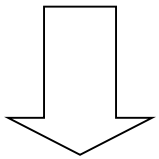
Quantitative evaluation of Dependability

Quantitative evaluation of Dependability

- Faults are the cause of errors and failures. Does the arrival time of faults fit a **probability distribution**?
  If so, what are the parameters of that distribution?

- Consider the time to failure of a system or component.
  It is not exactly predictable - **random variable**.



probability theory

Quantitative evaluation of failure rate, Mean Time To Failure (MTTF), Mean Time To Repair (MTTR), Reliability function (R(t)), Availability function (A(t)) and Safety function (S(t))

Quantitative definition of dependability attributes

Reliability - R(t)

conditional probability that the system performs correctly throughout the *interval of time* [t0, t], given that the system was performing correctly at the *instant* of time t0

Availability - A(t)

the probability that the system is operating correctly and is available to perform its functions at the *instant* of time t

Safety – S(t)

the probability that the system either behaves correctly or will discontinue its functions in a manner that causes no harm throughout the *interval of time* [t0, t], given that the system was performing correctly at the *instant* of time t0

Definitions

Reliability R(t)

$$R(0) = 1 \quad R(\infty) = 0$$

Failure probability Q(t)

$$Q(t) = 1 - R(t)$$

Failure probability density function f(t)

the failure density function f(t) at time t is the number of failures in Δt

$$f(t) = \frac{dQ(t)}{dt} = \frac{-\,dR(t)}{dt}$$

Failure rate function $\lambda(t)$

the failure rate $\lambda(t)$ at time t is defined by the number of failures during Δt in relation to the number of correct components at time t

$$\lambda(t) = \frac{f(t)}{R(t)} = \frac{-\,dR(t)}{dt}\,\frac{1}{R(t)}$$

Hardware Reliability

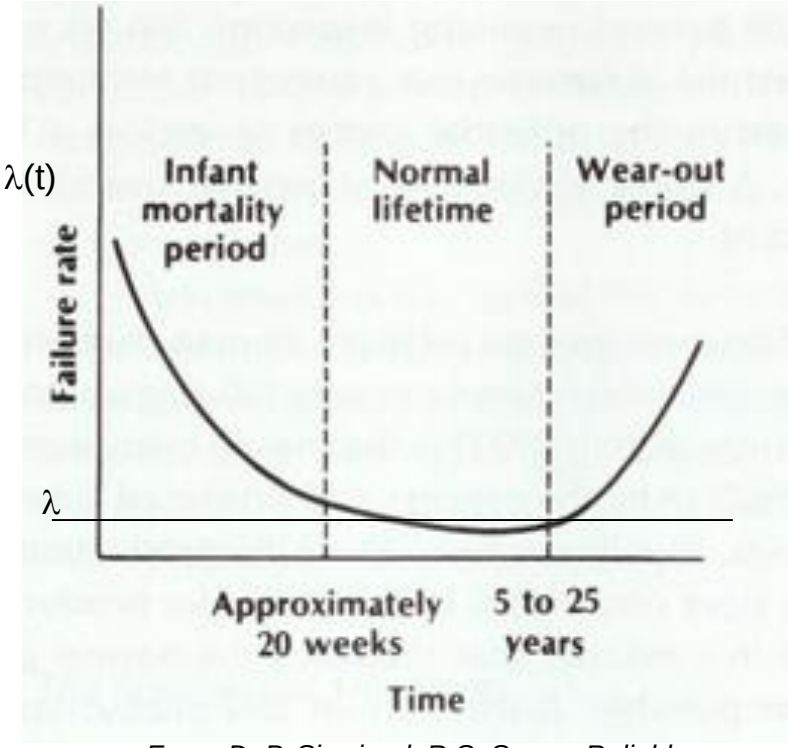
$\lambda(t)$ is a function of time
(bathtub-shaped curve)

$\lambda(t)$ constant > 0 in the useful life period

Constant failure rate λ

(usually expressed in number of failures for million hours)

λ = 1/2000
one failure every 2000 hours



From: D. P. Siewiorek R.S. Swarz, Reliable Computer Systems, Prentice Hall, 1992

Early life phase: there is a higher failure rate, calleld infant mortality, due to the failures of weaker components. Often these infant mortalities result from defetct or stress introduced in the manufacturing process.

Operational life phase: the failure rate is approximately constant.

Wear-out phase: time and use cause the failure rate to increase.

Hardware Reliability

Constant failure rate

$$\lambda(t) = \lambda$$

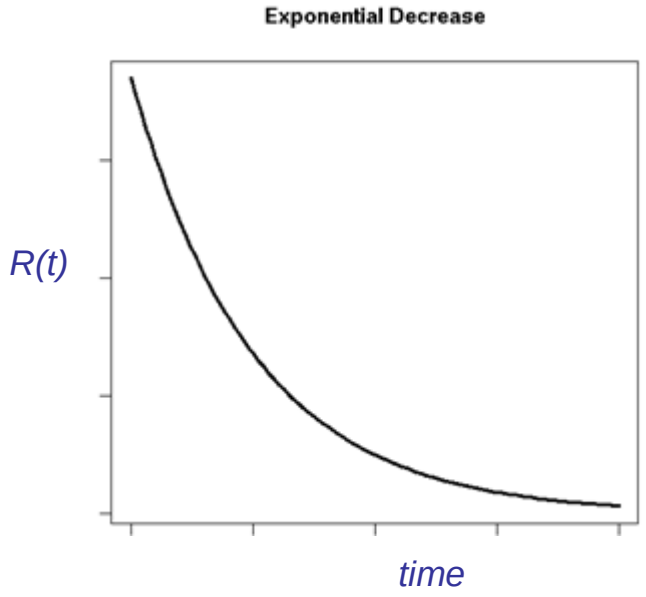
$$\lambda(t) = \frac{f(t)}{R(t)} = \frac{-dR(t)}{dt}\frac{1}{R(t)}$$

Reliability function

$$R(t) = e^{-\lambda t}$$

Probability density function

$$f(t) = \lambda e^{-\lambda t}$$



the exponential relation between reliability and time is known as ***exponential failure law***

Time to failure of a component

Time to failure of a component can be modeled by a **random variable X**

$F_X(t) = P[X <= t]$ (cumulative distribution function)

$F_X(t)$ *unreliability* of the component at time t

Reliability of the component at time t is given by

$R(t) = P[X > t] = 1 - P[X <= t] = 1 - F_X(t)$

reliability function

R(t) is the probability of not observing any failure before time t

Hardware Reliability

Mean time to failure (MTTF)

is the expected time that a system will operate before the first failure occurs (e.g., 2000 hours)

$$MTTF = \int_0^\infty t\, f(t)\, dt = \int_0^\infty t\, \lambda e^{-\lambda t}\, dt = \frac{1}{\lambda}$$

λ = 1/2000	0.0005 per hour
MTTF = 2000	time to the first failure 2000 hours

Failure in time (FIT)

measure of failure rate in 10^9 device hours

1 FIT means 1 failure in 10^9 device hours

Failure Rate

- Handbooks of failure rate data for various components are available from government and commercial sources.

- Reliability Data Sheet of product

- **Commercially available databases**
  - Military Handbook MIL-HDBK-217F
  - Telcordia,
  - PRISM User's Manual,
  - International Eletrotechnical Commission (IEC) Standard 61508
  - …

  Databases used to obtain reliability parameters in
  ''Traditional Probabilistic Risk Assessment Methods
  for Digital Systems'',
  U.S. Nuclear Regulatory Commission,
  NUREG/CR-6962, October 2008

Distribution model for permanent faults

MIL-HBDK-217 (*Reliability Prediction of Electronic Equipment* -Department of Defence) is a model for chip failure. Statistics on electronic components failures are studied since 1965 (periodically updated).

Typical component failure rates in the range 0.01-1.0 per million hours.

Failure rate for a single chip :

$$\lambda = \pi_L \pi_Q (C_1 \pi_T \pi_V + C_2 \pi_E)$$

π_L = learning factor, based on the maturity of the fabrication process

π_Q = quality factor, based on incoming screening of components

π_T = temperature factor, based on the ambient operating temperature and the type of semiconductor process

π_E = environmental factor, based on the operating environment

π_V = voltage stress derating factor for CMOS devices

C_1, C_2 = complexity factors, based on the number of gates, or bits for memories in the component and number of pins.

From Reliable Computer Systems.D. P. Siewiorek R.S. Swarz, Prentice Hall, 1992

Model-based evaluation of dependability

a model is an abstraction of the system that highlights the important features for the objective of the study

Methodologies that employ combinatorial models
Reliability Block Diagrams, Fault tree, ….

State space representation methodologies
Markov chains, Petri-nets, SANs, …

Model-based evaluation of dependability

Combinatorial methods

offer simple and intuitive methods of the construction and solutions of models

independent components

each component is associated a failure rate

model construction is based on the structure of the systems (series/parallel connections of components)

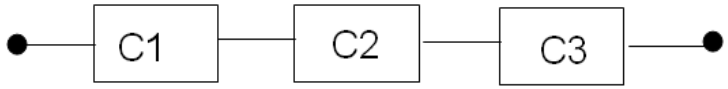
inadequate to deal with systems that exhibits complex dependencies among components and repairable systems

Combinatorial models

Series: all components must be operational (a)

$R_i(t)$ reliability of module i at time t

$R_{series}(t) = \Pi_{i=1}^{n} R_i(t)$
where Π is the product



(a)

If each individual component i satisfies the exponential failure law with constant failure rate λ_i:

$$R_{series}(t) = e^{-\lambda_1 t} ... e^{-\lambda_n t} = e^{-\sum_{i=1}^{n} \lambda_i t}$$

Unreliability function

$$Q_{series}(t) = 1 - R_{series}(t) = 1 - \Pi_{i=1}^{n} R_i(t) = 1 - \Pi_{i=1}^{n} [1 - Q_i(t)]$$

Combinatorial models

If the system does not contain any redundancy, that is any component must function properly for the system to work, and if component failures are independent, then

- the **system reliability** is the product of the component reliability, and it is exponential

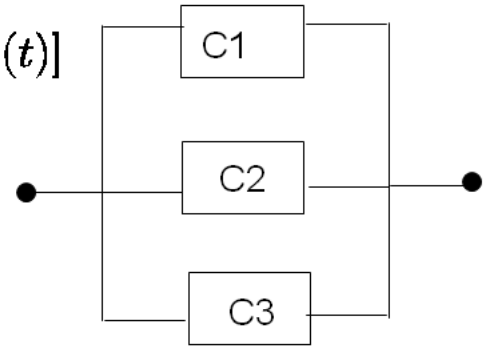
- the **failure rate of the system** is the sum of the failure rates of the individual components

Combinatorial models

Parallel: at least one of the components must be operational (b)

$$Q_{parallel}(t) = \Pi_{i=1}^{n} Q_i(t)$$
$$R_{parallel}(t) = 1 - Q_{parallel}(t) = 1 - \Pi_{i=1}^{n} Q_i(t) = 1 - \Pi_{i=1}^{n}[1 - R_i(t)]$$

Note the duality between Q and R in the two cases



(b)

M-of-N systems - a generalisation of parallel model
at least M modules of N are required to function

Assume N identical modules and M of those are required for the system to function properly, the expression for reliability of M-of-N substems can be written as:

$$R_{M-of-N}(t) = \sum_{i=0}^{N-M} \frac{N!}{(N-i)!i!} R^{N-i}(t)(1 - R(t))^i$$

i number of faulty components

$$\binom{N}{i} = \frac{N!}{(N-i)!\ i!}$$

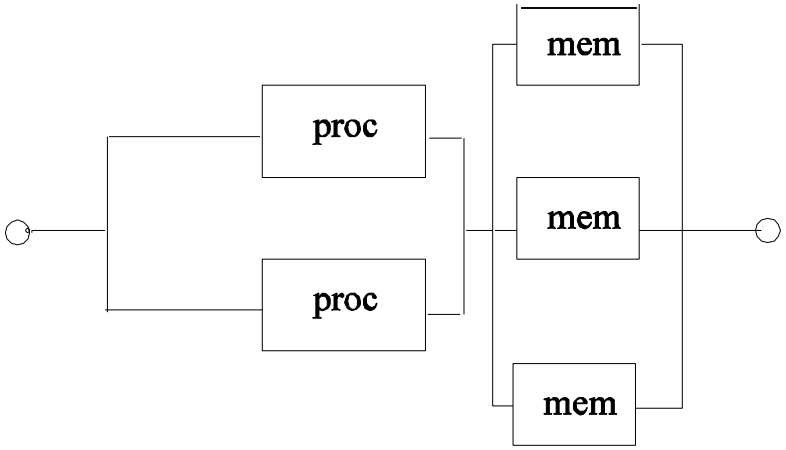
Binomial coefficient

Combinatorial models

If the system contain redundancy, that is a subset of components must function properly for the system to work, and if component failures are independent, then

- the **system reliability** is the reliability of a series/parallel combinatorial model

Series/Parallel models

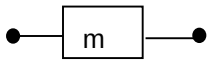


Multiprocessor with 2 processors and three shared memories

-> analysis under different conditions

TMR

Simplex system



λ failure rate of module m

$$R_m = e^{-\lambda t}$$

$$R_{simplex} = e^{-\lambda t}$$

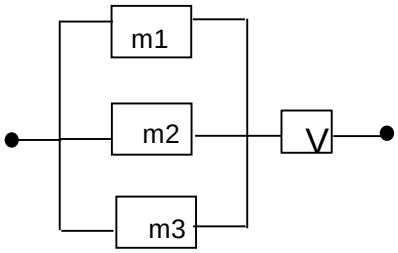
TMR system

$$R_V(t) = 1$$

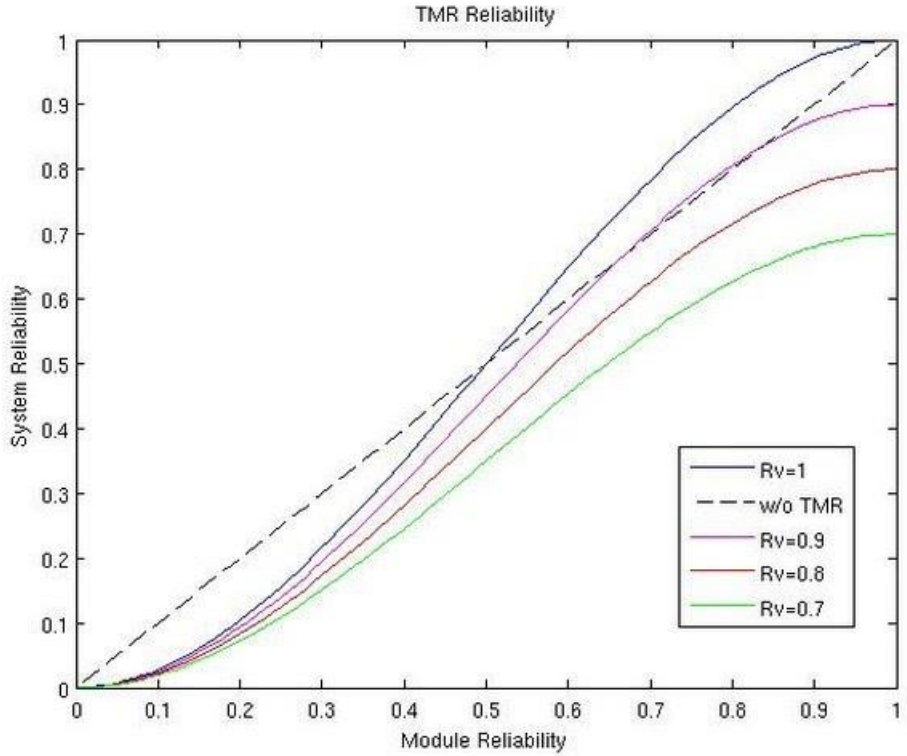
$$R_{TMR} = \sum_{i=0}^{1} \binom{3}{i} (e^{-\lambda t})^{3-i} (1 - e^{-\lambda t})^i$$

$$= (e^{-\lambda t})^3 + 3(e^{-\lambda t})^2 (1 - e^{-\lambda t})$$

$$R_{TMR} > R_m \text{ if } R_m > 0.5$$



2 of 3



From www.google.com

TMR: reliability function and mission time

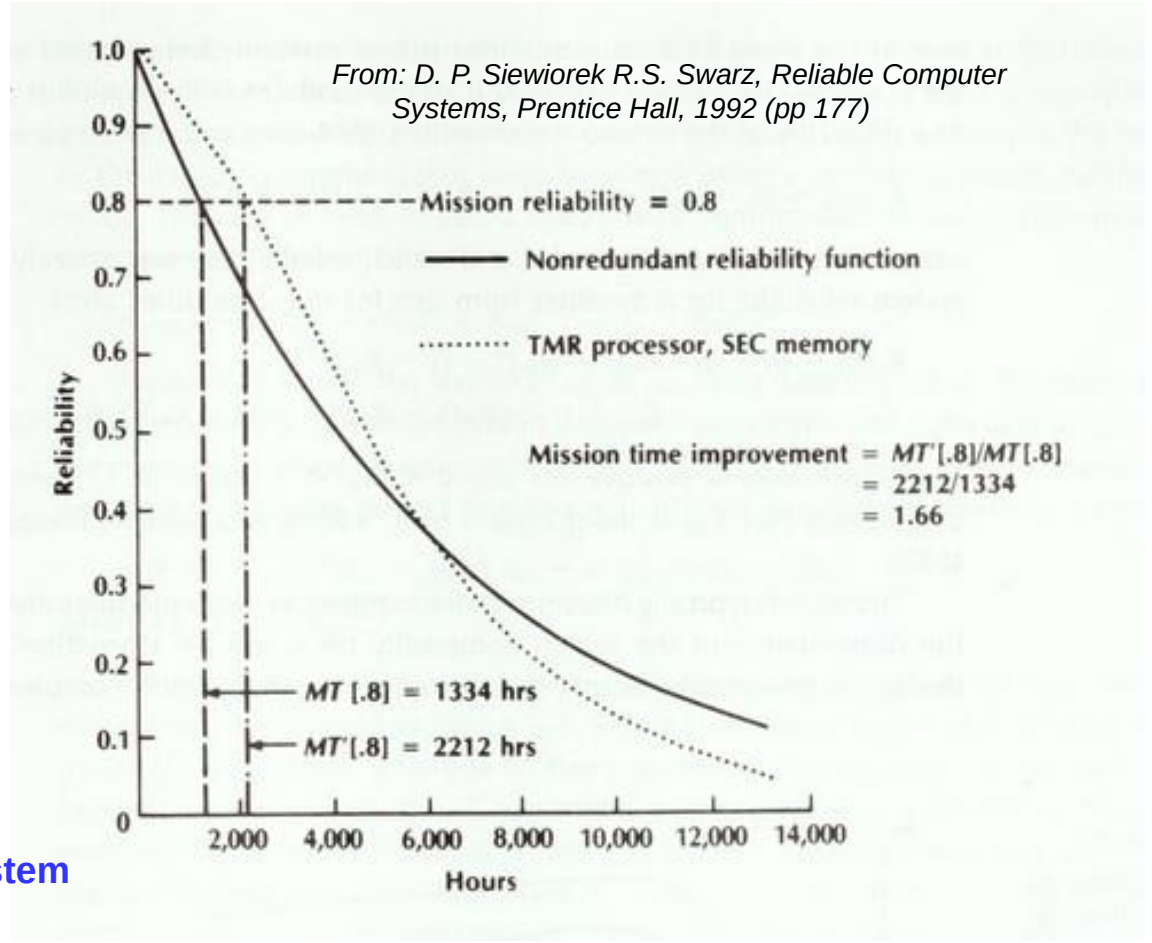
$R_{simplex} = e^{-\lambda t}$

$MTTF_{simplex} = \frac{1}{\lambda}$

TMR system

$R_{TMR} = 3e^{-2\lambda t} - 2e^{-3\lambda t}$

$MTTF_{TMR} = \frac{3}{2\lambda} - \frac{2}{3\lambda} = \frac{5}{6\lambda} < \frac{1}{\lambda}$



TMR worse than a simplex system !

TMR has a higher reliability for the first 6.000 hours of system life

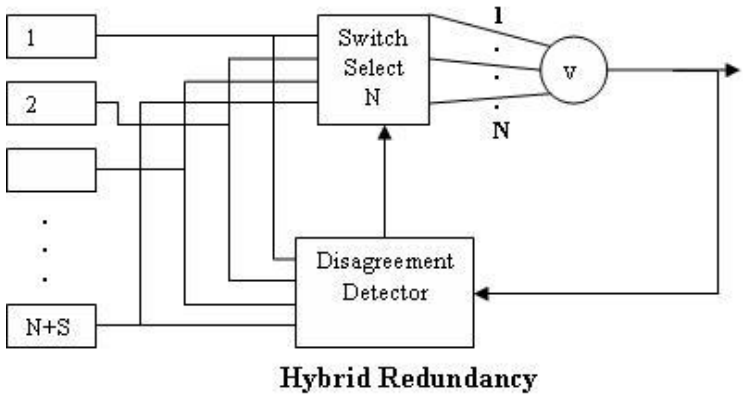
TMR operates at or above 0.8 reliability 66 percent longer than the simplex system

- **S shape curve is typical of redundant systems (there is the well known knee): above the knee the redundant system has components that tolerate failures; after the knee there is a sharper decrease of the reliability function in the redundant system (the system has exhausted redundancy, there is more hardware to fail than in the non redundant system)**

Hybrid redundancy with TMR

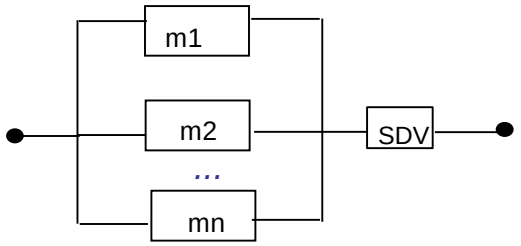
Symplex system
λ failure rate m
$R_m = e^{-\lambda t}$
$R_{sys} = e^{-\lambda t}$

Hybrid system
n=N+S total number of components
S number of spares



Hybrid Redundancy

Let N = 3 $R_{SDV}(t) = 1$
λ failure rate of on line comp
λ failure rate of spare comp



The first system failure occurs if 1) all the modules fail; 2) all but one modules fail

$R_{Hybrid} = R_{SDV}(1- Q_{Hybrid})$

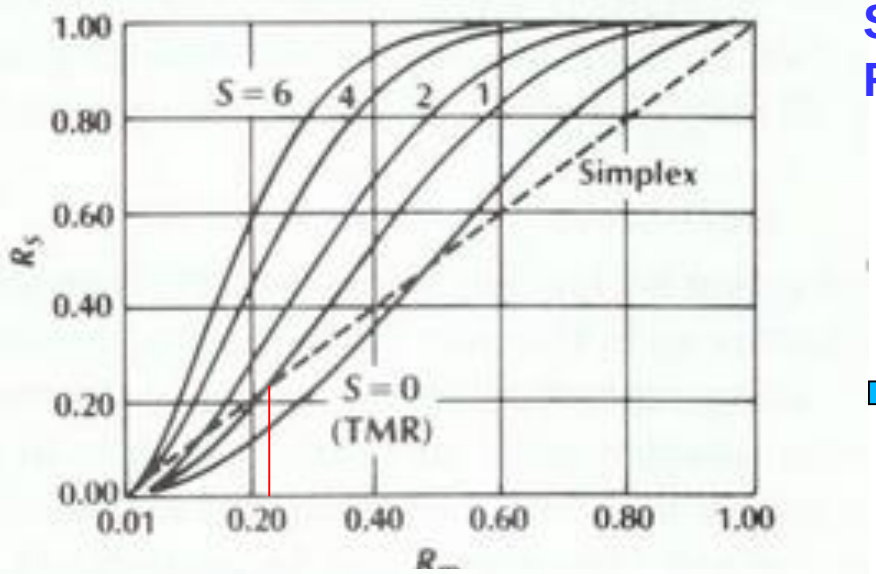
$R_{Hybrid} = (1 – ((1-R_m)^n + n(R_m)(1-R_m)^{n-1}))$

$R_{Hybrid(n+1)} – R_{Hybrid(n)} > 0$

adding modules increases the system reliability under the assumption R_{SDV} *independent of n*

Hybrid redundancy with TMR

Hybrid TMR system reliability R_S vs individual module reliability R_m



S is the number of spares

R_{SDV} =1

Figure 1. system with standby failure rate equal to on-line failure rate

the TMR with one spare is more reliable than simplex system if R_m>0.23

From: D. P. Siewiorek R.S. Swarz, Reliable Computer Systems, Prentice Hall, 1992 (pp 177)

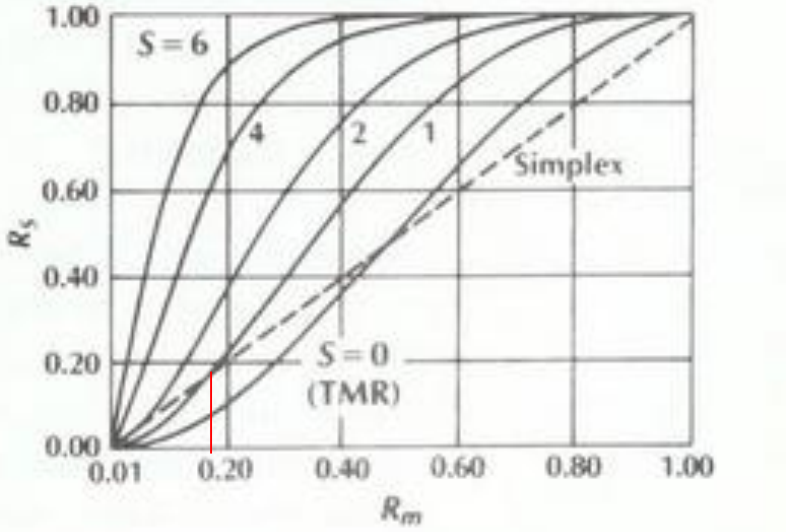


Figure 2. system with standby failure rate equal to 10% of on line failure rate

the TMR with one spare is more reliable than simplex system if R_m>0.17

From: D. P. Siewiorek R.S. Swarz, Reliable Computer Systems, Prentice Hall, 1992 (pp 177)

Fault Trees

Fault Trees

- FT considers the combination of events that may lead to an unsdesirable situation of the system
  (the delivery of improper service for a Reliability study,
  catastrophic failures for a Safety study)

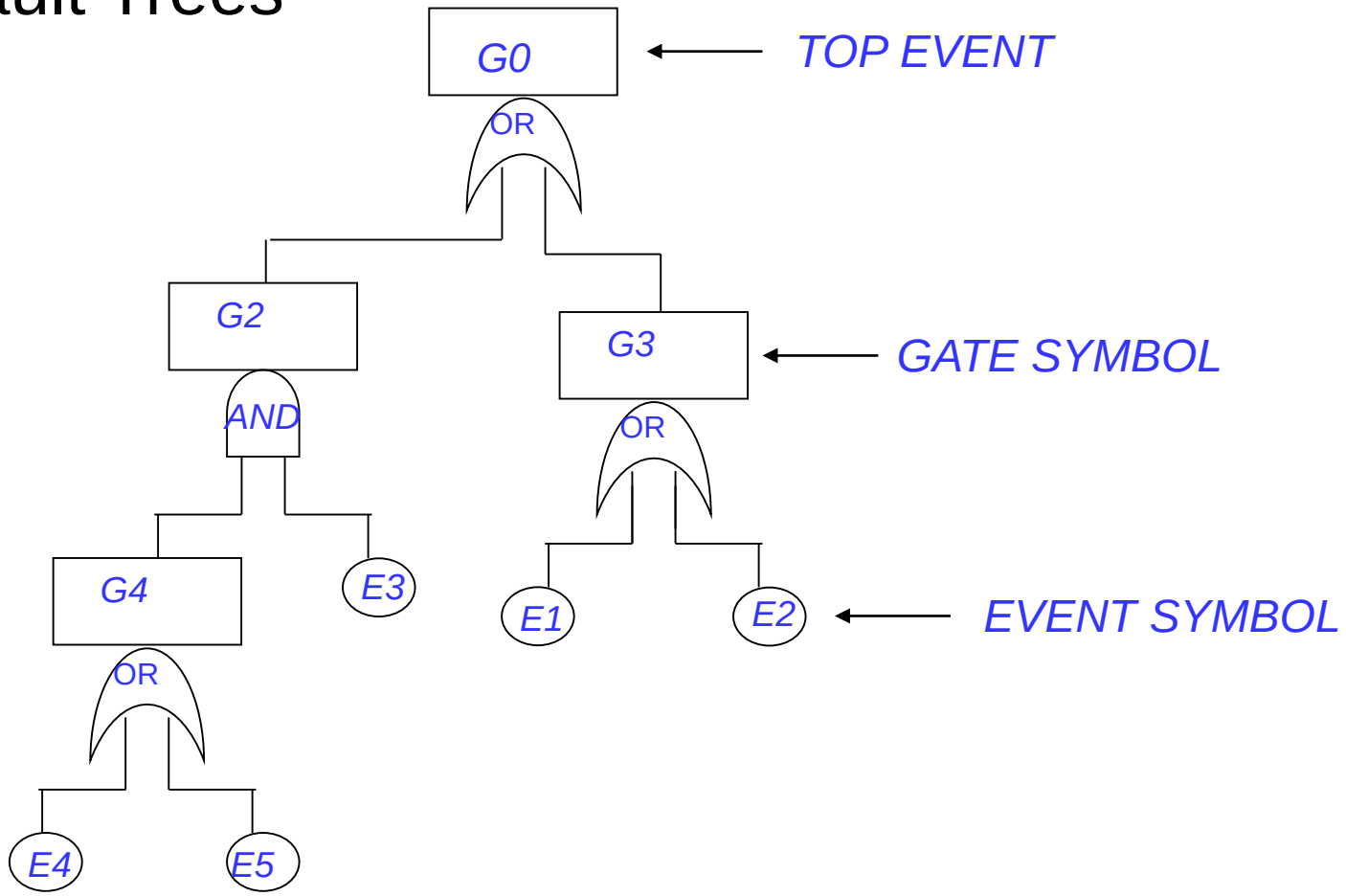
- Describe the scenarios of occurrence of events at abstract level

- Hierarchy of levels of events linked by logical operators

- The analysis of the fault tree evaluates the probability of occurrence of the root event, in terms of the status of the leaves (faulty/non faulty)

- Applicable both at design phase and operational phase

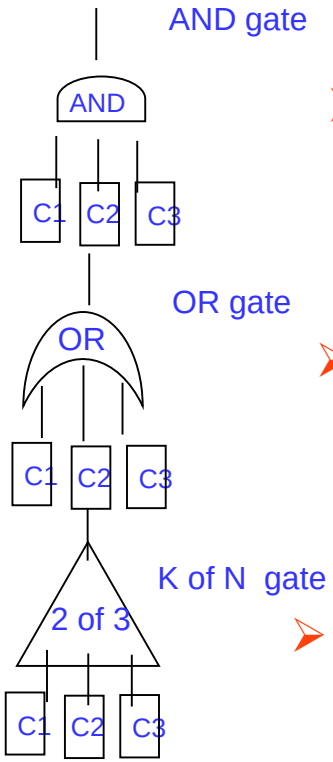
Fault Trees



Describes the Top Event (status of the system) in terms of the status (faulty/non faulty) of the Basic events (system's components)

Fault Trees

- Components are leaves in the tree
- Component faulty corresponds to logical value **true**, otherwise **false**
- Nodes in the tree are boolen AND, OR and k of N gates
- The system fails if the root is true

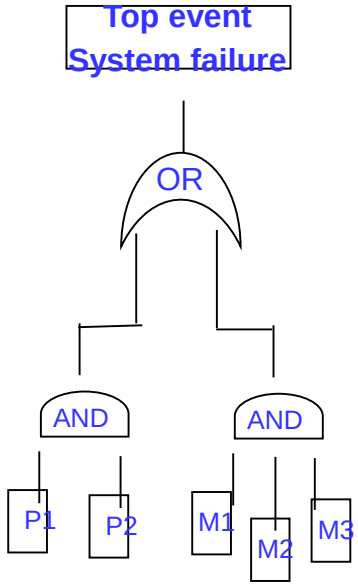


- True if all the components are true (faulty)
- True if at least one of the components is true (faulty)
- True if at least k of the components are true (two or three components) (faulty)

Example:

Multiprocessor with 2 processors and three shared memories

-> the computer fail if all the memories fail or all the processors fail



A cut is defined as a set of elementary events that, according to the logic expressed by the FT, leads to the occurrence of the root event.

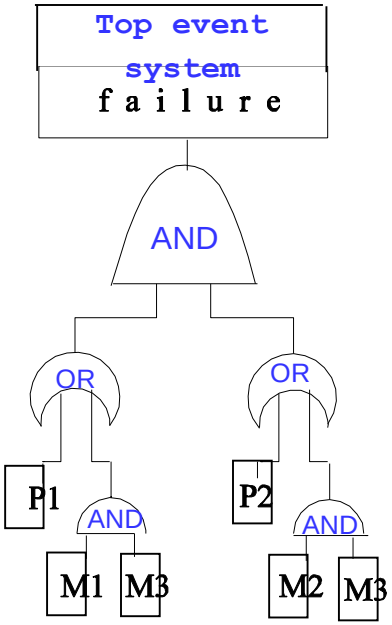
To estimate the probability of the root event, compute the probability of occurrence for each of the cuts and combine these probabilities

Conditioning Fault Trees

- If the same component appears more than once in a fault tree, it violates the independent failure assumption (conditioned fault tree)

- Example

  Multiprocessor with 2 processors and three memories: M1 private memory of P1 M2 private memory of P2, M3 shared memory.



- Assume every process has its own private memory plus a shared memory.
- Operational condition: at least one processor is active and can access to its private or shared memory.
- **repeat** instruction:given a component C whether or not the component is input to more than one gate, the component is unique
  M3 is a shared memory

