Exercise session: Fault Trees and Event Trees

Course: Resilience and Security of Critical Infrastructures

Maria Valentina Clavijo Mesa

- Emergency cooling system
- Electrical pump
- Simple network system
- Power distribution system

- A computational tool useful for FTA

Emergency cooling system

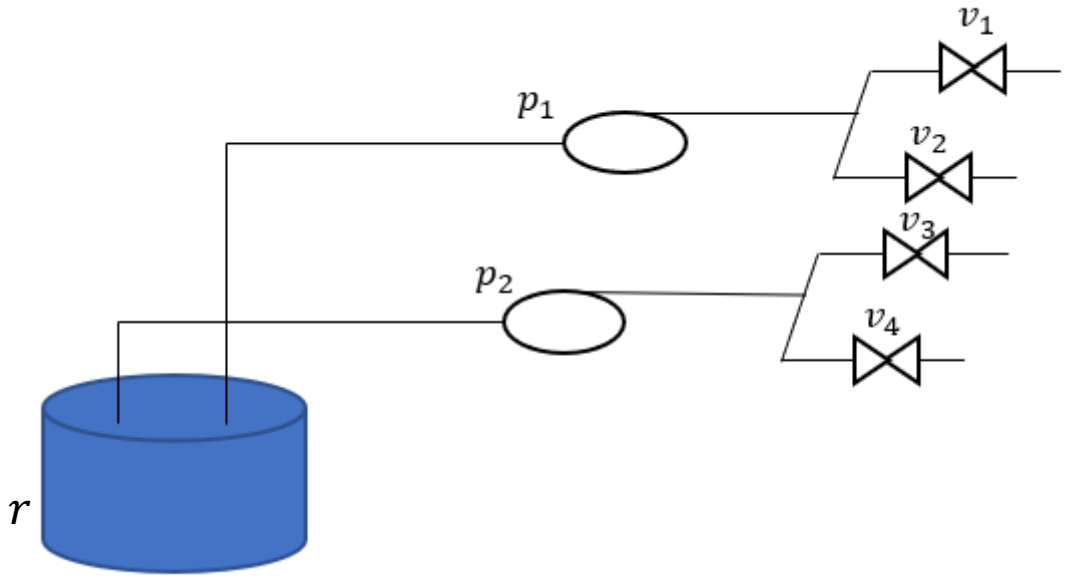
The following system is designed to deliver emergency cooling to a nuclear reactor.

In the event of an accident, ***the protection system delivers an actuation signal*** to two identical pumps and four identical valves.

The pumps then start, the valves open, and liquid coolant is delivered to the reactor.

The following failure probabilities are found to be significant:

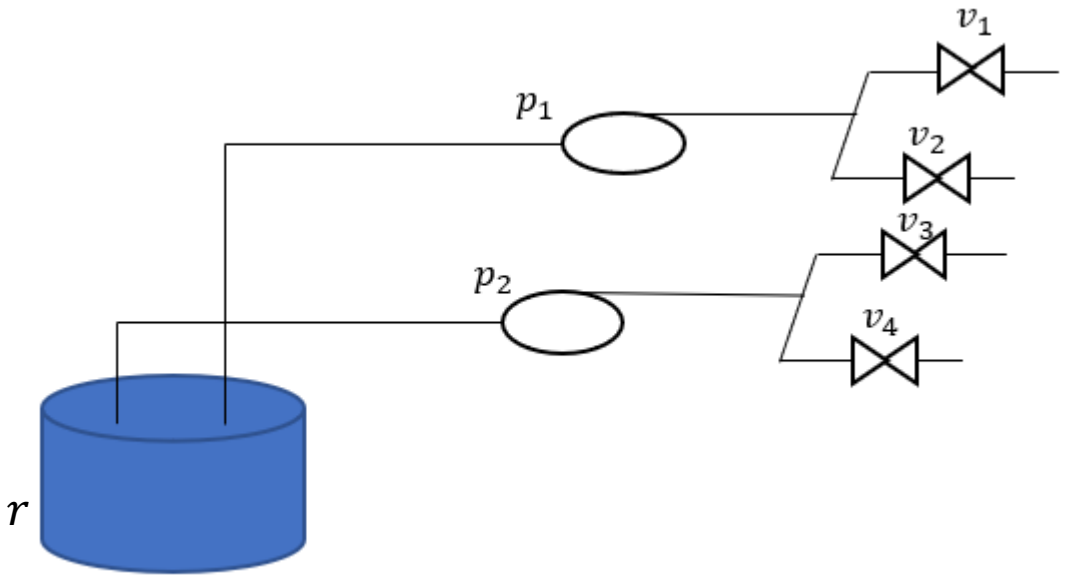
- Probability that the protection system (ps) will not deliver signal: P_{ps}=0.05
- Probability that a pump (p) fails: P_p=0.1
- Probability that a valve (p) fails: P_v=0.02
- Probability that the reservoir (r) is empty: P_r=0.08



1. Emergency cooling system

Exercise session: Fault Trees and Event Trees

- Draw a Fault Tree (FT) for *the failure of the system to deliver cooling to the nuclear reactor*.
- Identify the *minimal cut-sets* of the system.
- Compute *the probability of the top event*.



