From Stochastic Processes to Stochastic Petri Nets

Serge Haddad

LSV CNRS & ENS Cachan & INRIA Saclay

Advanced Course on Petri Nets, the 16th September 2010, Rostock

Plan

Discrete Event Stochastic Process

Intuitively

An execution of a discrete event stochastic process (DESP) is an infinite sequence of events: $e_1, e_2, \ldots$ interleaved with (possibly null) delays.
(generated by some operational model)

Formally

A discrete event stochastic process is defined by two families of random variables:

- $S_0, S_1, S_2, \ldots$ such that S_0 is the initial state and S_i is the state of the system after the occurrence of e_i.
- $T_0, T_1, T_2, \ldots$ such that T_0 is the time elapsed before the occurrence of e_0 and T_i is the time elapsed between the occurrences of e_i and e_{i+1}.

Hypotheses and notations

- The process diverges almost surely, i.e. $Pr(\sum_{i \in \mathbb{N}} T_i = \infty) = 1$.
- thus $N(\tau) = \min(\{n \mid \sum_{k \leq n} T_k > \tau\})$ is defined almost everywhere and $X(\tau) = S_{N(\tau)}$ is the observable state at time τ.
- When $Pr(S_0 = s) = 1$, one says that the process starts in s.

Discrete Event Stochastic Process

Intuitively

An execution of a discrete event stochastic process (DESP) is an infinite sequence of events: $e_1, e_2, \ldots$ interleaved with (possibly null) delays.
(generated by some operational model)

Formally

A discrete event stochastic process is defined by two families of random variables:

- $S_0, S_1, S_2, \ldots$ such that S_0 is the initial state and S_i is the state of the system after the occurrence of e_i.
- $T_0, T_1, T_2, \ldots$ such that T_0 is the time elapsed before the occurrence of e_0 and T_i is the time elapsed between the occurrences of e_i and e_{i+1}.

Hypotheses and notations

- The process diverges almost surely, i.e. $Pr(\sum_{i \in \mathbb{N}} T_i = \infty) = 1$.
- thus $N(\tau) = \min(\{n \mid \sum_{k \leq n} T_k > \tau\})$ is defined almost everywhere and $X(\tau) = S_{N(\tau)}$ is the observable state at time τ.
- When $Pr(S_0 = s) = 1$, one says that the process starts in s.

Discrete Event Stochastic Process

Intuitively

An execution of a discrete event stochastic process (DESP) is an infinite sequence of events: $e_1, e_2, \ldots$ interleaved with (possibly null) delays.
(generated by some operational model)

Formally

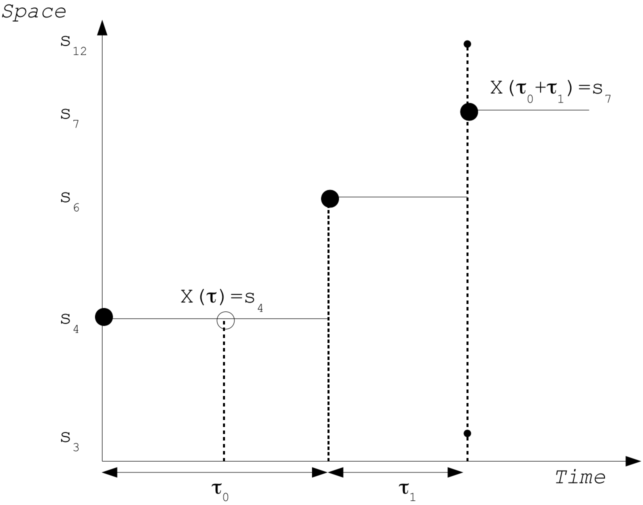
A discrete event stochastic process is defined by two families of random variables:

- $S_0, S_1, S_2, \ldots$ such that S_0 is the initial state and S_i is the state of the system after the occurrence of e_i.
- $T_0, T_1, T_2, \ldots$ such that T_0 is the time elapsed before the occurrence of e_0 and T_i is the time elapsed between the occurrences of e_i and e_{i+1}.

Hypotheses and notations

- The process diverges almost surely, i.e. $Pr(\sum_{i \in \mathbb{N}} T_i = \infty) = 1$.
- thus $N(\tau) = \min(\{n \mid \sum_{k \leq n} T_k > \tau\})$ is defined almost everywhere and $X(\tau) = S_{N(\tau)}$ is the observable state at time τ.
- When $Pr(S_0 = s) = 1$, one says that the process starts in s.

An Execution of a Process



$T_0 = \tau_0$

$T_1 = \tau_1$

$T_2 = 0$

$T_3 = 0$

$S_0 = s_4$

$S_1 = s_6$

$S_2 = s_3$

$S_3 = s_{12}$

$S_4 = s_7$

Analysis of DESP

Two kinds of analysis

- **Transient analysis**: computation of measures depending on the elapsed time since the initial state.
- **Steady-state analysis**: computation of measures depending on the long-run behaviour of the system *(requires to establish its existence)*.

Performance indices

- A *performance index* is a function from states to numerical values.
- The measure of an index f w.r.t. to a state distribution π is given by:
  $$\sum_{s \in S} \pi(s) \cdot f(s)$$
- When range f is $\{0, 1\}$ it is an *atomic property* and its measure can be rewritten:
  $$\sum_{s \models f} \pi(s)$$

More on performance indices in the next talk

Analysis of DESP

Two kinds of analysis

- **Transient analysis**: computation of measures depending on the elapsed time since the initial state.
- **Steady-state analysis**: computation of measures depending on the long-run behaviour of the system *(requires to establish its existence)*.

Performance indices

- A *performance index* is a function from states to numerical values.
- The measure of an index f w.r.t. to a state distribution π is given by:
  $$\sum_{s \in S} \pi(s) \cdot f(s)$$
- When range f is $\{0, 1\}$ it is an *atomic property* and its measure can be rewritten:
  $$\sum_{s \models f} \pi(s)$$

More on performance indices in the next talk

Discrete Time Markov Chain (DTMC)

A DTMC is a stochastic process which fulfills:

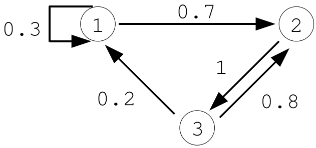
- For all n, T_n is the constant 1
- The process is *memoryless*

$$Pr(S_{n+1} = s_j \mid S_0 = s_{i_0}, ..., S_{n-1} = s_{i_{n-1}}, S_n = s_i)$$

$$= Pr(S_{n+1} = s_j \mid S_n = s_i)$$

$$\equiv \mathtt{P}[i, j]$$

A DTMC is defined by S_0 and $\mathtt{P}$



$$\mathbb{P} \begin{pmatrix} 0.3 & 0.7 & 0.0 \\ 0.0 & 0.0 & 1.0 \\ 0.2 & 0.8 & 0.0 \end{pmatrix}$$

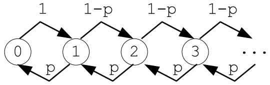
Analysis of a DTMC: the State Status

The transient analysis is easy *(and effective in the finite case)* :

$$\pi_n = \pi_0 \cdot \mathrm{P}^n \text{ with } \pi_n \text{ the distribution of } S_n$$

Classification of states w.r.t. the asymptotic behaviour of the DTMC

- A state is *transient* if the probability of a return after a visit is strictly less than one. Hence the probability of its occurrence will go to zero. $(p < 1/2)$
- A state is *recurrent null* if the probability of a return after a visit is one but the mean time of this return is infinite. Hence the probability of its occurrence will go to zero. $(p = 1/2)$
- A state is *recurrent non null* if the probability of a return after a visit is one and the mean time of this return is finite. $(p > 1/2)$



State Status in finite DTMC

In a finite DTMC

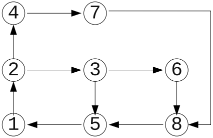
- The status of a state only depends on the graph associated with the chain.
- A state is transient iff it belongs to a non terminal *strongly connected component* (scc) of the graph.
- A state is recurrent non null iff it belongs to a terminal scc.

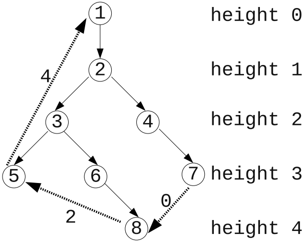
Analysis of a DTMC: Periodicity

Irreducibility and Periodicity

- A chain is *irreducible* if its graph is strongly connected.
- The *periodicity* of an irreducible chain is the greatest integer p such that the set of states can be partionned in p subsets $\mathcal{S}_0, \ldots, \mathcal{S}_{p-1}$ where every transition goes from $\mathcal{S}_i$ to $\mathcal{S}_{i+1\%p}$ for some i.

How to compute the periodicity? Build a rooted tree by any traversal of the graph. (On the fly) associate a value $h_i - h_j + 1$ to every edge (i, j) and compute the gcd of these values.





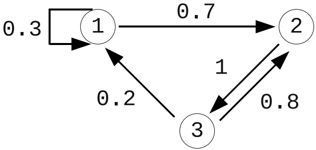
periodicity=gcd(0,2,4)=2

Analysis of a DTMC: a Particular Case

A particular case

The chain is irreducible and *aperiodic* (i.e. its periodicity is 1)

- π_∞ exists and its value is independent from π_0.
- π_∞ is the unique solution of $X = X \cdot \mathtt{P} \wedge X \cdot \mathbf{1} = 1$.
  where one can omit an arbitrary equation of the first system.



$$\pi_\infty \begin{pmatrix} 1/8 & 7/16 & 7/16 \end{pmatrix}$$

$$\pi_1 = 0.3\pi_1 + 0.2\pi_2 \quad \pi_2 = 0.7\pi_1 + 0.8\pi_3 \quad \pi_3 = \pi_2$$

Analysis of a DTMC: the "General" Case

Almost general case: every terminal scc is aperiodic

- π_∞ exists.
- $\pi_\infty = \sum_{s \in S} \pi_0(s) \sum_{i \in I} \mathtt{preach}_i[s] \cdot \pi^i_\infty$ where:
  1. S is the set of states,
  2. $\{\mathcal{C}_i\}_{i \in I}$ is the set of terminal scc,
  3. π^i_∞ is the steady-state distribution of $\mathcal{C}_i$,
  4. and $\mathtt{preach}_i[s]$ is the probability to reach $\mathcal{C}_i$ starting from s.

Computation of the reachability probability for transient states

- Let T be the set of transient states
  (i.e. not belonging to a terminal scc)
- Let $\mathrm{P}_{T,T}$ be the submatrix of P restricted to transient states
- Let $\mathrm{P}_{T,i}$ be the submatrix of P transitions from T to $\mathcal{C}_i$
- Then $\mathtt{preach}_i = (\sum_{n \in \mathbb{N}} (\mathrm{P}_{T,T})^n) \cdot \mathrm{P}_{T,i} \cdot \mathbf{1} = (Id - \mathrm{P}_{T,T})^{-1} \cdot \mathrm{P}_{T,i} \cdot \mathbf{1}$

Analysis of a DTMC: the "General" Case

Almost general case: every terminal scc is aperiodic

- π_∞ exists.
- $\pi_\infty = \sum_{s \in S} \pi_0(s) \sum_{i \in I} \mathtt{preach}_i[s] \cdot \pi_\infty^i$ where:
  1. S is the set of states,
  2. $\{\mathcal{C}_i\}_{i \in I}$ is the set of terminal scc,
  3. π_∞^i is the steady-state distribution of $\mathcal{C}_i$,
  4. and $\mathtt{preach}_i[s]$ is the probability to reach $\mathcal{C}_i$ starting from s.

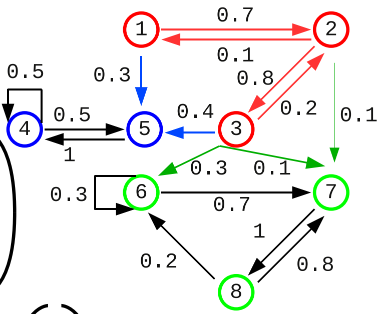
Computation of the reachability probability for transient states

- Let T be the set of transient states
  (i.e. not belonging to a terminal scc)
- Let $\mathtt{P}_{T,T}$ be the submatrix of $\mathtt{P}$ restricted to transient states
- Let $\mathtt{P}_{T,i}$ be the submatrix of $\mathtt{P}$ transitions from T to $\mathcal{C}_i$
- Then $\mathtt{preach}_i = (\sum_{n \in \mathbb{N}} (\mathtt{P}_{T,T})^n) \cdot \mathtt{P}_{T,i} \cdot \mathbf{1} = (Id - \mathtt{P}_{T,T})^{-1} \cdot \mathtt{P}_{T,i} \cdot \mathbf{1}$

Illustration: SCC and Matrices

$T=\{1,2,3\}, C_1=\{4,5\}, C_2=\{6,7,8\}$

$$P_{T,T} = \begin{pmatrix} 0.0 & 0.7 & 0.0 \\ 0.1 & 0.0 & 0.8 \\ 0.0 & 0.2 & 0.0 \end{pmatrix}$$



$$P_{T,1}.\mathbf{1} = \begin{pmatrix} 0.0 & 0.3 \\ 0.0 & 0.0 \\ 0.0 & 0.4 \end{pmatrix} \begin{pmatrix} 1.0 \\ 1.0 \\ 1.0 \end{pmatrix} = \begin{pmatrix} 0.3 \\ 0.0 \\ 0.4 \end{pmatrix}$$

$$P_{T,2}.\mathbf{1} = \begin{pmatrix} 0.0 & 0.0 & 0.0 \\ 0.0 & 0.1 & 0.0 \\ 0.3 & 0.1 & 0.0 \end{pmatrix} \begin{pmatrix} 1.0 \\ 1.0 \\ 1.0 \end{pmatrix} = \begin{pmatrix} 0.0 \\ 0.1 \\ 0.4 \end{pmatrix}$$

Continuous Time Markov Chain (CTMC)

A CTMC is a stochastic process which fulfills:

- Memoryless state change

$$Pr(S_{n+1} = s_j \mid S_0 = s_{i_0}, ..., S_{n-1} = s_{i_{n-1}}, T_0 < \tau_0, ..., T_n < \tau_n, S_n = s_i)$$
$$= Pr(S_{n+1} = s_j \mid S_n = s_i) \equiv \mathtt{P}[i,j]$$

- Memoryless transition delay

$$Pr(T_n < \tau \mid S_0 = s_{i_0}, ..., S_{n-1} = s_{i_{n-1}}, T_0 < \tau_0, ..., T_{n-1} < \tau_{n-1}, S_n = s_i)$$
$$= Pr(T_n < \tau \mid S_n = s_i) = 1 - e^{-\lambda_i \tau}$$

Notations and properties

- $\mathtt{P}$ defines an *embedded* DTMC (the chain of state changes)
- Let $\pi(\tau)$ the distribution de $X(\tau)$, for δ going to 0 the following assertion holds:

$$\pi(\tau+\delta)(s_i) \approx \pi(\tau)(s_i)(1-\lambda_i\delta) + \sum_j \pi(\tau)(s_j)\lambda_j\delta\mathtt{P}[j,i]$$

- Hence, let $\mathtt{Q}$ *the infinitesimal generator* defined by:
$\mathtt{Q}[i,j] \equiv \lambda_i\mathtt{P}[i,j]$ for $j \neq i$ and $\mathtt{Q}[i,i] \equiv -\sum_{j\neq i}\mathtt{Q}[i,j]$
Then: $\frac{d\pi}{d\tau} = \pi \cdot \mathtt{Q}$

Continuous Time Markov Chain (CTMC)

A CTMC is a stochastic process which fulfills:

- Memoryless state change

$$Pr(S_{n+1} = s_j \mid S_0 = s_{i_0}, ..., S_{n-1} = s_{i_{n-1}}, T_0 < \tau_0, ..., T_n < \tau_n, S_n = s_i)$$
$$= Pr(S_{n+1} = s_j \mid S_n = s_i) \equiv \mathtt{P}[i, j]$$

- Memoryless transition delay

$$Pr(T_n < \tau \mid S_0 = s_{i_0}, ..., S_{n-1} = s_{i_{n-1}}, T_0 < \tau_0, ..., T_{n-1} < \tau_{n-1}, S_n = s_i)$$
$$= Pr(T_n < \tau \mid S_n = s_i) = 1 - e^{-\lambda_i \tau}$$

Notations and properties

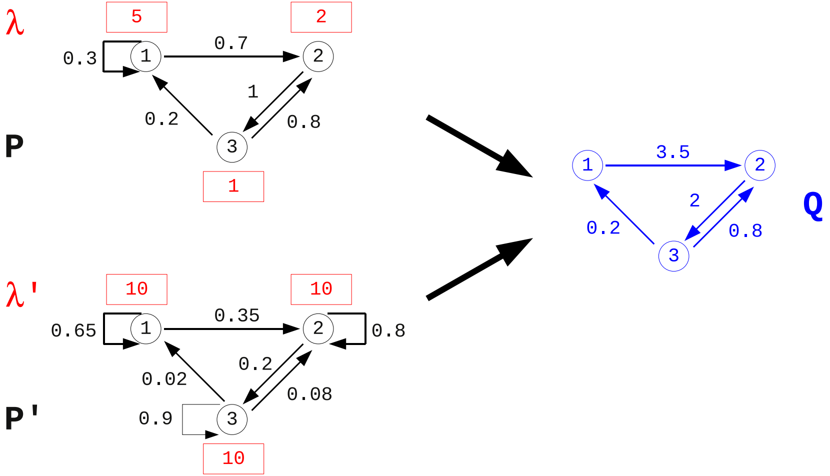
- $\mathtt{P}$ defines an *embedded* DTMC (the chain of state changes)
- Let $\pi(\tau)$ the distribution de $X(\tau)$, for δ going to 0 the following assertion holds:

$$\pi(\tau + \delta)(s_i) \approx \pi(\tau)(s_i)(1 - \lambda_i \delta) + \sum_j \pi(\tau)(s_j)\lambda_j \delta \mathtt{P}[j, i]$$

- Hence, let $\mathtt{Q}$ *the infinitesimal generator* defined by:
  $\mathtt{Q}[i, j] \equiv \lambda_i \mathtt{P}[i, j]$ for $j \neq i$ and $\mathtt{Q}[i, i] \equiv -\sum_{j \neq i} \mathtt{Q}[i, j]$
  Then: $\frac{d\pi}{d\tau} = \pi \cdot \mathtt{Q}$

CTMC: Illustration and Uniformization

A CTMC



A *uniform version* of the CTMC *(equivalent w.r.t.* $X(\tau)$*)*

Analysis of a CTMC

Transient Analysis

- Construction of a uniform version of the CTMC $(\lambda, \mathtt{P})$ such that $\mathtt{P}[i,i] > 0$ for all i.
- Computation by case decomposition w.r.t. the number of transitions:

$$\pi(\tau) = \pi(0) \sum_{n \in \mathbb{N}} (e^{-\lambda\tau}) \frac{\tau^n}{n!} \mathtt{P}^n$$

Steady-state analysis

- The steady-state distribution of visits is given by the steady-state distribution of $(\lambda, \mathtt{P})$ (by construction, the terminal scc are aperiodic) ...
- equal to the steady-state distribution since the sojourn times follow the same distribution.
- A particular case: $\mathtt{P}$ irreducible
  the steady-state distribution π is the unique solution of $X \cdot \mathtt{Q} = 0 \wedge X \cdot \mathbf{1} = 1$
  where one can omit an arbitrary equation of the first system.

Analysis of a CTMC

Transient Analysis

- Construction of a uniform version of the CTMC $(\lambda, \mathtt{P})$ such that $\mathtt{P}[i,i] > 0$ for all i.
- Computation by case decomposition w.r.t. the number of transitions:

$$\pi(\tau) = \pi(0) \sum_{n \in \mathbb{N}} (e^{-\lambda\tau}) \frac{\tau^n}{n!} \mathtt{P}^n$$

Steady-state analysis

- The steady-state distribution of visits is given by the steady-state distribution of $(\lambda, \mathtt{P})$ (by construction, the terminal scc are aperiodic) ...
- equal to the steady-state distribution since the sojourn times follow the same distribution.
- A particular case: $\mathtt{P}$ irreducible
the steady-state distribution π is the unique solution of $X \cdot \mathtt{Q} = 0 \wedge X \cdot \mathbf{1} = 1$ where one can omit an arbitrary equation of the first system.

Markovian Renewal Process

A Markovian Renewal Process (MRP) fulfills:

- a *relative* memoryless property

$$Pr(S_{n+1} = s_j, T_n < \tau \mid S_0 = s_{i_0}, ..., S_{n-1} = s_{i_{n-1}}, T_0 < \tau_0, ..., S_n = s_i)$$
$$= Pr(S_{n+1} = s_j, T_n < \tau \mid S_n = s_i) \equiv \mathtt{Q}[i, j, \tau]$$

- The embedded chain is defined by: $\mathtt{P}[i, j] = \lim_{\tau \to \infty} \mathtt{Q}[i, j, \tau]$
- The sojourn time `Soj` has a distribution defined by:

$$Pr(\mathtt{Soj}[i] < \tau) = \sum_j \mathtt{Q}[i, j, \tau]$$

Analysis of a MRP

- The steady-state distribution (if there exists) π is deduced from the steady-state distribution of the embedded chain π' by:

$$\pi(s_i) = \frac{\pi'(s_i) E(\mathtt{Soj}[i])}{\sum_j \pi'(s_j) E(\mathtt{Soj}[j])}$$

- Transient analysis is much harder ... but the reachability probabilities only depend on the embedded chain.

