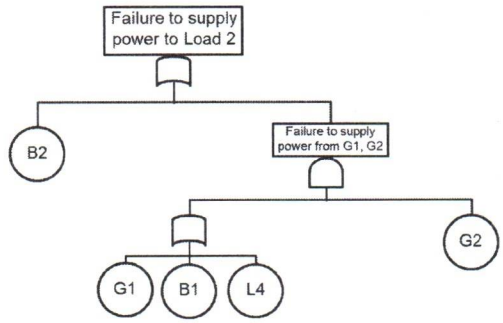
FT Example 4: IEEE14 Bus Power Distribution System

Find the Mcs for the top event "failure to supply power to bus 2" (Load2)



FT Example 4: IEEE14 Bus Power Distribution System

Find the Mcs for the top event "failure to supply power to bus 2" (Load2)

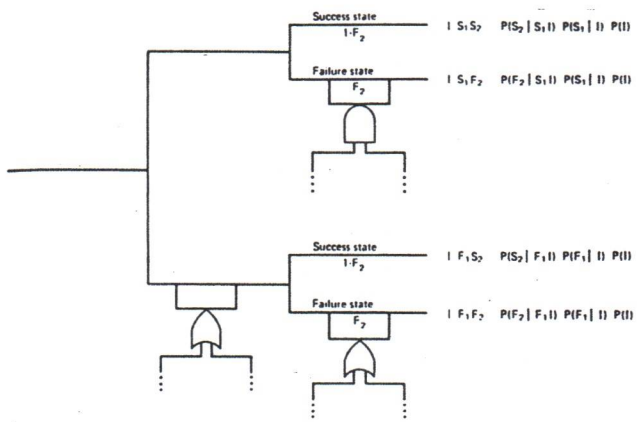


$M_1 = \{B_2\}$
$M_2 = \{G_1, G_2\}$
$M_3 = \{B_1, G_2\}$
$M_4 = \{L_4, G_2\}$

FT: comments

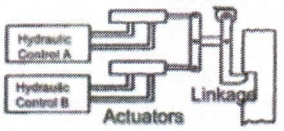
1. **Straightforward modelization via few, simple logic operators;**
2. **Focus on one top event of interest at a time;**
3. **Providing a graphical communication tool whose analysis is transparent;**
4. **Providing an insight into system behaviour;**
5. **Minimal cut sets are a synthetic result which identifies the critical components.**

ETA+FTA



Quantitative analysis (static)

FT quantitative analysis: Example 2



5 mcs:

$P(M_1) = P(X_L = 1) = 0.01$

$P(M_2) = P(X_A X_B = 1) = 0.1 \cdot 0.1 = 0.01$

$P(M_3) = P(X_A X_{HB}) = 0.1 \cdot 0.1 = 0.01$

$P(M_4) = P(X_{HA} X_B = 1) = 0.1 \cdot 0.1 = 0.01$

$P(M_5) = P(X_{HA} X_{HB}) = 0.1 \cdot 0.1 = 0.01$

⇩

Rare event approximation:

$$P[\Phi(\underline{X}) = 1] \le \sum_{j=1}^{mcs} P[M_j] = 0.05$$

$$P[\Phi(\underline{X}) = 1] \ge \sum_{j=1}^{mcs} P[M_j] - \sum_{i=1}^{mcs-1} \sum_{j=i+1}^{mcs} P[M_i M_j] = 0.0464$$

underestimate (the real value (even if there is an error) should be 0.048)

As we go on with the expansion we gain precision but: