Bayesian Belief Networks in Reliability

Prof. Luigi Portinale, Ph.D.

Department of Computer Science

University of Piemonte Orientale "A. Avogadro"

Alessandria (Italy)

Dipartimento di Informatica

AMEDEO AVOGADRO
università degli studi del piemonte orientale

Overview

- Dependability/Reliability issues
- Main Model Types for Reliability
- Probabilistic Graphical Models (BN and DBN)
  - Modeling
  - Computing
- From (Dynamic) Fault Trees to (Dynamic) Bayesian Nets
  - Modeling
  - Computing
- Case Studies
- Tools
- Open Issues

Overview

- Dependability/Reliability issues
- Main Model Types for Reliability
- Probabilistic Graphical Models (BN and DBN)
  - Modeling
  - Computing
- From (Dynamic) Fault Trees to (Dynamic) Bayesian Nets
  - Modeling
  - Computing
- Case Studies
- Tools
- Open Issues

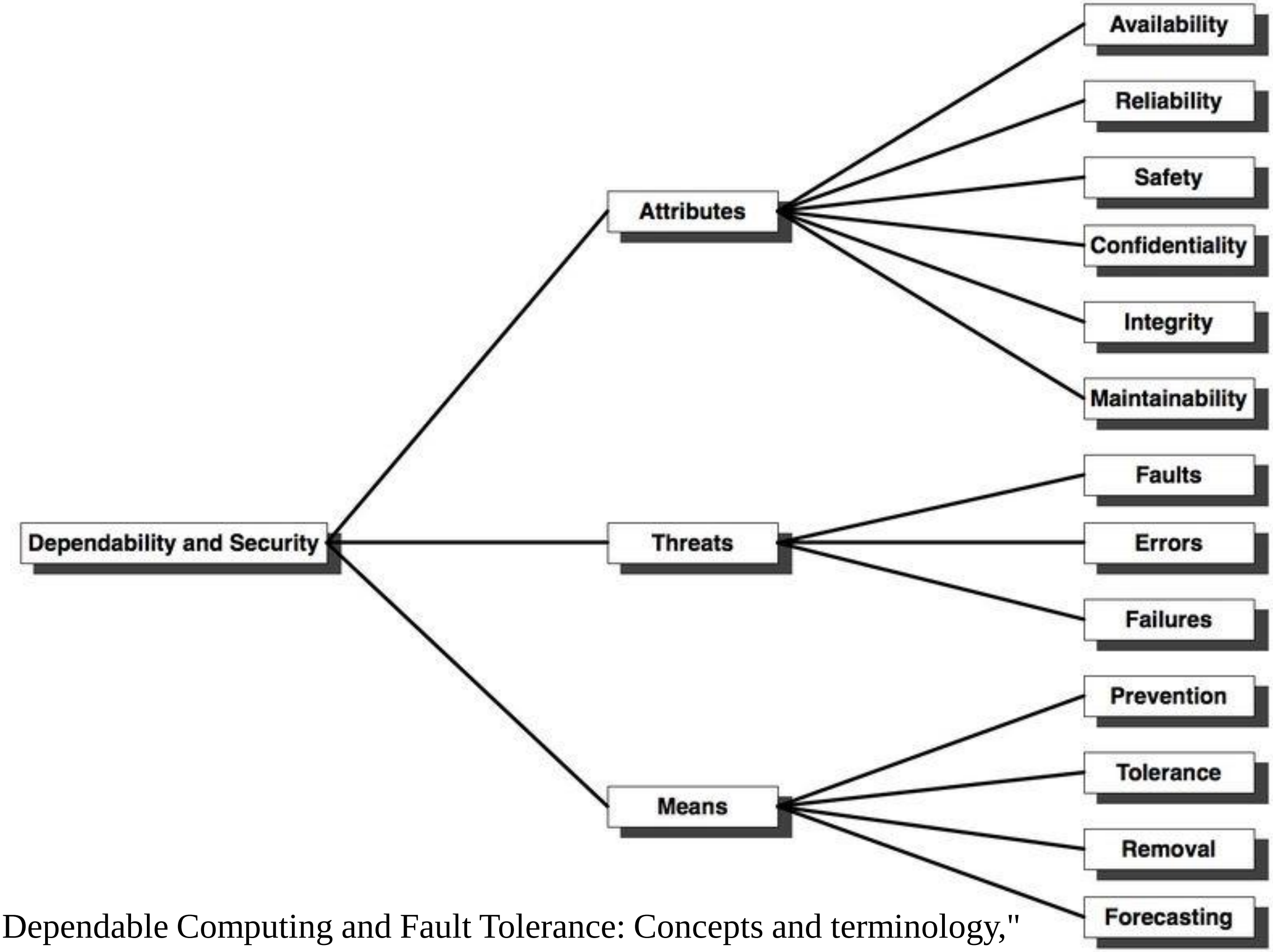
Dependability vs Reliability

We adopt the term dependability to identify the ability of a system to deliver service that can justifiably be trusted.

Dependability is an integrating concept that encompasses various attributes:

- **Reliability**: continuity of correct service.
- **Availability**: readiness for correct service.
- **Maintainability**: ability to undergo modifications and repairs.
- **Safety**: absence of catastrophic consequences.
- (**Security**)

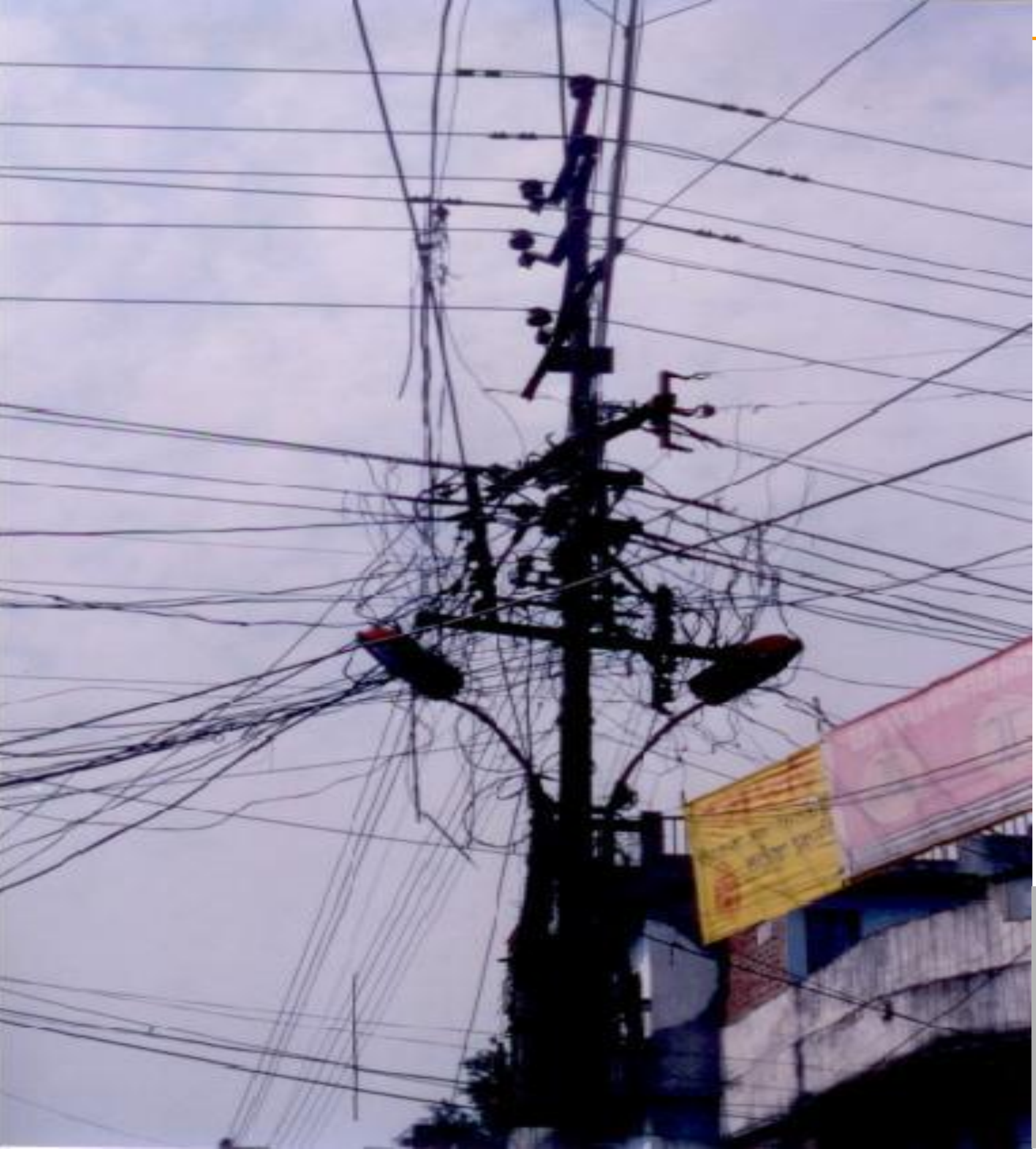
Dep/Sec Taxonomy



J.C. Laprie. "Dependable Computing and Fault Tolerance: Concepts and terminology," in Proc. 15th IEEE Int. Symp. on Fault-Tolerant Computing, 1985

22/10/1895: Gare Montparnasse.

What dependability theory and practice wants to avoid



Are these connections reliable ?

Some technicalities…

- **Reliability:** R(*t*)
  probability that the system performs the required function in the interval (*0*, *t*) given the stress and environmental conditions in which it operates. e.g. $R(t) = 1 - e^{-\lambda t}$

- **Unreliability:** U(*t*) = 1–R(*t*)
  probability that the system is not performing the required function at time *t*.

- **Availability:**

$$A = \frac{E[\text{Uptime}]}{E[\text{Uptime}] + E[\text{Downtime}]}$$

MTTF

MTTR

$$X(t) = \begin{cases} 1, & \text{sys functions at time } t \\ 0, & \text{otherwise} \end{cases} \qquad A(t) = \Pr[X(t) = 1] = E[X(t)]. \qquad A = \lim_{t\to\infty} A(t).$$

A(*t*)=R(*t*) if repear is absent

Some technicalities…

- **Failure**: a system deviation from the correct/expected service (*failure modes*)
- **Fault**: a cause of a failure (a defect in the system)
- **Error**: a discrepancy between the intended behaviour of a system component and its actual behaviour
- **Fault-Error-Failure chain:** a fault, when activated, can lead to an error (which is an invalid state) and the invalid state generated by an error may lead to another error or a failure (which is an observable deviation from the specified behaviour at the system boundary)
- The chain can actually be a loop (having faults causing failures, causing other faults, causing other failures, etc…)

Reliability Evaluation

- **Measurement-based** evaluation
  - It requires the observation of the behaviour of the system physical components.
  - It may be expensive or unpractical.
- **Model-based** evaluation
  - A model is a convenient abstraction of the system.
  - A model has a certain degree of accuracy.
  - A model can be the object of analysis or simulation.
  - **Models classification:**
    - Combinatorial models
    - State space based models
    - Models with conditional local dependencies

Overview

- Dependability/Reliability issues
- Main Model Types for Reliability
- Probabilistic Graphical Models (BN and DBN)
  - Modeling
  - Computing
- From (Dynamic) Fault Trees to (Dynamic) Bayesian Nets
  - Modeling
  - Computing
- Case Studies
- Tools
- Open Issues

Modeling Properties

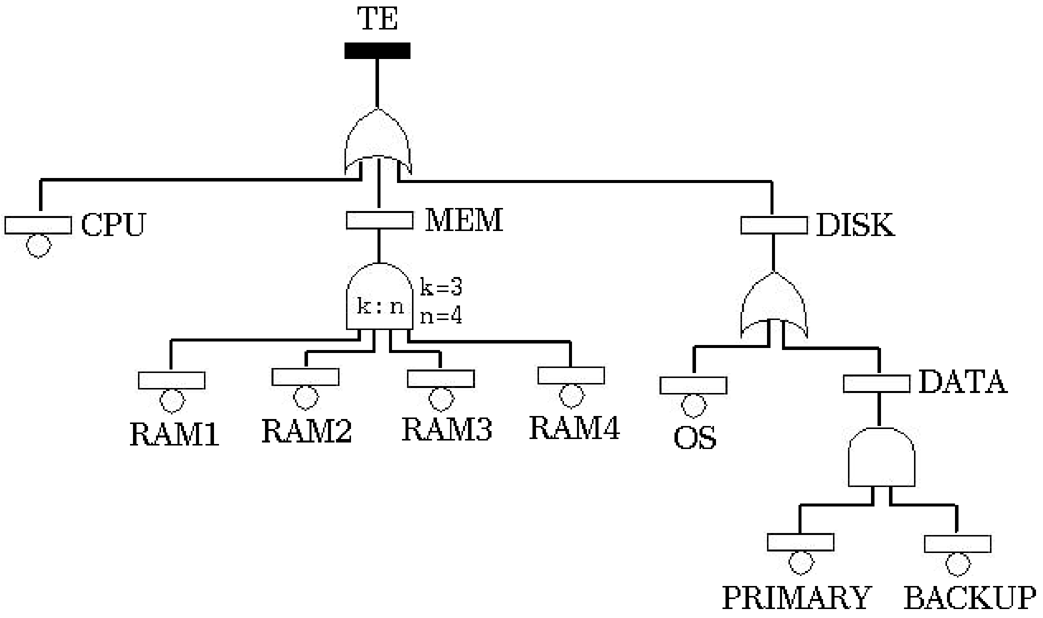
- Several modeling paradigms are available. The usability of a model can be classified according to two main properties:
- **The Modeling Power -** Refers to the ability of the model to allow an accurate and faithful representation of the system;
- **The Decision Power -** Refers to the ability of the model to be analytically tractable and to provide results with a low space and time complexity.

Model Types

- **Combinatorial models** assume that components are statistically independent: poor modeling power coupled with high analytical tractability.
- **State-space models** rely on the specification of the whole set of the possible system states and of the possible transitions among them.
- **Local dependencies:** between combinatorial and state space models, research is currently carried on to include localized dependencies

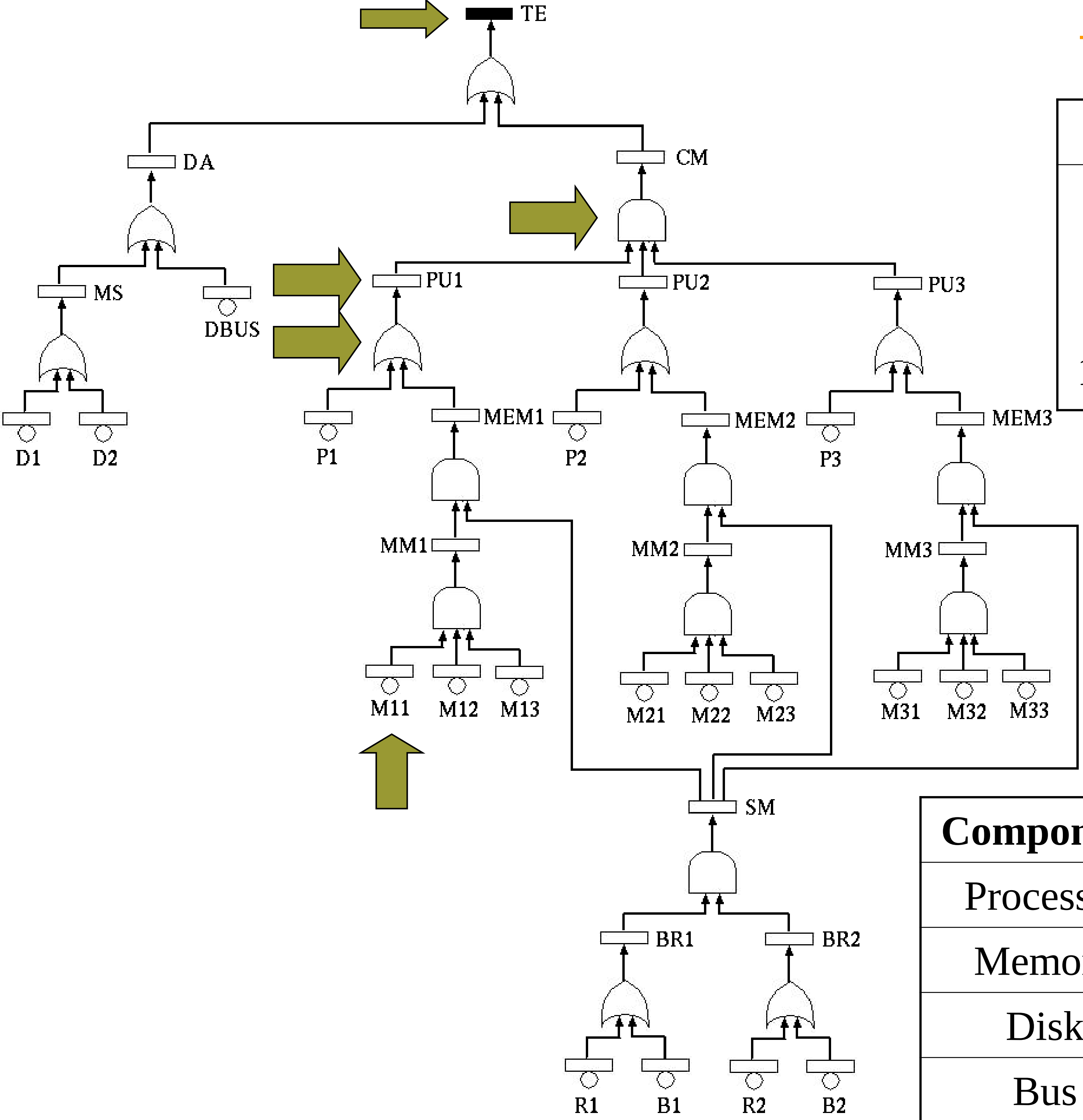
Combinatorial Models

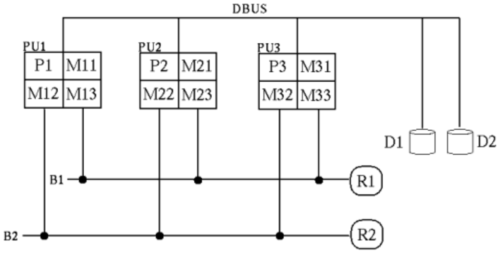
- They represent the structure of the system in terms of logical connection of working (failed) components in order to obtain the system success (failure).
  - **Fault Trees**, Reliability Block Diagrams, Reliability Graphs
  - Easy to use, concise, analytically tractable
  - Limited modeling power (binary independent components)



Fault Tree

Time	Unreliability
4000 h	6.387520E-3
6000 h	9.565979E-3
8000 h	1.273428E-2
10000 h	1.589248E-2

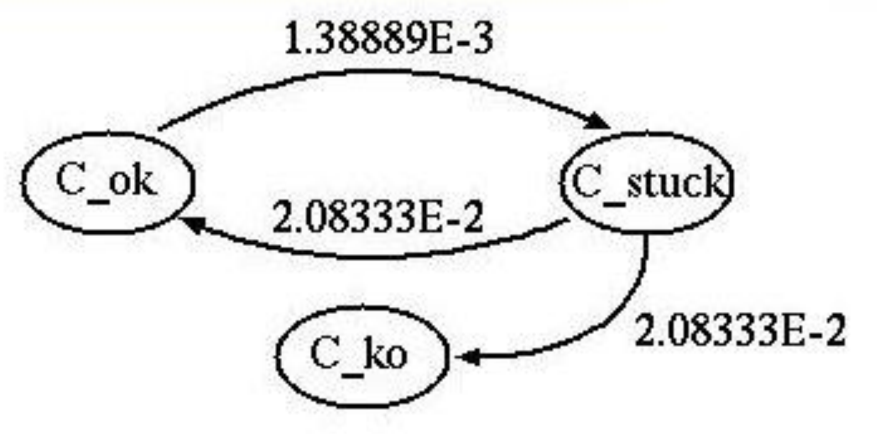




Component	Failure rate (λ)
Processor	5.0E-7 1/h
Memory	3.0E-8 1/h
Disk	8.0E-7 1/h
Bus	2.0E-9 1/h

State Space Models

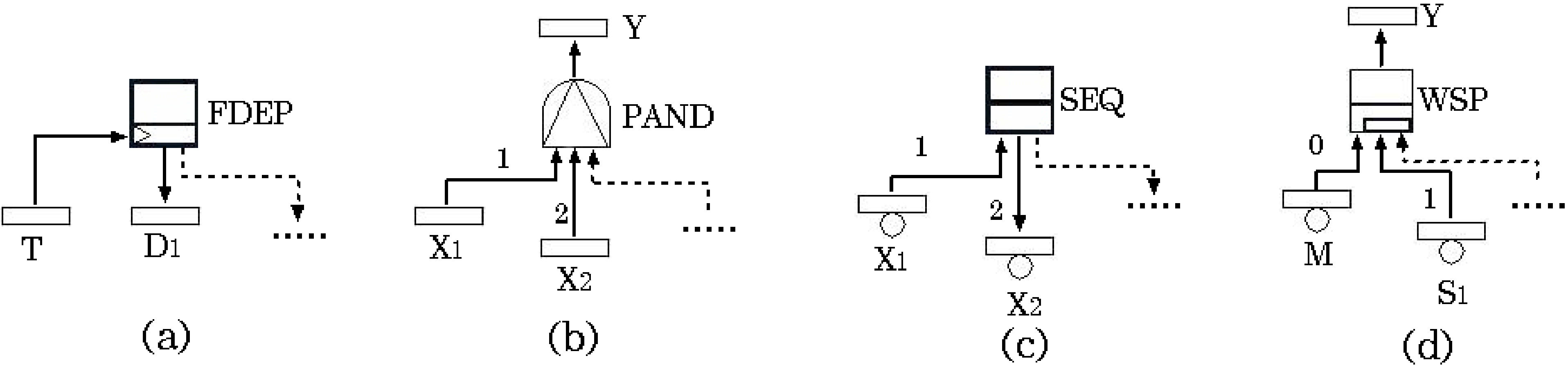
- They enumerate the set of meaningful states and state transitions of the system
  - **Markov Chains**, Markov Decision Processes, Petri Nets
  - State space may be over-specified with respect to the modeling needs
  - Dynamic behavior of the system may lead to the explosion of the state space size



Local Dependencies: Dynamic Fault Trees

- A dependency arises when the failure behaviour of a component depends on the state of the system.

- DFTs are characterized by the dynamic gates
  - Functional dependencies (FDEP gate)
  - Temporal dependencies (SEQ gate, PAND gate)
  - (Warm) spare components (WSP gate): multi-state components

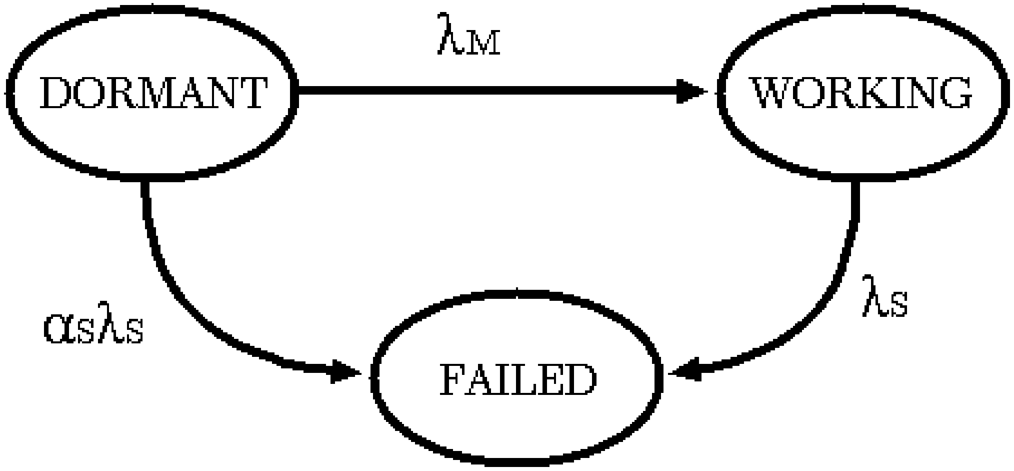


J. B. Dugan, S. J. Bavuso, M. A. Boyd, “Dynamic Fault-Tree Models for Fault-Tolerant Computer Systems”, *IEEE Transactions on Reliability*, vol 41, 1992, pp 363-377

Modeling Spare Dependencies

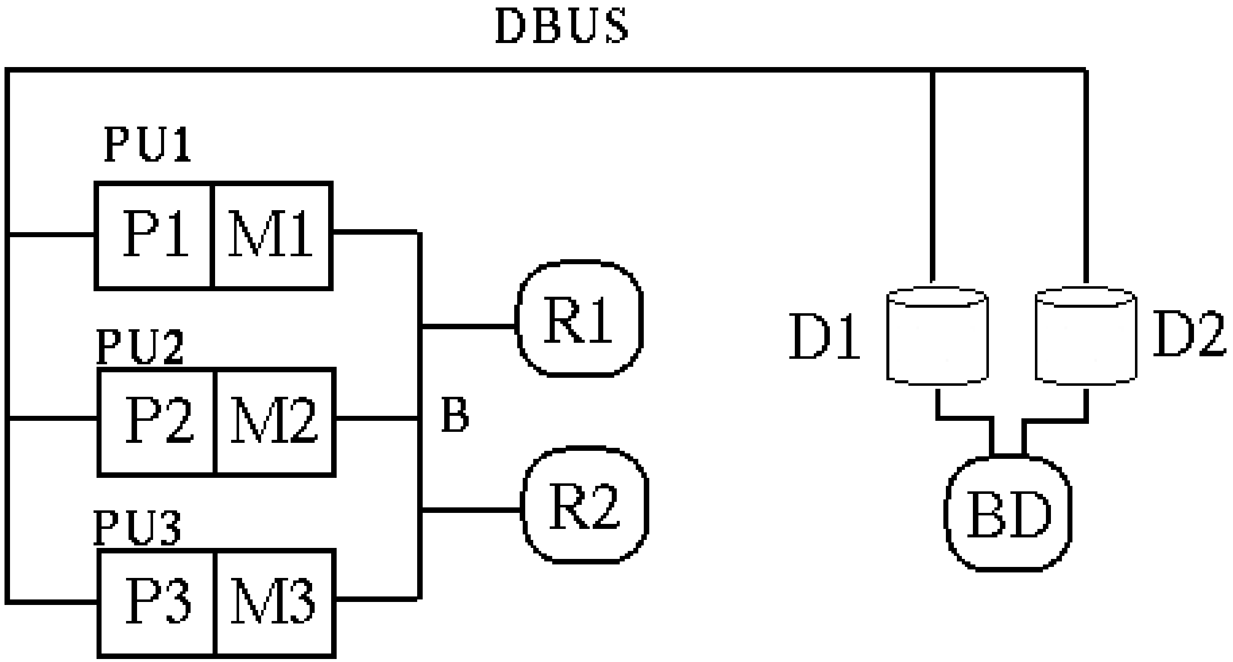
- M is the main component; S is its spare component.

- States of S:
  - Stand-by (dormant): $\alpha_s\lambda_s$
  - Working: λ_s
  - Failed
- λ_s is the failure rate
- α_s is the dormancy factor



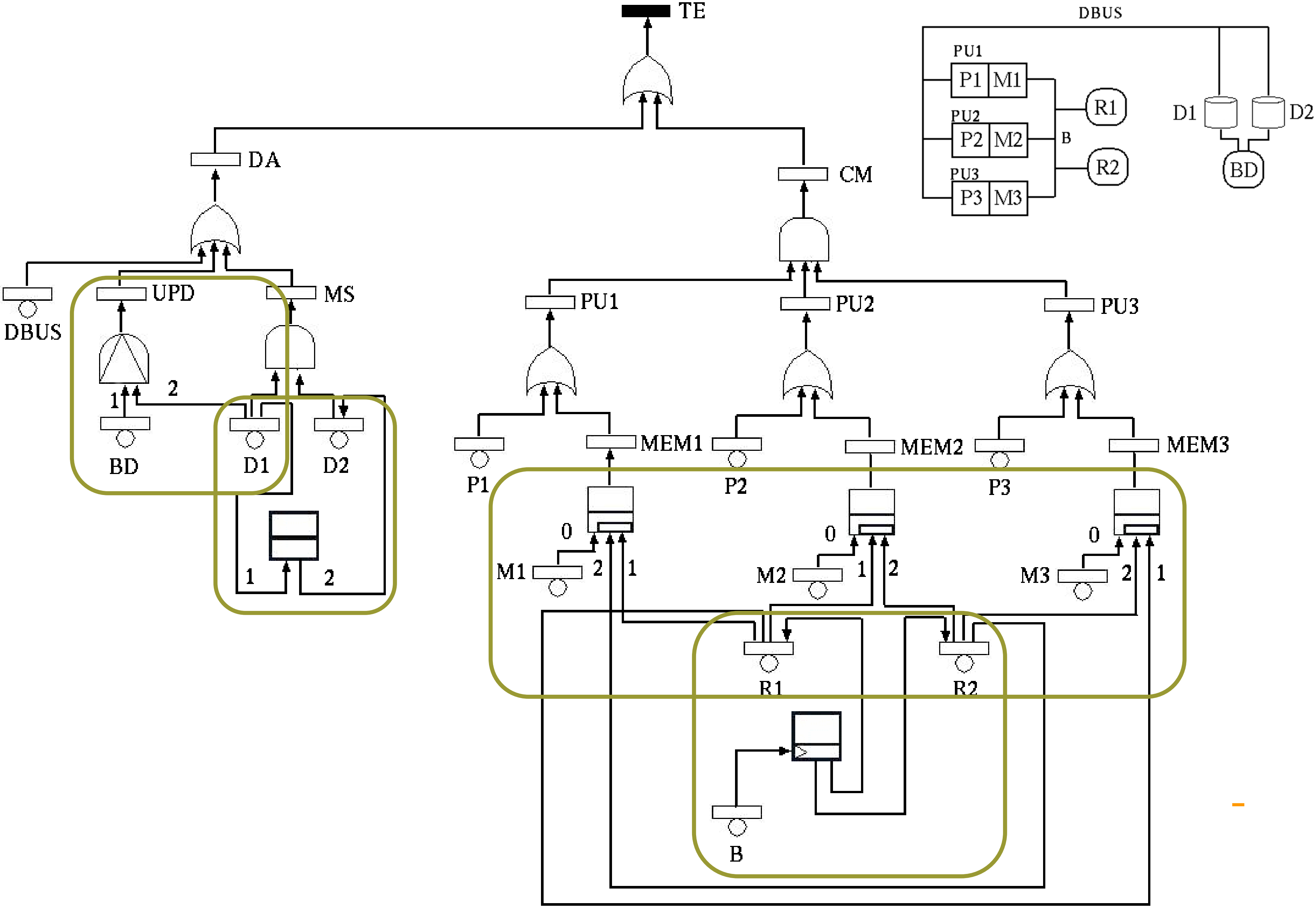
- Warm spare: $0 < \alpha < 1$
- Cold spare: $\alpha = 0$
- Hot spare: $\alpha = 1$

Example:
Multiprocessor Computing System



- R1 and R2 are warm spare memories. R1 and R2 functionally depend on the bus B.
- D1 is the primary disk; D2 is the backup disk. D2 can not fail before D1.
- BD is the device updating periodically D2. The failure of BD is relevant if it happens before the failure of D2.

