Stochastic Structured Modeling and Performance Analysis of a Multiprocessor System

Ihab Sbeity Brigitte Plateau

{first name. last name}@imag.fr

ID-IMAG Laboratory, Grenoble, France
ACI Sure-Paths Project / Mescal Project

Introduction

Motivations

Motivations - 2

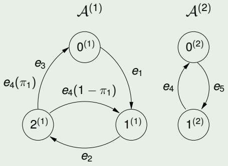
- Stochastic Automata Networks (SAN) is an efficient way to handle the problems above
  - SAN is a structured decomposition of a MARKOV chain.
  - SAN allows the use of PH distributions.
  - SAN permits to model easly the dependencies (functional rate).

SAN formalism (B.Plateau, 1980)

Goal

- Modeling very large and complex MARKOV chains in a compact and structured manner : *Tensor* or *Kronecker Algebra*.

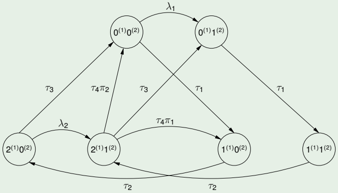
Example



Event	Rate
e_1	τ_1
e_2	τ_2
e_3	τ_3
e_4	τ_4
e_5	f_5

$$f_5 = [(st(\mathcal{A}^{(1)}) == 0^{(1)}) * \lambda_1] + [(st(\mathcal{A}^{(1)}) == 2^{(1)}) * \lambda_2]$$

(a) : A sample SAN Model



(b) : Transition rate diagram of the corresponding MARKOV chain

Tensor Formula

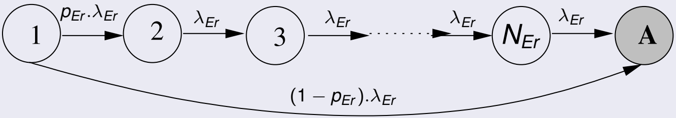
- The generator of the Markov chain is given by :

$$Q = \oplus_{i=1}^{N} Q^{(i)} + \sum_{e \in \mathcal{E}_s} \otimes_{i=1}^{N} Q_{e^+}^{(i)} + \otimes_{i=1}^{N} Q_{e^-}^{(i)}$$

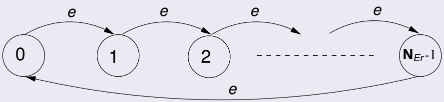
Advantages

- Less memory is required for the storage of the chain.
- Resolution is done using the elementary matrices and powerful algorithms (shuffle algorithms)
- SAN and PH Distributions (I. Sbeity *et al*, 2005) : PH-SAN models

Erlang Distribution



Erlang Distribution as a SAN automatan

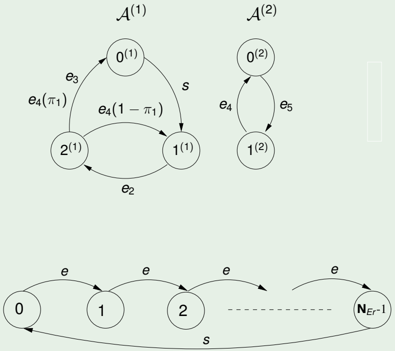


- State 0 is a special state.
- Event e has a rate $\tau_e.\tilde{f}$.
- $\tilde{f}$ equal to 1 if the Erlang event is fireable in the "base model".

Building the new SAN model

- Connection between PH (Erlang) automata and the base automata can be done systematically.
- The connection procedure takes into account the Preemption Policy of the PH distribution.

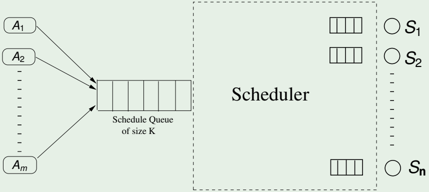
Example



System architecture

The System model

- Queuing architecture of an n-site distributed system.



System features

- Scheduling policy satisfies a mean average of service time.
- Sites may fail. We consider a *crach fault* assumption
  - transient failure : the failed site has to be rebooted.
  - permanent failure : the failed site has to be repaired.

Rates & Distributions

- The availability model - **Sites**
  - Failure distribution time is exponential, and depends of the number of *active* sites $N(active)$ (rate = $flt \times N(active)$).
  - Reboot distribution time is exponential (rate = *reb*).
  - Repair time follows a general distribution (rate = *rep*).
- The performance model - **Sites + Task Schedule Queue**
  - Poissonian arrival process of tasks with mean $1/\lambda$ (rate = λ).
  - Service time of mean value $1/\mu$, and depends of the number of *active* sites (rate = $\mu \times N(active)$)

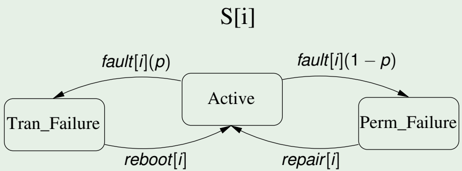
Non-Markovian distribution approximations

Exponential, Erlang-3 and Erlang-8 distributions

the performance model

(n=7 sites and Erlang-3) $\Rightarrow$ $|$state-space$|$ $\approx$ 8 millions.

Availability SAN model



event / dist / rate

- *fault*[*i*] / exponential / $flt \times N(A)$
- *reboot*[*i*] / exponential / *reb*
- *repair*[*i*] / general / *rep*

Global failure

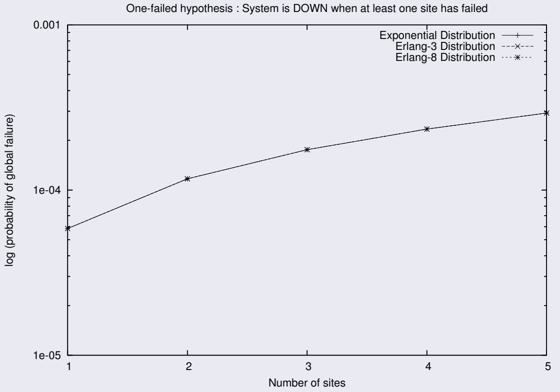
- Different "System is Down" assumptions are considered :
  -

Availability analysis

parameters

- $f = 10^{-5}$ (failure mean time = $\frac{10^5}{N(A)}$ mn)
- $reb = 1$ (rebooting mean time = 1 mn)
- $rep = 0.033$ (repairing mean time is about 30 mn)

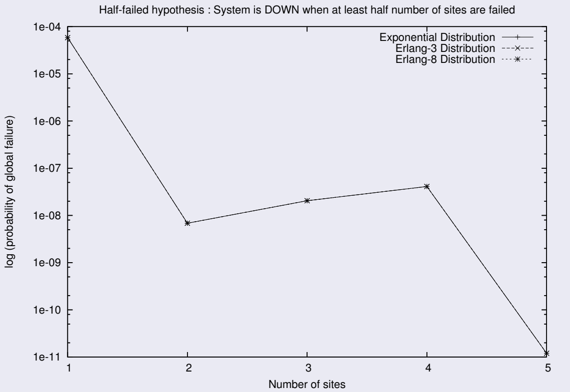
One-failed hypothesis



parameters

- $f = 10^{-5}$ (failure mean time = $\frac{10^5}{N(A)}$ mn)
- $reb = 1$ (rebooting mean time = 1 mn)
- $rep = 0.033$ (repairing mean time is about 30 mn)

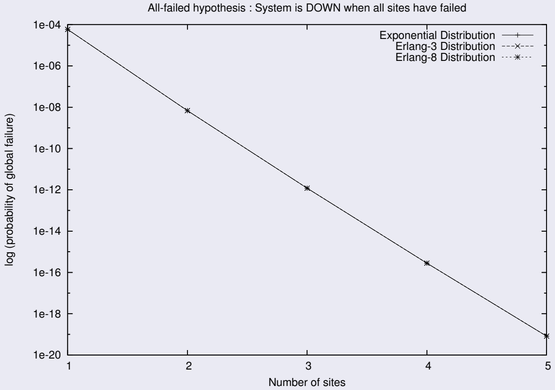
Half-failed hypothesis



parameters

- $f = 10^{-5}$ (failure mean time = $\frac{10^5}{N(A)}$ mn)
- $reb = 1$ (rebooting mean time = 1 mn)
- $rep = 0.033$ (repairing mean time is about 30 mn)