- Probability that the protection system (ps) will not deliver signal: P_{ps}=0.05
- Probability that a pump (p) fails: P_p=0.1
- Probability that a valve (p) fails: P_v=0.02
- Probability that the reservoir (r) is empty: P_r=0.08

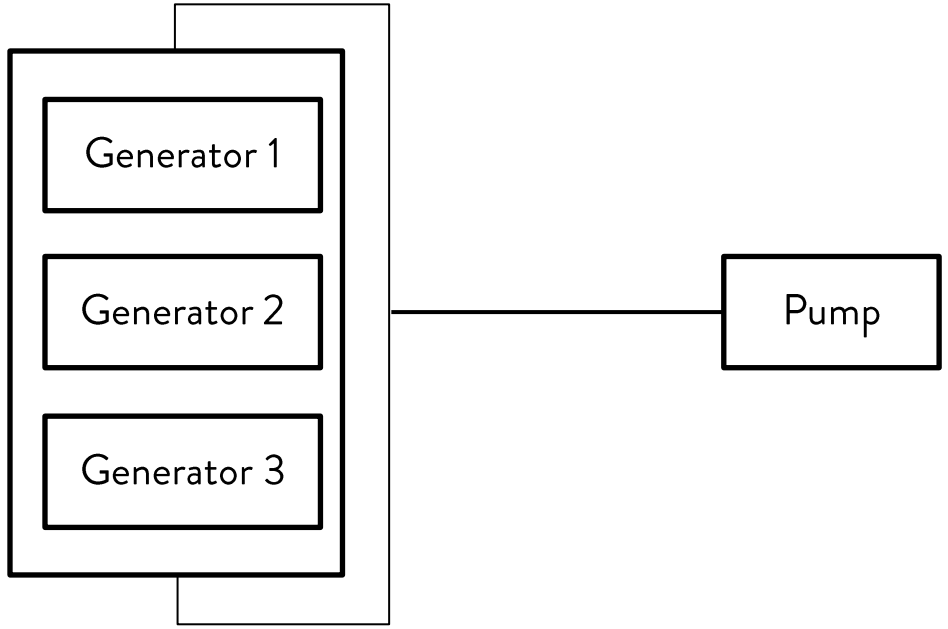
Electrical pump system

An electrical pump (P) is powered by a group of three electric generators (G). Due to the high-power demand of the pump, at least two of the generators must be in operation.

Design the ET to calculate the probability of system operation, following the structure in the header event table. Use the given reliability data for quantification:

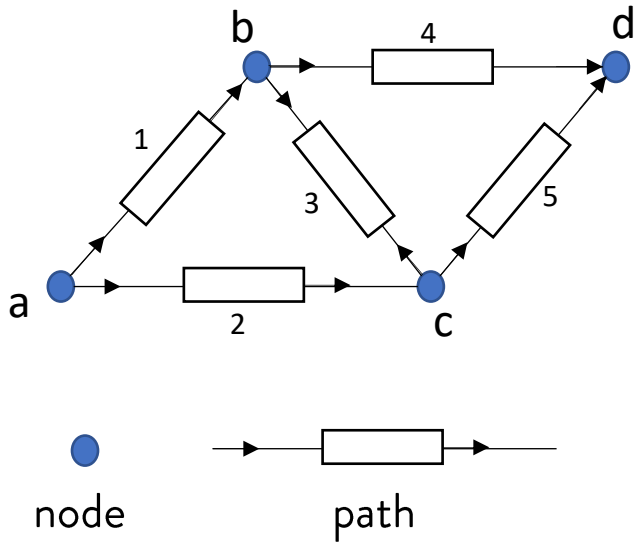
$$R_P = 0.9, \qquad R_{G1} = R_{G2} = R_{G3} = 0.85$$

P	G1	G2	G3



Simple network system

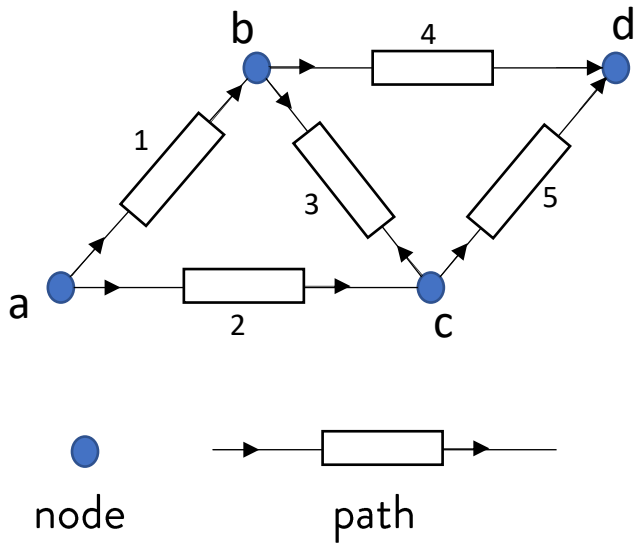
Consider the following network system. All components have an equal failure rate of $\lambda = 10^{-4} days^{-1}$. The system fails when there is no connection between node a and node d.



