Mathematical models for computer systems behaviour

Goals : predict computer system behaviours
- performances measurements,
- comparison of systems,
- dimensioning,

Methodology :
- modelling environment (stochastic process)
- modelling system (automaton)
- behaviour = reaction of automaton on stochastic stimuli

Organisation

Automata + probabilistic transitions :

Discrete time Markov chains

Automata + probabilistic transitions + time:

Poisson processes, continuous time Markov chains

State space structure :

Simple queues, product form queuing networks

Stochastic automata networks

Simulation of Markov chains :

Direct simulation/perfect simulation

Applications

- **Operating systems**
- **Networks protocols**
- **Manufacturing systems**
- **Production lines**
- **Middlewares**
- **...**

Scientific domains

Applied mathematics :

stochastic processes, ergodic theory,...

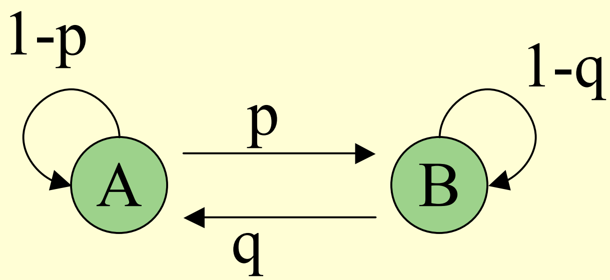
Markov processes

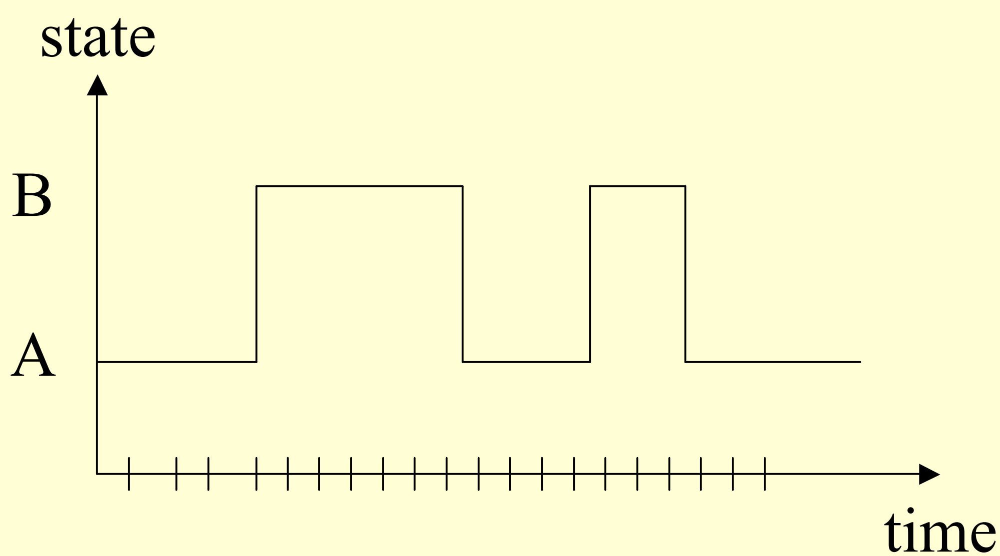
References :