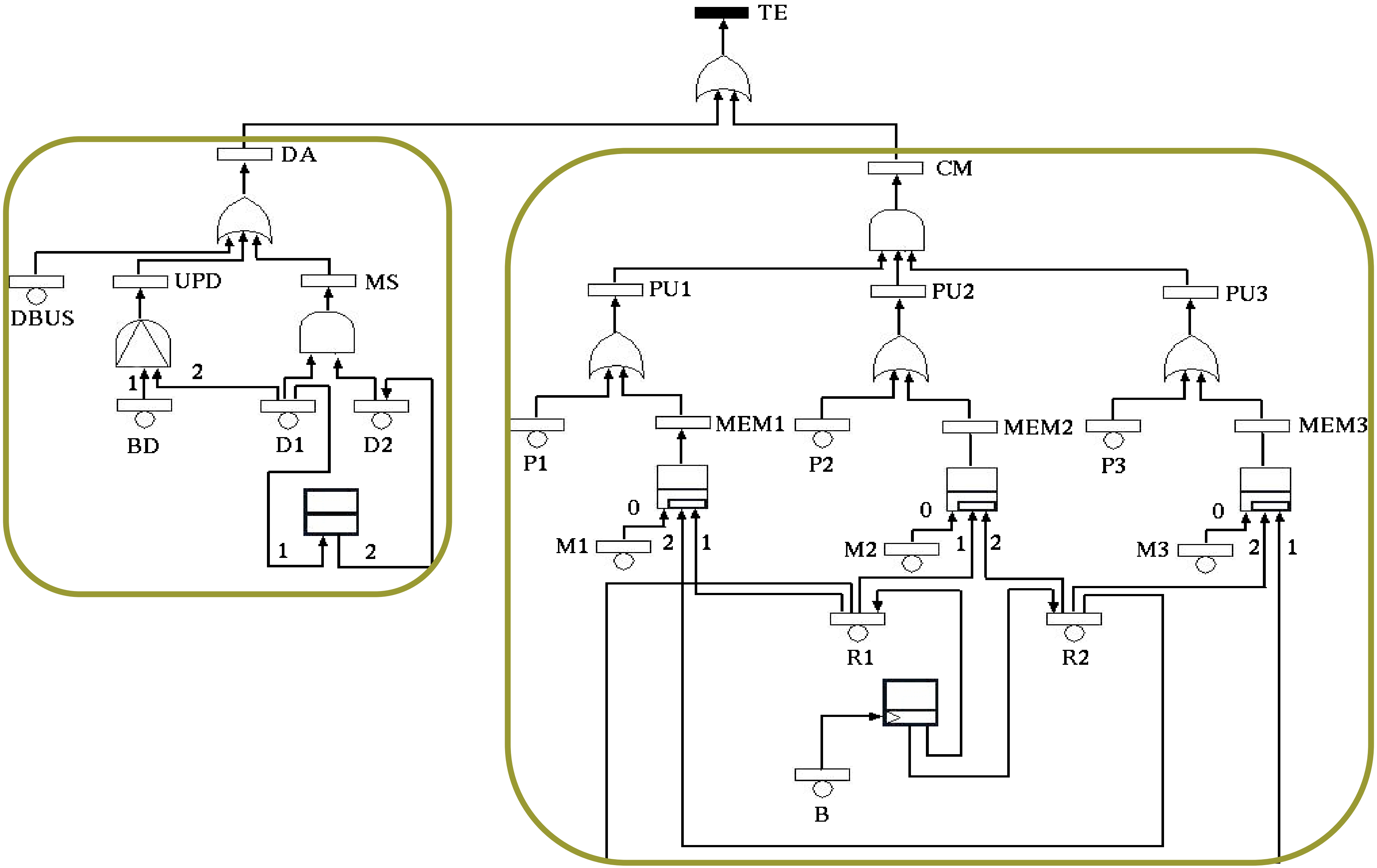
Example: Dynamic FT



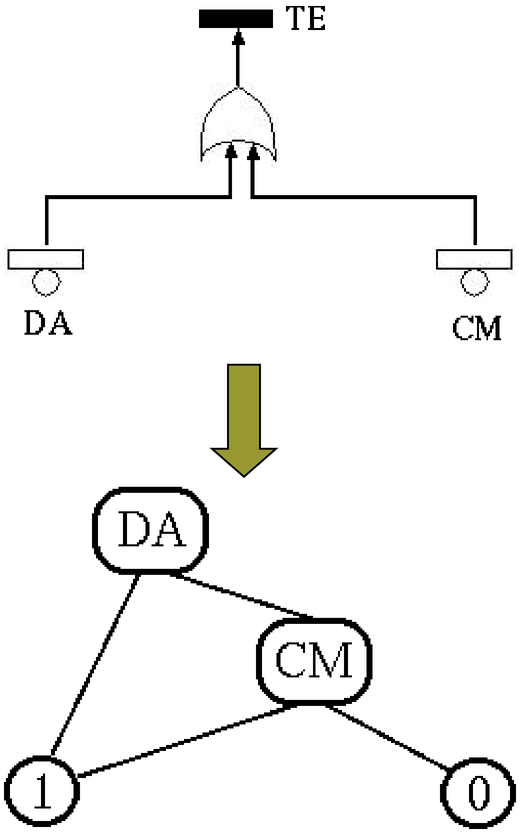
DFT Analysis

- Due to dependencies, DFTs need state space analysis.
- State space analysis can be limited to *dynamic modules* (**Modularization**).
- Modules analyzed through standard MC or through PN (e.g. GSPN)

Example: dynamic modules



Example: analysis results



Time	Pr(DA) GSPN	Pr(CM) GSPN	Pr(TE) BDD
2000 h	5.3904E-6	9.99E-10	5.3914E-6
4000 h	1.3555E-5	7.976E-9	1.3563E-5
6000 h	2.4486E-5	2.6879E-8	2.4512E-5
8000 h	3.8172E-5	6.3617E-8	3.8236E-5
10000 h	5.4605E-5	1.2406E-7	5.473E-5

- Module DA: 14 states ⇒ < 1 sec.
- Module CM: 487 states ⇒ < 1 sec.
- Whole DFT: 7806 states ⇒ 12 sec.
  - Pentium 4, 2 Mhz, 512 MB

Overview

- Dependability/Reliability issues
- Main Model Types for Reliability
- Probabilistic Graphical Models (BN and DBN)
  - Modeling
  - Computing
- From (Dynamic) Fault Trees to (Dynamic) Bayesian Nets
  - Modeling
  - Computing
- Case Studies
- Tools
- Open Issues

Probabilistic Graphical Models

- Static Models
  - Bayesian Networks (aka Causal Networks, Probabilistic Networks, Belief Networks,…)
  - Influence Diagrams
- Dynamic Models
  - Dynamic Bayesian Networks (2TBN)
  - Dynamic Decision Networks

Probabilistic Graphical Models

- Static Models
  - Bayesian Networks (aka Causal Networks, Probabilistic Networks, Belief Networks,…)
  - Influence Diagrams
- Dynamic Models
  - Dynamic Bayesian Networks (2TBN)
  - Dynamic Decision Networks

Bayesian Networks

- Bayesian (or Belief) Networks (BN) are a widely used formalism from AI (Artificial Intelligence) for representing uncertain knowledge in probabilistic systems, applied to a variety of real-world problems *[J. Pearl, Probabilistic Reasoning in Inteligence Systems, Morgan Kaufmann, 1988]*
- BN are defined by a directed acyclic graph in which (discrete) random variables are assigned to each node, together with the quantitative conditional dependence on the parent nodes (Conditional Probability Table or CPT)

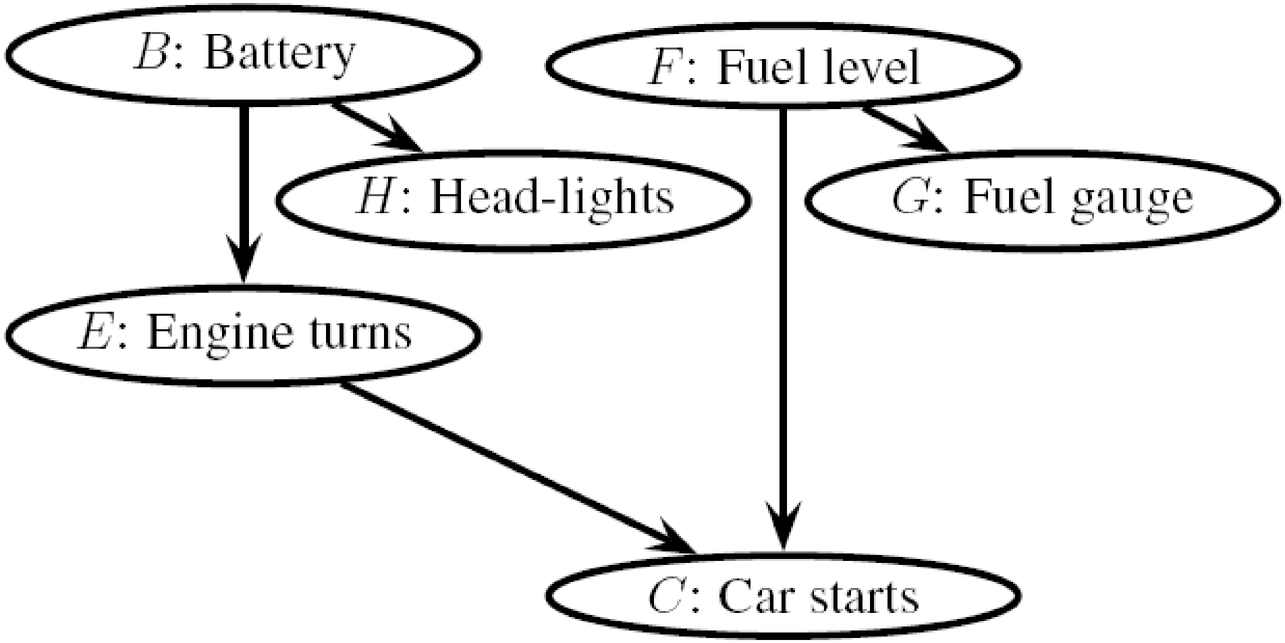
BN: definition

- A Bayesian Network is a pair <*G*,*P*> where
  - *G* is a Directed Acyclic Graph (DAG) with
    - nodes representing (discrete) random variables
    - an oriented arc *X* → *Y* represents a dependency relation of *Y* from *X* (*X* influences *Y*, *Y* depends on *X*, *X* causes *Y*, etc…)
  - *P* is a probability distribution over the random variables represented by the nodes $X_1, \ldots X_n$ of the DAG such that

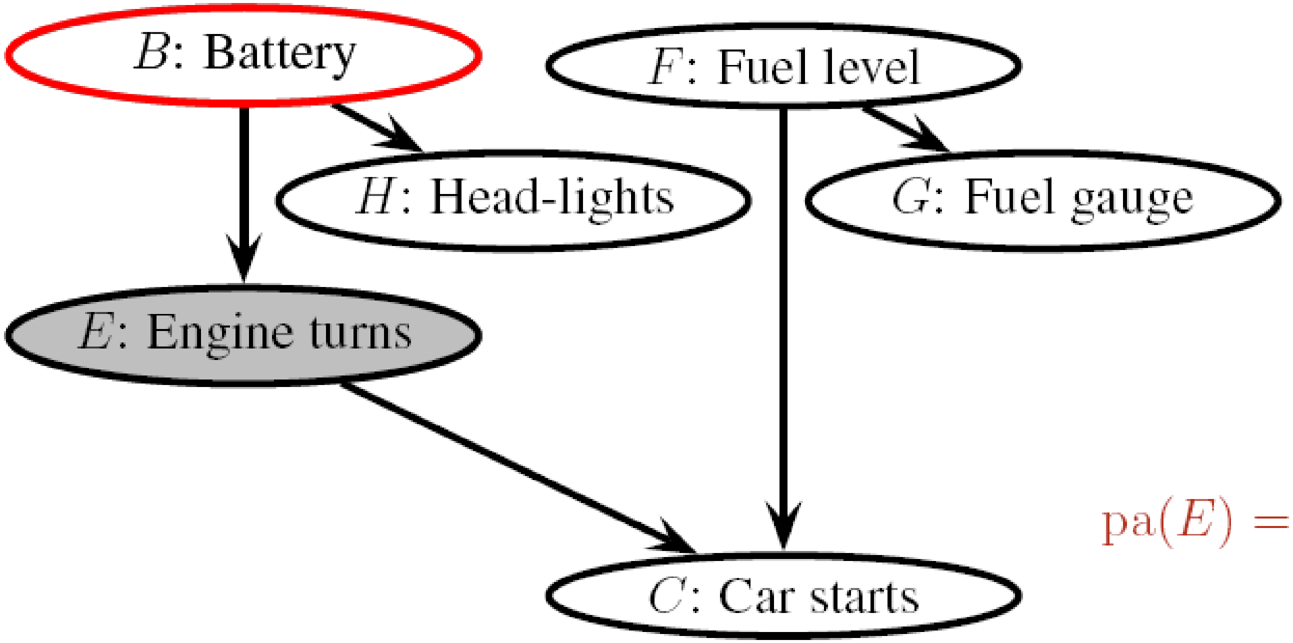
$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i \mid pa(X_i))$$

⇨ Specification of a CPT local to each node

Example: car start (H. Langseth)

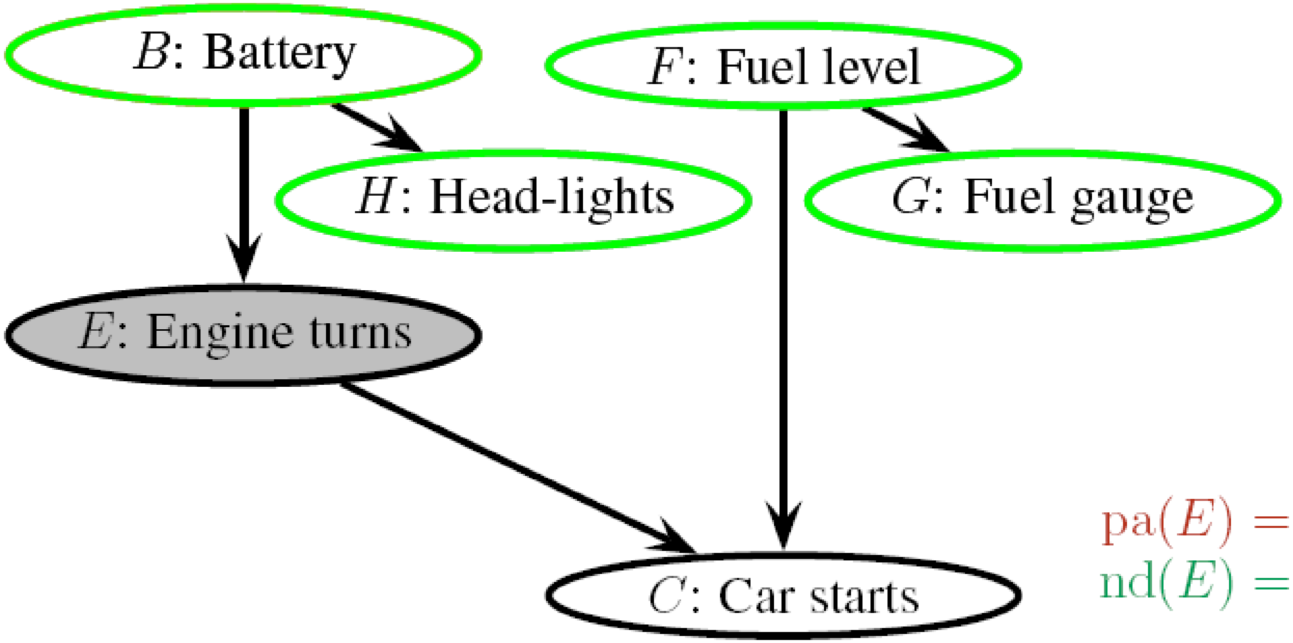


$$P(B, F, H, G, E, C)$$



$\mathrm{pa}(E) = \{B\}$

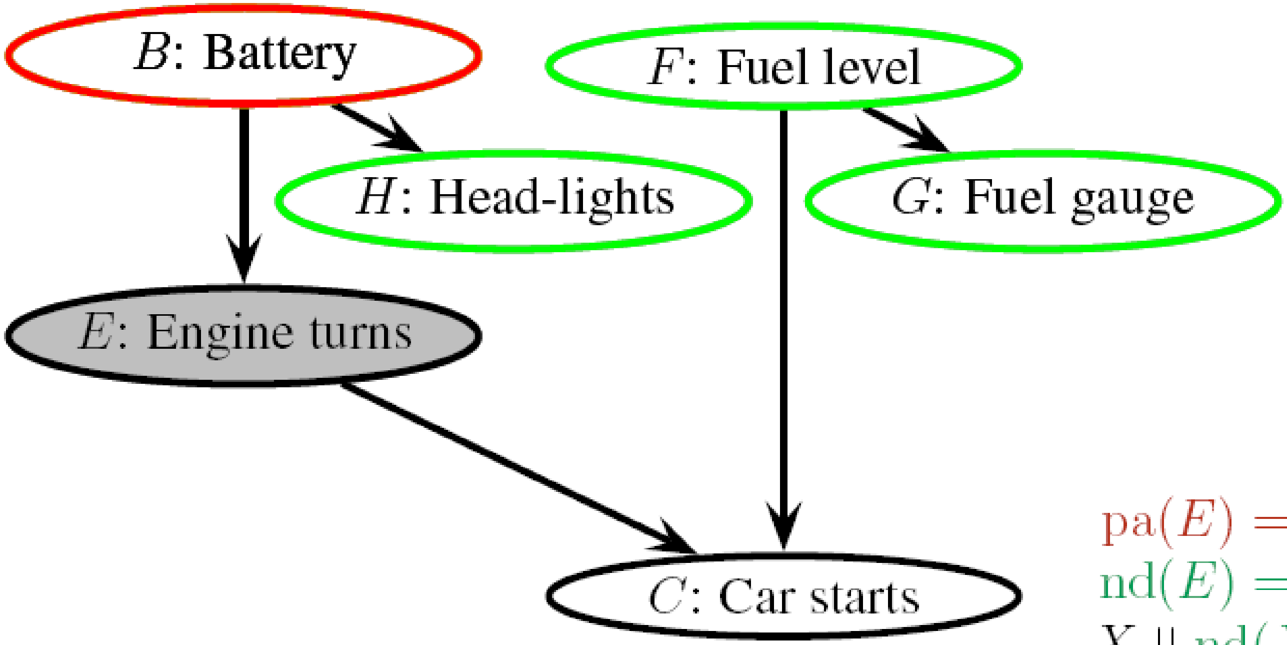
$P(B, F, H, G, E, C)$



$\mathrm{pa}(E) = \{B\}$

$\mathrm{nd}(E) = \{H, G, F, B\}$

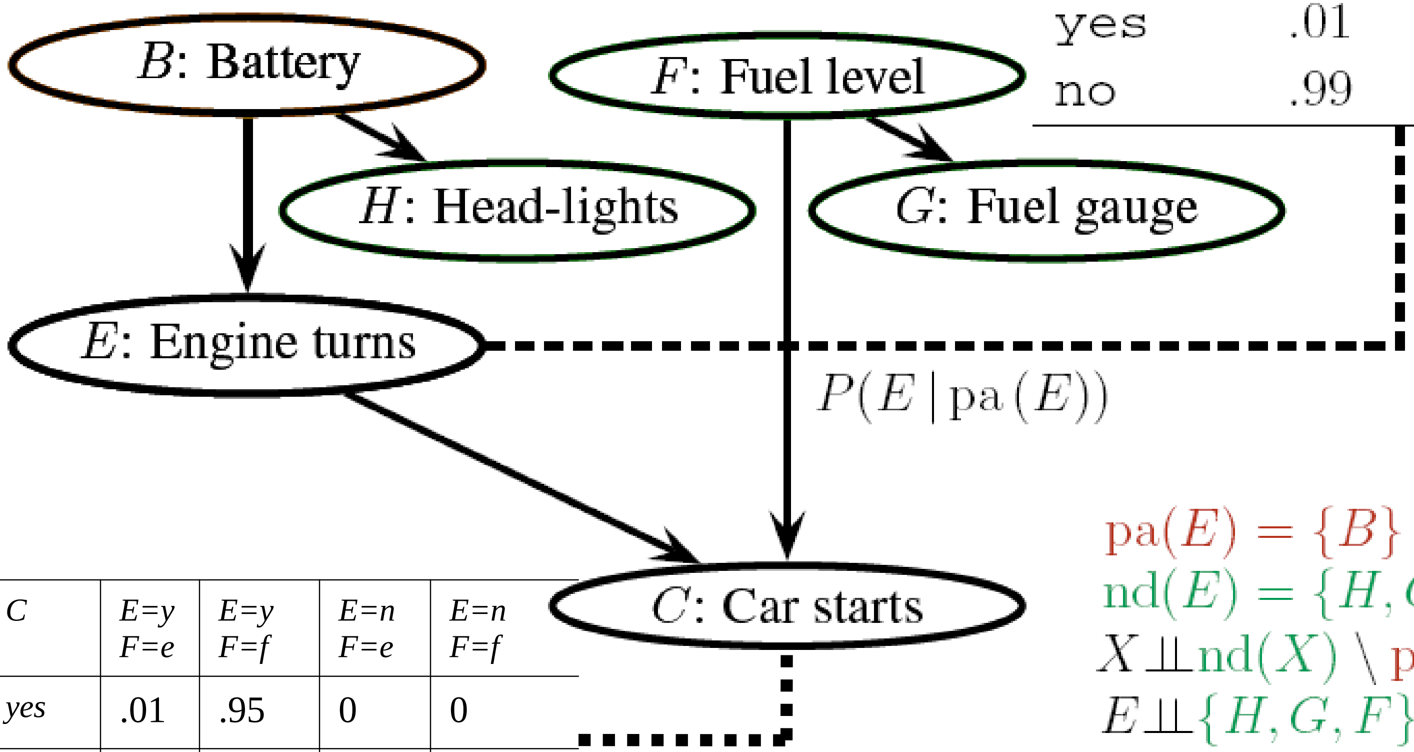
$P(B, F, H, G, E, C)$



$\mathrm{pa}(E) = \{B\}$
$\mathrm{nd}(E) = \{H, G, F, B\}$
$X \perp\!\!\!\perp \mathrm{nd}(X) \setminus \mathrm{pa}(X) \mid \mathrm{pa}(X)$
$E \perp\!\!\!\perp \{H, G, F\} \mid B$
Other d-sep. rules: Pearl(88)

$P(B, F, H, G, E, C)$

E	$B =$empty	$B \neq$empty
yes	.01	.97
no	.99	.03



C	*E=y* *F=e*	*E=y* *F=f*	*E=n* *F=e*	*E=n* *F=f*
yes	.01	.95	0	0
no	.99	.05	1	1

$\mathrm{pa}(E) = \{B\}$

$\mathrm{nd}(E) = \{H, G, F, B\}$

$X \perp\!\!\!\perp \mathrm{nd}(X) \setminus \mathrm{pa}(X) \mid \mathrm{pa}(X)$

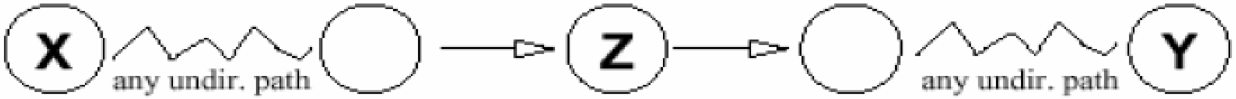
$E \perp\!\!\!\perp \{H, G, F\} \mid B$

Other d-sep. rules: Pearl(88)

$$P(B, F, H, G, E, C) = P(B)P(F)P(H \mid B)P(G \mid F) \cdot P(E \mid B)P(C \mid E, F)$$

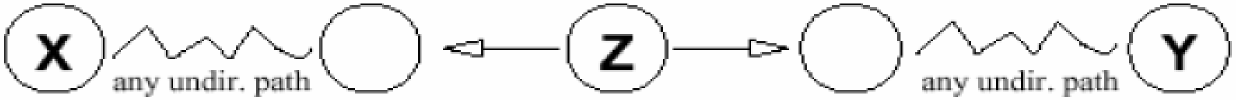
Blocking: Graphical View

(1)



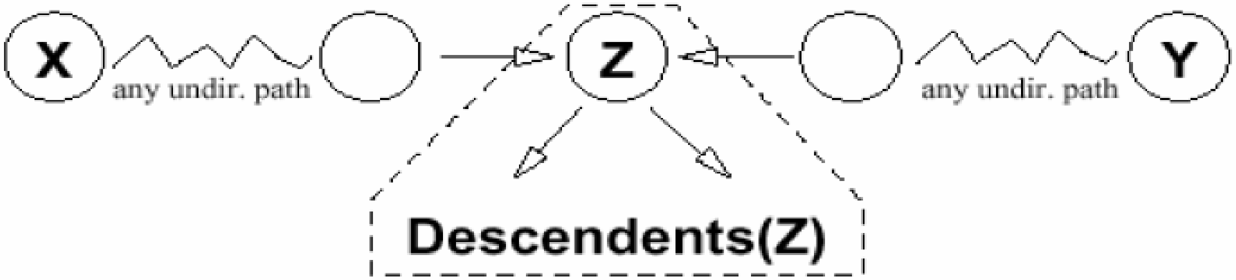
If Z in evidence, the path between X and Y blocked

(2)



If Z in evidence, the path between X and Y blocked

(3)

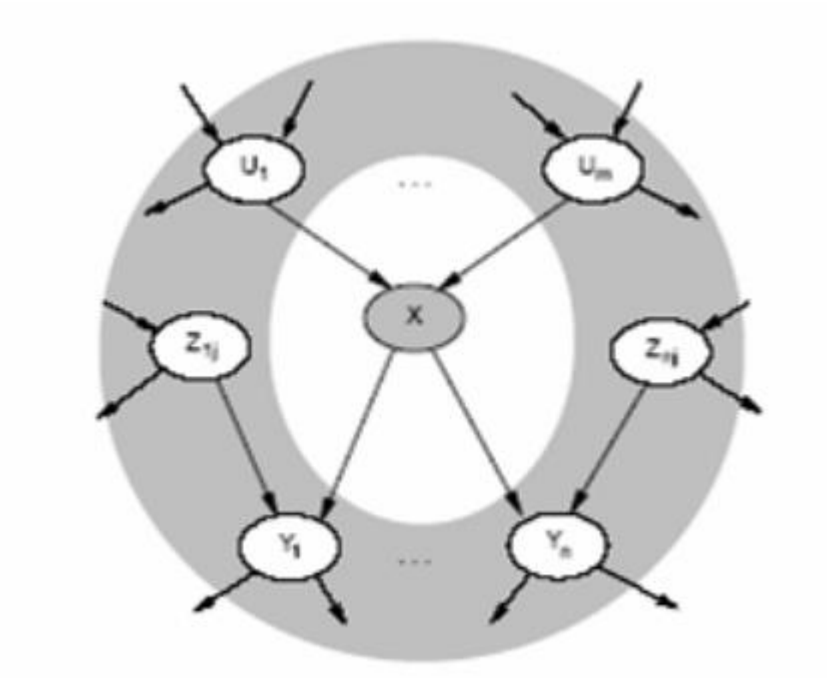


If Z is ***not*** in evidence and***no*** descendent of Z is in evidence, then the path between X and Y is blocked

More on independence: Markov Blanket

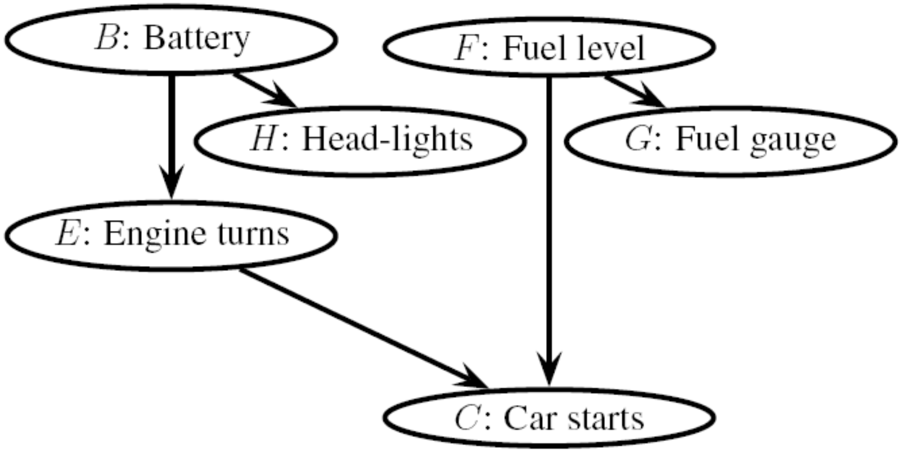
MB(X)= parents(X) U children(X) U mates(X)