3.A. Simple network system

Exercise session: Fault Trees and Event Trees

- Assuming the nodes are perfect (i.e., they cannot fail), develop a Fault Tree (FT) for the event *'No signal at D given a signal at A'*.
- Identify the *minimal cut-sets* of the network system.
- Evaluate *the system unreliability for a mission time of 1 year*.

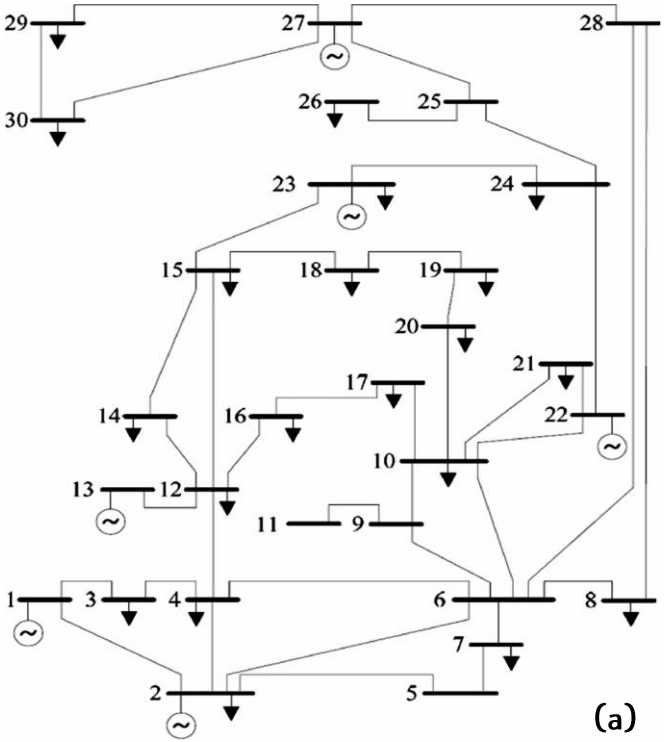


- Assuming the nodes are perfect (i.e., they cannot fail), build *the ET to calculate the probability of network operation*, starting from an initiating event and following the given header events.

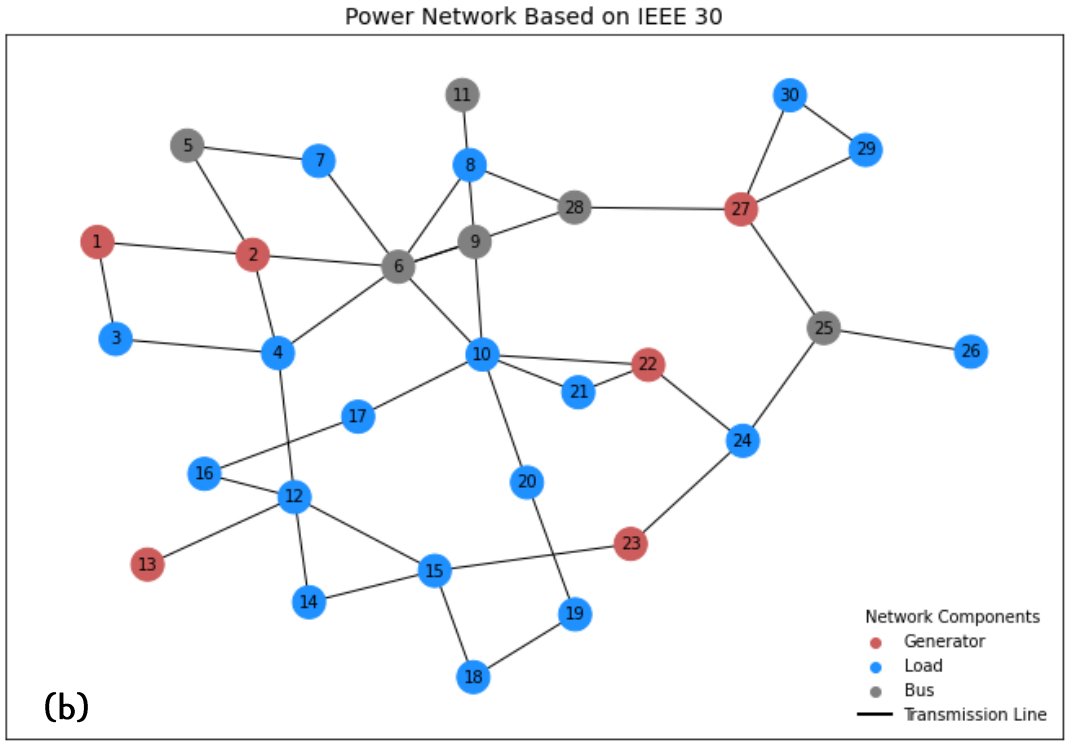
1	2	3	4	5

Power distribution system

The standard IEEE 30-bus power distribution system is shown in Figure (a). Only the major components are considered: generators (6 generators), loads (20 loads), and power delivery paths consisting of lines (L) and buses (B). The stochastic network is shown in Figure (b)