R. Nelson Probability theory with...

R. Jain The art of computer systems performance analysis

A first example : Flip-flop





$$P(X_{n+1} = B | X_n = A) = p$$

$$P(X_{n+1} = A | X_n = A) = 1 - p$$

$$P(X_{n+1} = A | X_n = B) = q$$

$$P(X_{n+1} = B | X_n = B) = 1 - q$$

$$P = \begin{pmatrix} 1 - p & p \\ q & 1 - q \end{pmatrix}$$

Transition matrix

Long run behaviour

Linear recurrence equation

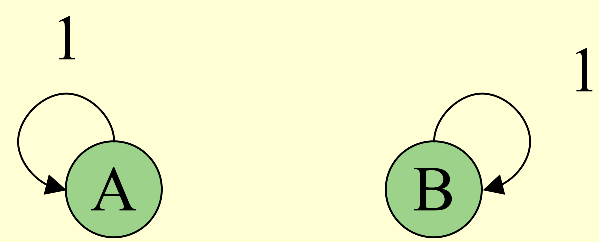
$$\pi_n = P(X_n = A)$$

$$\pi_{n+1} = \pi_n(1-p) + (1-\pi_n)q = q + (1-p-q)\pi_n$$

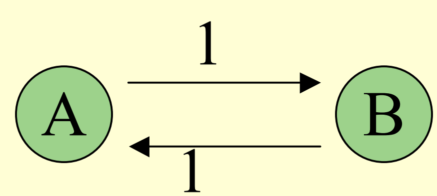
Case 1 : |1-p-q| <1

$$\pi_n = \frac{q}{p+q} + \left(\pi_0 - \frac{q}{p+q}\right)(1-p-q)^n$$

Case 2 : 1-p-q =1 => p=q=0



Case 3 : 1-p-q =-1 => p=q=1



Results

Convergence

$$\pi_A = \lim P(X_n = A) = \frac{q}{p+q};$$

$$\pi_B = \lim P(X_n = B) = \frac{p}{p+q}$$

Geometric

$$(1 - p - q)^n$$

Asymptotic satisfies

$$\pi_A = \pi_A(1-p) + \pi_B q$$

$$\pi_B = \pi_A p + \pi_B(1-q)$$

Lack of memory

Ergodic convergence

$$\pi_A = \lim P\left(X_n = A\right) = \lim \frac{1}{n} \sum_{i=1}^{n} 1_{\left(X_i = A\right)};$$

$$\pi_B = \lim P\left(X_n = B\right) = \lim \frac{1}{n} \sum_{i=1}^{n} 1_{\left(X_i = A\right)}$$

Performances indexes :

- link utilisation
- communication delays

Discrete time Markov chains

$\{X_n\}_{n \in N}$ Trajectory of the system
Discrete state space
Step by step evolution

$$P(X_{n+1} = j \mid X_n = i, X_{n-1} = i_{n-1}, \ldots, X_0 = i_0) = P(X_{n+1} = j \mid X_n = i)$$

Lack of memory

Homogeneity (in time)

$$P(X_{n+1} = j \mid X_n = i) = P(X_1 = j \mid X_0 = i) = p_{i,j}$$

Transition matrix (stochastic) $\mathrm{P} = ((p_{i,j}))$ $p_{i,j} \geq 0, \quad \sum_j p_{i,j} = 1.$