Conditioning Fault Trees

If a component C appears multiple times in the FT

$$Q_s(t) = Q_{S|C\ Fails}(t)\ Q_C(t) + Q_{S|C\ not\ Fails}(t)\ (1-Q_C(t))$$

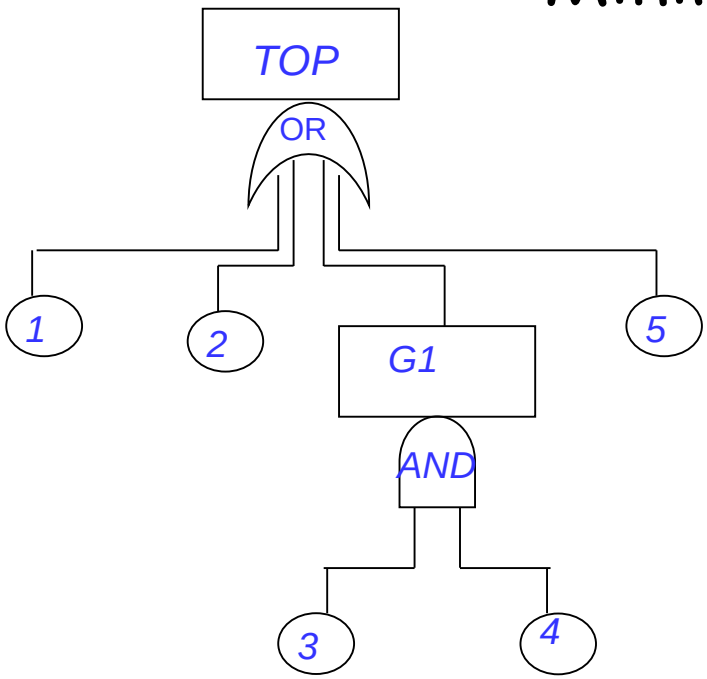
where

S|C Fails is the system given that C fails

and

S|C not Fails is the system given that C has not failed

Minimal cut sets



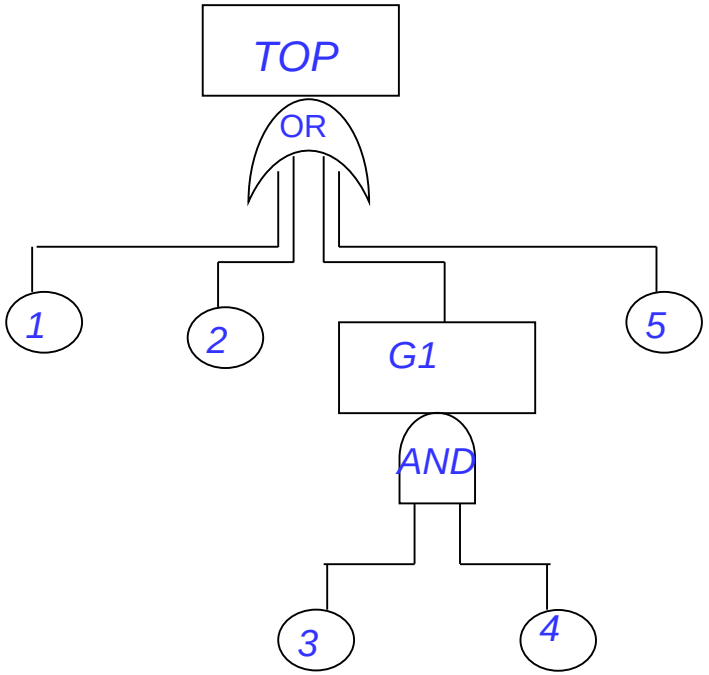
A cut is defined as a set of elementary events that, according to the logic expressed by the FT, leads to the occurrence of the root event.

Cut Sets

Top = {1}, {2} , {G1} , {5} = {1}, {2} , {3, 4} , {5}

Minimal Cut Sets

Top = {1}, {2} , {3, 4} , {5}



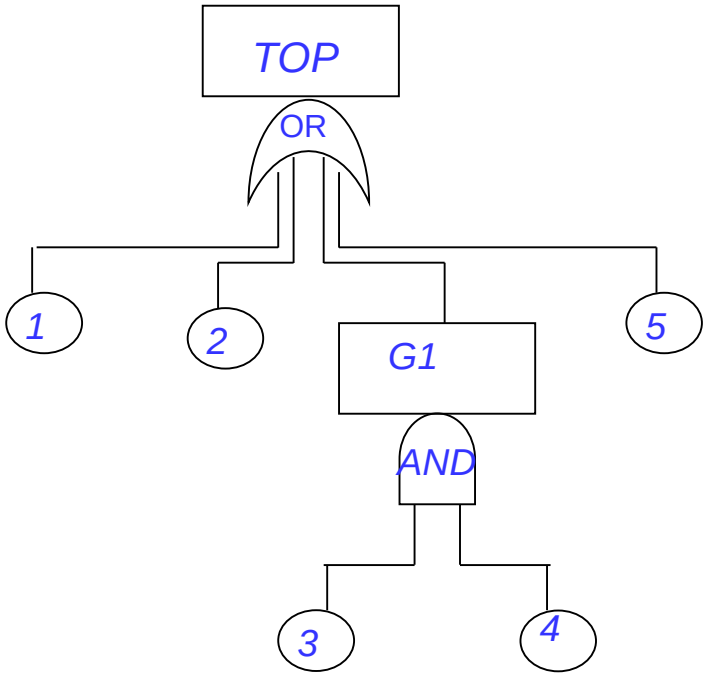
Minimal Cut Sets
Top = {1}, {2} , {3, 4} , {5}

independent faults of the components

$Q_{Si}(t)$ = probability that all components in the minimal cut set *Si* are faulty

$Q_{Si}(t) = q_1(t)\, q_2(t) \ldots q_{ni}(t)$ with *Si* ={1, 2, …, ni }

The numerical solution of the FT is performed by computing the probability of occurrence for each of the cuts, and by combining those probabilities to estimate the probability of the root event



Minimal Cut Sets

Top = {1}, {2} , {3, 4} , {5}

$S_1 = \{1\}$

$S_2 = \{2\}$

$S_3 = \{3, 4\}$

$S_4 = \{5\}$

$$Q_{Top}(t) = Q_{S1}(t) + \ldots + Q_{Sn}(t)$$

n number of mininal cut sets

Fault Trees

- Definition of the Top event
- Analysis of failure models of components
- Minimal cut set

  minimal set of events that leads to the top event

  -> critical path of the system

Analysis:

- Failure probability of Basic events
- Failure probability of minimal cut sets
- Failure probability of Top event
- Single point of failure of the system: minimal cuts with a single event

Model-based evaluation of dependability

State-based models: Markov models

Characterize the state of the system at time t:

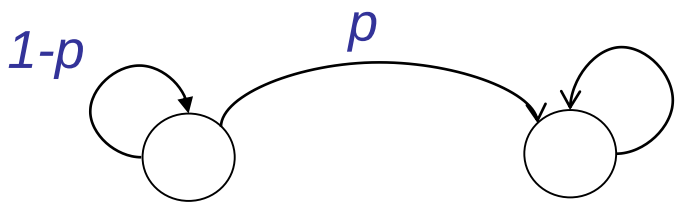
- identification of system states
- identification of transitions that govern the changes of state within a system

Each state represents a distinct combination of failed and working modules

The system goes from state to state as modules fail and repair.

The state transitions are characterized by the probability of failure and the probability of repair

Markov model:
graph where nodes are all the possible states and arcs are the possible transitions between states (labeled with a probability function)



Reliability/Availability modelling

Each state represents a distinct combination of working and failed components
As time passes, the system goes from state to state as modules fails and are repaired

Model-based evaluation of dependability

Markov models (a special type of random process) :

Basic assumption: the system behavior at any time instant depends only on the current state (independent of past values)

Main points:

- systems with arbitrary structures and complex dependencies can be modeled

- assumption of independent failures no longer necessary

- can be used for both reliability and availability modeling

Random process

In a general random process $\{X_t\}$, the value of the random variable X_{t+1} may depend on the values of the previous random variables $X_{t0}\, X_{t1}$ .....X_t.

Markov process

the state of a process at time t+1 depends only on the state at time t, and is independent *on any state before t.*

$$\mathcal{P}\{X_{t+1} = j | X_0 = k_0, ..., X_{t-1} = k_{t-1}, X_t = i\} = \mathcal{P}\{X_{t+1} = j | X_t = i\}$$

Markov property: "the current state is enough to determine the future state"

Markov chain

A **Markov chain** is a Markov process X with *discrete state space S*.