X is independent of any other nodes of the network given *MB(X)*

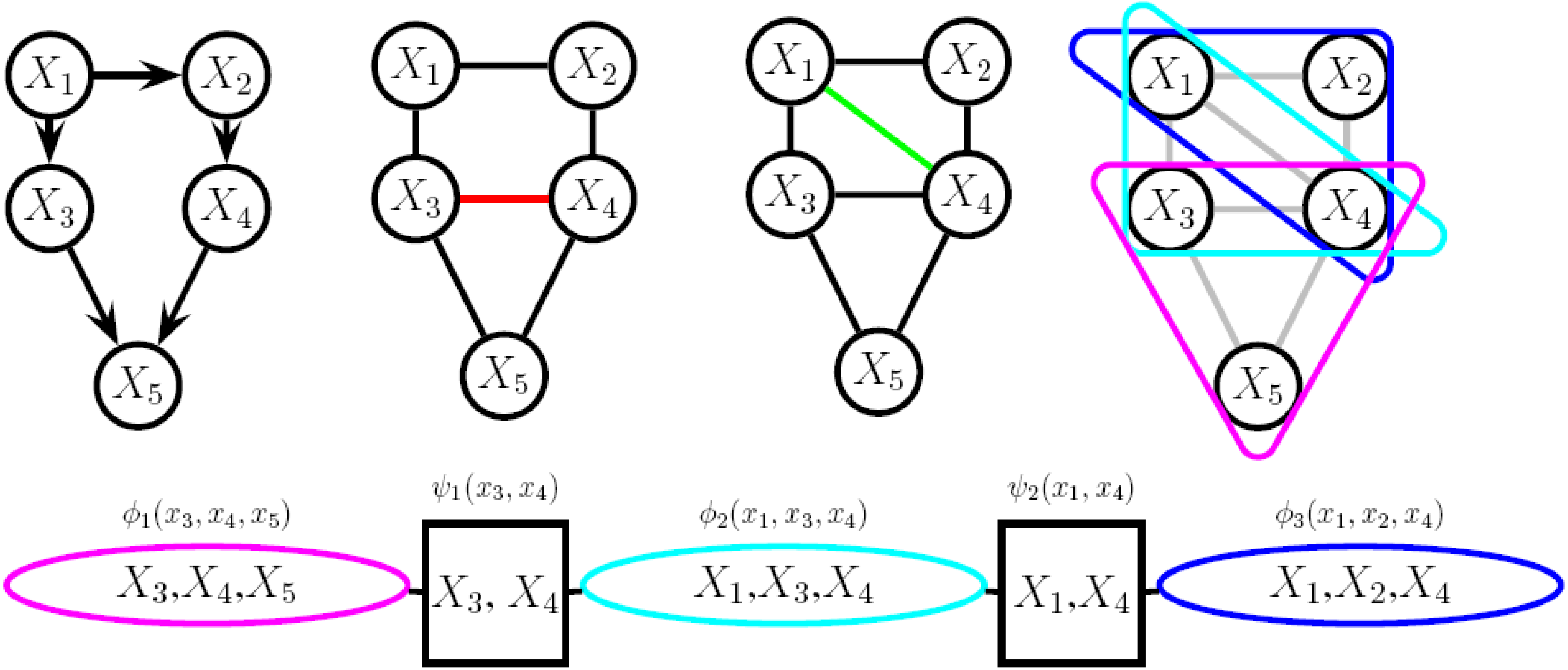


Inference: probabilistic computations

- Diagnostic inference
  - Pr(cause | effect)
    - Pr(B | C)
    - Pr(F| G)
- Predictive inference
  - Pr(effect | cause)
    - Pr(C | B)
    - Pr(C | F)
- Combined Inference
  - Pr(intermediate|cause, effect)
    - Pr(E | B, C)
- Exact algorithms (*Clustering*, *Conditioning*, *Variable Elimination*) or approximated algorithms (*Stochastic Simulation*) for BN inference



Clustering Computation Scheme

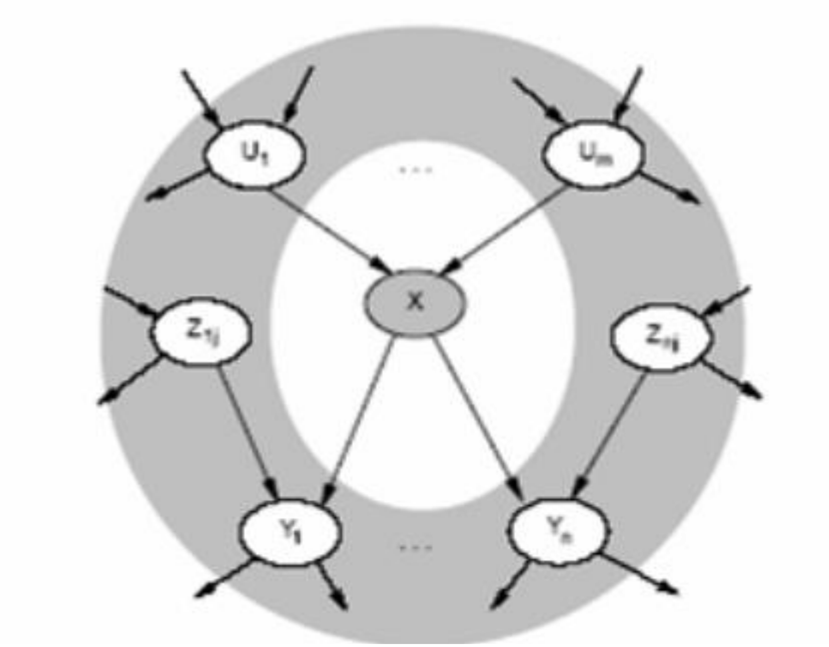


Junction (Join) tree

Advantage: dealing with 3 variables instead of 5

Approximate Inference: MCMC (Gibbs sampling)

Each node *X* is independent from the rest of thje network given the *MB(X)* → sample a value of *X* from the net distribution, given a specific instance of *MB(X)*



Probability given the Markov blanket is calculated as follows:

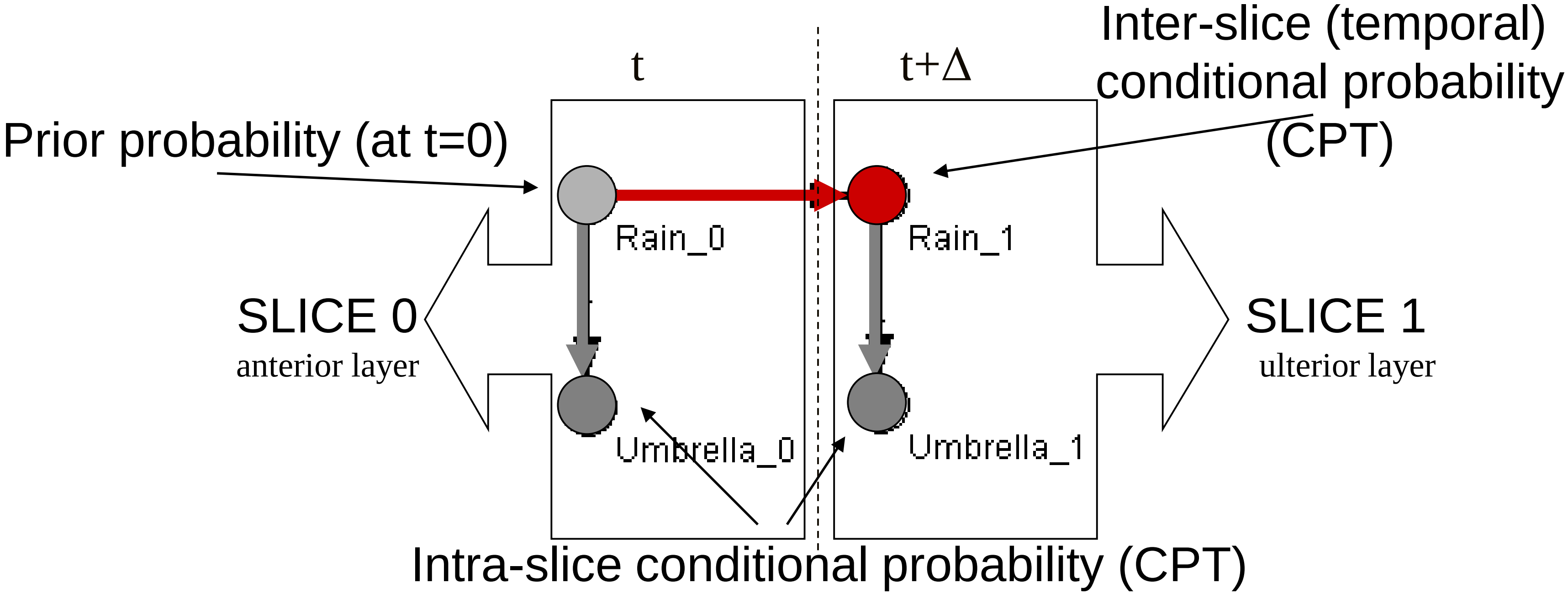
$$P(x_i'|mb(X_i)) = P(x_i'|parents(X_i))\Pi_{Z_j\in Children(X_i)}P(z_j|parents(Z_j))$$

Gibbs sampling

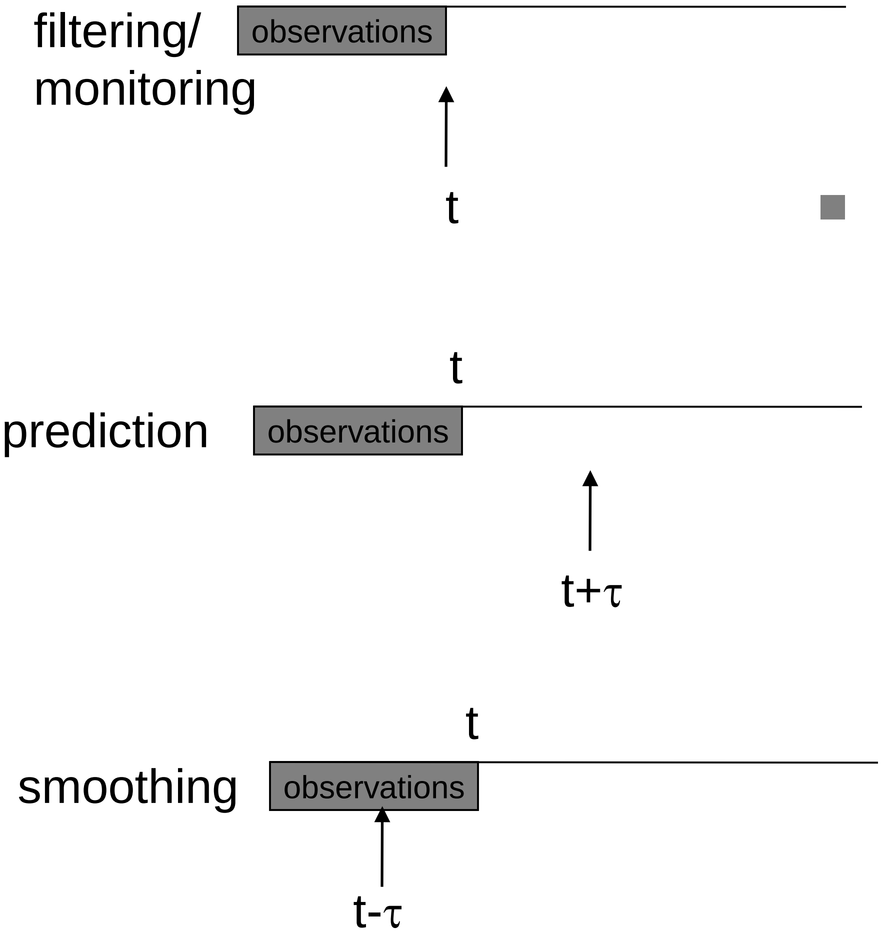
```
set $X_1,X_2, \dots X_n$ as a random instance
for $j=1$ to $MaxRun$ do
    for $i=1$ to $n$ do
        if $X_i$ in evidence then $X_i$=observation
                else sample $X_i$ from $P(X_i|MB(X_i))$
Estimate $P(e|D)=\frac{\#runs(e)}{\#runs}$
```

Dynamic Bayesian Networks

- DBN introduce a **discrete** temporal dimension:
  - The system is represented at several time slices
  - Conditional dependencies among variables at different slices, are introduced to capture the temporal evolution.
  - Time invariance is assumed: typically 2 time slices (t, t+Δ) are assumed in DBN: Markovian assumption (2TBN)


Inter-slice (temporal) conditional probability (CPT)
t
t+Δ
Prior probability (at t=0)
Rain_0
Rain_1
SLICE 0
anterior layer
SLICE 1
ulterior layer
Umbrella_0
Umbrella_1
Intra-slice conditional probability (CPT)

Inference in DBN



- Algorithms
  - 1.5 Junction tree (Murphy 02)
  - BK approximation (Boyen-Koller 98)
  - Particle filtering simulation

Overview

- Dependability/Reliability issues
- Main Model Types for Reliability
- Probabilistic Graphical Models (BN and DBN)
  - ❑ Modeling
  - ❑ Computing
- From Fault Trees to Bayesian Nets
  - ❑ Modeling
  - ❑ Computing
- Case Studies
- Tools
- Open Issues

BN vs FTA

BNs may improve both the modeling and the analysis power wrt FT:

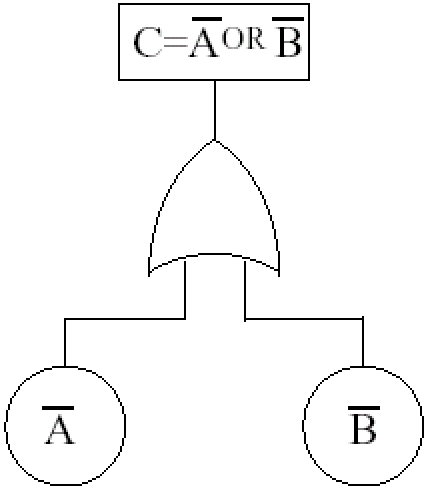
Modeling Issues:

- Local conditional dependencies, probabilistic gates, multi-state variables, dependent failures, uncertainty in model parameters.

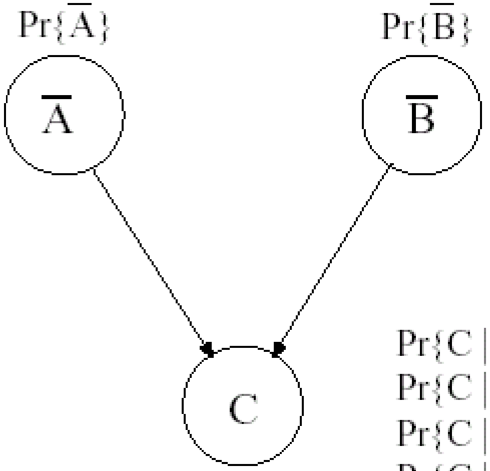
Analysis Issues:

- A forward (or predictive) analysis
- A backward (diagnostic) analysis, the posterior probability of any set of variables is computed.

OR gate vs BN node



FAULT - TREE: OR Gate

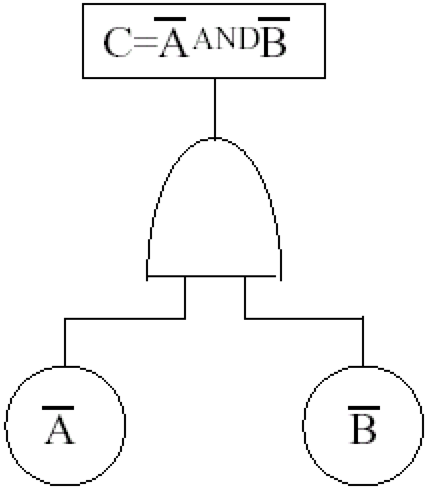


$$\left.\begin{array}{l} Pr\{C \mid \overline{A}, \overline{B}\}=1 \\ Pr\{C \mid \overline{A}, B\}=1 \\ Pr\{C \mid A, \overline{B}\}=1 \end{array}\right\} \textbf{cpt}$$

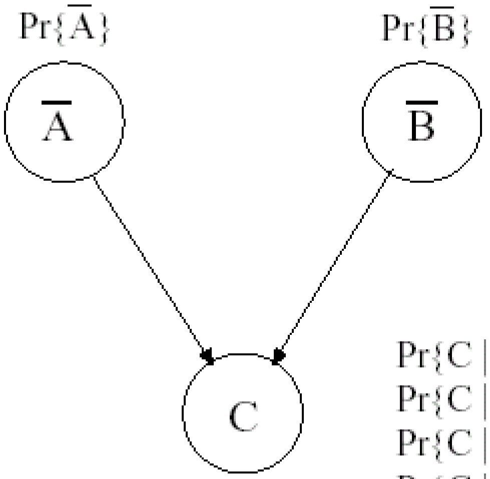
$Pr\{C \mid A, B\}=0$

BAYESIAN NETWORK: OR Node

AND gate vs BN node



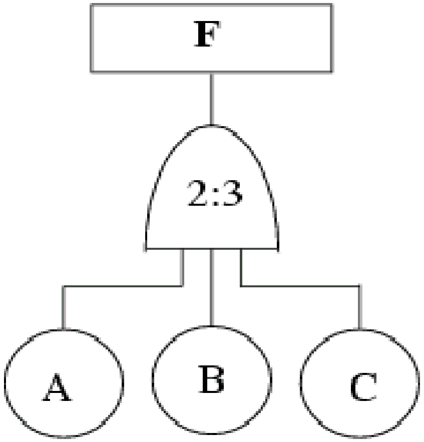
FAULT - TREE: AND Gate

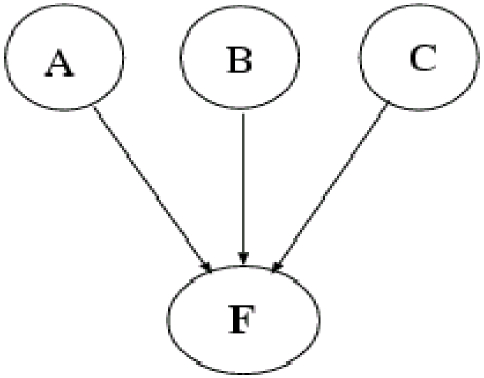


$$\left.\begin{array}{l}\Pr\{C \mid \overline{A}, \overline{B}\}=1\\ \Pr\{C \mid \overline{A}, B\}=0\\ \Pr\{C \mid A, \overline{B}\}=0\\ \Pr\{C \mid A, B\}=0\end{array}\right\}\ \textbf{cpt}$$

BAYESIAN NETWORK: AND Node

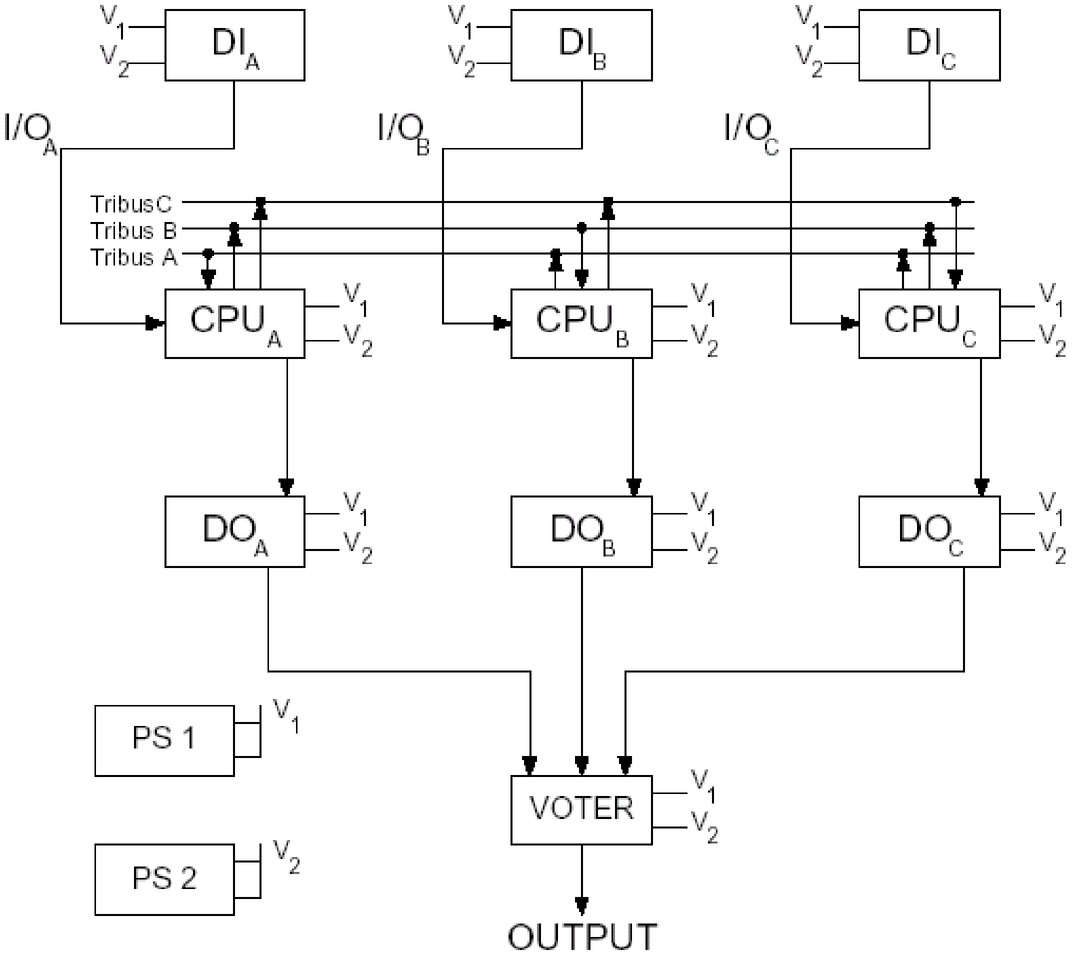
k:n gate vs BN node



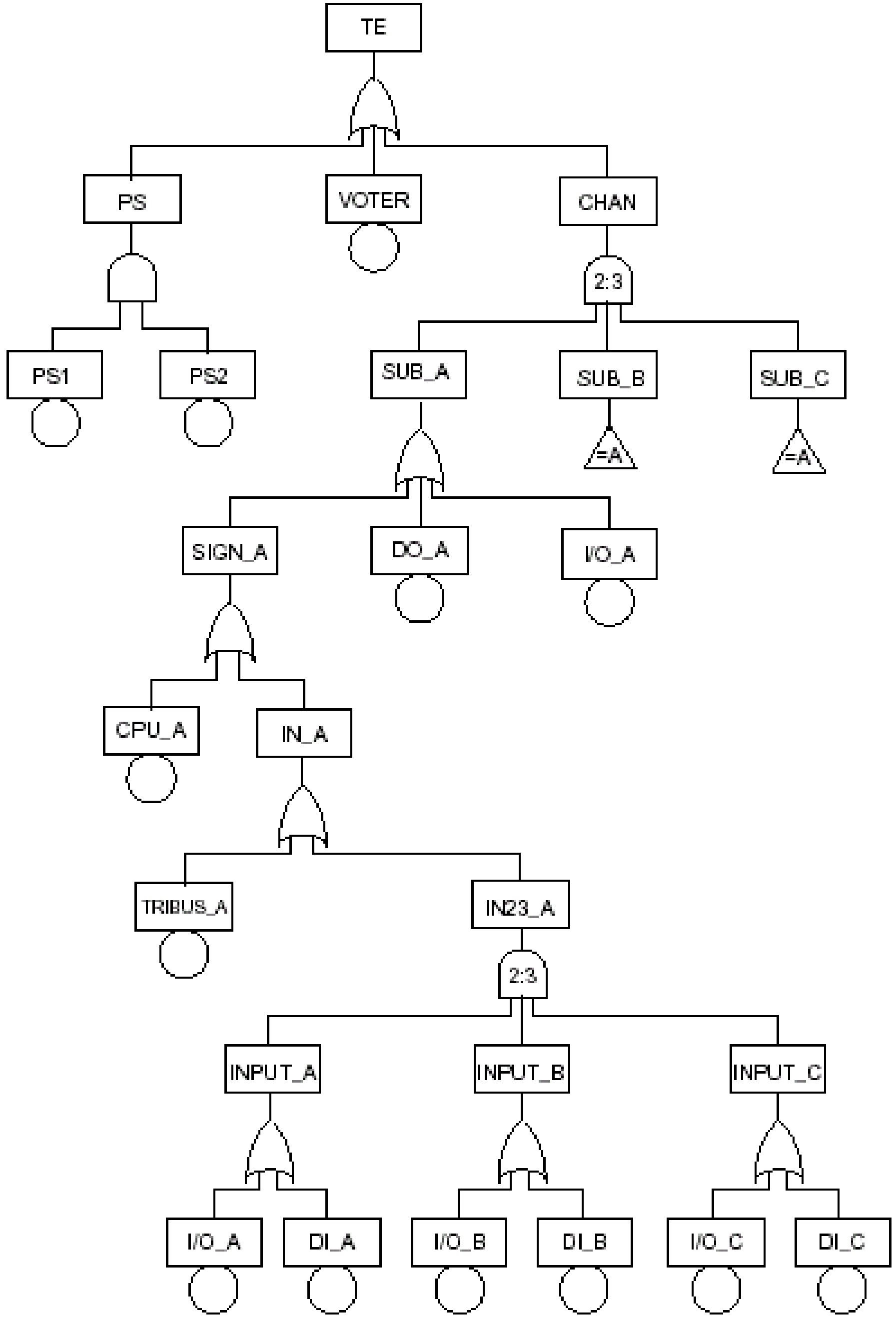


$$\left.\begin{array}{lcl}
Pr\{F=1|A=0,B=0,C=0\} & = & 0 \\
Pr\{F=1|A=0,B=0,C=1\} & = & 0 \\
Pr\{F=1|A=0,B=1,C=0\} & = & 0 \\
Pr\{F=1|A=1,B=0,C=0\} & = & 0 \\
Pr\{F=1|A=0,B=1,C=1\} & = & 1 \\
Pr\{F=1|A=1,B=0,C=1\} & = & 1 \\
Pr\{F=1|A=1,B=1,C=0\} & = & 1 \\
Pr\{F=1|A=1,B=1,C=1\} & = & 1
\end{array}\right\}\ \textbf{cpt}$$

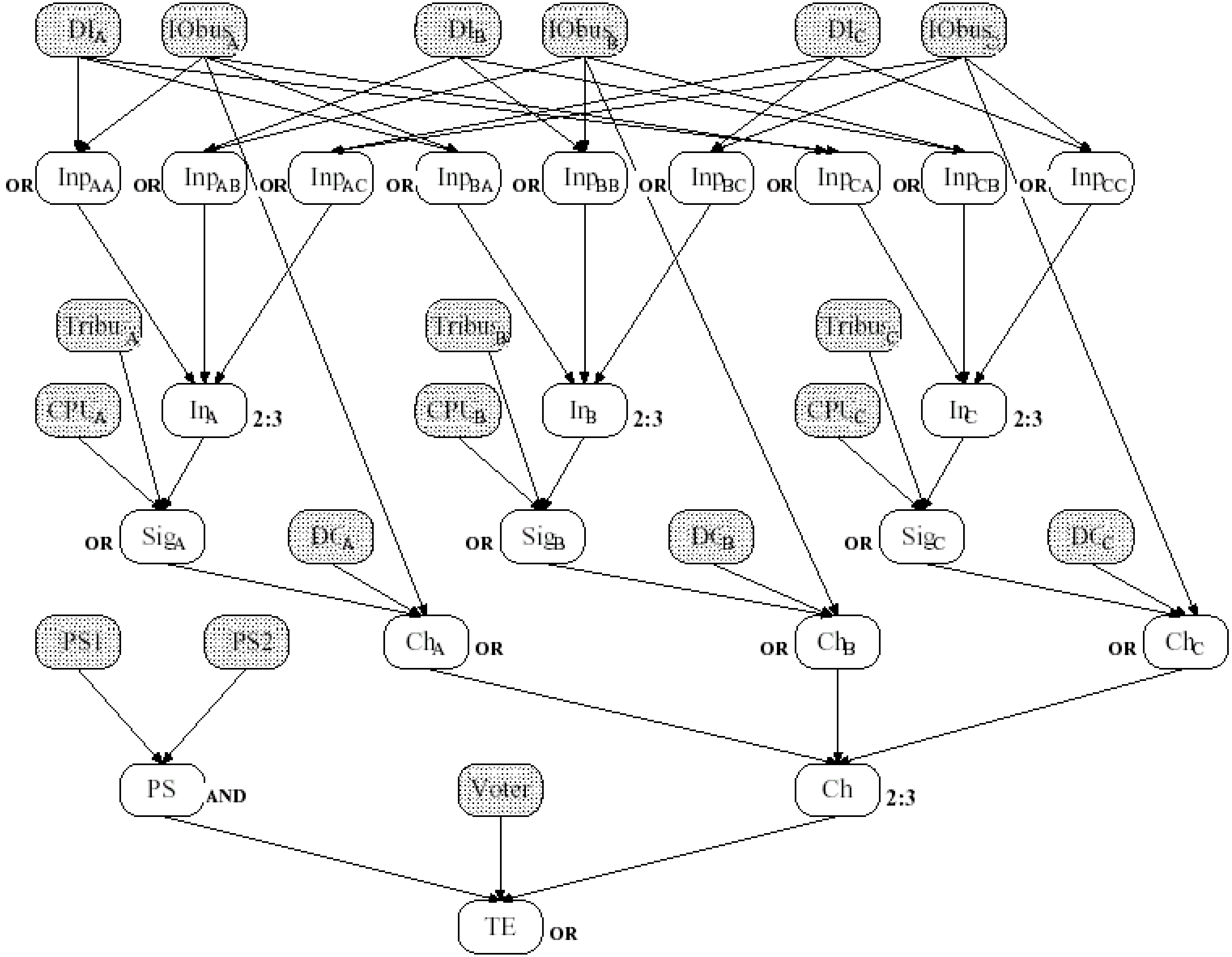
Example: a PLC architecture



PLC:
the FT

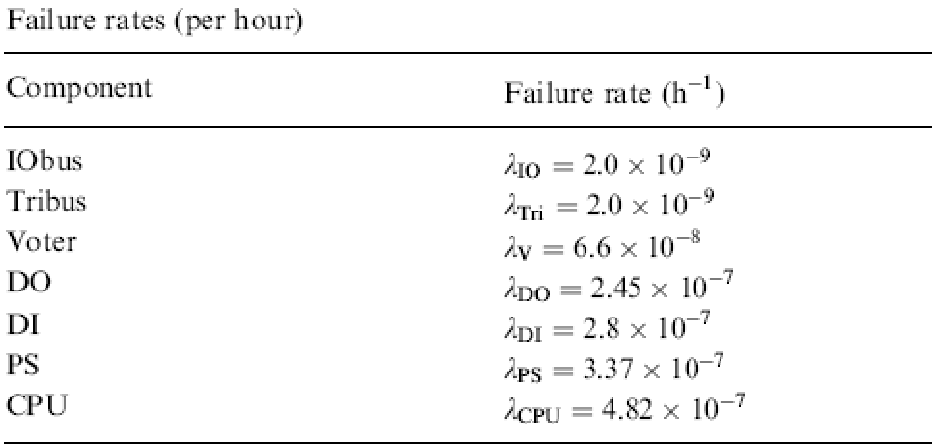

TE
PS
VOTER
CHAN
2:3
PS1
PS2
SUB_A
SUB_B
SUB_C
=A
=A
SIGN_A
DO_A
I/O_A
CPU_A
IN_A
TRIBUS_A
IN23_A
2:3
INPUT_A
INPUT_B
INPUT_C
I/O_A
DI_A
I/O_B
DI_B
I/O_C
DI_C