Markovian Renewal Process

A Markovian Renewal Process (MRP) fulfills:

- a *relative* memoryless property

$$Pr(S_{n+1} = s_j, T_n < \tau \mid S_0 = s_{i_0}, ..., S_{n-1} = s_{i_{n-1}}, T_0 < \tau_0, ..., S_n = s_i) \\ = Pr(S_{n+1} = s_j, T_n < \tau \mid S_n = s_i) \equiv \mathtt{Q}[i, j, \tau]$$

- The embedded chain is defined by: $\mathtt{P}[i, j] = \lim_{\tau \to \infty} \mathtt{Q}[i, j, \tau]$
- The sojourn time Soj has a distribution defined by:

$$Pr(\mathtt{Soj}[i] < \tau) = \sum_j \mathtt{Q}[i, j, \tau]$$

Analysis of a MRP

- The steady-state distribution (if there exists) π is deduced from the steady-state distribution of the embedded chain π' by:

$$\pi(s_i) = \frac{\pi'(s_i) E(\mathtt{Soj}[i])}{\sum_j \pi'(s_j) E(\mathtt{Soj}[j])}$$

- Transient analysis is much harder ... but the reachability probabilities only depend on the embedded chain.

Plan

Stochastic Processes and Markov Chains

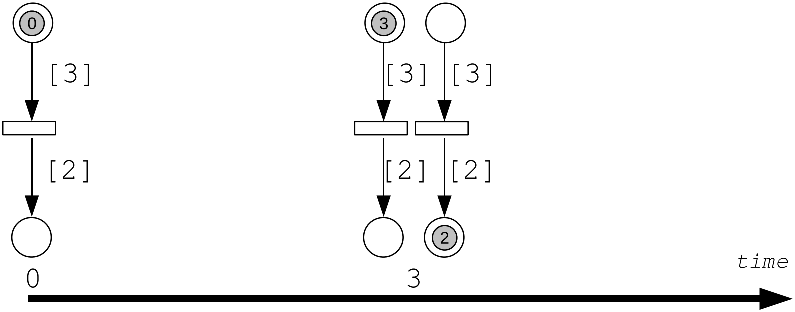
2. A Semantic for Stochastic Petri Nets

Time and Probability in Petri Nets

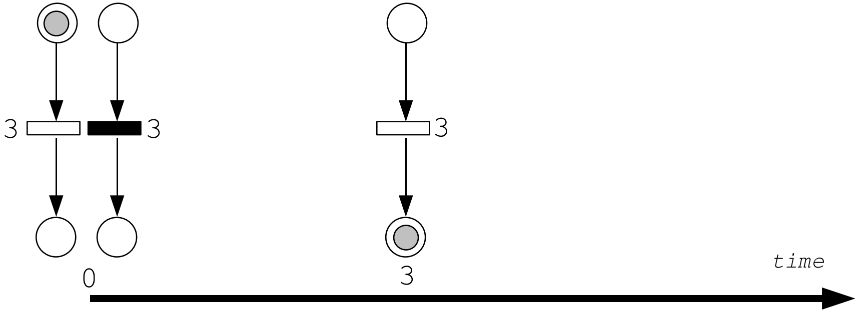
How to introduce time in nets?

- Age of a token
- Firing duration of a transition
- etc.
- Firing delay of a transition with instantaneous firing

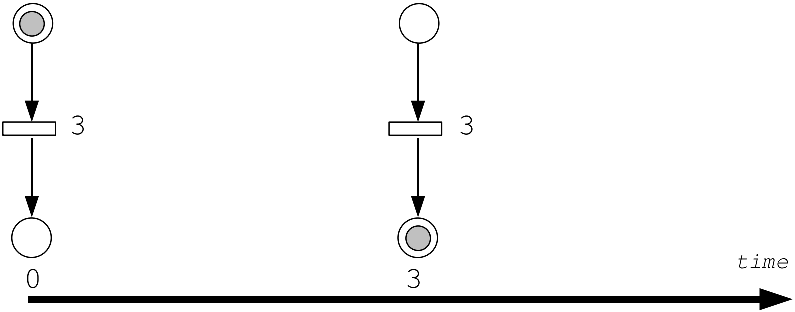
A Token-Based Semantic



A Duration-Based Semantic



A Delay-Based Semantic



A Semantic for Stochastic Petri Nets

- The initial distribution is concentrated on the the initial marking
- A distribution with outcomes in $\mathbb{R}_{\geq 0}$ is associated with every transition.

... but these distributions are not sufficient to define a stochastic process.

Policies for a net

One needs to define:

- The *choice* policy.
  What is the next transition to fire?
- The *service* policy.
  What is the influence of the enabling degree of a transition on the process?
- The *memory* policy.
  What become the samplings of distributions that have not be used?

A Semantic for Stochastic Petri Nets

- The initial distribution is concentrated on the the initial marking
- A distribution with outcomes in $\mathbb{R}_{\geq 0}$ is associated with every transition.

... but these distributions are not sufficient to define a stochastic process.

Policies for a net

One needs to define:

- The *choice* policy.
  What is the next transition to fire?
- The *service* policy.
  What is the influence of the enabling degree of a transition on the process?
- The *memory* policy.
  What become the samplings of distributions that have not be used?

Choice Policy

In the net, associate a distribution D_i and a weight w_i with every transition t_i

Preselection w.r.t. a marking m

- Normalize weights w_i of the enabled transitions s.t. $w'_i \equiv w_i / (\sum_{m[t_j\rangle} w_j)$
- Sample the distribution defined by the w'_i's. Let t_i be the selected transition
- Sample the distribution D_i giving the value d_i.

versus

Race policy with postselection w.r.t. a marking m

- For every enabled transition t_i, sample the distribution D_i giving the value d_i.
- Let T' be the subset of enabled transitions with the smallest delays. Normalize weights w_i of transitions of T' s.t. $w'_i \equiv w_i / (\sum_{t_j \in T'} w_j)$
- Sample the distribution defined by the w'_i's. Let t_i be the selected transition. *(postselection can also be handled with priorities)*

Then t_i is the next transition to fire with delay d_i.

Choice Policy

In the net, associate a distribution D_i and a weight w_i with every transition t_i

Preselection w.r.t. a marking m

- Normalize weights w_i of the enabled transitions s.t. $w'_i \equiv w_i / (\sum_{m[t_j\rangle} w_j)$
- Sample the distribution defined by the w'_i's. Let t_i be the selected transition
- Sample the distribution D_i giving the value d_i.

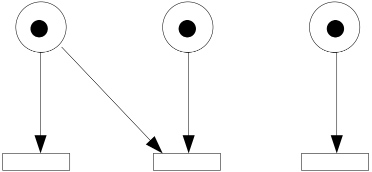
versus

Race policy with postselection w.r.t. a marking m

- For every enabled transition t_i, sample the distribution D_i giving the value d_i.
- Let T' be the subset of enabled transitions with the smallest delays. Normalize weights w_i of transitions of T' s.t. $w'_i \equiv w_i / (\sum_{t_j \in T'} w_j)$
- Sample the distribution defined by the w'_i's. Let t_i be the selected transition. *(postselection can also be handled with priorities)*

Then t_i is the next transition to fire with delay d_i.

Choice Policy: Illustration

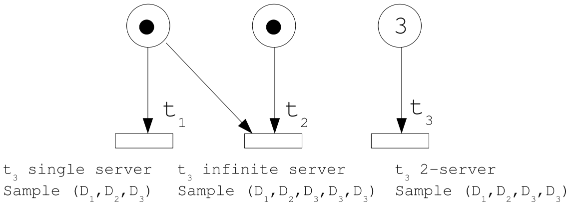


Preselection	Race Policy
Sample (1/5,2/5,2/5)	Sample (D_1,D_2,D_3)
Outcome $\mathbf{t_1}$	Outcome(**3.2**,6.5,**3.2**)
Sample D_1	Sample(1/3,-,2/3)
Outcome **4.2**	Outcome $\mathbf{t_3}$

Server Policy

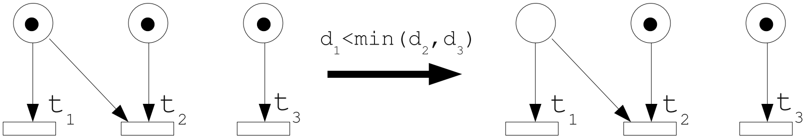
A transition can be viewed as server for firings:

- A *single server* transition t allows a single instance of firings in m if $m[t\rangle$.
- An *infinite server* transition t allows d (*the enabling degree*) instances of firings in m where $d = \min(\left\lfloor \frac{m(p)}{Pre(p,t)} \right\rfloor \mid p \in {}^{\bullet}t)$.
- A *multiple server* transition t with bound b allows $\min(b, d)$ instances of firings in m.



This can be generalised by marking-dependent rates *(see the next talk)*.