A Markov chain is homogeneous if it has *steady-state transition probabilities*:

$$\mathcal{P}\{X_{t+1} = j | X_t = i\} = \mathcal{P}\{X_1 = j | X_0 = i\} \ \forall t \geq 0$$

The probability of transition from state i to state j does not depend by the time. This probability is called p_{ij}

$$p_{ij} = \mathcal{P}\{X_1 = j | X_0 = i\}$$

We consider only *homogeneous* Markov chains

Discrete-time Markov chains (DTMC)

Continuous-time Markov chains (CTMC)

Discrete-time Markov model of a simplex system with repair

$\{X_t\}$ t=0, 1, 2, …. S={0, 1}

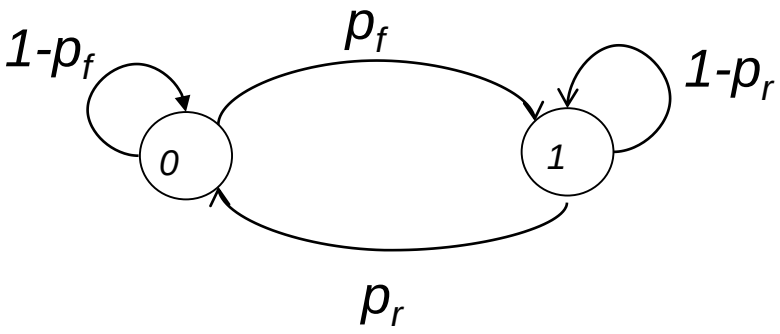
State 0 : working
State 1: failed

- all state transitions occur at fixed intervals
- probabilities assigned to each transition

p_f *Failure probability*

p_r *Repair probability*

The probability of state transition depends only on the current state



Graph model

$$P = \begin{array}{c} \\ 0 \\ 1 \end{array} \begin{array}{c} \begin{array}{cc} 0 & 1 \end{array} \\ \begin{bmatrix} 1-p_f & p_f \\ p_r & 1-p_r \end{bmatrix} \end{array}$$

new state

current state

Transition Probability Matrix

- Pij = probability of a transition from state i to state j
- Pij >=0
- the sum of each row must be one

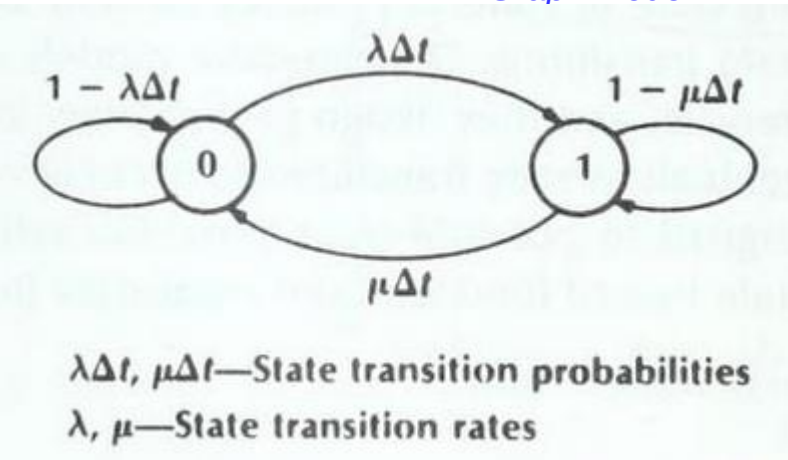
Continuous-time Markov model of a simplex system with repair

derived from the discrete time model, taking the limit as the time-step interval approaches zero

λ failure rate, μ repair rate

state 0: working
state 1: failed

Graph model



$$P = \begin{bmatrix} 1 - \lambda\Delta t & \lambda\Delta t \\ \mu\Delta t & 1 - \mu\Delta t \end{bmatrix}$$

Transition Matrix P

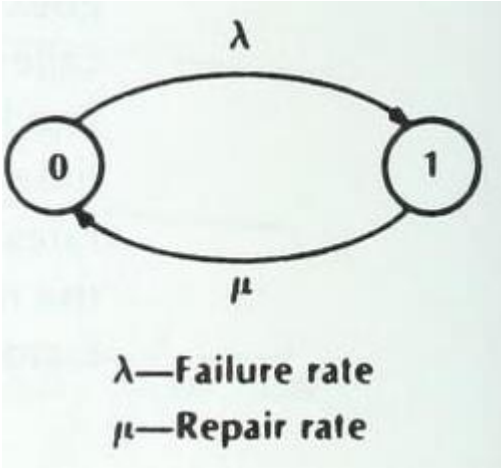
Continuous-time Markov models

Matrix form:

T matrix

$$[\dot{p}_0(t), \dot{p}_1(t)] = [p_0(t), p_1(t)] \begin{bmatrix} -\lambda & \lambda \\ \mu & -\mu \end{bmatrix}$$

The set of equations can be written by inspection of a transition diagram without self-loops and Δt's:



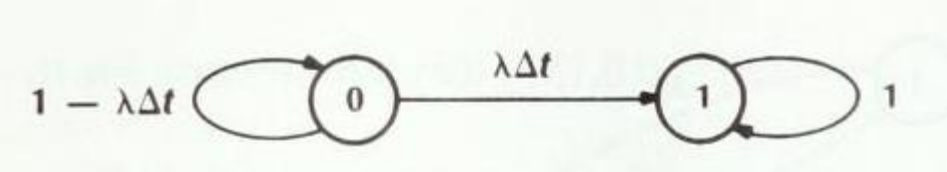
Continuous time Markov model graph

The change in state 0 is minus the flow out of state 0 times the probability of being in state 0 at time t, plus the flow into state 0 from state 1 times the probability of being in state 1.

Continuous-time Markov models: Reliability

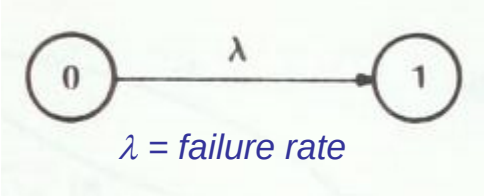
Markov model making the system-failed state a trapping state

Single system without repair



λΔt = state transition probability

Continuous time Markov model graph



λ = failure rate

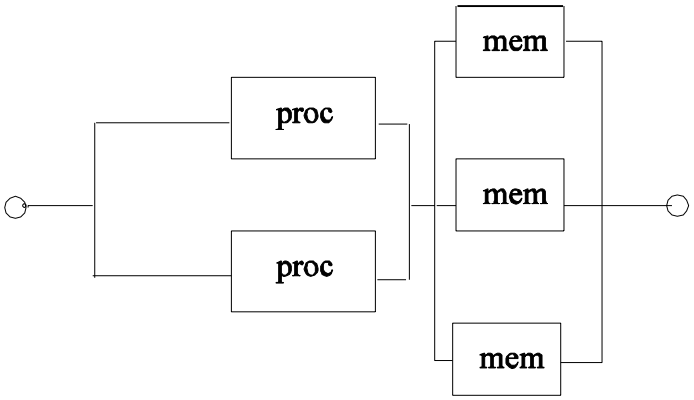
T matrix

$$\begin{bmatrix} -\lambda & \lambda \\ 0 & 0 \end{bmatrix}$$

T matrix can be built by inspection

An example of modeling (CTMC)

Multiprocessor system with 2 processors and 3 shared memories system. System is operational if at least one processor and one memory are operational.