Chapman-Kolmogorov equations

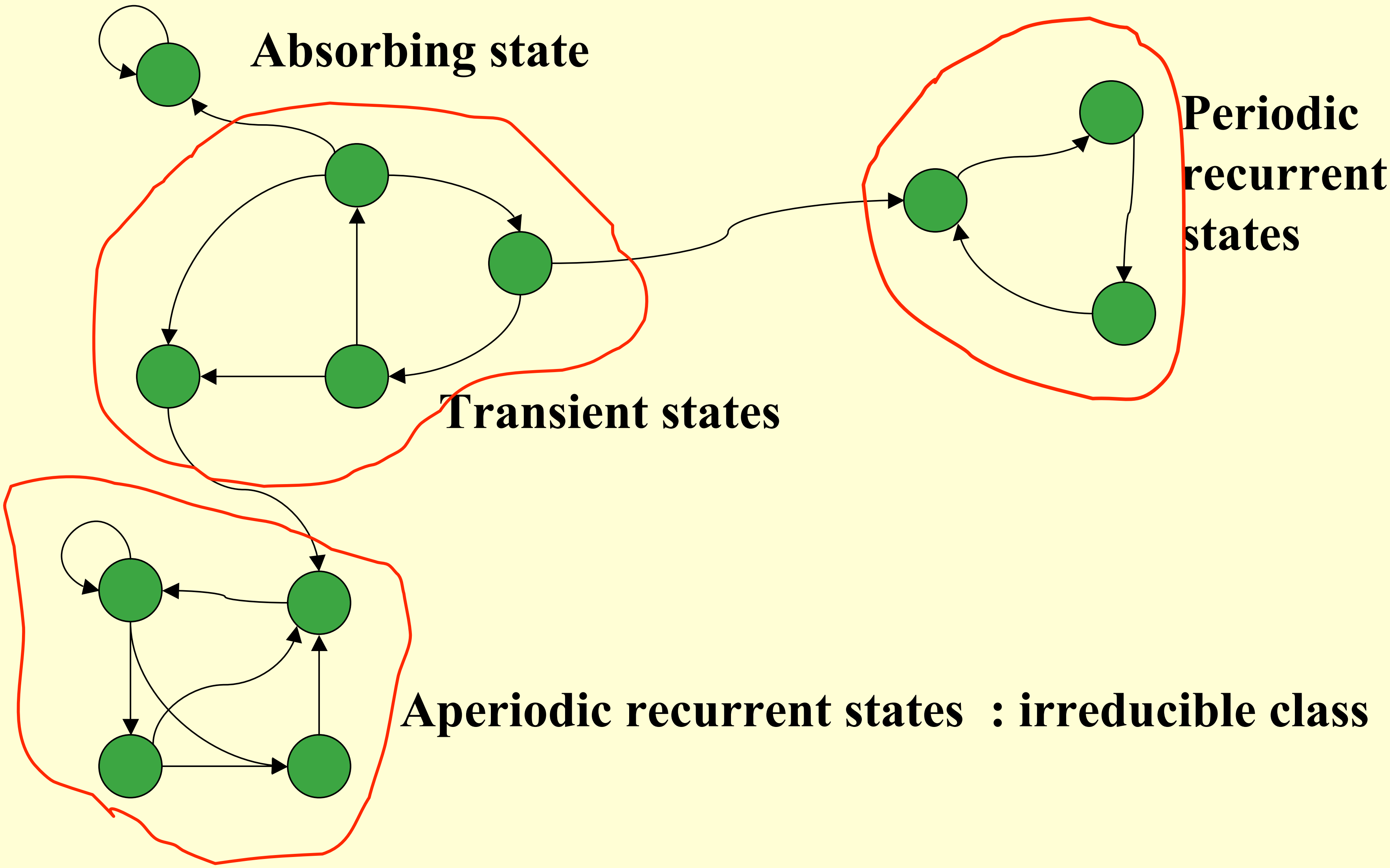
$$P(X_2 = j \mid X_0 = i) = \sum_k P(X_2 = j \mid X_1 = k) P(X_1 = k \mid X_0 = i) = \sum_k p_{i,k} p_{k,j}$$

Iteration ---> product of matrices

Asymptotic behaviour

$$\lim_{n \to \infty} \mathrm{P}^n = ??$$

Classification of states



Convergence theorem

$\{X_n\}_{n \in N}$ **Homogeneous, aperiodic, irreducible Markov chain (finite state space)**

$$\lim P\left(X_n = j \middle| X_0 = i\right) = \pi_j$$

$\pi = \left(\pi_1, \pi_2, \ldots, \pi_N\right)$ **Steady-state vector**

Unique solution of the linear system $\pi = \pi \mathrm{P}$

Geometric convergence (module of the second eigenvalue)

Ergodic theorem

$\{X_n\}_{n\in N}$ **Homogeneous, aperiodic, irreducible Markov chain (finite state space)**

For any function f (cost function)

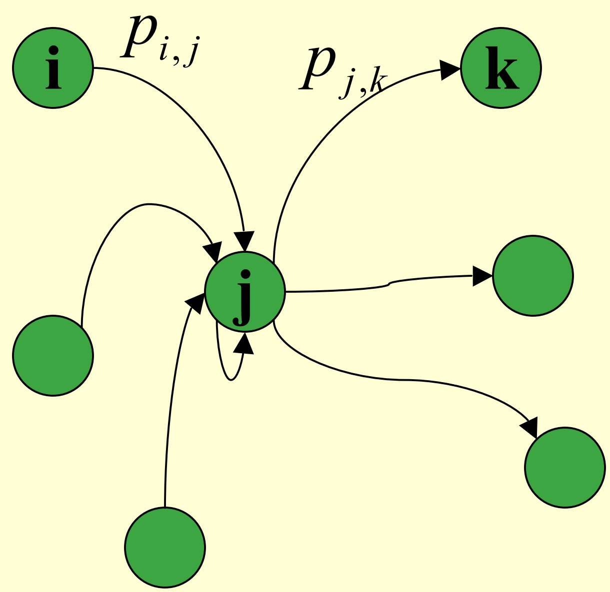
$$\lim \frac{1}{n}\sum_{k=0}^{n-1} \mathrm{f}(X_k) = \sum_i \pi_i \mathrm{f}(i)$$