PLC: the BN


DI_A
IObus_A
DI_B
IObus_B
DI_C
IObus_C
OR Inp_AA
OR Inp_AB
OR Inp_AC
OR Inp_BA
OR Inp_BB
OR Inp_BC
OR Inp_CA
OR Inp_CB
OR Inp_CC
Tribus_A
Tribus_B
Tribus_C
CPU_A
In_A 2:3
CPU_B
In_B 2:3
CPU_C
In_C 2:3
OR Sig_A
DC_A
OR Sig_B
DC_B
OR Sig_C
DC_C
PS1
PS2
Ch_A OR
OR Ch_B
OR Ch_C
PS AND
Voter
Ch 2:3
TE OR

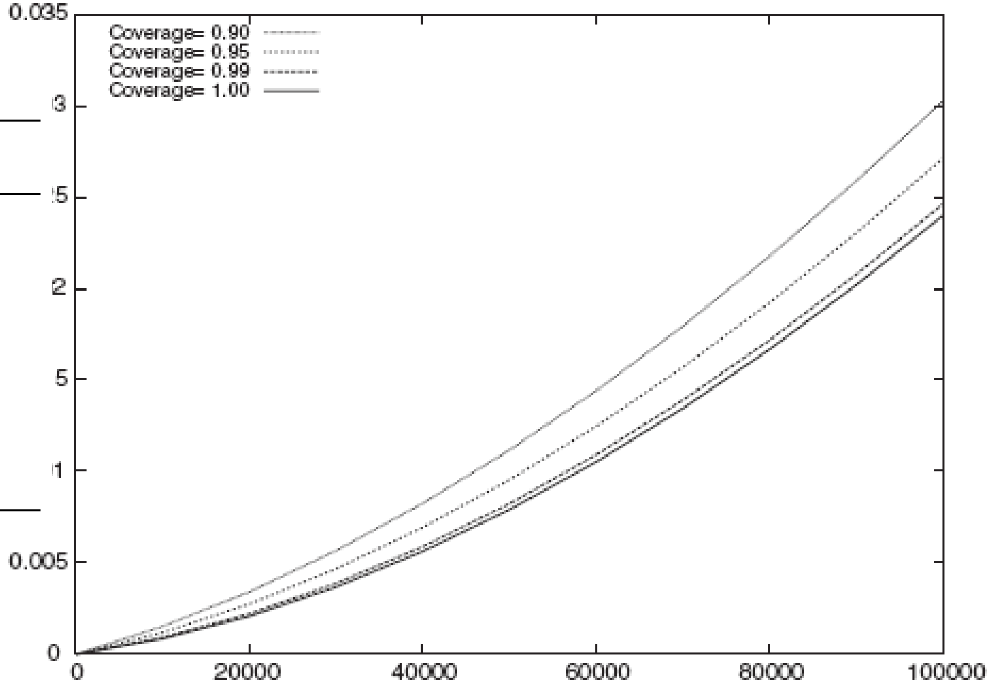
Analysis Tasks

- Probability of TE at time t (system’s unreliability)
  - Query: *P(TE)* using the probability of basic events (i.e. BN roots) computed at time t (e.g. $P(C=true)=1-e^{-\lambda t}$)



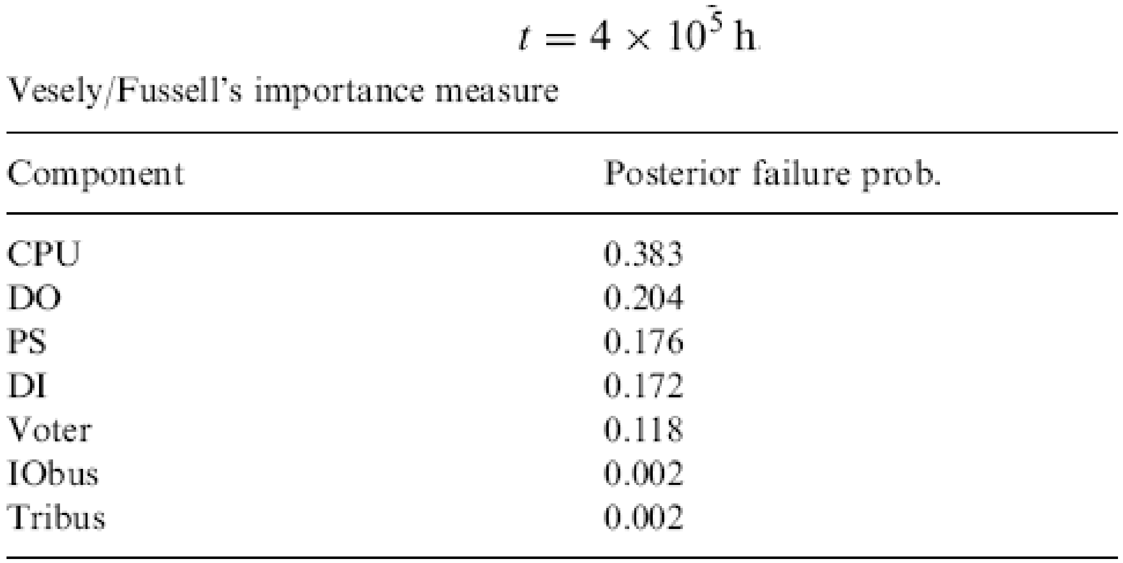
Failure rates (per hour)

Component	Failure rate (h^{-1})
IObus	$\lambda_{IO} = 2.0 \times 10^{-9}$
Tribus	$\lambda_{Tri} = 2.0 \times 10^{-9}$
Voter	$\lambda_{V} = 6.6 \times 10^{-8}$
DO	$\lambda_{DO} = 2.45 \times 10^{-7}$
DI	$\lambda_{DI} = 2.8 \times 10^{-7}$
PS	$\lambda_{PS} = 3.37 \times 10^{-7}$
CPU	$\lambda_{CPU} = 4.82 \times 10^{-7}$



Analysis Tasks

- Posterior probability of each component *C* given the system failure (Fussell-Vesely importance) at time *t*
  - Query: *P(C | TE)* by using priors on roots at time *t*



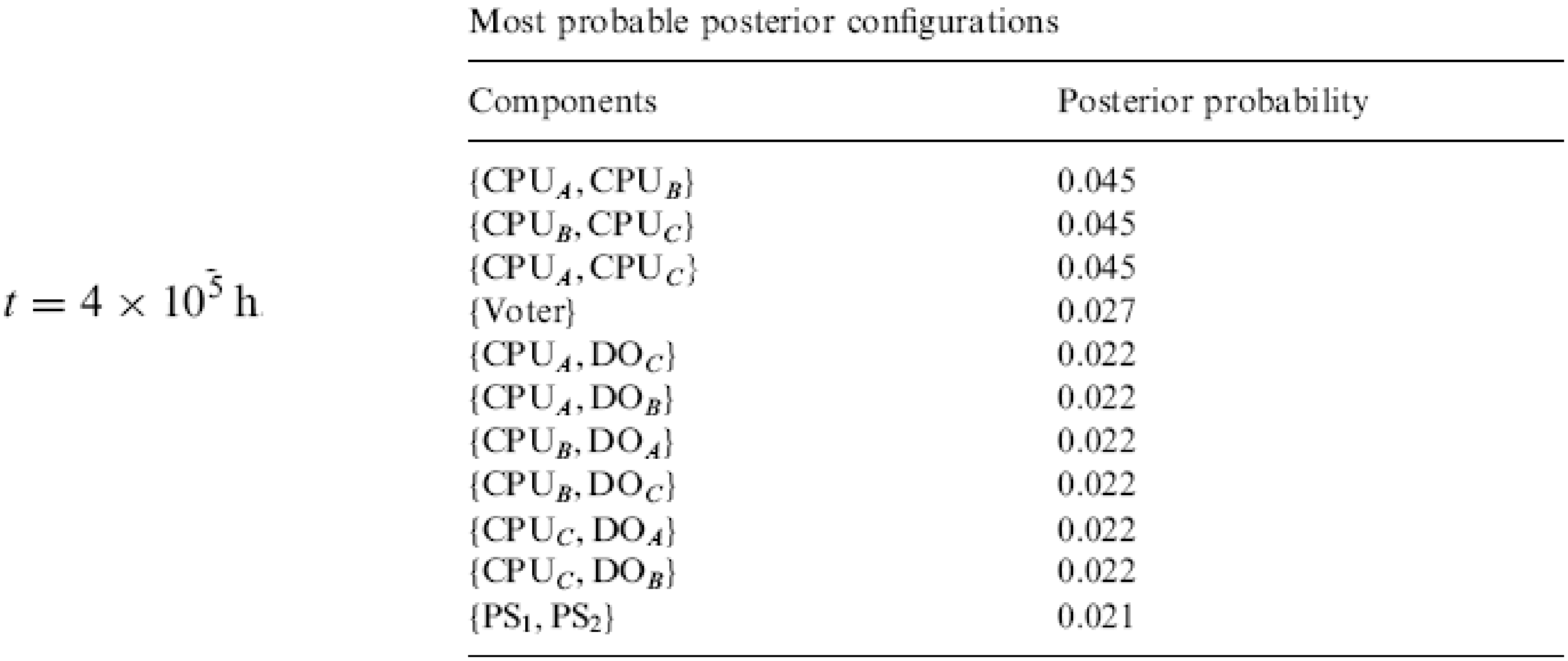
$t = 4 \times 10^5$ h

Vesely/Fussell's importance measure

Component	Posterior failure prob.
CPU	0.383
DO	0.204
PS	0.176
DI	0.172
Voter	0.118
IObus	0.002
Tribus	0.002

Analysis Tasks

- Posterior probability of a set of components given the system failure at time *t*
  - Query $P(C_1, \dots C_n \mid TE)$ at time *t*



$t = 4 \times 10^5$ h

Most probable posterior configurations

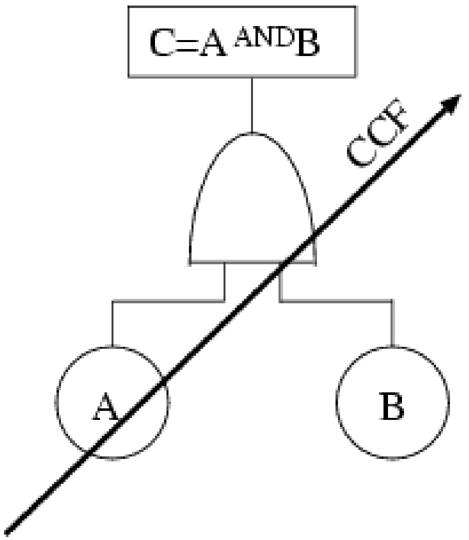
Components	Posterior probability
$\{\mathrm{CPU}_A, \mathrm{CPU}_B\}$	0.045
$\{\mathrm{CPU}_B, \mathrm{CPU}_C\}$	0.045
$\{\mathrm{CPU}_A, \mathrm{CPU}_C\}$	0.045
$\{\mathrm{Voter}\}$	0.027
$\{\mathrm{CPU}_A, \mathrm{DO}_C\}$	0.022
$\{\mathrm{CPU}_A, \mathrm{DO}_B\}$	0.022
$\{\mathrm{CPU}_B, \mathrm{DO}_A\}$	0.022
$\{\mathrm{CPU}_B, \mathrm{DO}_C\}$	0.022
$\{\mathrm{CPU}_C, \mathrm{DO}_A\}$	0.022
$\{\mathrm{CPU}_C, \mathrm{DO}_B\}$	0.022
$\{\mathrm{PS}_1, \mathrm{PS}_2\}$	0.021

Advanced Modeling Features

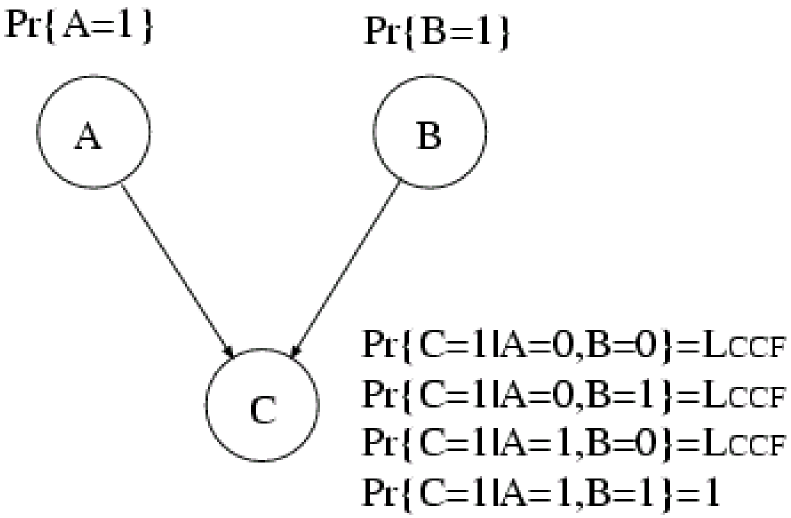
BN can also improve the modeling power wrt FT

- Probabilistic Gates
- Multi-state Variables
- Sequentially Dependent Faults
- Parameter Uncertainty

Probabilistic Gates: Common Cause Failure

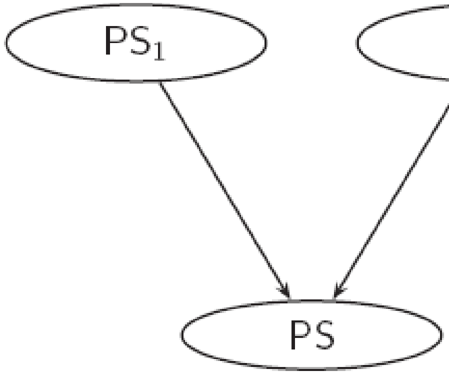


FAULT - TREE: AND Gate
With Common Cause Failures

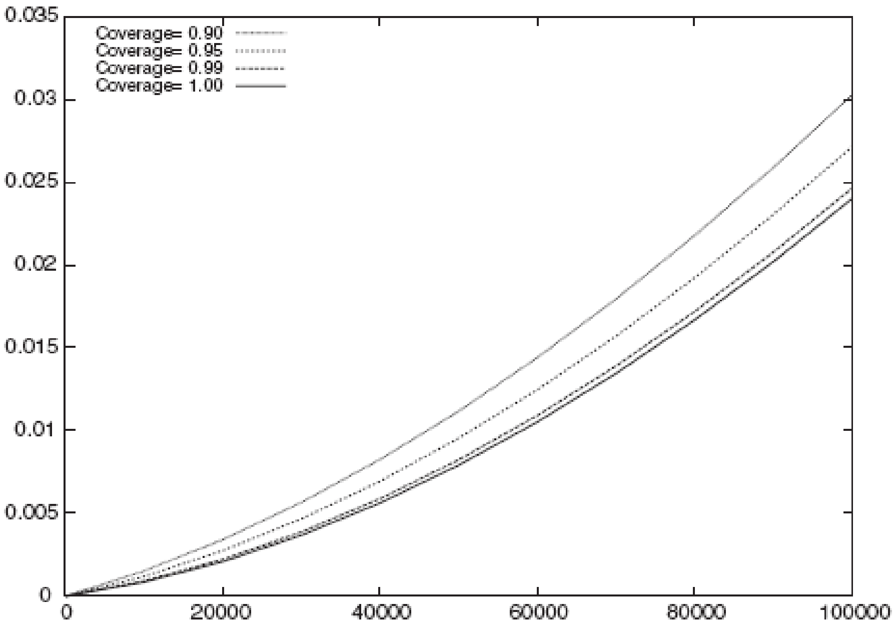


BAYESIAN NETWORK: AND Node
With Common Cause Failures

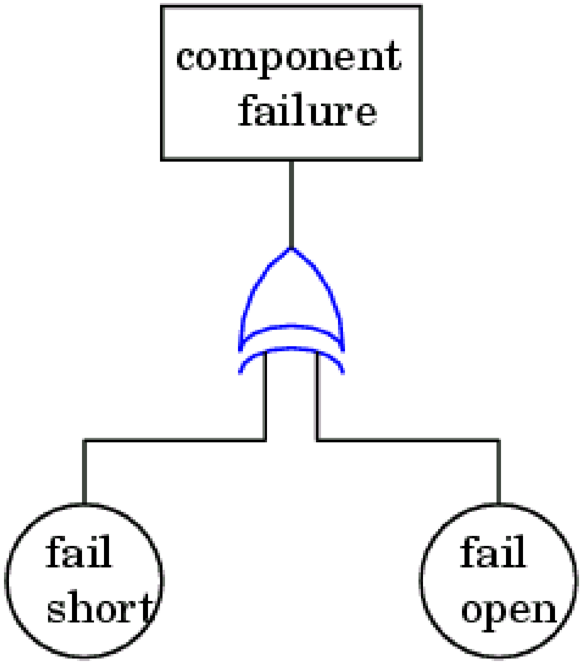
Probabilistic Gates: Coverage

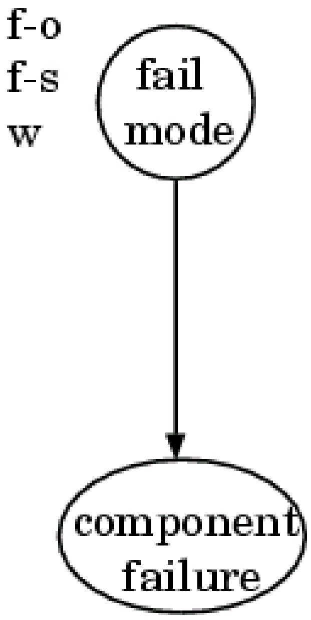


$$
\begin{aligned}
P(\mathsf{PS}=\mathtt{F}|\mathsf{PS}_1=\mathtt{W},\mathsf{PS}_2=\mathtt{W}) &= 0\\
P(\mathsf{PS}=\mathtt{F}|\mathsf{PS}_1=\mathtt{W},\mathsf{PS}_2=\mathtt{F}) &= 1-c\\
P(\mathsf{PS}=\mathtt{F}|\mathsf{PS}_1=\mathtt{F},\mathsf{PS}_2=\mathtt{W}) &= 1-c\\
P(\mathsf{PS}=\mathtt{F}|\mathsf{PS}_1=\mathtt{F},\mathsf{PS}_2=\mathtt{F}) &= 1
\end{aligned}
$$



Multi-State Variables





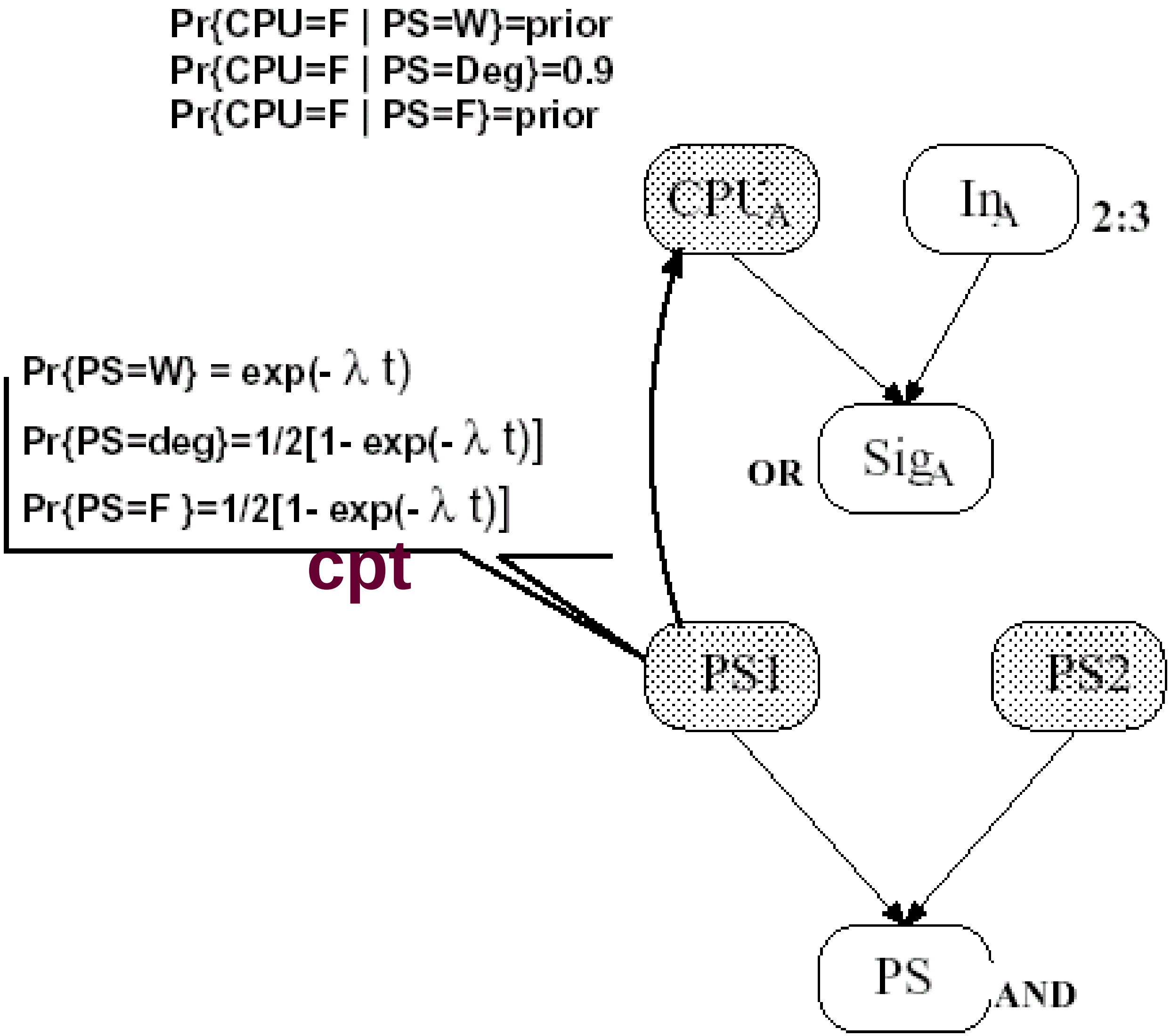
prior

Pr{f-o}=a
Pr{f-s}=b
Pr{w}=1-a-b

Pr{$\overline{C}$ | f-o}=1
Pr{$\overline{C}$ | f-s}=1
Pr{$\overline{C}$ | w }=0

cpt

Sequentially dependent failures



Pr{CPU=F | PS=W}=prior
Pr{CPU=F | PS=Deg}=0.9
Pr{CPU=F | PS=F}=prior

Pr{PS=W} = exp(- λ t)

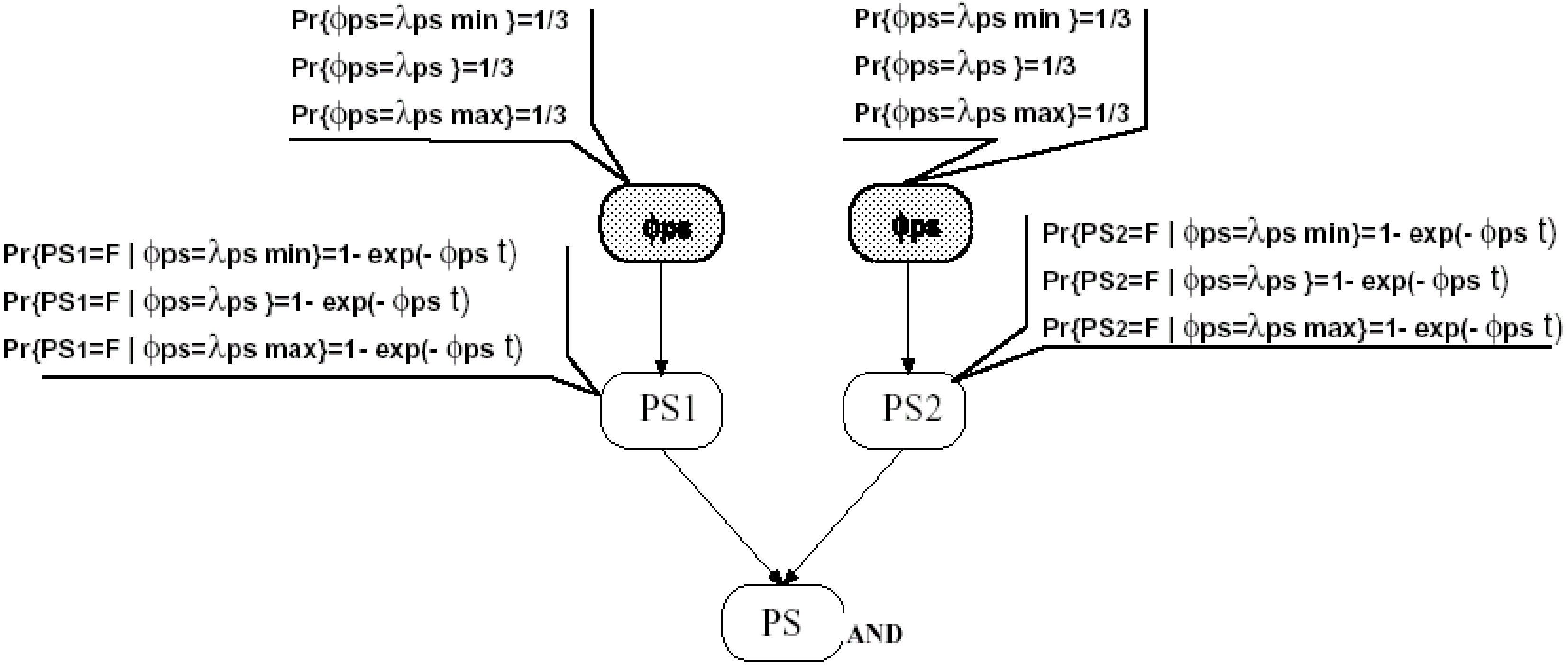
Pr{PS=deg}=1/2[1- exp(- λ t)]

Pr{PS=F }=1/2[1- exp(- λ t)]

cpt

Parameter Uncertainty

λps min= 0.9 λps
λps
λps max=1.1 λps



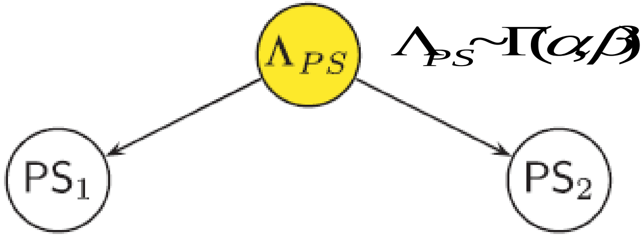
Pr{ϕps=λps min }=1/3
Pr{ϕps=λps }=1/3
Pr{ϕps=λps max}=1/3

Pr{ϕps=λps min }=1/3
Pr{ϕps=λps }=1/3
Pr{ϕps=λps max}=1/3

Pr{PS1=F | ϕps=λps min}=1- exp(- ϕps t)
Pr{PS1=F | ϕps=λps }=1- exp(- ϕps t)
Pr{PS1=F | ϕps=λps max}=1- exp(- ϕps t)

Pr{PS2=F | ϕps=λps min}=1- exp(- ϕps t)
Pr{PS2=F | ϕps=λps }=1- exp(- ϕps t)
Pr{PS2=F | ϕps=λps max}=1- exp(- ϕps t)