λ_m failure rate for memory

λ_p failure rate for processor

X random process that represents the number of operational memories and the number of operational processors at time t

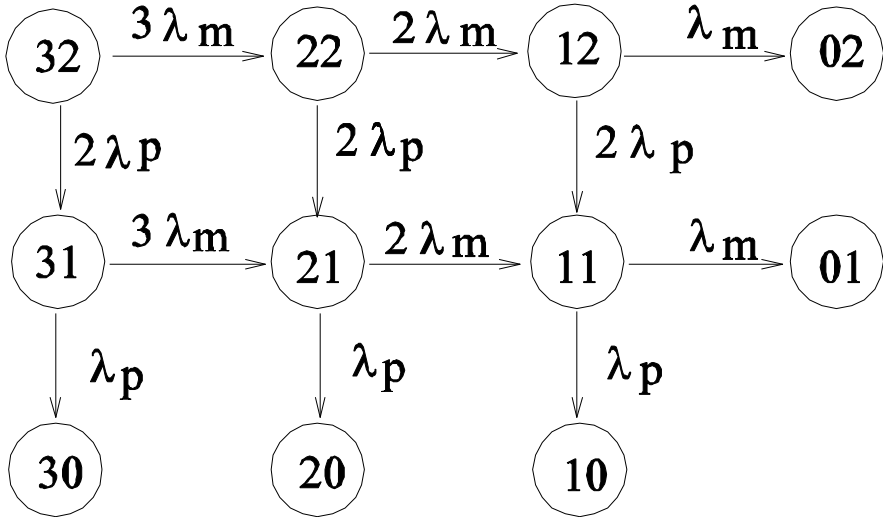
Given a state (i, j):

i is the number of operational memories;

j is the number of operational processors

S = {(3,2), (3,1), (3,0), (2,2), (2,1), (2,0), (1,2), (1,1), (1,0), (0,2), (0,1)}

Reliability modeling



λ_m failure rate for memory
λ_p failure rate for processor

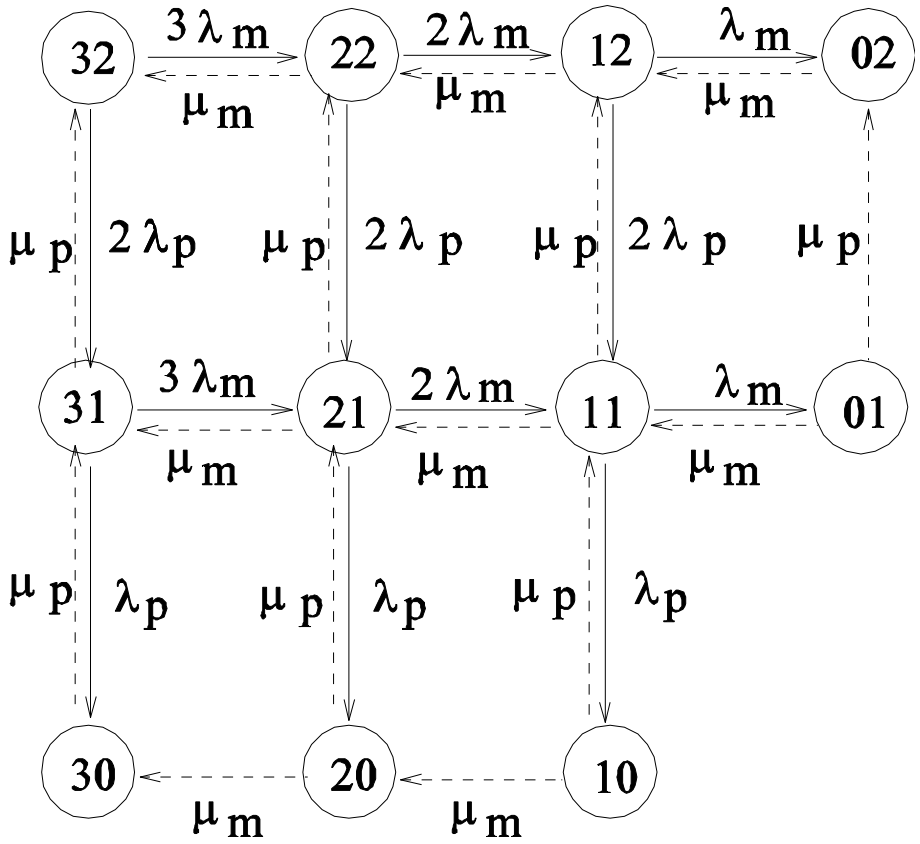
(3, 2) -> (2,2) failure of one memory

(3,0), (2,0), (1,0), (0,2), (0,1) are absorbent states

Availability modeling

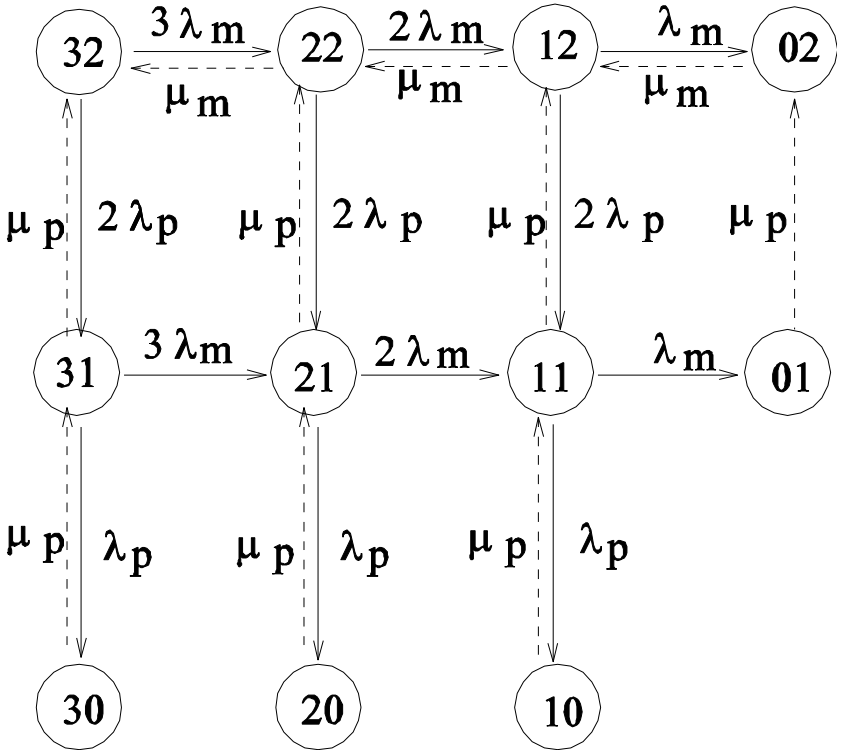
- Assume that faulty components are replaced and we evaluate the probability that the system is operational at time t
- Constant repair rate μ (number of expected repairs in a unit of time)
- Strategy of repair:
  only one processor or one memory at a time can be substituted
- The behaviour of components (with respect of being operational or failed) is not independent: it depends on whether or not other components are in a failure state.

- Strategy of repair:
  only one component can be substituted at a time



λm failure rate for memory
λp failure rate for processor
μm repair rate for memory
μp repair rate for processor

- An alternative strategy of repair:
  only one component can be substituted at a time and processors have higher priority
- exclude the lines μm representing memory repair in the case where there has been a process failure



State occupacy vector

$\pi^{(t)} = [\pi_0^{(t)}, \pi_1^{(t)}, \pi_2^{(t)}, \ldots]$ state occupancy vector

Transient analysis

$\pi_j^{(t)}$ prob of being in state j at time t

Steady-state behaviour

$$\lim_{t \to \infty} \pi_j^{(t)}$$

Many solution methods exist

Moebius tool

Instructions to obtain the Moebius tool:
1. visit http://www.mobius.illinois.edu/ .
2. Click "Login" in the menu bar.
3. In the login page, click "Create an account".
4. Follow instructions to obtain a license. IN PARTICULAR:
a) use an unipi.it email address if possible (not commercial addresses like gmail);
b) in a comment field, say that you attend a course on the Mobius tool held by Cinzia Bernardeschi (owner of an academic licence).
5. Within 48 hours, you should receive a confirmation letter with the link to download the tool.
6. Versions are available for Ubuntu Linux, Mac OSX, and Windows, either 32 or 64 bit.