Estimation of the steady-state distribution by -> simulation

$$\lim \frac{1}{n}\sum_{k=0}^{n-1} 1_{X_k = i} = \pi_i$$

Equilibrium equations

Interpretation :

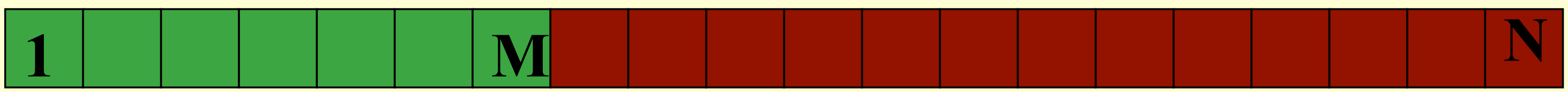


$$\sum_i \pi_i p_{i,j} = \pi_j \sum_k p_{j,k} = \pi_j$$

$$\Leftrightarrow \pi = \pi \mathrm{P}$$

Example : cache management

Cache space **Memory space**



Optimisation : Time to access cache << Time to access memory

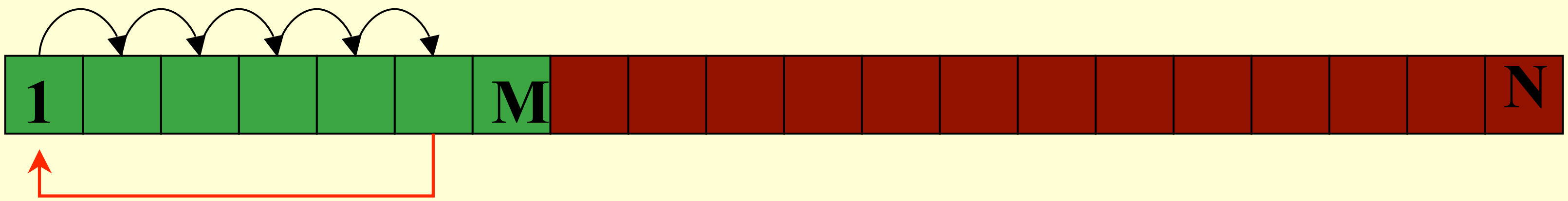
Cache replacement policy

M = cache size

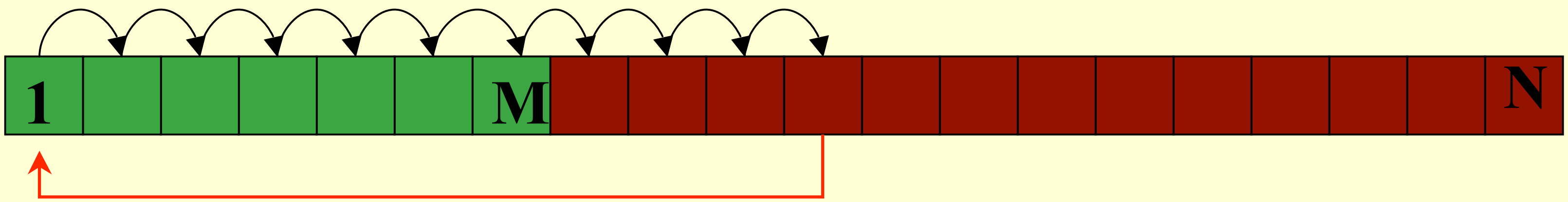
N =global memory size

LRU policy

Cache hit