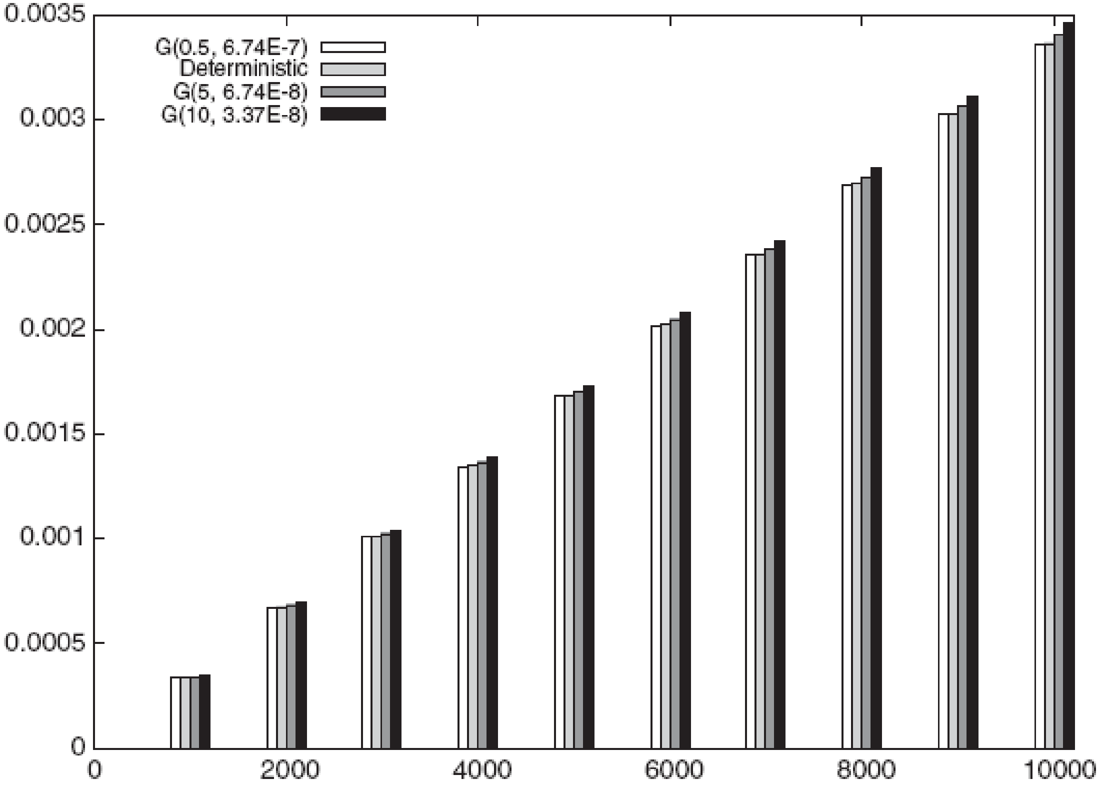
Parameter Ucertainty: example



$\Lambda_{PS} \sim \Gamma(\alpha,\beta)$

$$E[\Lambda_{\mathrm{PS}}] = \alpha \cdot \beta = 3.37\ 10^{-7}$$

$$P(\mathrm{PS}_i = \mathrm{F} \mid \Lambda_{\mathrm{PS}} = \lambda_{\mathrm{PS}}) = 1 - \exp(-\lambda_{\mathrm{PS}} \cdot t)$$



Overview

- Dependability/Reliability issues
- Main Model Types for Reliability
- Probabilistic Graphical Models (BN and DBN)
  - Modeling
  - Computing
- From Dynamic Fault Trees to Dynamic Bayesian Nets
  - Modeling
  - Computing
- Case Studies
- Tools
- Open Issues

DFT analysis

- DFT relaxes assumptions holding for FT
- DFT analysis must capture the system evolution during the time. Solutions:
  - DFT → BDD + CTMC (modular approach)
    - Dynamic module → Continuous Time Markov Chains (CTMC)
      - Univ. of Virginia
    - Dynamic module → (Colored) Stochastic Petri Nets → CTMC
      - Univ. del Piemonte Orientale
  - DFT → algebraic formula including ◁ operator
      - ENS Cachan
  - DFT → I/O Interactive Markov Chains
      - Univ. of Twente
  - **DFT → Dynamic Bayesian Networks (DBN)**

DBN for DFT analysis

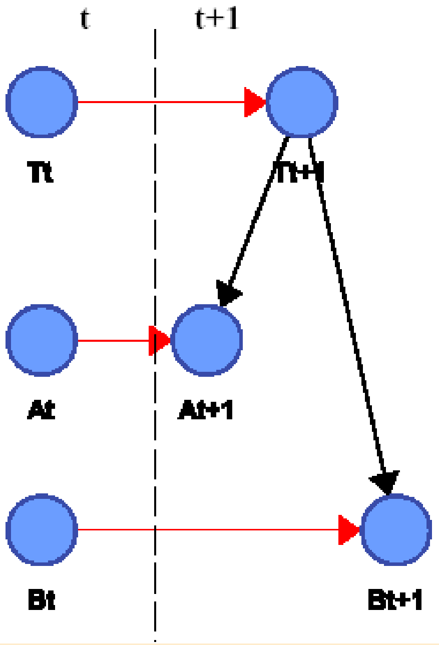
- DBN remove the assumption on binary events
  - Multistate components
- DBN remove the assumption on statistical independence
  - Event dependency
- DBN remove the assumption on Boolean gates (AND, OR)
  - Noisy OR, noisy AND
  - Dynamic gates
- DBN provide a more flexible forward and backward analysis, possibly based on observations
  - Forward (predictive) analysis: Pr(TE), Pr(Sub), Pr(TE|A), Pr(Sub|A)
  - Backward (diagnostic) analysis: Pr(A|TE), Pr(Sub|TE), …
- DBN avoid the state space generation
  - The model does not enumerate all the system states and transitions

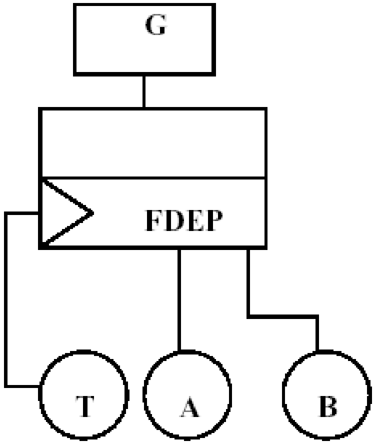
DFT conversion into DBN

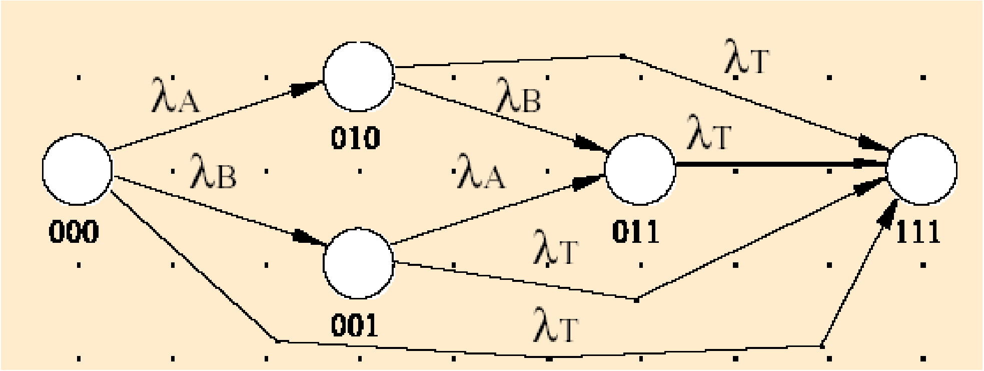
- Modular approach:
  - First, every single gate is converted into DBN
  - Then, the resulting DBNs are connected together in correspondance to the nodes they share.
    - Connection of DBN1 with DBN2
    - An adjustment to the CPT of a node is required when new arcs enter the node:
      - add all the parents derived from DBN1 and DBN2 as columns in the new CPT;
      - in every entry of the table, set the probability of failure of the node using some in teraction rules (Noisy-Or, MSP,…)

- The connection of all the DBNs corresponding to the single gates, provides the DBN expressing the DFT model.

Functional Dependency Gate







$Pr\{T(t+\Delta)=1|T(t)=1\}=1$

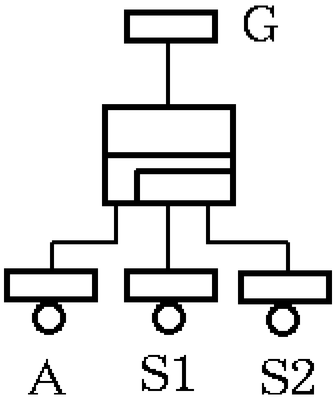
$Pr\{T(t+\Delta)=1|T(t)=0\}=1-e^{-\lambda_T \Delta t}$

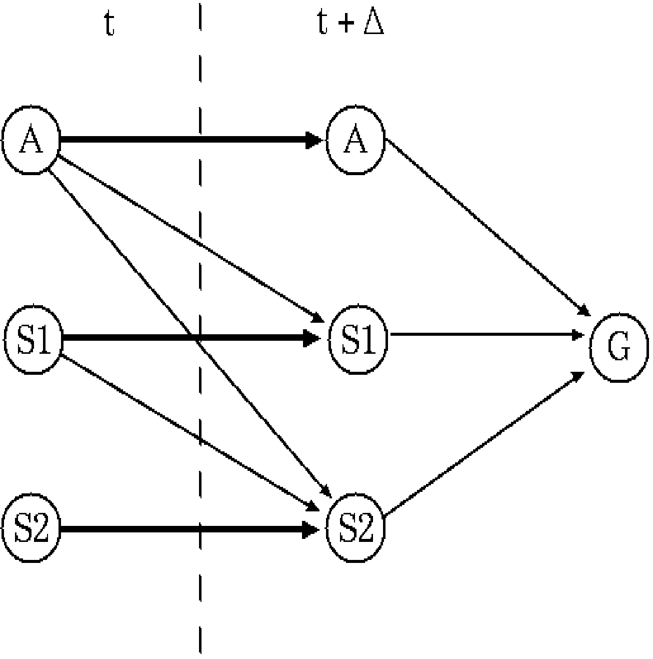
$Pr\{A(t+\Delta)=1|A(t)=1\}=1$

$Pr\{A(t+\Delta)=1|A(t)=0,T(t+\Delta)=0\}=1-e^{-\lambda_A \Delta t}$

$Pr\{A(t+\Delta)=1|A(t)=0,T(t+\Delta)=1\}=1$

Warm Spare Gate





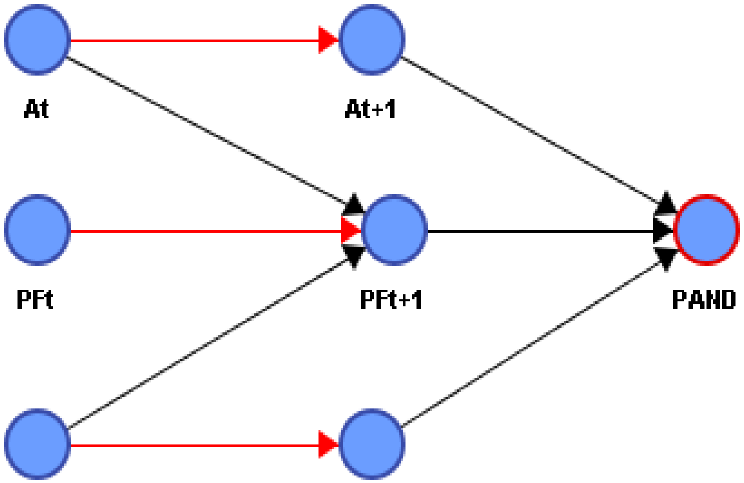
- A is the main component
  - failure rate: λ
- S1, S2 are the warm spare components
  - stand by → αλ   α is the dormancy factor (0<α<1)
  - working → λ

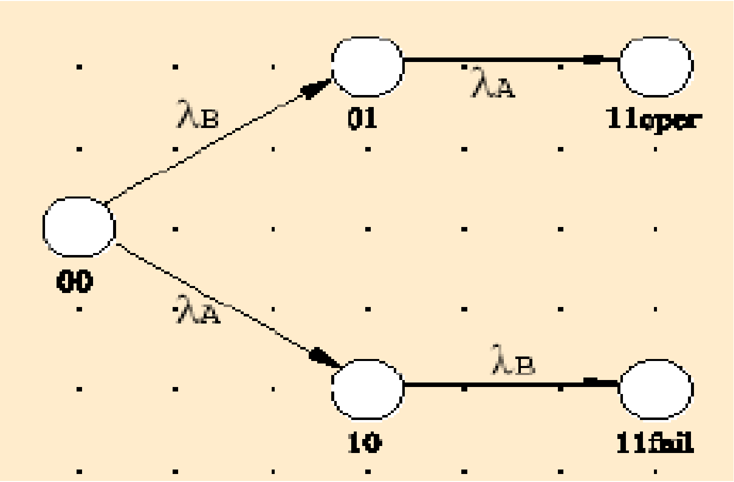
$$Pr\{A(t+\Delta)=1|A(t)=1\}=1$$
$$Pr\{A(t+\Delta)=1|A(t)=0\}=1-e^{-\lambda_A\Delta}$$

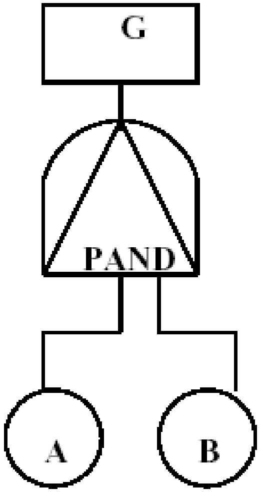
$$Pr\{S1(t+\Delta)=1|S1(t)=1\}=1$$
$$Pr\{S1(t+\Delta)=1|A(t)=0,S1(t)=0\}=1-e^{-\alpha\lambda_{S_1}\Delta}$$
$$Pr\{S1(t+\Delta)=1|A(t)=1,S1(t)=0\}=1-e^{-\lambda_{S_1}\Delta}$$

$$Pr\{S2(t+\Delta)=1|S2(t)=1\}=1$$
$$Pr\{S2(t+\Delta)=1|A(t)=0,S1(t)=0,S2(t)=0\}=1-e^{-\alpha\lambda_{S_2}\Delta}$$
$$Pr\{S2(t+\Delta)=1|A(t)=0,S1(t)=1,S2(t)=0\}=1-e^{-\alpha\lambda_{S_2}\Delta}$$
$$Pr\{S2(t+\Delta)=1|A(t)=1,S1(t)=0,S2(t)=0\}=1-e^{-\alpha\lambda_{S_2}\Delta}$$
$$Pr\{S2(t+\Delta)=1|A(t)=1,S1(t)=1,S2(t)=0\}=1-e^{-\lambda_{S_2}\Delta}$$

Priority AND Gate







$$Pr\{A(t+\Delta)=1|A(t)=1\}=1$$
$$Pr\{A(t+\Delta)=1|A(t)=0\}=1-e^{-\lambda_A \Delta t}$$
$$Pr\{B(t+\Delta)=1|B(t)=1\}=1$$
$$Pr\{B(t+\Delta)=1|B(t)=0\}=1-e^{-\lambda_B \Delta t}$$
$$Pr\{PF(t+\Delta)=1|*,PF(t)=1\}=0$$
$$Pr\{PF(t+\Delta)=1|\ A(t)=0,\ B(t)=0,PF(t)=0\}=0$$
$$Pr\{PF(t+\Delta)=1|\ A(t)=1,\ B(t)=0,PF(t)=0\}=1$$
$$Pr\{PF(t+\Delta)=1|\ A(t)=0,\ B(t)=1,PF(t)=0\}=0$$
$$Pr\{PF(t+\Delta)=1|\ A(t)=1,\ B(t)=1,PF(t)=0\}=1$$

Combining Modules: noisy-or

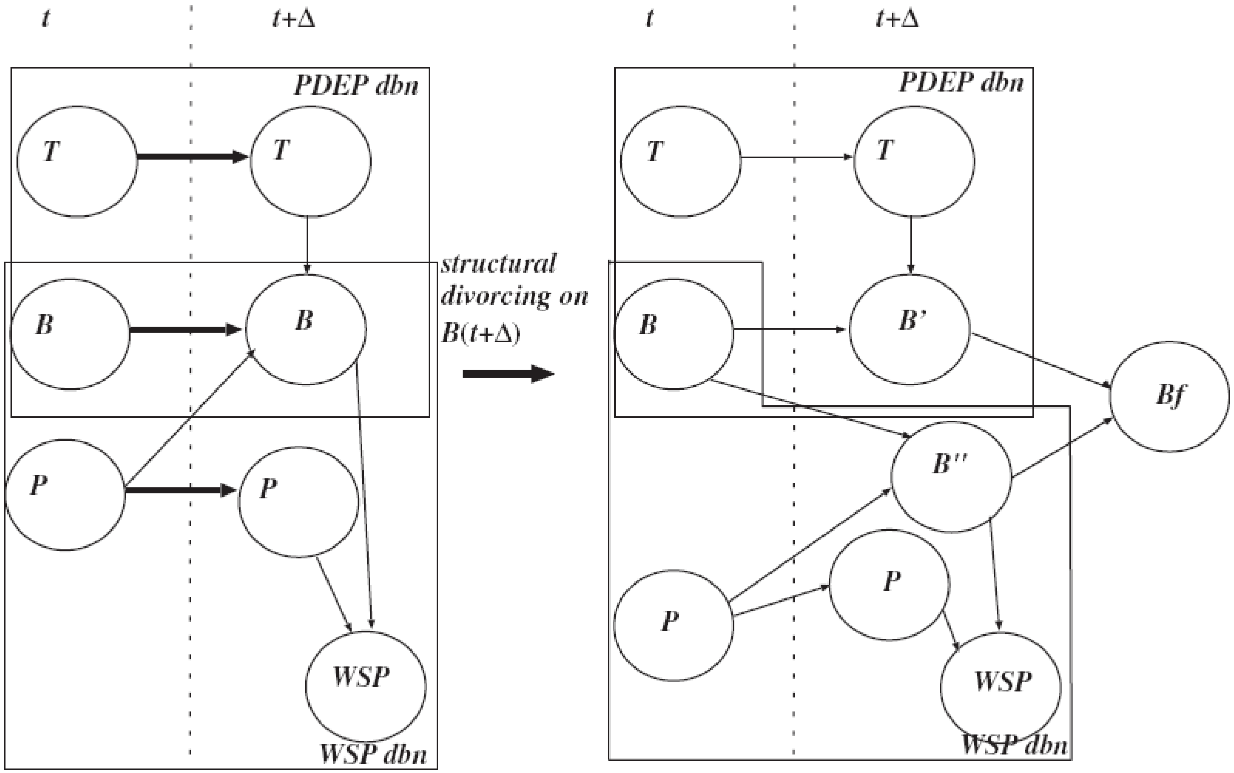
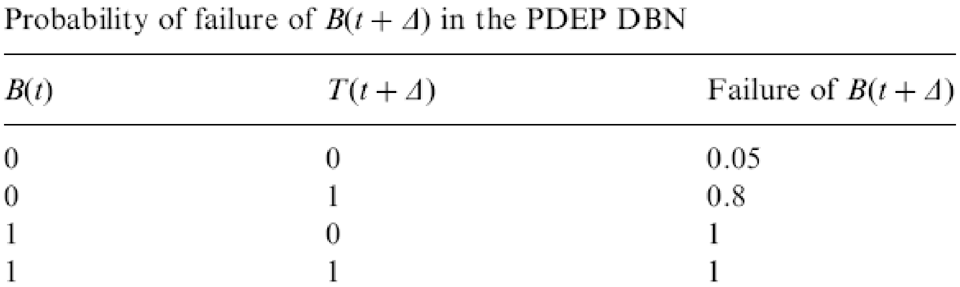


Fig. 6. The DBN for a PDEP triggering the spare of a WSP.

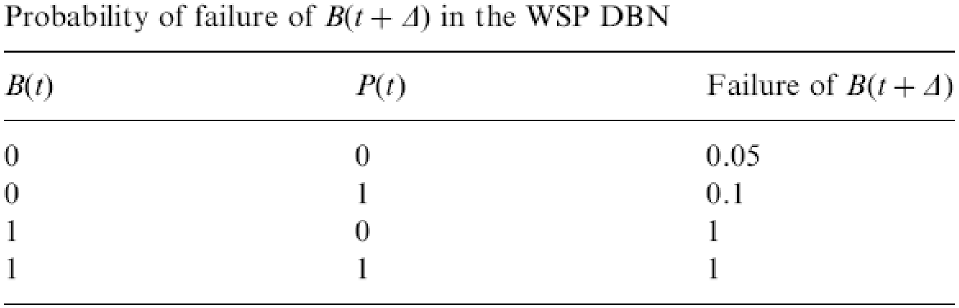
$$
\begin{aligned}
&P[B(t+\Delta)=1 \mid B(t)=0,\, T(t+\Delta)=1,\, P(t)=1]\\
&\quad = P[B_{\mathrm{f}}(t+\Delta)=1 \mid B'(t+\Delta)=\text{“01”},\, B''(t+\Delta)=\text{“01”}]\\
&\quad = 1-((1-p_{\mathrm{d}})(1-\lambda)) = 1-0.2\,0.9 = 0.82. \qquad (1)
\end{aligned}
$$

Combining Modules: MSP



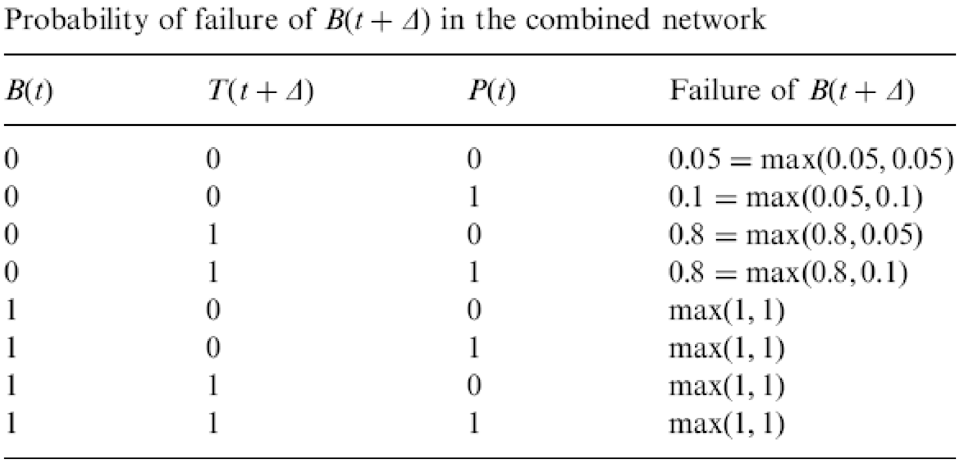
Probability of failure of $B(t+\Delta)$ in the PDEP DBN

$B(t)$	$T(t+\Delta)$	Failure of $B(t+\Delta)$
0	0	0.05
0	1	0.8
1	0	1
1	1	1



Probability of failure of $B(t+\Delta)$ in the WSP DBN

$B(t)$	$P(t)$	Failure of $B(t+\Delta)$
0	0	0.05
0	1	0.1
1	0	1
1	1	1



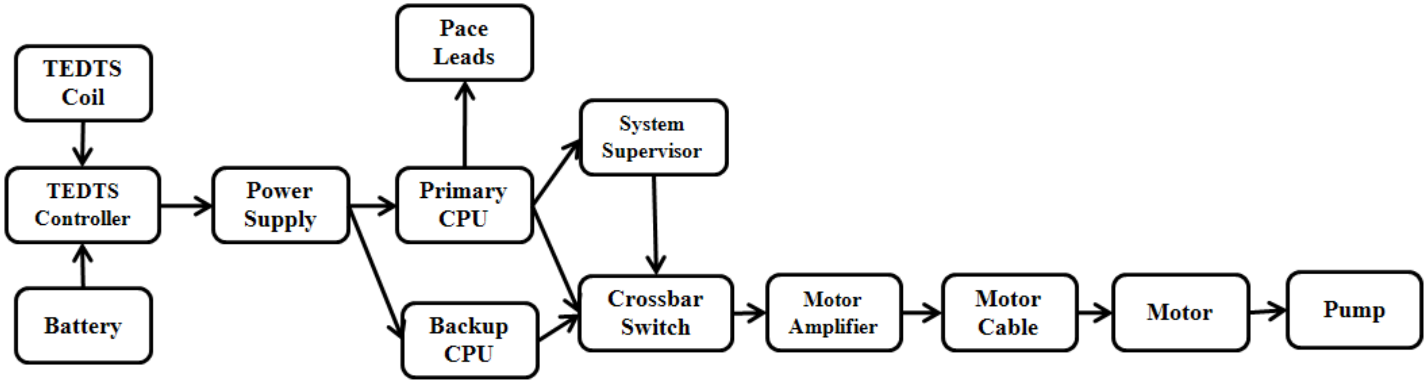
Probability of failure of $B(t+\Delta)$ in the combined network

$B(t)$	$T(t+\Delta)$	$P(t)$	Failure of $B(t+\Delta)$
0	0	0	$0.05 = \max(0.05, 0.05)$
0	0	1	$0.1 = \max(0.05, 0.1)$
0	1	0	$0.8 = \max(0.8, 0.05)$
0	1	1	$0.8 = \max(0.8, 0.1)$
1	0	0	$\max(1, 1)$
1	0	1	$\max(1, 1)$
1	1	0	$\max(1, 1)$
1	1	1	$\max(1, 1)$

Overview

- Dependability/Reliability issues
- Main Model Types for Reliability
- Probabilistic Graphical Models (BN and DBN)
  - ❑ Modeling
  - ❑ Computing
- From (Dynamic) Fault Trees to (Dynamic) Bayesian Nets
  - ❑ Modeling
  - ❑ Computing
- Case Studies
- Tools
- Open Issues

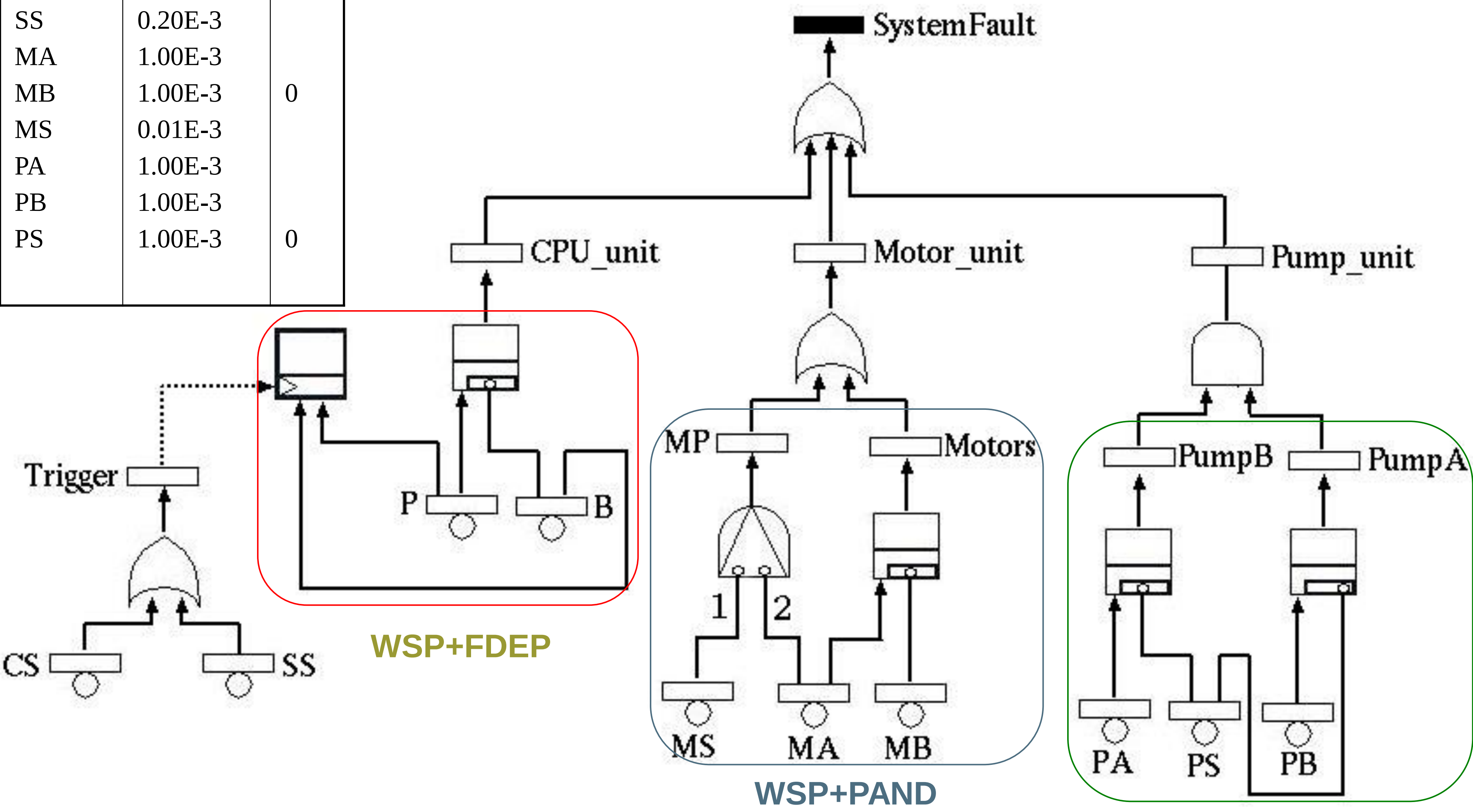
Cardiac Assist System (Dugan et al)



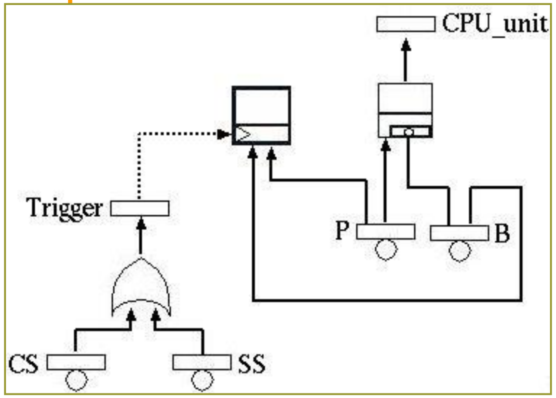
- The failure of either one of the modules causes the whole system failure:
  - The CPU module consists of the primary cpu P and a warm spare B:
    - Both P and B are functionally dependent on a cross switch CS and a system supervision SS
    - Both P and B are considered as repairable
  - The Motor module consists of the primary motor MA and a cold spare MB:
    - MB turns into operation when the MA fails, because of a motor switching component MS
      - if MS fails before MA, then the spare cannot become operational
  - The Pump module is composed by two primary pumps PA and PB running in parallel and a cold spare PS