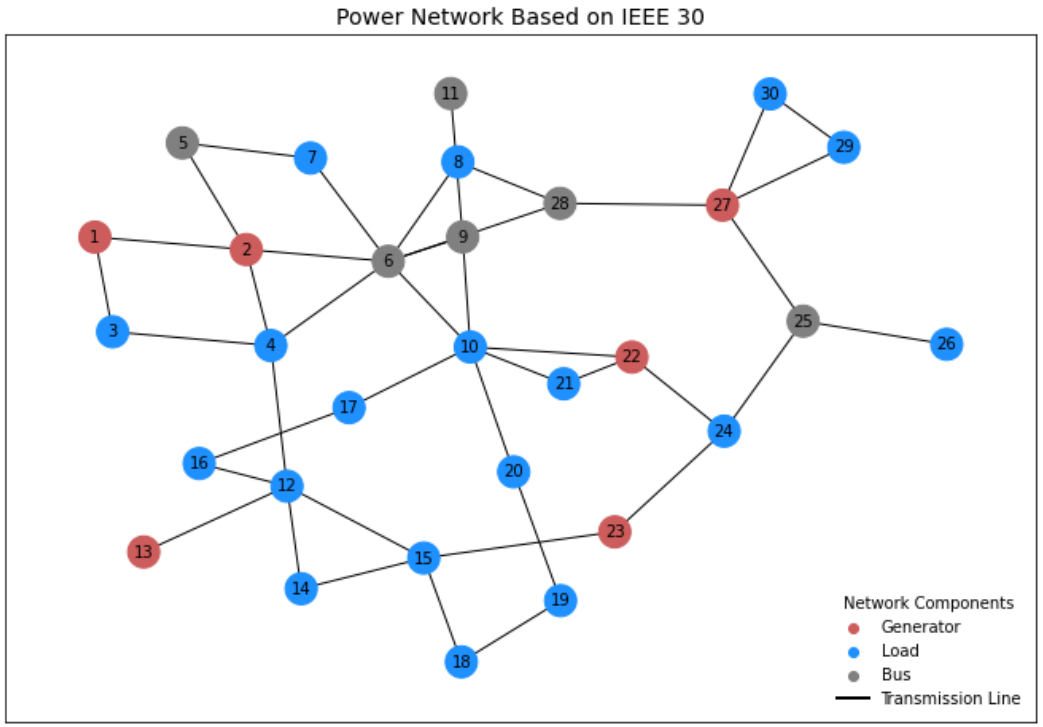


(a)

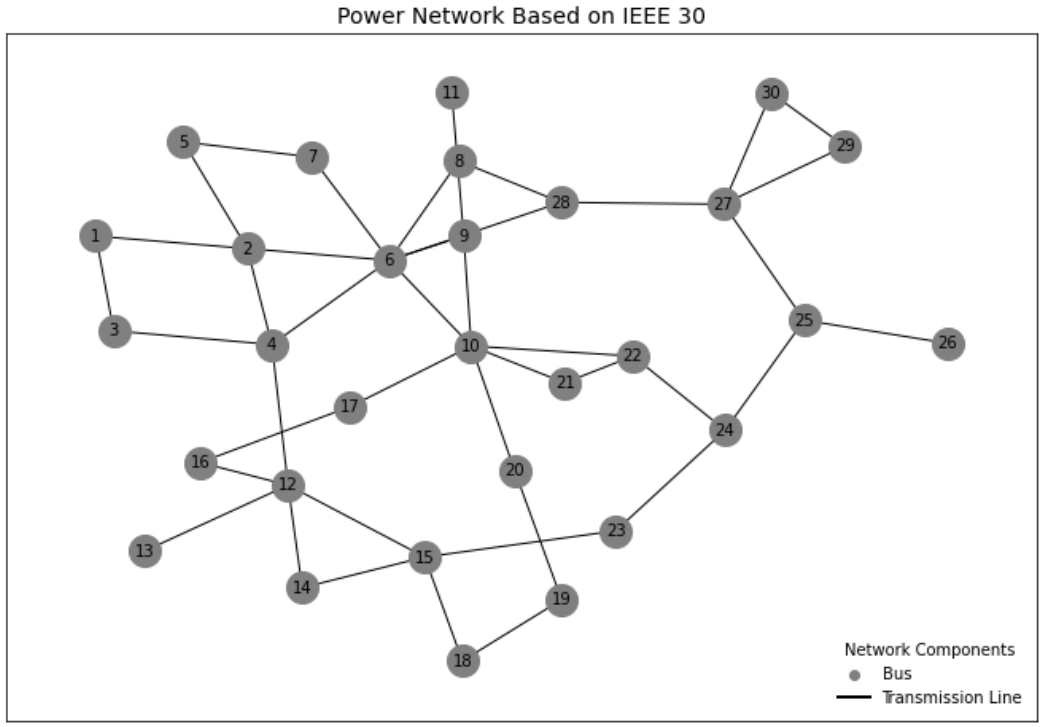


(b)