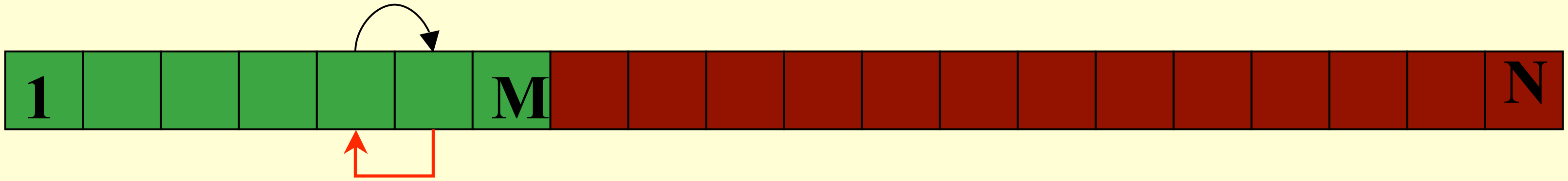


Cache miss

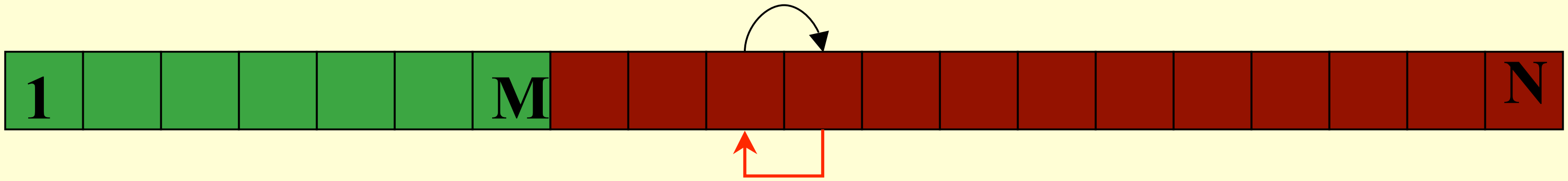


Move ahead policy

Cache hit



Cache miss



Environment model

Program = sequence of memory accesses

hypothesis 1 :

independent sequence

same distribution

State space : permutations of (1,…,N)

size : N!

State space reduction

Hypothesis 2 :

one reference A is more frequent

others are equally distributed

(uniform)

$$a = P(\text{reference A}),$$

$$b = P(\text{reference other than A}),$$

$$a + (N-1)b = 1.$$

Markov model

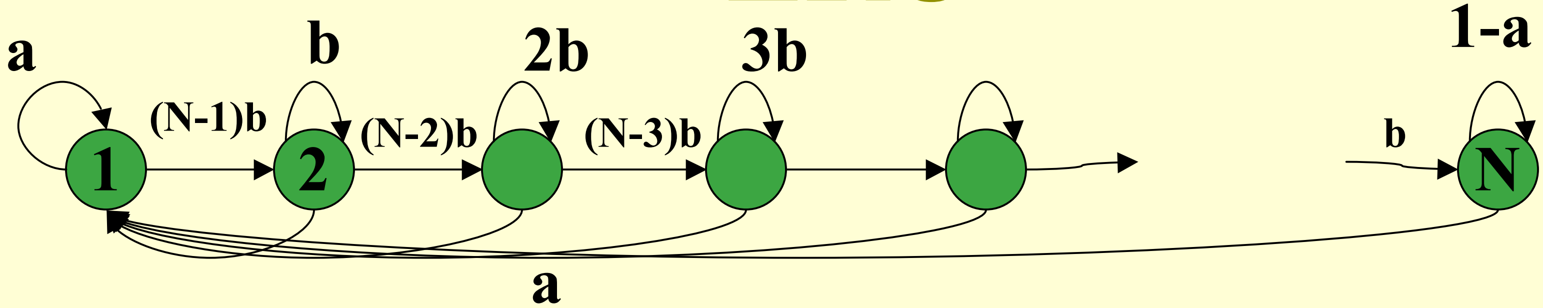
$\{X_n\}_{n \in N}$ X_n = position of reference A at step n