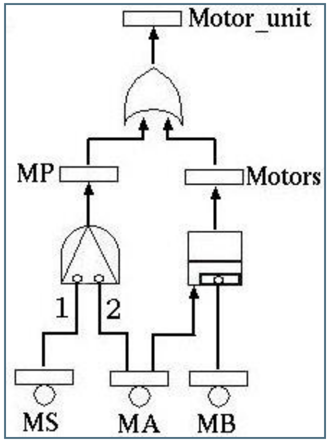
Comp.	λ (1/h)	α
P	0.50E-3	
B	0.50E-3	0.5
CS	0.20E-3	
SS	0.20E-3	
MA	1.00E-3	
MB	1.00E-3	0
MS	0.01E-3	
PA	1.00E-3	
PB	1.00E-3	
PS	1.00E-3	0

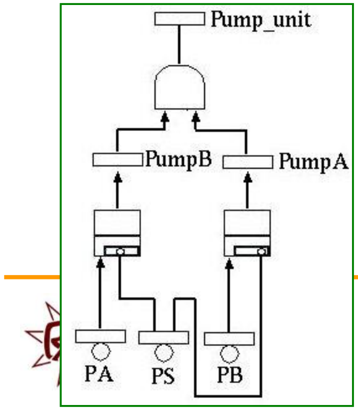
DFT of the case study CAS

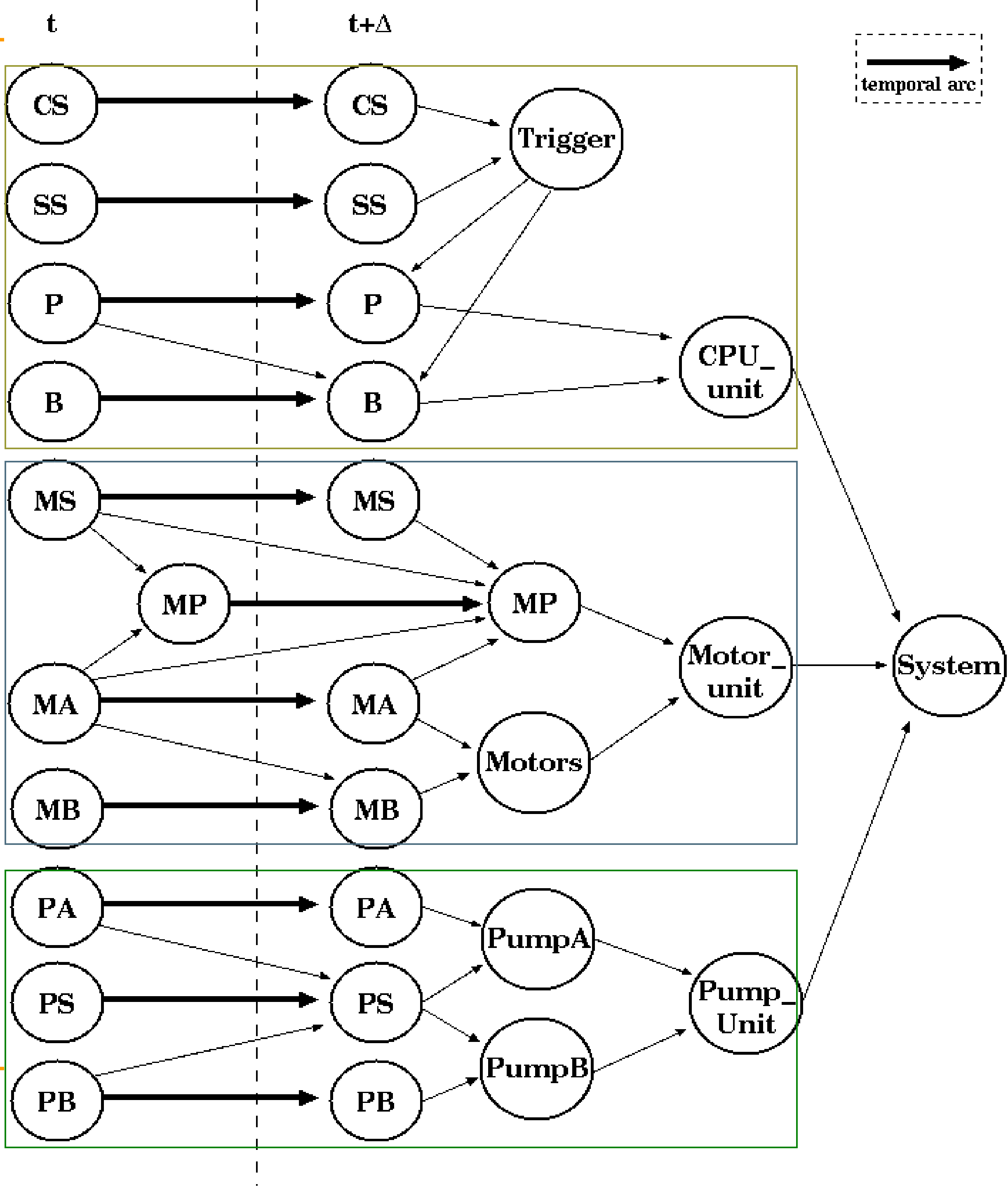


DBN→DFT

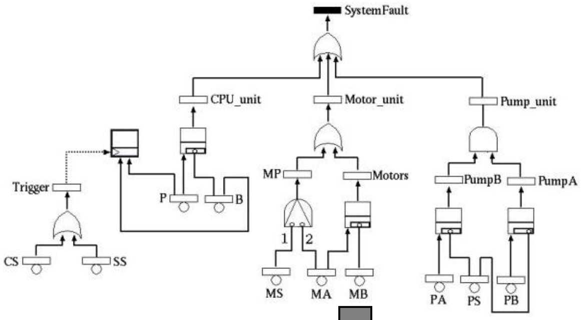

CPU_unit
Trigger
P
B
CS
SS

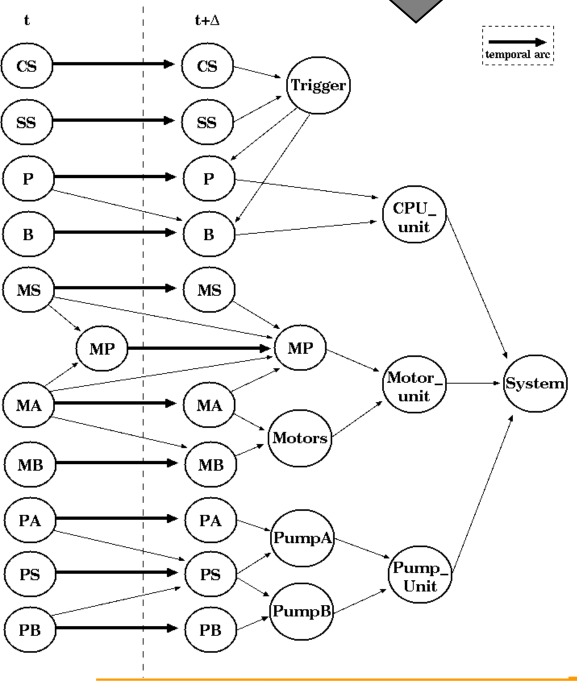

Motor_unit
MP
Motors
1
2
MS
MA
MB


Pump_unit
PumpB
PumpA
PA
PS
PB

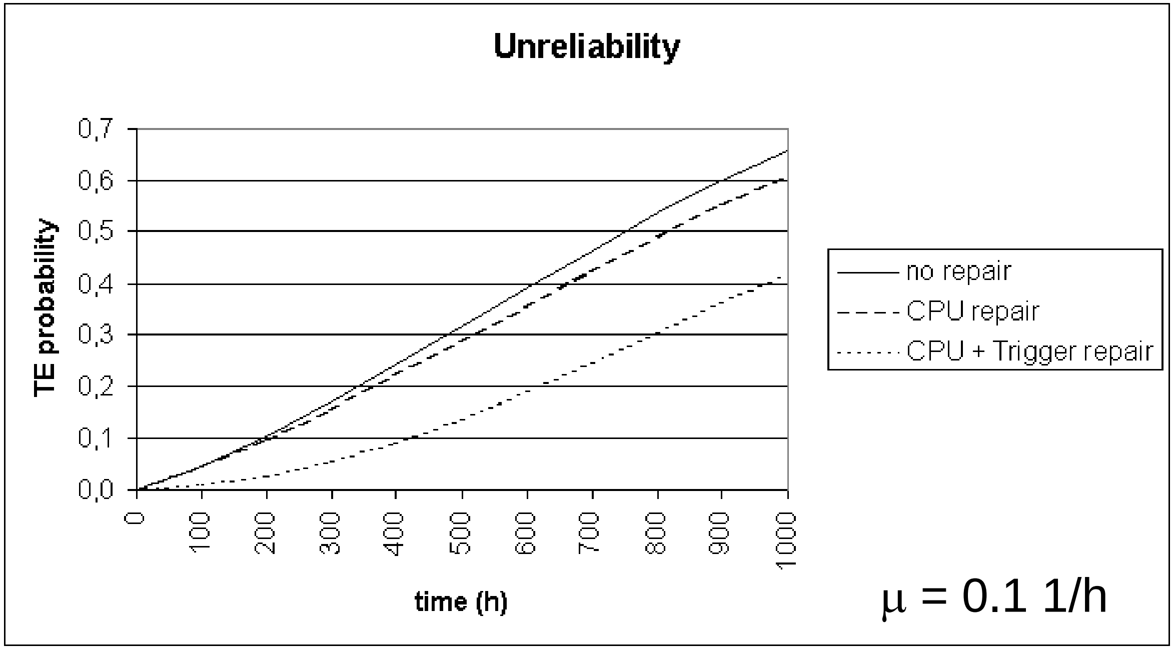

t
t+Δ
temporal arc
CS
CS
SS
SS
Trigger
P
P
B
B
CPU_unit
MS
MS
MP
MP
MA
MA
Motor_unit
Motors
MB
MB
System
PA
PA
PumpA
PS
PS
Pump_Unit
PB
PB
PumpB

Inference results



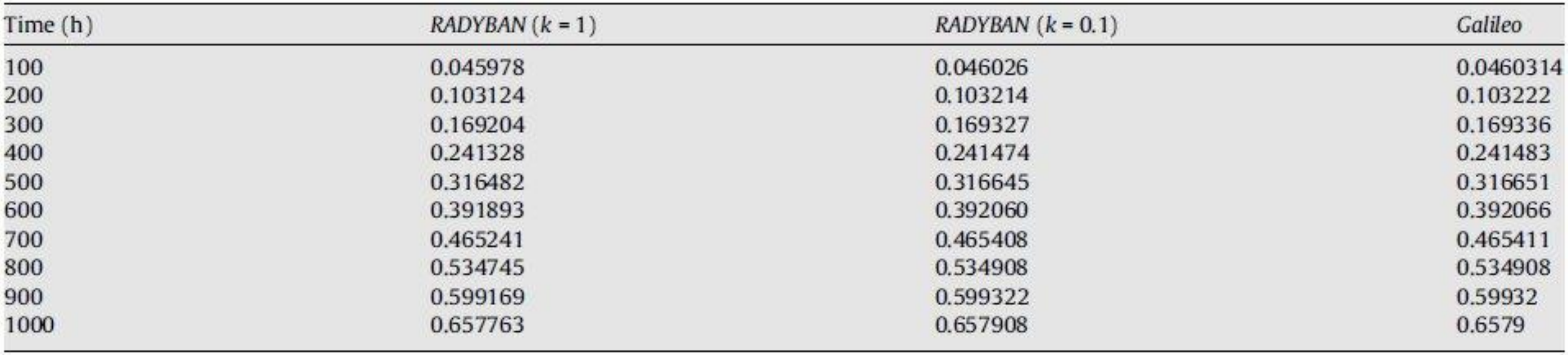


Case of filtering
with no observations

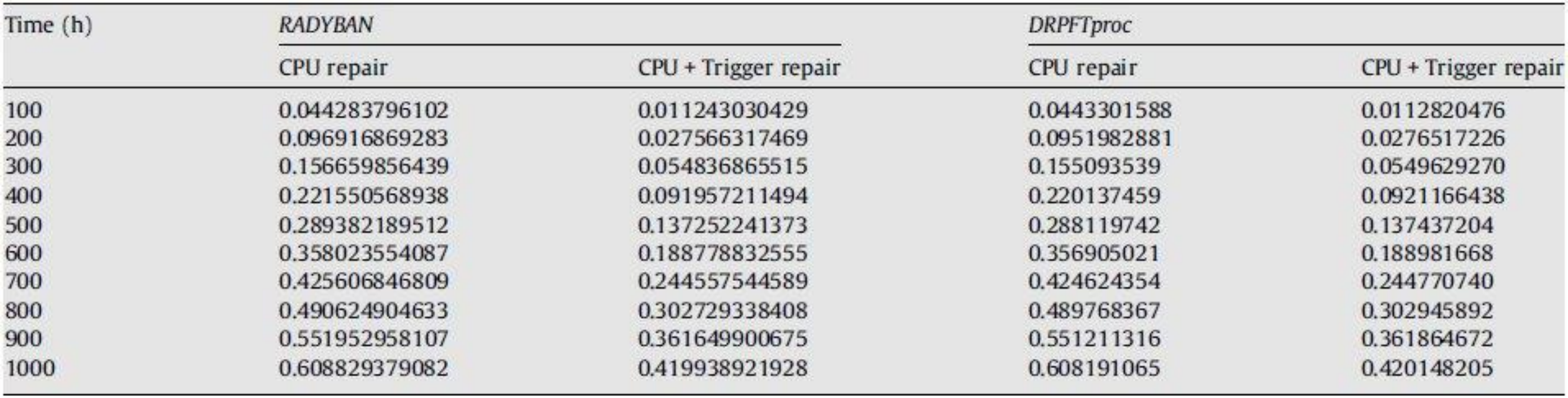


μ = 0.1 1/h

Results comparison

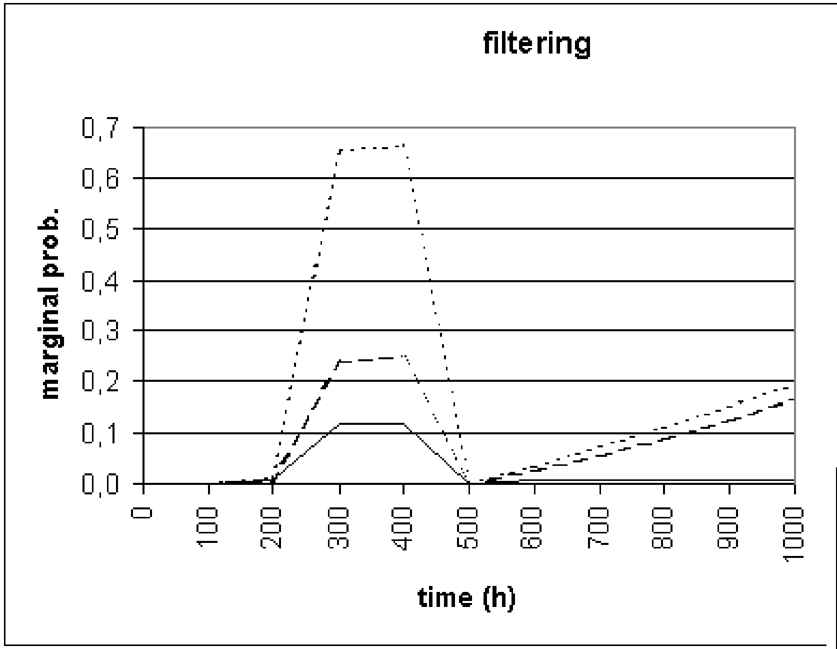


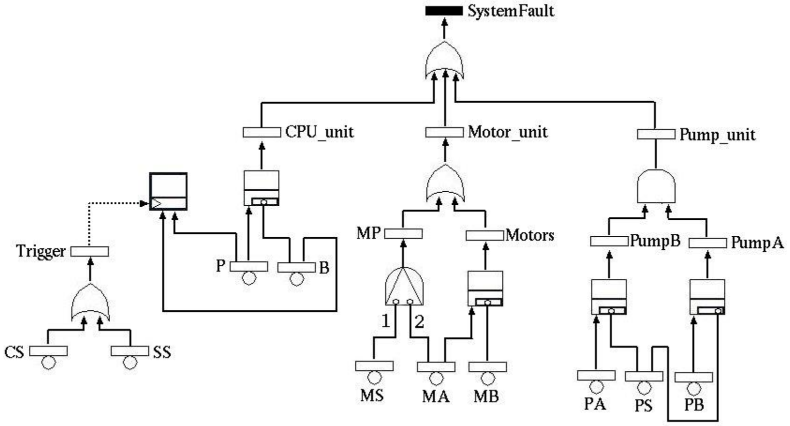
Time (h)	*RADYBAN* (*k* = 1)	*RADYBAN* (*k* = 0.1)	*Galileo*
100	0.045978	0.046026	0.0460314
200	0.103124	0.103214	0.103222
300	0.169204	0.169327	0.169336
400	0.241328	0.241474	0.241483
500	0.316482	0.316645	0.316651
600	0.391893	0.392060	0.392066
700	0.465241	0.465408	0.465411
800	0.534745	0.534908	0.534908
900	0.599169	0.599322	0.59932
1000	0.657763	0.657908	0.6579

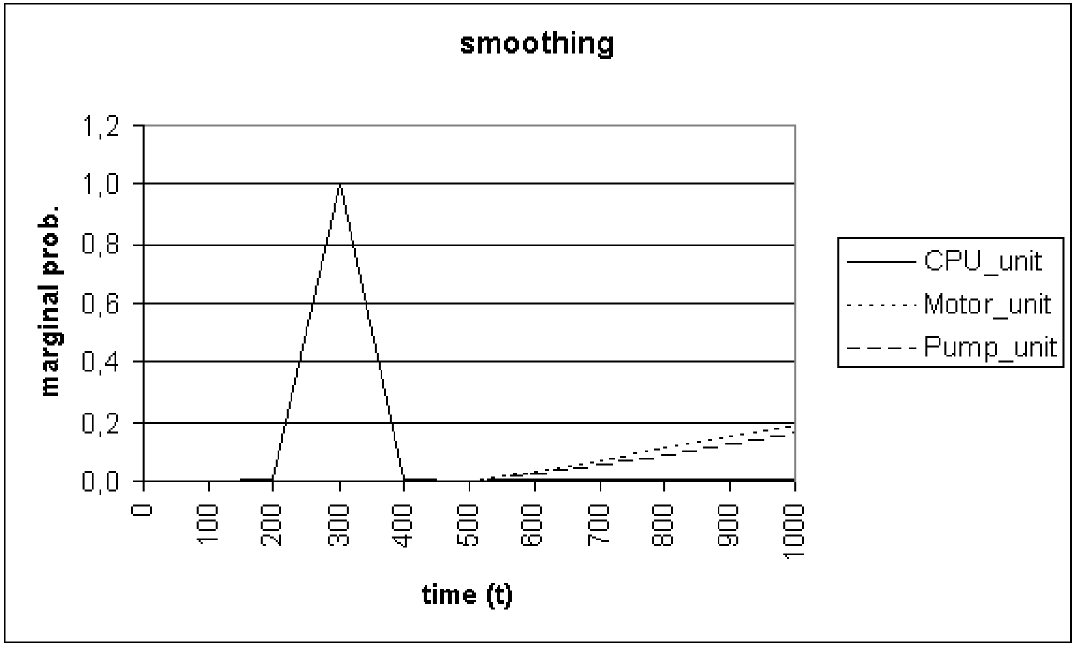


Time (h)	*RADYBAN*		*DRPFTproc*	
	CPU repair	CPU + Trigger repair	CPU repair	CPU + Trigger repair
100	0.044283796102	0.011243030429	0.0443301588	0.0112820476
200	0.096916869283	0.027566317469	0.0951982881	0.0276517226
300	0.156659856439	0.054836865515	0.155093539	0.0549629270
400	0.221550568938	0.091957211494	0.220137459	0.0921166438
500	0.289382189512	0.137252241373	0.288119742	0.137437204
600	0.358023554087	0.188778832555	0.356905021	0.188981668
700	0.425606846809	0.244557544589	0.424624354	0.244770740
800	0.490624904633	0.302729338408	0.489768367	0.302945892
900	0.551952958107	0.361649900675	0.551211316	0.361864672
1000	0.608829379082	0.419938921928	0.608191065	0.420148205

Inference with observations

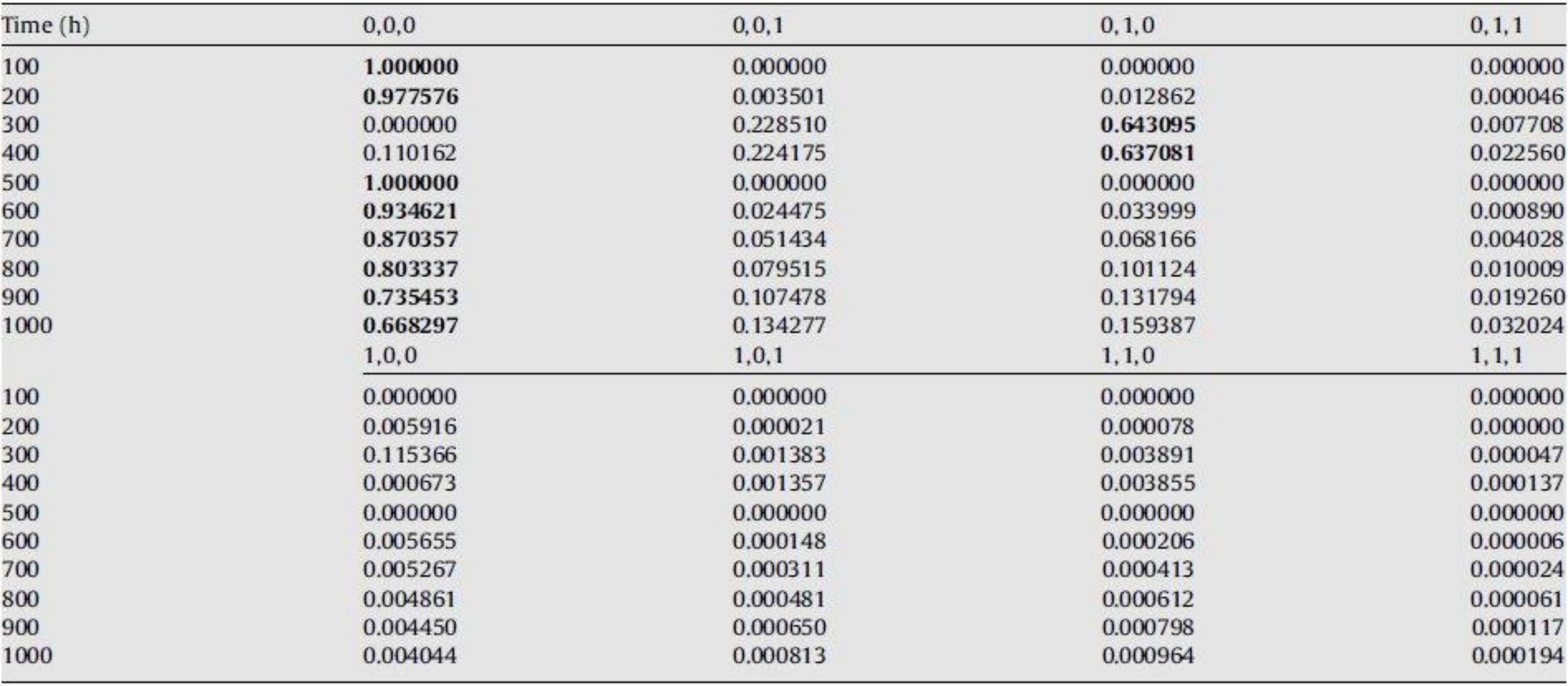






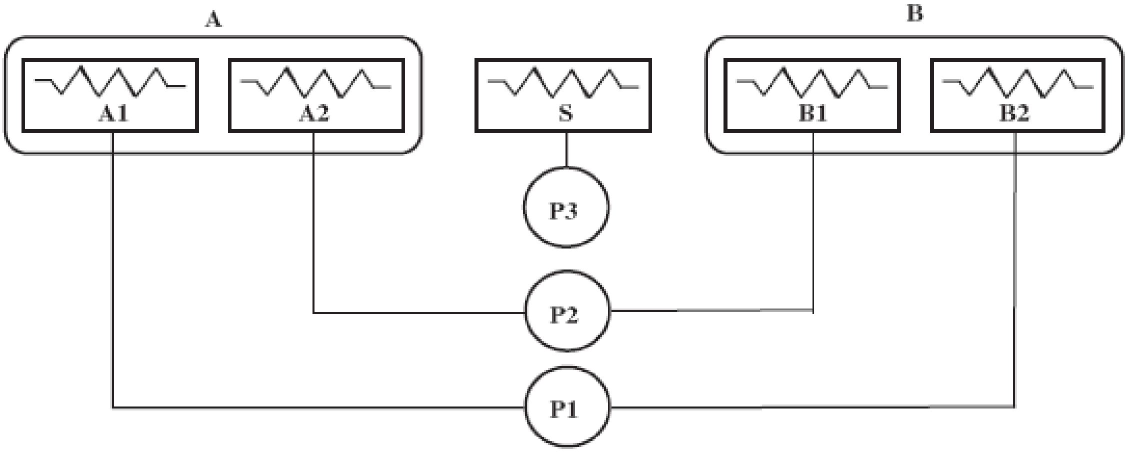
- P, B, CS, SS are repairable
- The system was observed
  - operational at t1=100 h
  - failed at t2=300 h
  - operational t3=500 h

Joint probabilities assuming observations



Time (h)	0,0,0	0,0,1	0,1,0	0,1,1
100	**1.000000**	0.000000	0.000000	0.000000
200	**0.977576**	0.003501	0.012862	0.000046
300	0.000000	0.228510	**0.643095**	0.007708
400	0.110162	0.224175	**0.637081**	0.022560
500	**1.000000**	0.000000	0.000000	0.000000
600	**0.934621**	0.024475	0.033999	0.000890
700	**0.870357**	0.051434	0.068166	0.004028
800	**0.803337**	0.079515	0.101124	0.010009
900	**0.735453**	0.107478	0.131794	0.019260
1000	**0.668297**	0.134277	0.159387	0.032024
	1,0,0	1,0,1	1,1,0	1,1,1
100	0.000000	0.000000	0.000000	0.000000
200	0.005916	0.000021	0.000078	0.000000
300	0.115366	0.001383	0.003891	0.000047
400	0.000673	0.001357	0.003855	0.000137
500	0.000000	0.000000	0.000000	0.000000
600	0.005655	0.000148	0.000206	0.000006
700	0.005267	0.000311	0.000413	0.000024
800	0.004861	0.000481	0.000612	0.000061
900	0.004450	0.000650	0.000798	0.000117
1000	0.004044	0.000813	0.000964	0.000194

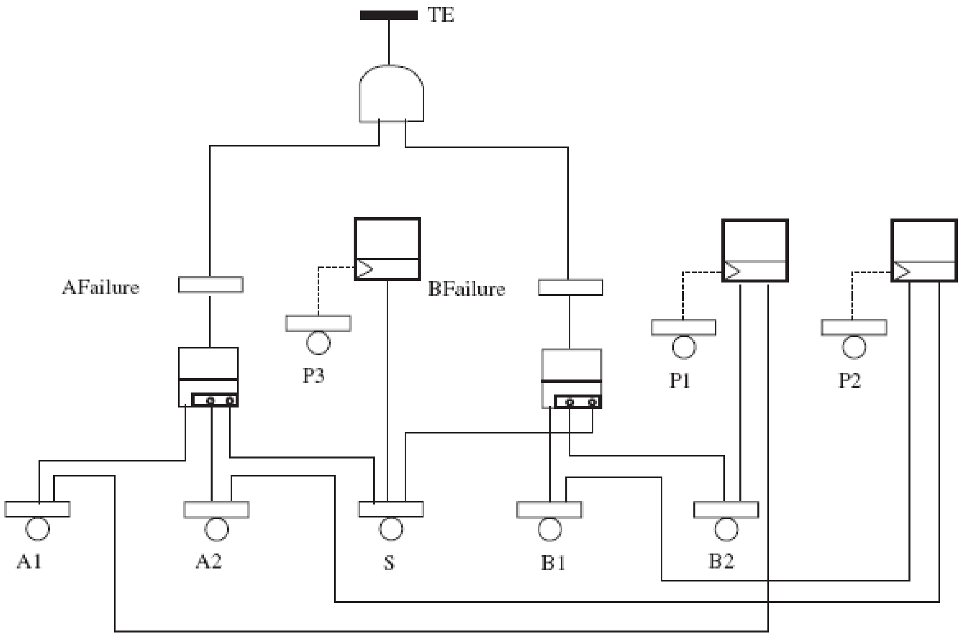
Active Heat Rejection System (Boudali-Dugan)