Power distribution system: IEEE 30-bus power distribution system



A load is a power consumer that does not fail but can be disconnected due to upstream failures

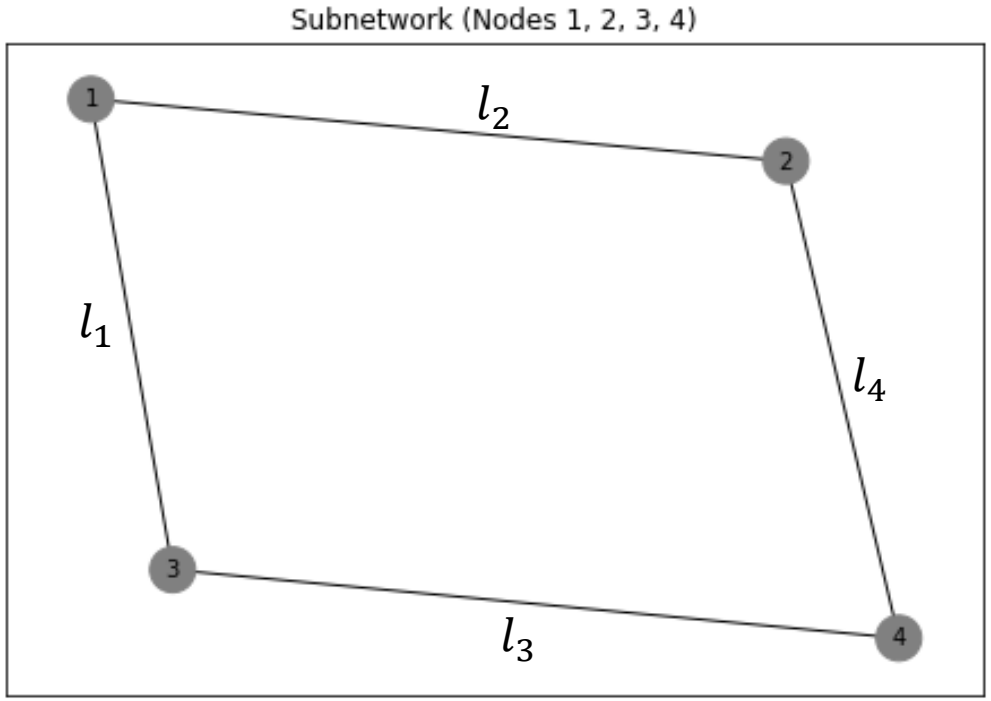


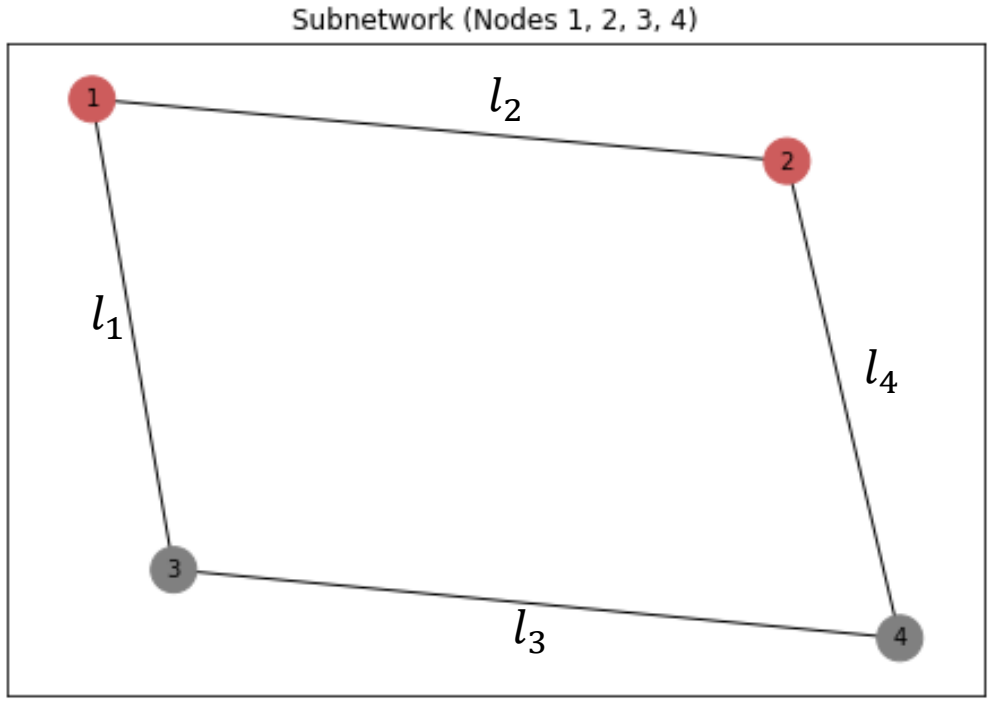
All nodes have buses that act as hubs where electrical power is either injected (from generators), withdrawn (by loads), or transferred across the network. If a bus fails, the corresponding node becomes inoperable