Markov chain : homogeneous, aperiodic, irreducible

=> convergence and ergodicity

=> computation of the steady-state

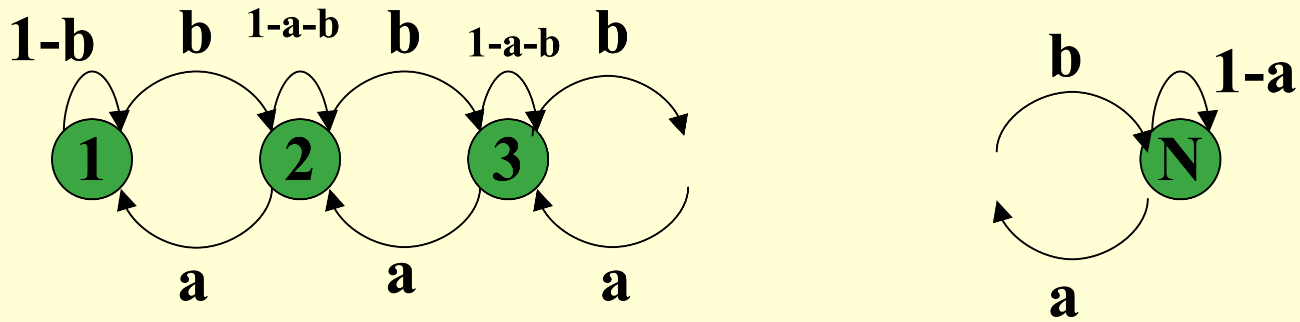
LRU



$$P_{LRU}=\begin{bmatrix} a & (N-1)b & 0 & 0 & & 0 \\ a & b & (N-2)b & 0 & & 0 \\ a & 0 & 2b & (N-3)b & & 0 \\ a & 0 & 0 & & & 0 \\ a & 0 & 0 & & & 0 \\ a & 0 & 0 & & & 0 \\ a & 0 & 0 & & (N-2)b & b \\ a & 0 & 0 & & 0 & (N-1)b \end{bmatrix}$$

$$\pi_i.(N-i)b=\pi_{i+1}.(a+(N-i-1)b)$$

Move ahead



$$P_{MA} = \begin{bmatrix} 1-b & b & 0 & 0 & & & 0 \\ a & 1-a-b & b & 0 & & & 0 \\ 0 & a & 1-a-b & b & & & 0 \\ 0 & 0 & a & 1-a-b & & & 0 \\ 0 & 0 & 0 & a & & & 0 \\ 0 & 0 & 0 & 0 & & & 0 \\ 0 & 0 & 0 & 0 & a & 1-a-b & b \\ 0 & 0 & 0 & 0 & & a & (N-1)b \end{bmatrix}$$

$$\pi_i = C\left(\frac{b}{a}\right)^i$$

$$\pi_i(a+b) = \pi_{i+1}a + \pi_{i-1}.b \quad \Rightarrow \quad \pi_i b = \pi_{i+1} a$$

Numerical example

Toy example N=8 a=0.3 b=0.1

Steady-state probability vectors

LRU [0.30 0.23 0.18 0.12 0.08 0.05 0.03 0.01]

MA [0.67 0.22 0.07 0.02 0.01 0.01 0.00 0.00]

Self optimising algorithm

Rapid decreasing

Cache miss evaluation

Page A

M	LRU	MA
0	1.00	1.00
1	0.70	0.33
2	0.47	0.11
3	0.29	0.04
4	0.17	0.02
5	0.09	0.01
6	0.04	0.00
7	0.01	0.00
8	0.00	0.00

Global

M	LRU	MA
0	1.00	1.00
1	0.84	0.77
2	0.69	0.62
3	0.56	0.51
4	0.43	0.40
5	0.32	0.30
6	0.21	0.20
7	0.10	0.10
8	0.00	0.00

Example: conclusion

Self adapting algorithm,
"Minimise" cache miss
Move-ahead better than LRU

Speed of convergence ?

2nd module of eigenvalue
LRU : 0.7
MA : 0.96
LRU reaches stationary regime more quickly than MA !!!

Example: conclusion

Hypothesis 2 :

relaxing uniformity (ex Zipf law)

==> same behaviour (simulation)

Hypothesis 1 :

model of evolution of probability of references

==> depends on the "variability" of the process

compromise between rate of convergence and speed of evolution !

Generalizations

- Infinite state space

  Same behaviour (if non recurrent null)
- Transient analysis

  study of powers of P
- Generalized Markov Processes

  Timed states
    - computation of steady-state
    - weighted probabilities

Solving Markov chains

N< 50 Formal methods Maple

N<500 Classical numerical methods

(Gaussian elimination,…) Mathematica, Lapack…

N<100000 Iterative methods, preconditionning,

N< 10000000 Specific numerical algorithms

(sparse matrices, …) Marca, Peps,…

N>10000000 Simulation

Over … Approximations and analytical techniques

Links in computer science

Common formalisms with verification tools

Queuing networks

Petri nets

Process algebra

Automata networks

==> state space construction,
behaviours specifications