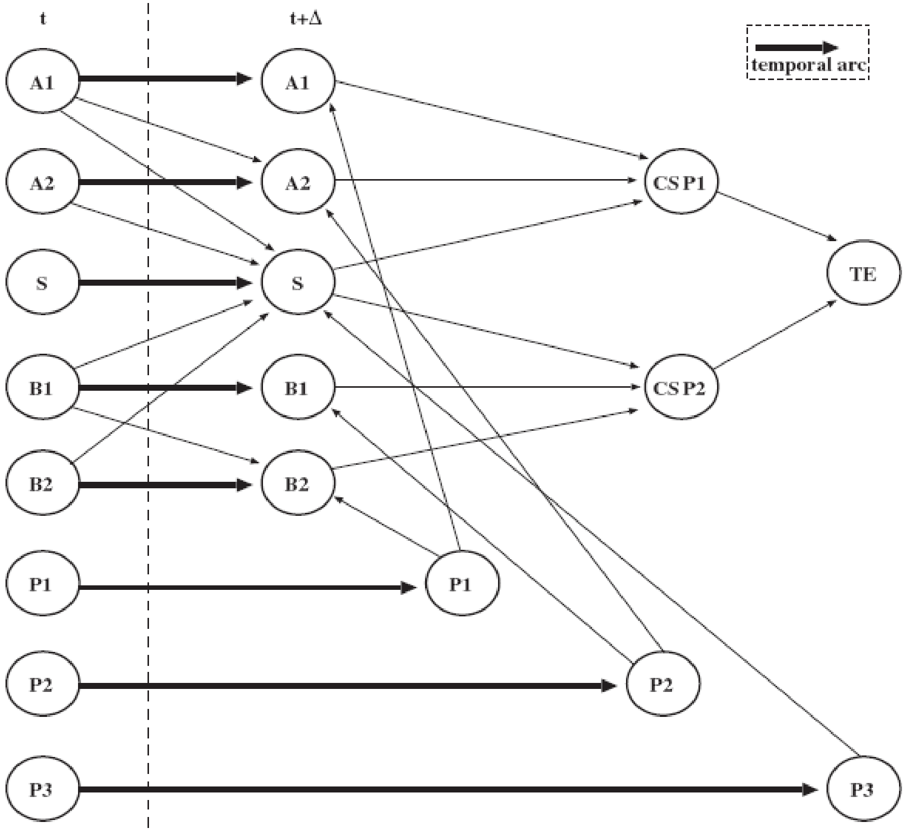


The failure rates in the AHRS example

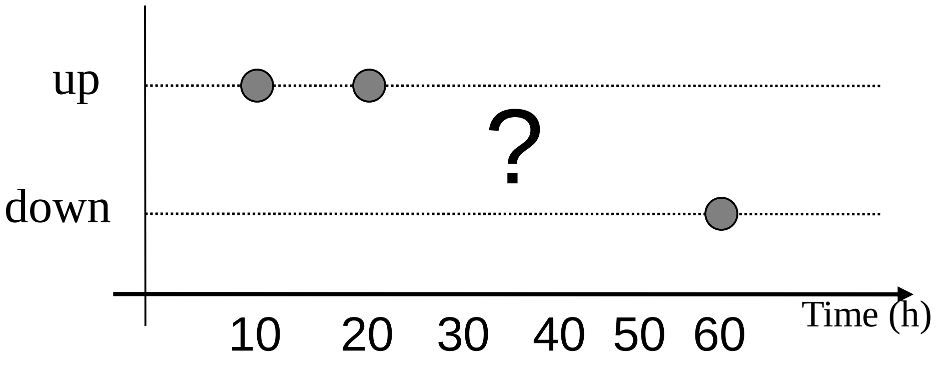
Component	Failure rate (λ) (h^{-1})
*A*1	0.001
*A*2	0.005
*B*1	0.002
*B*2	0.0035
S	0.005
*P*1, *P*2, *P*3	0.003

AHRS: the DFT and the DBN

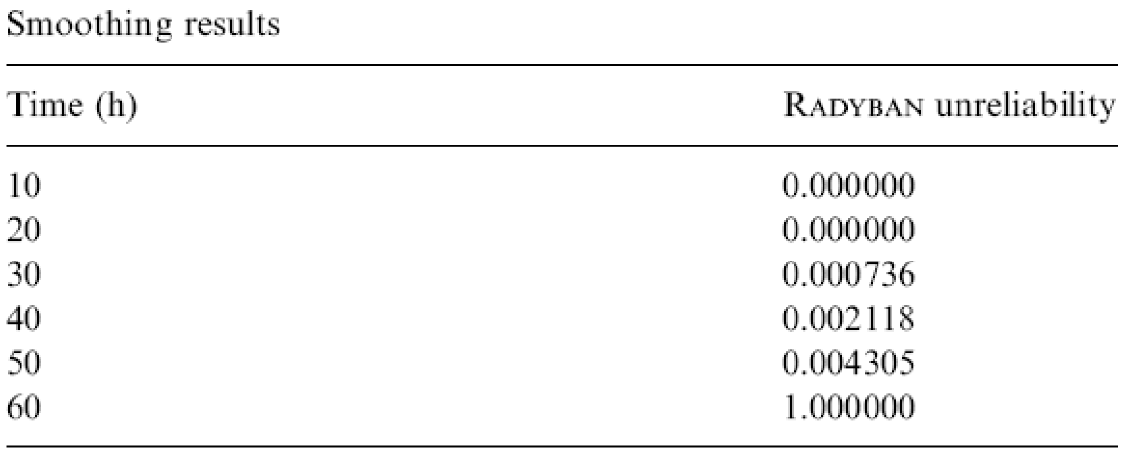




AHRS: smoothing results



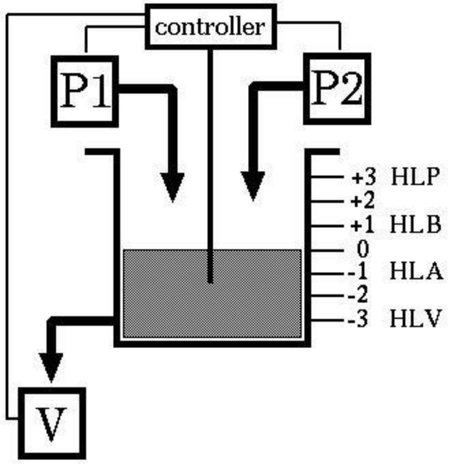
Observation stream
on system status (TE)

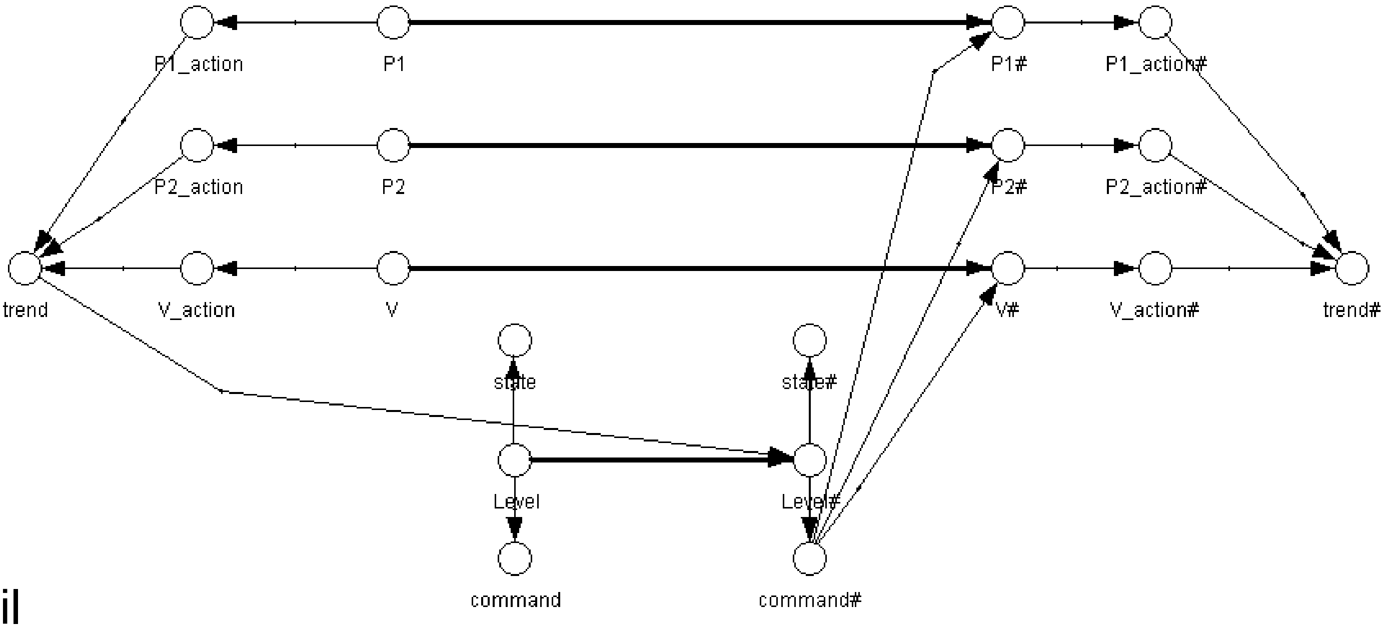


Smoothing results

Time (h)	RADYBAN unreliability
10	0.000000
20	0.000000
30	0.000736
40	0.002118
50	0.004305
60	1.000000

DBN model and analysis of a benchmark





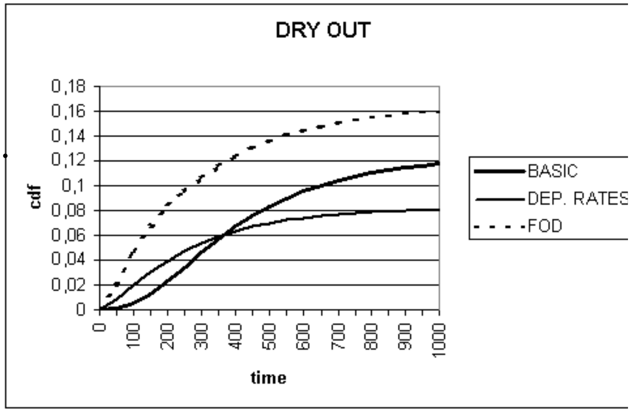
Each component may fail

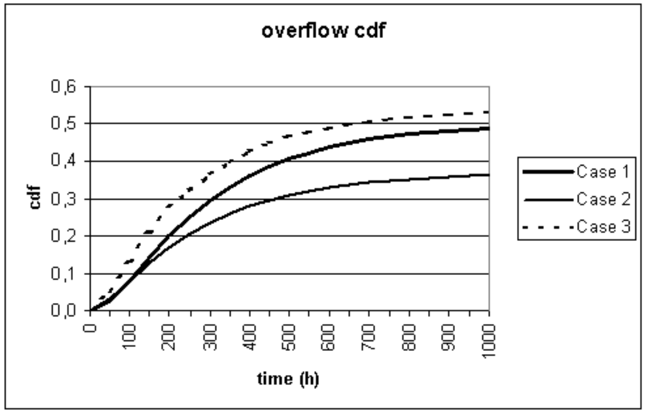
Control laws:

- H ≤ HLA ⇒ P1:ON, P2:ON, V:OFF.
- H ≥ HLB ⇒ P1:OFF, P2:OFF, V:ON.

Failure conditions:

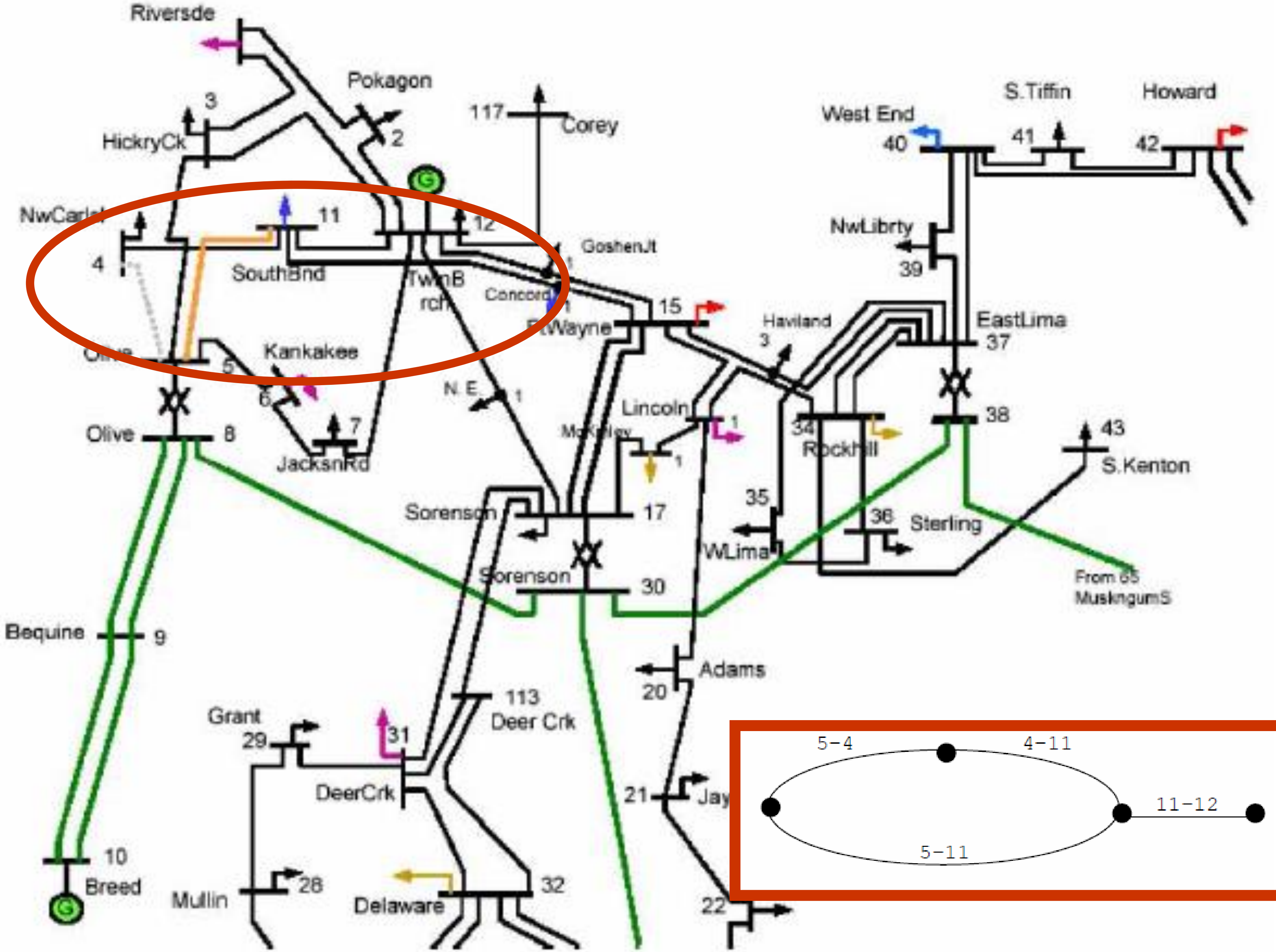
- Dry out (H<HLV)
- Overflow (H>HLP)





Cascading failures

- Interdependencies among complex system(s) components increase the risk of failures
- Cascading failures:
  - Failure in one component causes an overload in adjacent components, increasing their failure probability
  - If not compensated, the cascading overload/failure can cause a progressive disruption of the system
  - E.g. recent occurrence of large scale electrical blackouts


Riversde
Pokagon
2
3
HickryCk
117
Corey
West End
S.Tiffin
Howard
40
41
42
NwCarlsl
4
11
SouthBnd
12
TwinBrch
GoshenJt
1
Concord
1
FtWayne
15
NwLibrty
39
Haviland
3
EastLima
37
Olive
5
Kankakee
6
N. E.
1
Lincoln
1
38
43
S.Kenton
Olive
8
7
JacksnRd
McKinley
1
34
Rockhill
Sorenson
17
35
36
Sterling
WLima
Sorenson
30
From 65
MuskngumS
Bequine
9
Adams
20
113
Deer Crk
Grant
29
31
DeerCrk
21
Jay
10
Breed
Mullin
28
Delaware
32
22
5-4
4-11
11-12
5-11

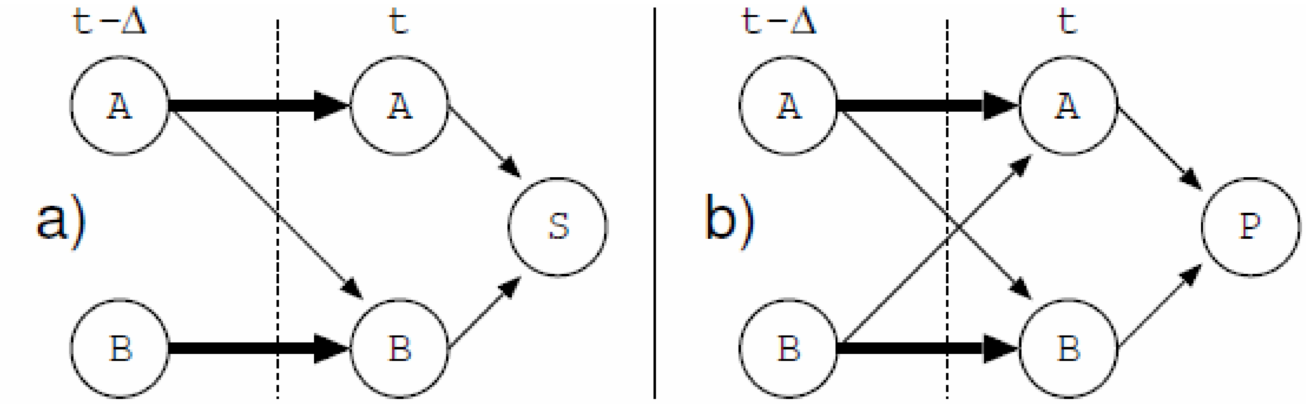
Issues and assumptions

- Line states
  - Working, outaged, overloaded
    - 3-state variables
  - Overload introduces temporal dependencies
- Outage probability
  - Negative exponential distribution
  - Working line: failure rate $\lambda=0.0001h^{-1}$
  - Overloaded line: increased failure rate $\alpha\lambda$ ($\alpha=1.2$), $\beta\lambda$ ($\beta=1.5$)

Methodology

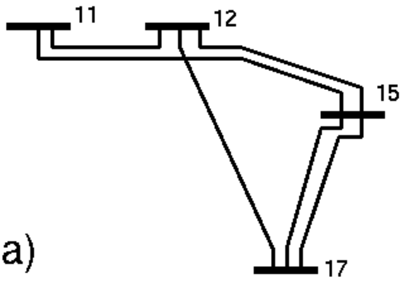
- Automatic conversion of the series/parallel diagram into a DBN
- Modular composition of
  - Series modules
  - Parallel modules
  - Generalization of OR and AND nodes, working with multi-state variables

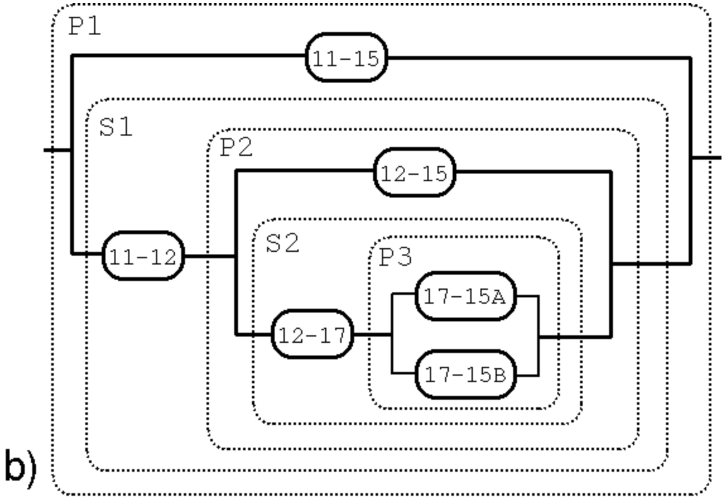
Basic modules

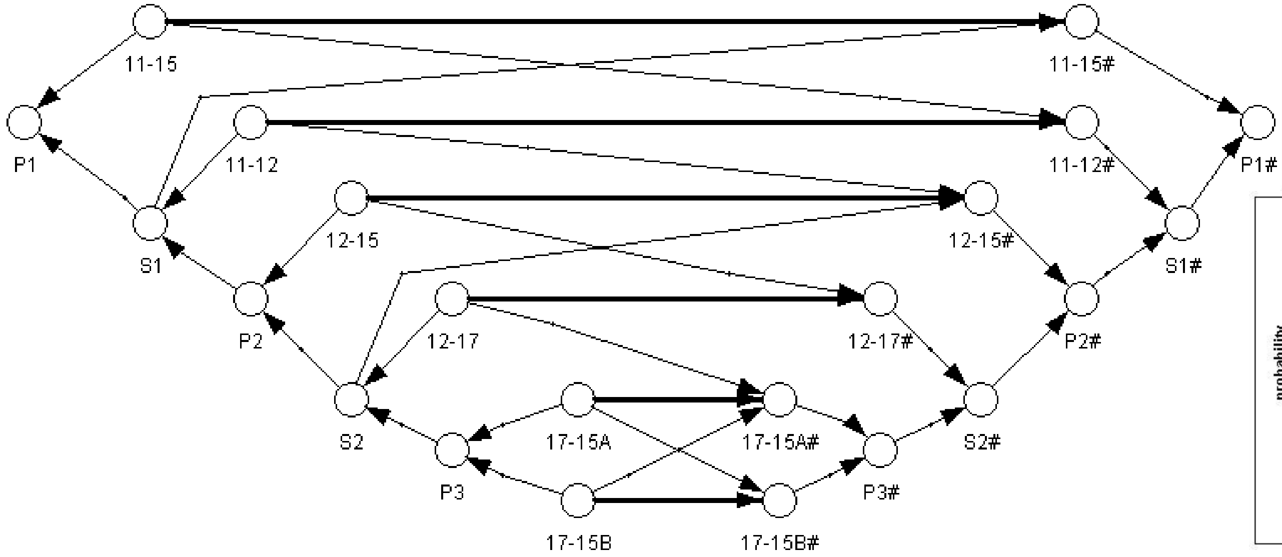


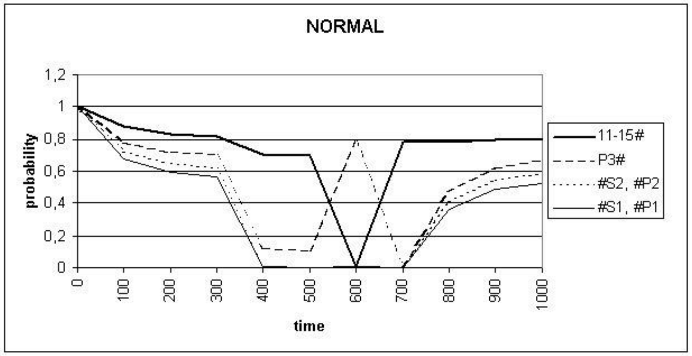
- Series:
  - if A is overloaded, S gets overloaded (B cannot be overloaded)
  - If A or B fails, S fails
- Parallel:
  - if A or B is overloaded, P gets overloaded
  - Is A and B fail, P fails
  - if only A or only B fails, P works properly

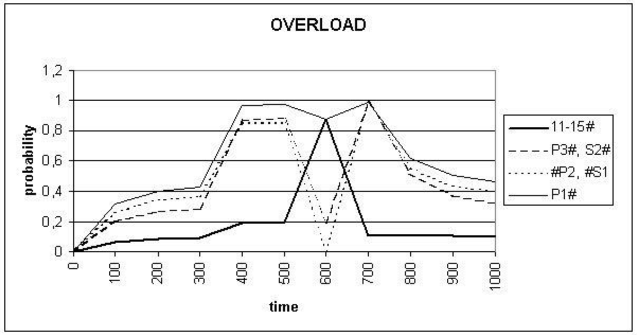
DBN model of cascading effects in a power grid

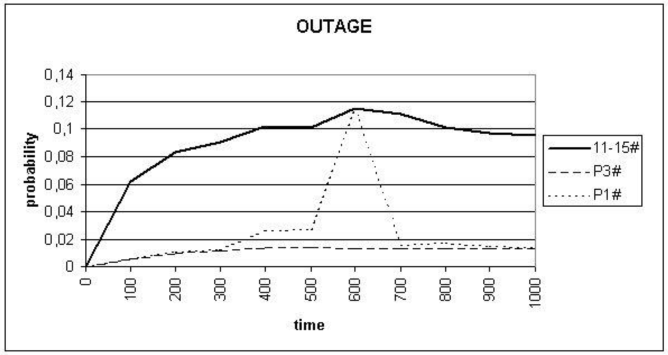








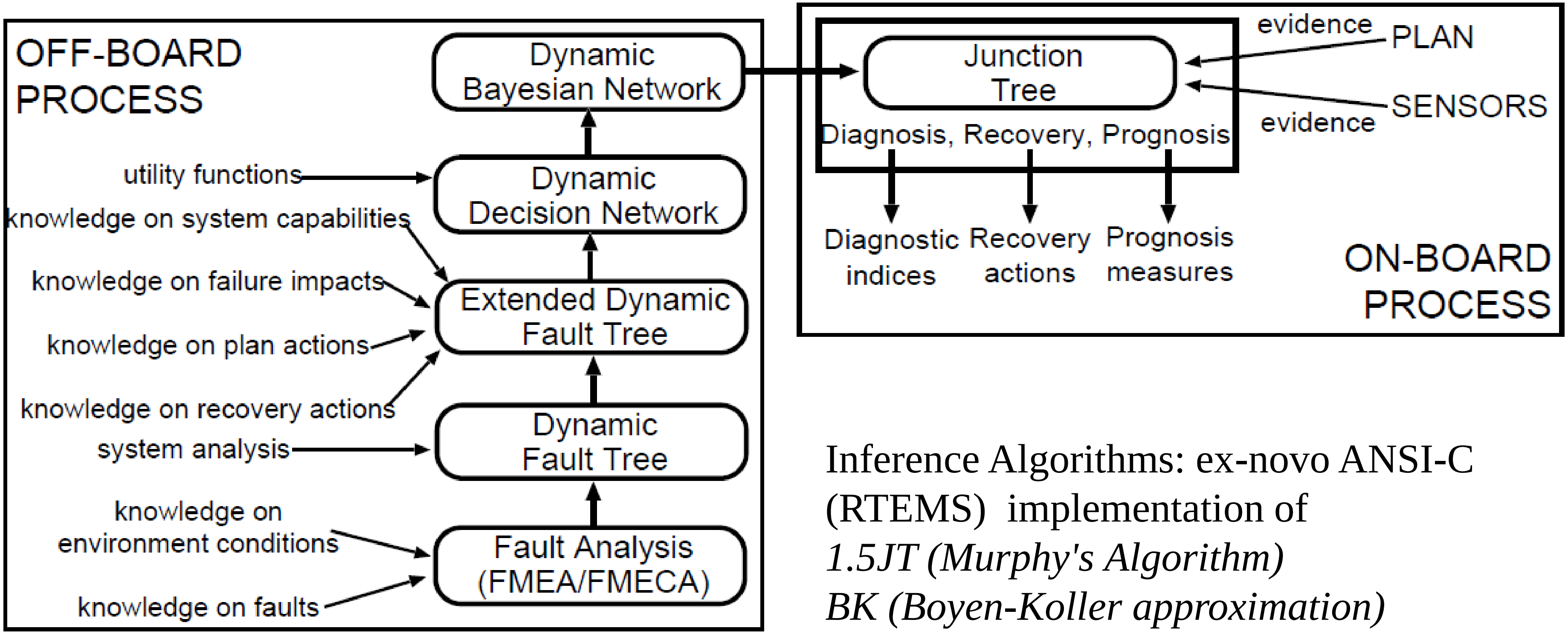




ARPHA: Anomaly Resolution for Prognostic Health management for Autonomy

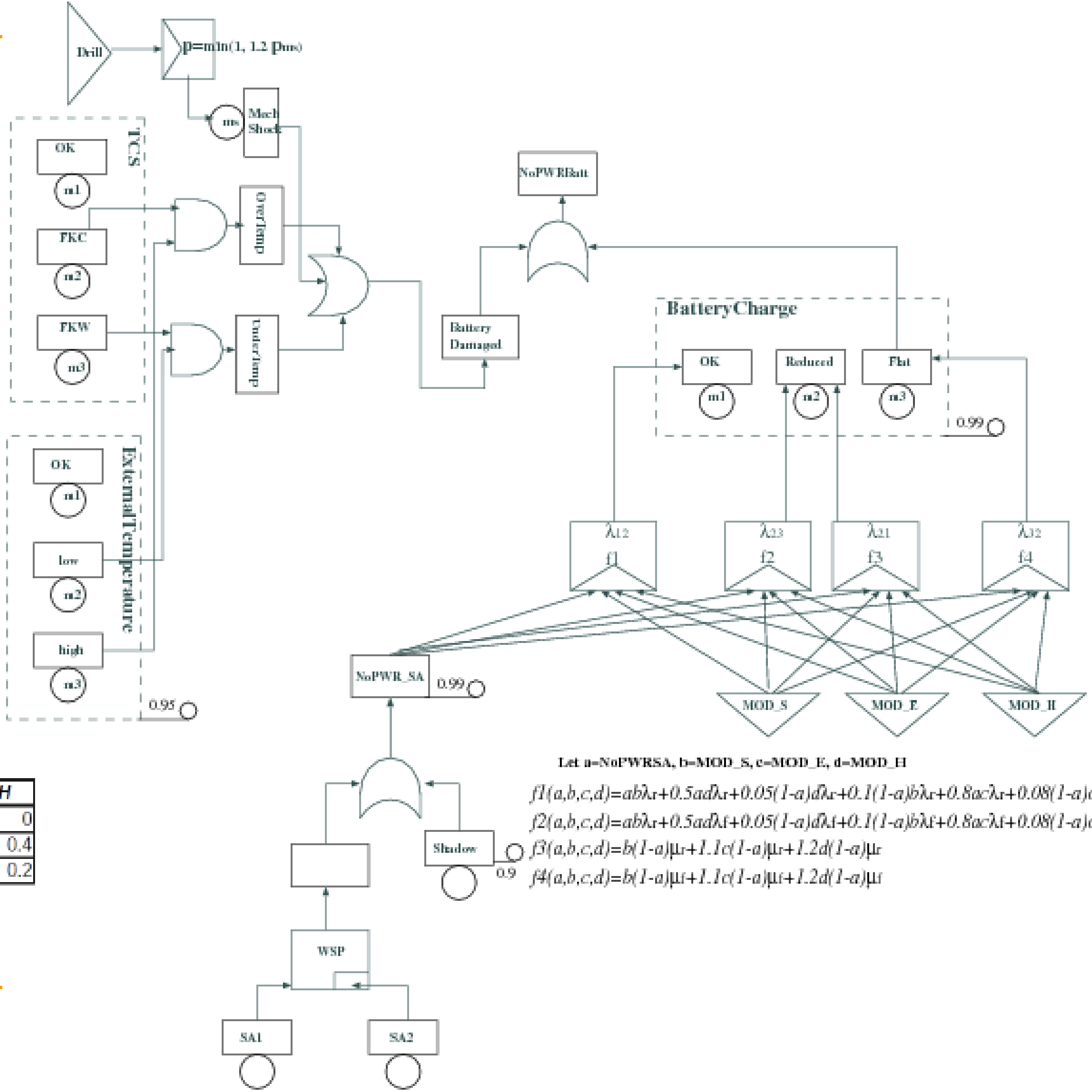
- Software architecture for FDIR analysis based on DBN inference
- Part of the VERIFIM study funded by ESA (partners U.P.O. and Thales/Alenia)
- Case study: Mars Rover power management subsystem reliability