All-failed hypothesis

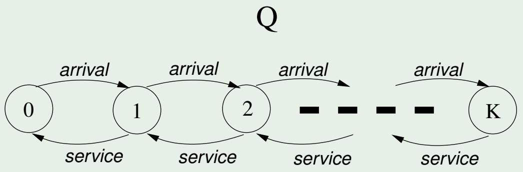


Performance analysis

The availability analysis

- Exponential distribution is robust to approximate the repair time distribution.
- The performance model :
  - repair time distribution may now be considered as exponential
  - a smaller state space

SAN model of the Task Schedule Queue



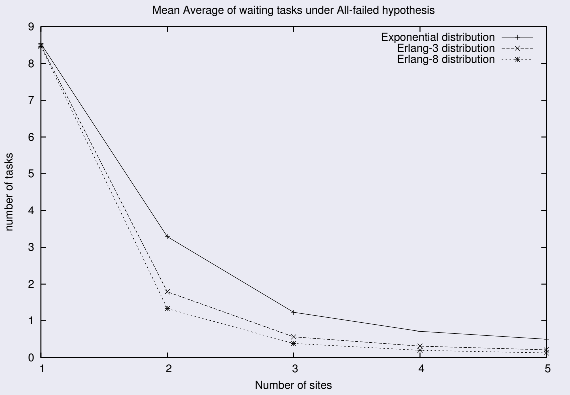
event / dist / rate

- *arrival* / exponential / λ
- *service* / general / $\mu \times N(A)$

parameters

- $\lambda = 5$
- $\mu = 3$
- $\rho = \frac{\lambda}{\mu} \times \frac{1}{N(A)}$ (the workload of the system)

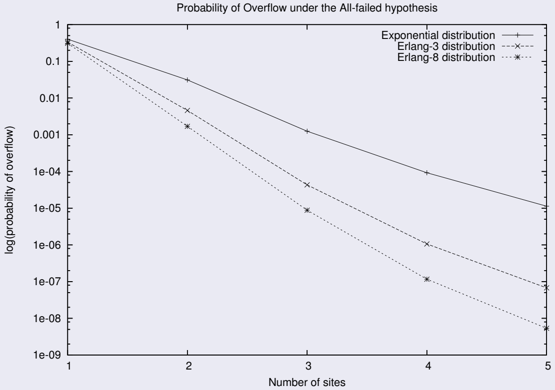
mean average of waiting tasks



parameters

- $\lambda = 5$
- $\mu = 3$
- $\rho = \frac{\lambda}{\mu} \times \frac{1}{N(A)}$ (the workload of the system)

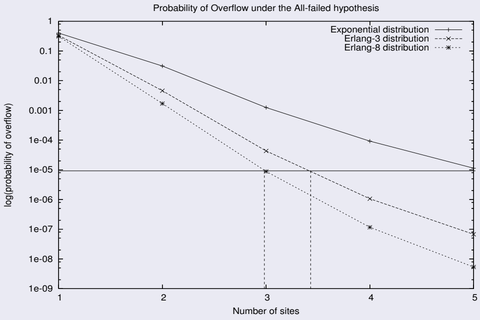
mean probability of overflow



QoS analyzis

- N_{QoS} the minimal number of sites which satisfies a QoS.
- QoS : " the logarithmic value of the probability of overflow is less than 10^{-5} "

mean probability of overflow



N_{QoS} value

- 3 with Erlang-8.
- 4 with Erlang-3.
- 6 with Exponential.

Conclusion

Conclusion

- PH Distribution to approximate non-Markovian distribution of a multiprocessors system.
- SAN formalism :
  - models systems with large state space.
  - allows PH Distribution to be used.
- Exponential distribution may be not robust to present non-Markovian distribution.
- Future works :
  - Others PH distributions than Erlang.
  - Others non-Markovian activities, e.g. the arrival process.