4.A. Power Distribution System

Exercise session: Fault Trees and Event Trees

- Draw a FT for the failure of the system to *supply power to the load demand at Bus 4 (load 4)*.
- Identify the *minimal cut-sets* and calculate the probability of the top event, assuming each element has a failure probability of 0.01





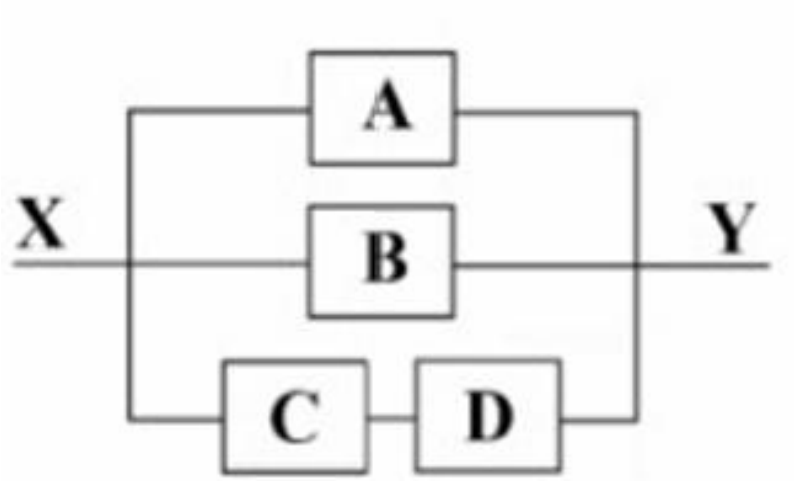
- Draw an ET for *the failure of the system to supply power to the load demand at bus 4 (load 4),* starting form an initiating event and following the given header events. Then, calculate the probability of network failure

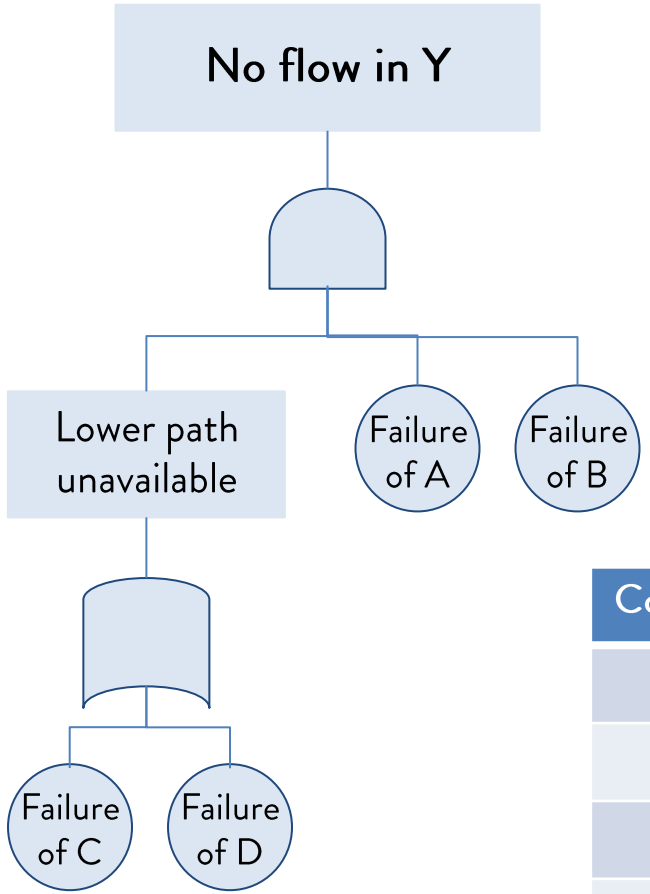
G1	B1	L1	B3	L3	L2	G2	B2	L5	B4

$$R_i = 0.99 \qquad Q_i = 0.01$$

A computational tool useful for FTA

Exercise session: Fault Trees and Event Trees





Component	Probability of failure
A	0.1
B	0.1
C	0.1
D	0.2

```
# Install and load the FaultTree package
install.packages("FaultTree", repos="http://cran.us.r-project.org")
library(FaultTree)
```