ARPHA Block Scheme



Inference Algorithms: ex-novo ANSI-C (RTEMS) implementation of
1.5JT (Murphy's Algorithm)
BK (Boyen-Koller approximation)

Extended DFT



Let a=NoPWRSA, b=MOD_S, c=MOD_E, d=MOD_H

$f1(a,b,c,d)=ab\lambda_r+0.5ad\lambda_r+0.05(1-a)d\lambda_r+0.1(1-a)b\lambda_r+0.8ac\lambda_r+0.08(1-a)c$

$f2(a,b,c,d)=ab\lambda_r+0.5ad\lambda_f+0.05(1-a)d\lambda_f+0.1(1-a)b\lambda_f+0.8ac\lambda_f+0.08(1-a)c$

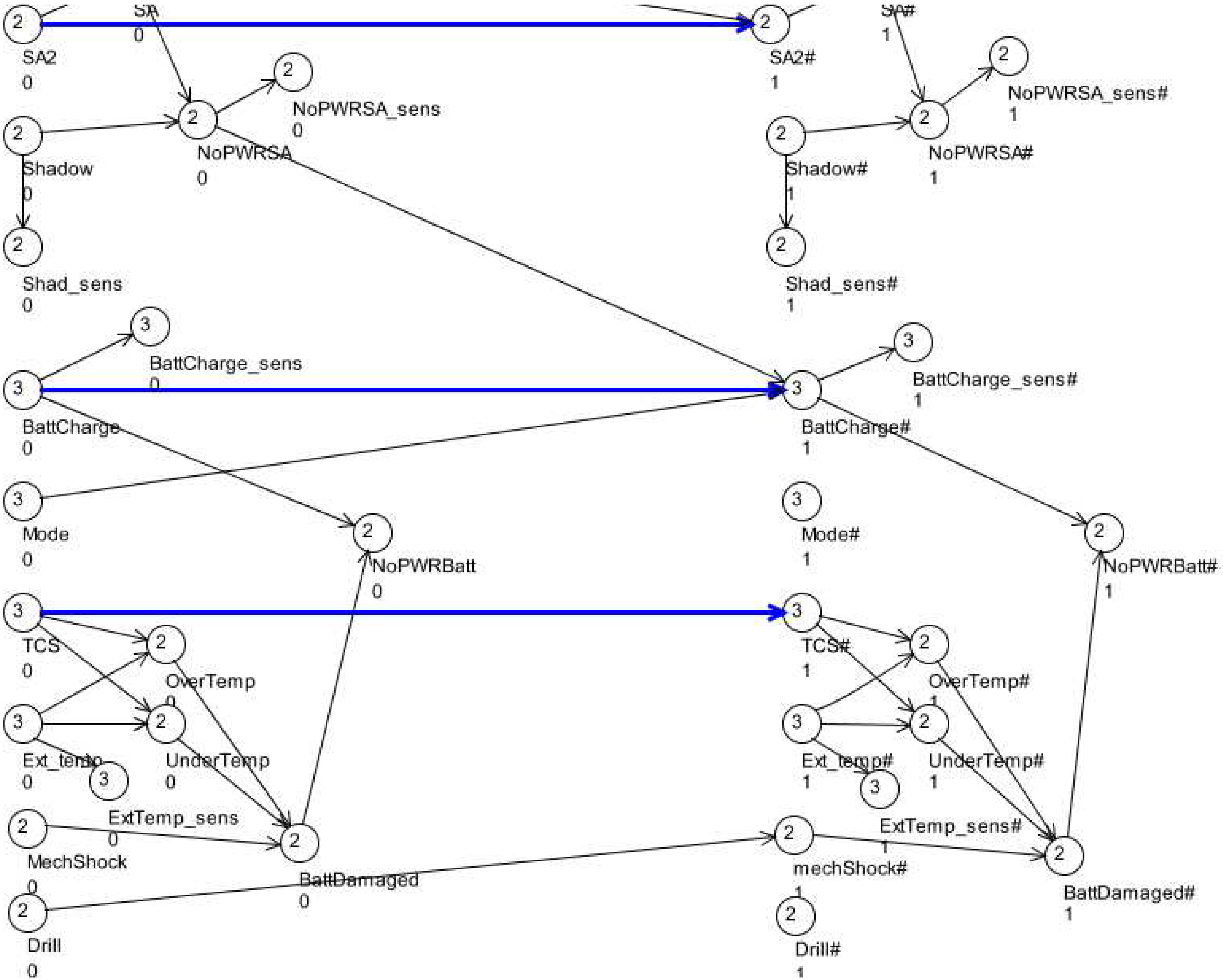
$f3(a,b,c,d)=b(1-a)\mu_r+1.1c(1-a)\mu_r+1.2d(1-a)\mu_r$

$f4(a,b,c,d)=b(1-a)\mu_f+1.1c(1-a)\mu_f+1.2d(1-a)\mu_f$

	MOD_S	*MOD_E*	*MOD_H*
Ok	1	0.8	0
Reduced	0.7	0.9	0.4
Flat	0	0	0.2

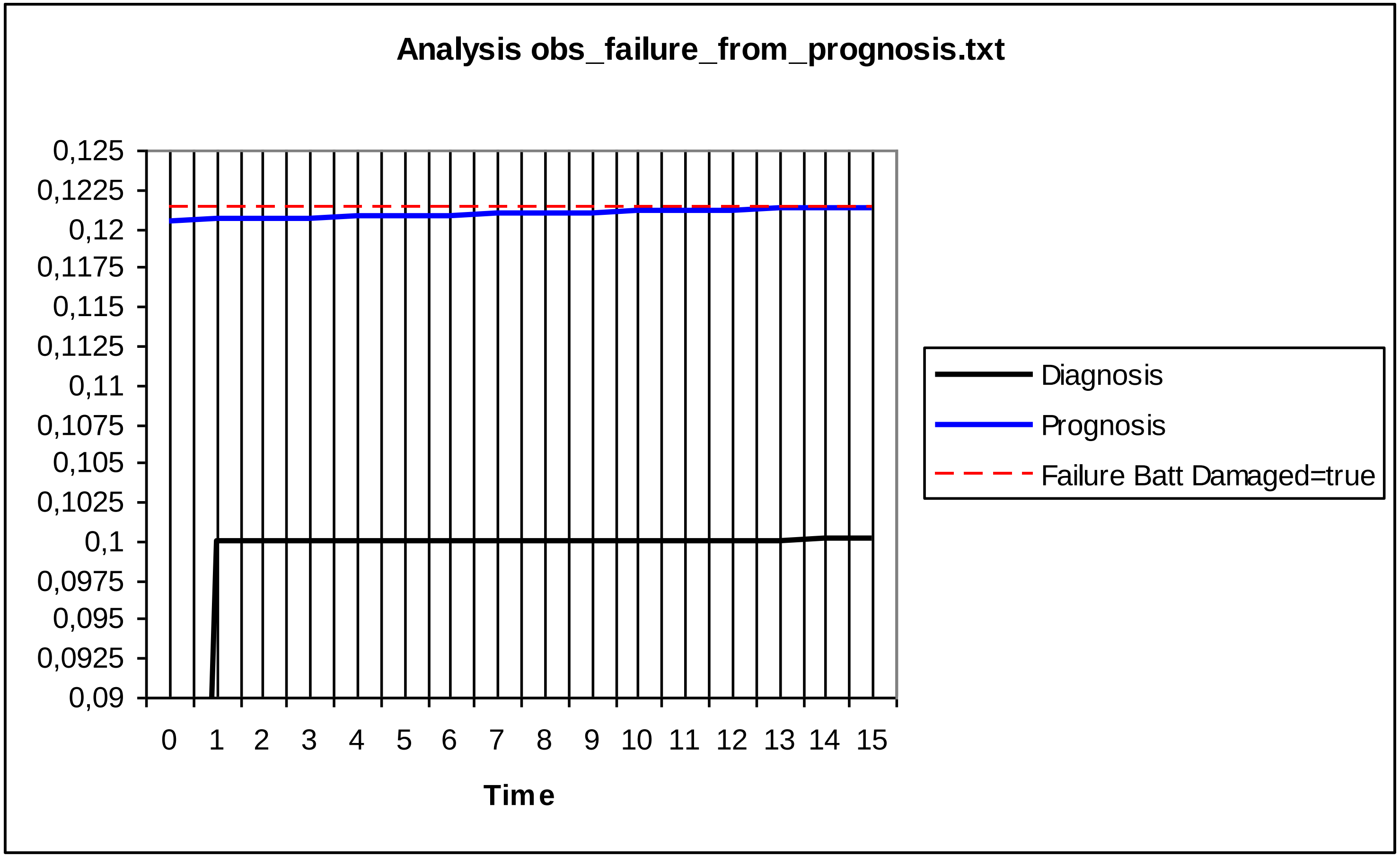
Utility table

DBN fragment



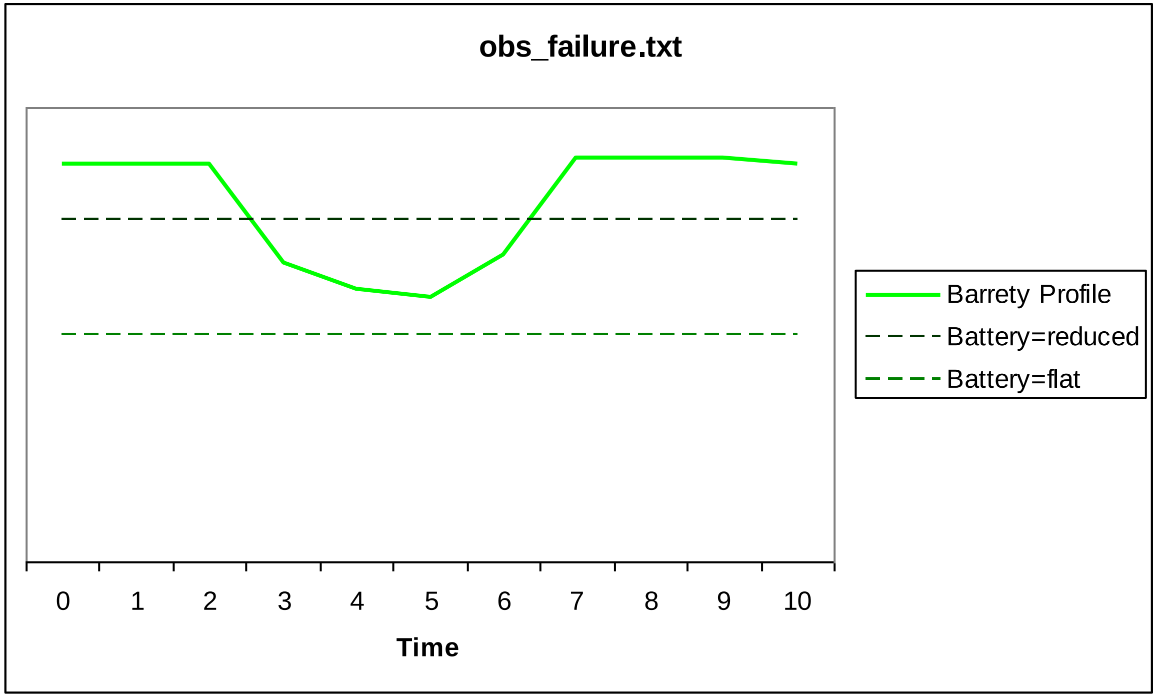
Diagnosis vs Prognosis

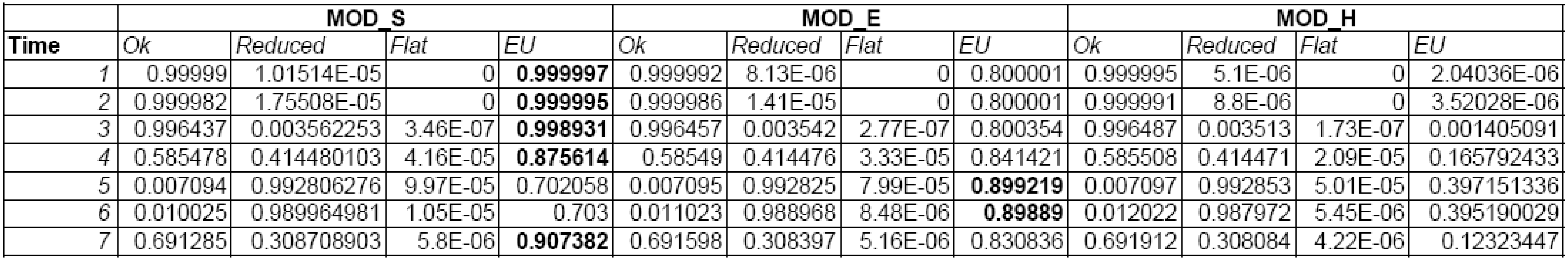
Sensor data with action (drill) at mission time 15 that will cause the damage of battery in the n_prog steps (180)



Diagnosis at time 15 says OK, but prognosis says “you'll got a problem in n steps”

Recovery as selection of best action





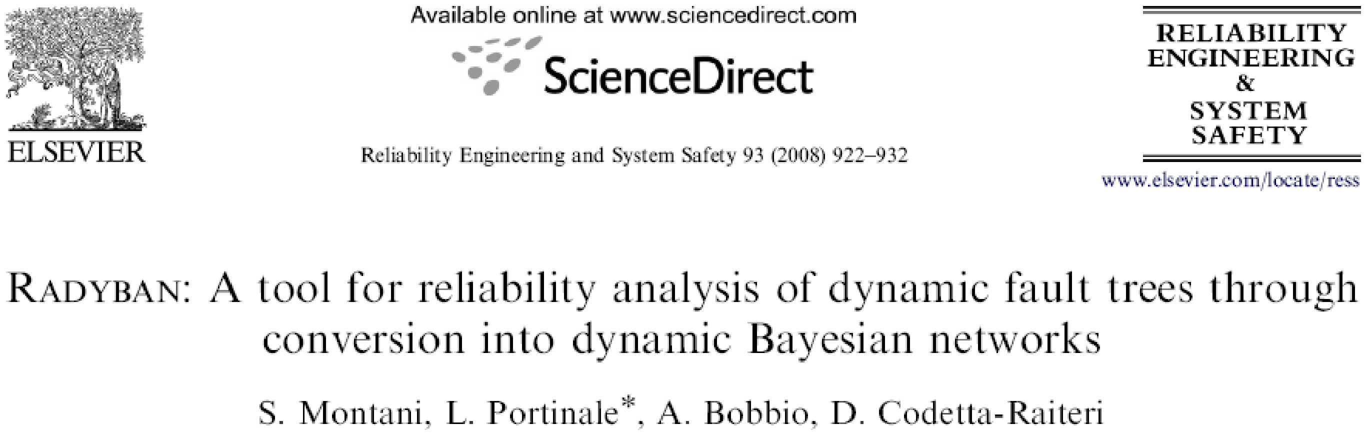
	MOD_S				MOD_E				MOD_H			
Time	*Ok*	*Reduced*	*Flat*	*EU*	*Ok*	*Reduced*	*Flat*	*EU*	*Ok*	*Reduced*	*Flat*	*EU*
1	0.99999	1.01514E-05	0	**0.999997**	0.999992	8.13E-06	0	0.800001	0.999995	5.1E-06	0	2.04036E-06
2	0.999982	1.75508E-05	0	**0.999995**	0.999986	1.41E-05	0	0.800001	0.999991	8.8E-06	0	3.52028E-06
3	0.996437	0.003562253	3.46E-07	**0.998931**	0.996457	0.003542	2.77E-07	0.800354	0.996487	0.003513	1.73E-07	0.001405091
4	0.585478	0.414480103	4.16E-05	**0.875614**	0.58549	0.414476	3.33E-05	0.841421	0.585508	0.414471	2.09E-05	0.165792433
5	0.007094	0.992806276	9.97E-05	0.702058	0.007095	0.992825	7.99E-05	**0.899219**	0.007097	0.992853	5.01E-05	0.397151336
6	0.010025	0.989964981	1.05E-05	0.703	0.011023	0.988968	8.48E-06	**0.89889**	0.012022	0.987972	5.45E-06	0.395190029
7	0.691285	0.308708903	5.8E-06	**0.907382**	0.691598	0.308397	5.16E-06	0.830836	0.691912	0.308084	4.22E-06	0.12323447

Overview

- Dependability/Reliability issues
- Main Model Types for Reliability
- Probabilistic Graphical Models (BN and DBN)
  - Modeling
  - Computing
- From (Dynamic) Fault Trees to (Dynamic) Bayesian Nets
  - Modeling
  - Computing
- Case Studies
- Tools
- Open Issues

RADYBAN: Reliability Analysis with DYnamic BAyesian Networks

- A tool aimed at exploiting DBN inference for reliability purposes
- Automatic compilation of a DFT into a DBN
- Graphical User Interface (both for DFT and DBN)
- Filtering and Smoothing inference (1.5JT and BK algorithms)
- Developed at the Computer Science Dept. of U.P.O.



ELSEVIER

Available online at www.sciencedirect.com

ScienceDirect

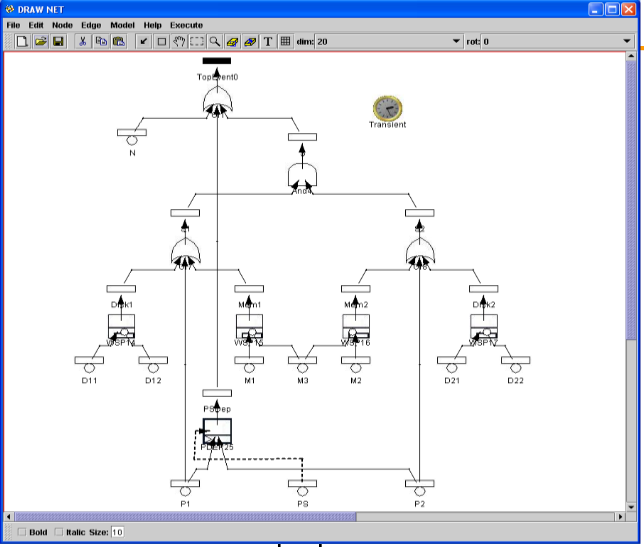
Reliability Engineering and System Safety 93 (2008) 922–932

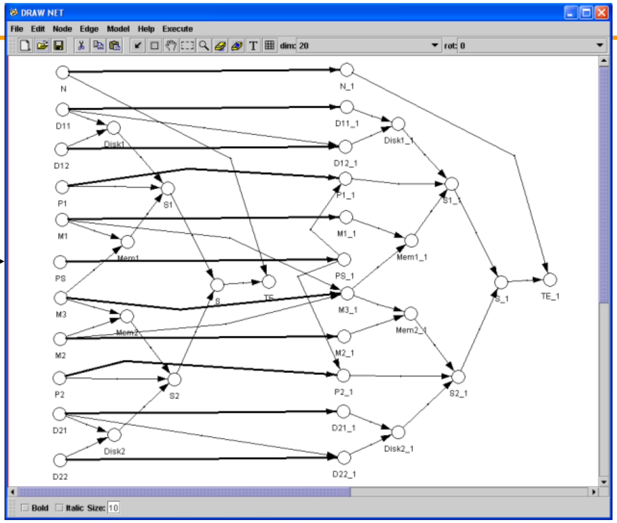
RELIABILITY ENGINEERING & SYSTEM SAFETY

www.elsevier.com/locate/ress

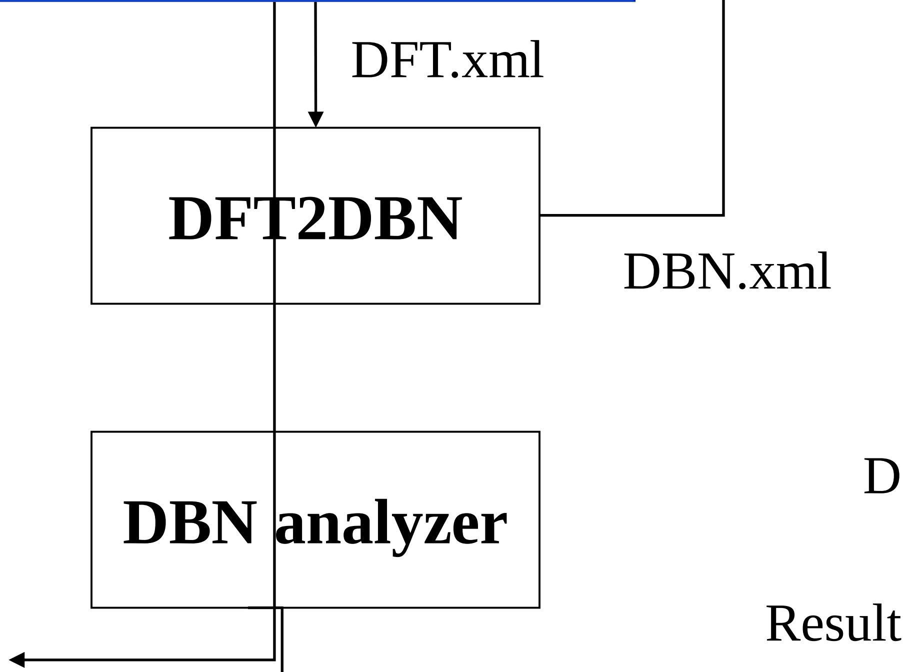
RADYBAN: A tool for reliability analysis of dynamic fault trees through conversion into dynamic Bayesian networks

S. Montani, L. Portinale*, A. Bobbio, D. Codetta-Raiteri



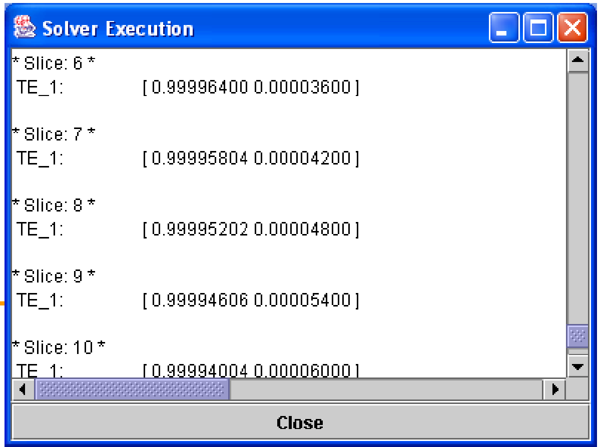


RADYBAN architecture and use

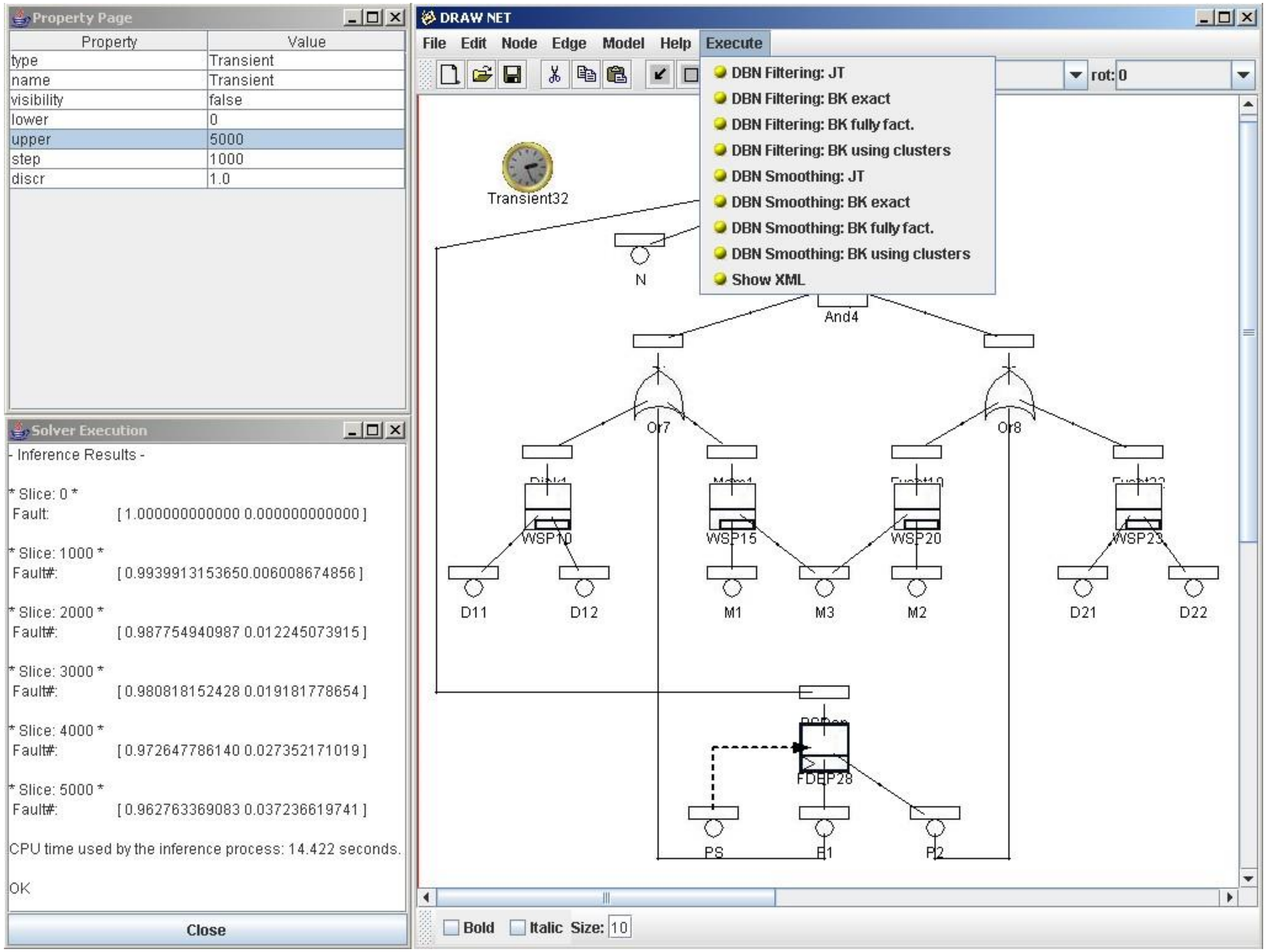


DBN.xml

Results



```
* Slice: 6 *
 TE_1:          [ 0.99996400 0.00003600 ]

* Slice: 7 *
 TE_1:          [ 0.99995804 0.00004200 ]

* Slice: 8 *
 TE_1:          [ 0.99995202 0.00004800 ]

* Slice: 9 *
 TE_1:          [ 0.99994606 0.00005400 ]

* Slice: 10 *
 TE_1:          [ 0.99994004 0.00006000 ]
```



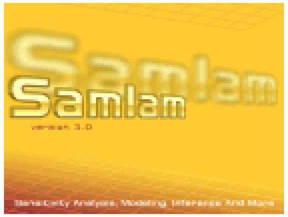
Property	Value
type	Transient
name	Transient
visibility	false
lower	0
upper	5000
step	1000
discr	1.0

```
- Inference Results -

* Slice: 0 *
 Fault:          [ 1.000000000000 0.000000000000 ]

* Slice: 1000 *
 Fault#:         [ 0.9939913153650.006008674856 ]

* Slice: 2000 *
 Fault#:         [ 0.987754940987 0.012245073915 ]

* Slice: 3000 *
 Fault#:         [ 0.980818152428 0.019181778654 ]

* Slice: 4000 *
 Fault#:         [ 0.972647786140 0.027352171019 ]

* Slice: 5000 *
 Fault#:         [ 0.962763369083 0.037236619741 ]

CPU time used by the inference process: 14.422 seconds.

OK
```

Draw-Net GUI
http://www.draw-net.com

INTEL PNL C++ libraries for DBN inference
http://sourceforge.net/projects/openpnl/

BN software tools



Overview

- Dependability/Reliability issues
- Main Model Types for Reliability
- Probabilistic Graphical Models (BN and DBN)
  - Modeling
  - Computing
- From (Dynamic) Fault Trees to (Dynamic) Bayesian Nets
  - Modeling
  - Computing
- Case Studies
- Tools
- Open Issues

Open Issues

- Dealing with continuous variables
  - Gaussian Bayesian Networks
  - Hybrid Bayesian Networks
- Dealing with Continous Time
  - CTBN or GCTBN
- Making the formalism more tailored to reliability practitioners and analysts (tools, tools and … more tools)

Acknowledgments

- Colleagues
  - Prof. Andrea Bobbio
  - Prof. Daniele Codetta-Raiteri
  - Prof. Stefania Montani
- Past Students
  - G. Vercellese
  - M. Varesio
  - S. Di Nolfo
- External collaborators
  - Ing. M. Minichino (ENEA)
  - Ing. E. Ciancamerla (ENEA)
  - Ing. A. Guiotto (Thales/Alenia)