Main functions in the FaultTree package

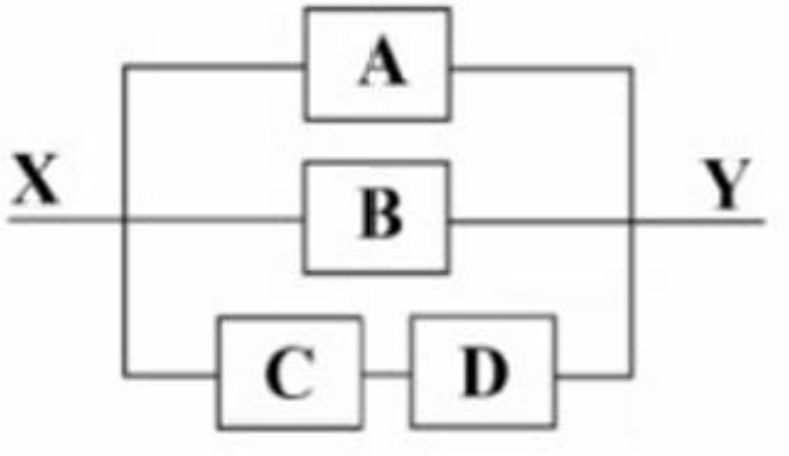
Function	Description
ftree.make()	Initializes a **new fault tree** with a **top event**
ftree.calc()	Performs **step-by-step calculations** from bottom to top
addLogic()	Introduces **intermediate events** that connect failure causes
addProbability()	Represents a **basic component failure** with an assigned probability
addDuplicate()	Used to **reuse events** without redefining them

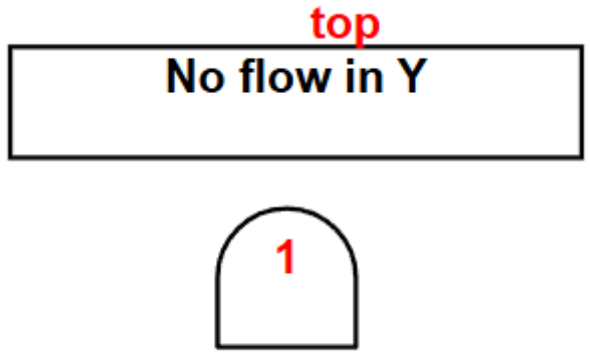
Install and load the FaultTree package

```
install.packages("FaultTree", repos="http://cran.us.r-project.org")
library(FaultTree)
```

Create the Fault Tree

```
system <- ftree.make(type="and", name="No flow in Y")
```

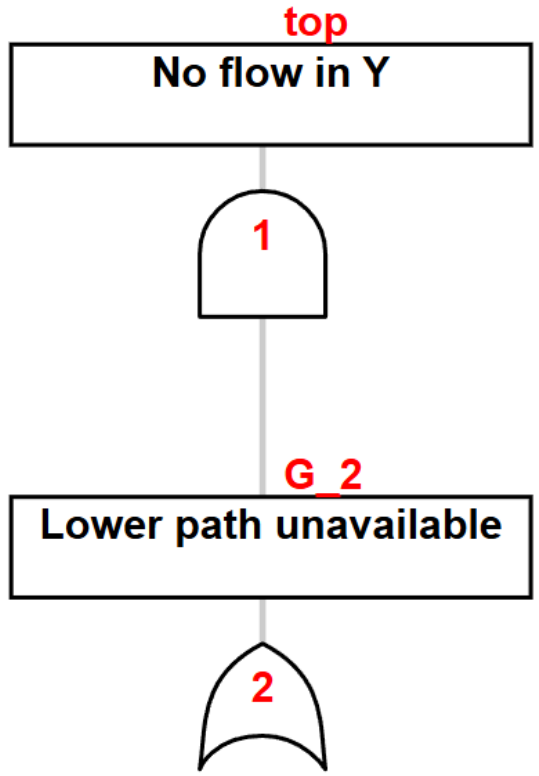




Visualize the FT

```
ftree2html(system, write_file = TRUE)
browseURL("system.html")
```

```
# Add a logical OR gate for path failure
system <- addLogic(system, at=1, type="or", name="Lower path unavailable")
```




```
# Visualize the FT
ftree2html(system, write_file = TRUE)
browseURL("system.html")
```

A computational tool useful for FTA

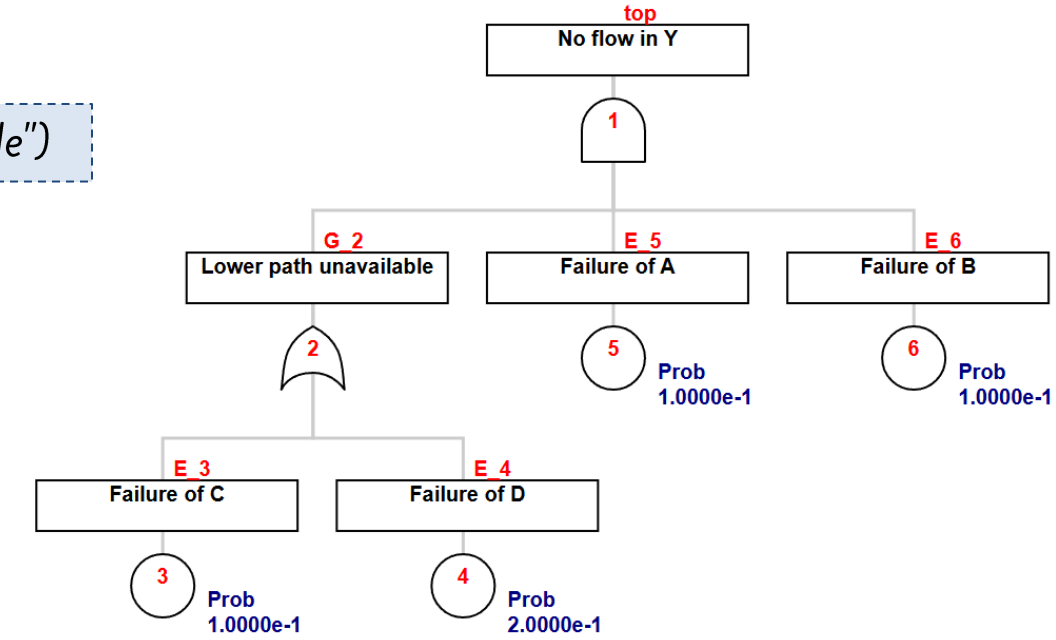
Exercise session: Fault Trees and Event Trees

 # Add a logical OR gate for path failure

```
system <- addLogic(system, at=1, type="or", name="Lower path unavailable")
```

 # Add basic failure probabilities

```
system <- addProbability(system, at=2, prob=0.1, name="Failure of C")
system <- addProbability(system, at=2, prob=0.2, name="Failure of D")
system <- addProbability(system, at=1, prob=0.1, name="Failure of A")
system <- addProbability(system, at=1, prob=0.1, name="Failure of B")
```



Visualize the FT

```
ftree2html(system, write_file = TRUE)
browseURL("system.html")
```

A computational tool useful for FTA

Exercise session: Fault Trees and Event Trees

 # Add a logical OR gate for path failure

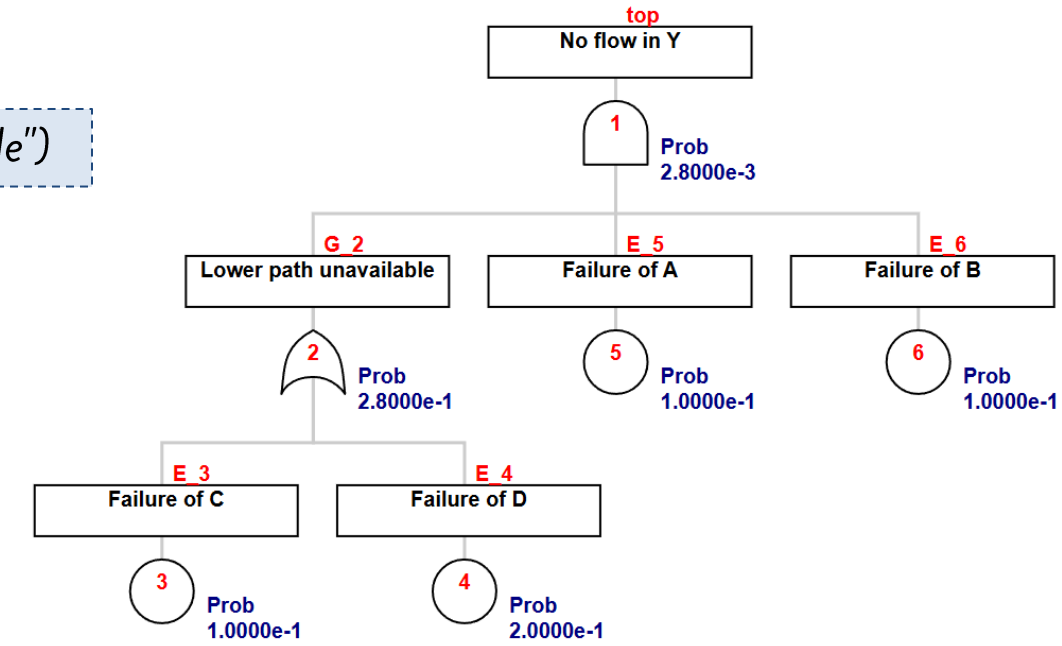
```
system <- addLogic(system, at=1, type="or", name="Lower path unavailable")
```

 # Add basic failure probabilities

```
system <- addProbability(system, at=2, prob=0.1, name="Failure of C")
system <- addProbability(system, at=2, prob=0.2, name="Failure of D")
system <- addProbability(system, at=1, prob=0.1, name="Failure of A")
system <- addProbability(system, at=1, prob=0.1, name="Failure of B")
```

 # Compute fault tree probabilities

```
system <- ftree.calc(system)
```



Visualize the FT

```
ftree2html(system, write_file = TRUE)
browseURL("system.html")
```

THANKS

Maria Valentina Clavijo Mesa

mariavalentina.clavijo@polimi.it