Memory Policy (1)



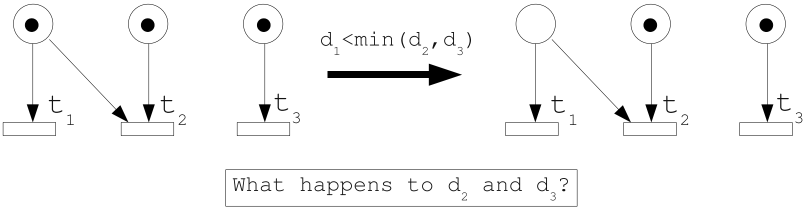
What happens to d_2 and d_3?

Resampling Memory

Every sampling not used is forgotten.

This could correspond to a “crash” transition.

Memory Policy (2)

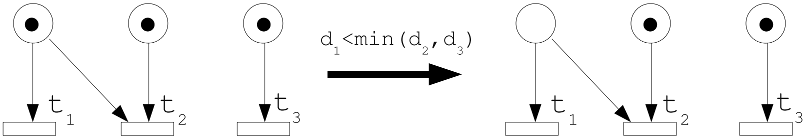


Enabling Memory

- The samplings associated with still enabled transitions are kept and decremented ($d_3' = d_3 - d_1$).
- The samplings associated with disabled transitions are forgotten (like d_2).

Disabling a transition could correspond to abort a service.

Memory Policy (3)



What happens to d_2 and d_3?

Age Memory

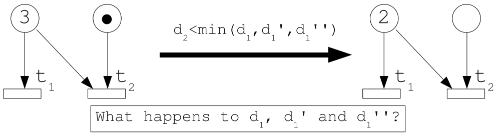
- All the samplings are kept and decremented ($d'_3 = d_3 - d_1 \;\; d'_2 = d_2 - d_1$).
- The sampling associated with a disabled transition is frozen until the transition become again enabled (like d'_2).

Disabling a transition could correspond to suspend a service.

Memory Policy (4)

Specification of memory policy

To be fully expressive, it should be defined w.r.t. any pair of transitions.



Interaction between memory policy and service policy

Assume enabling memory for t_1 when firing t_2 and infinite server policy for t_1. Which sample should be forgotten?

- The last sample performed,
- The first sample performed,
- The greatest sample, etc.

Warning: This choice may have a critical impact on the complexity of analysis.

Nets with Exponential Distributions

Hypotheses

The distribution of every transition t_i is an exponential distribution with density function $e^{-\lambda_i \tau}$ where the parameter λ_i is called *the rate* of the transition.

Observations

Given a marking m with transitions $t_1, \dots, t_k$ serving $n_1, \dots, n_k$ firings (depending on the service policy):

- The sojourn time in m is an exponential distribution with rate $\sum_i n_i \lambda_i$.
- The probability that t_i is the next transition to fire is $n_i \lambda_i / (\sum_j n_j \lambda_j)$.
- The residual time of a delay $d_j - d_i$ of transition t_j knowing that t_i has fired and that d_i is the shortest delay has for density function $e^{-\lambda_j \tau}$, the same as the initial delay. Thus the memory policy is irrelevant.
- The weights are not required since equality of two samples has a null probability (due to continuity of distributions).

The stochastic process is a CTMC whose states are markings and whose transitions are the transitions of the reachability graph allowing standard analysis methods.

Nets with Exponential Distributions

Hypotheses

The distribution of every transition t_i is an exponential distribution with density function $e^{-\lambda_i \tau}$ where the parameter λ_i is called *the rate* of the transition.

Observations

Given a marking m with transitions $t_1, \ldots, t_k$ serving $n_1, \ldots, n_k$ firings (depending on the service policy):

- The sojourn time in m is an exponential distribution with rate $\sum_i n_i \lambda_i$.
- The probability that t_i is the next transition to fire is $n_i \lambda_i / (\sum_j n_j \lambda_j)$.
- The residual time of a delay $d_j - d_i$ of transition t_j knowing that t_i has fired and that d_i is the shortest delay has for density function $e^{-\lambda_j \tau}$, the same as the initial delay. Thus the memory policy is irrelevant.
- The weights are not required since equality of two samples has a null probability (due to continuity of distributions).

The stochastic process is a CTMC whose states are markings and whose transitions are the transitions of the reachability graph allowing standard analysis methods.

Nets with Exponential Distributions

Hypotheses

The distribution of every transition t_i is an exponential distribution with density function $e^{-\lambda_i \tau}$ where the parameter λ_i is called *the rate* of the transition.

Observations

Given a marking m with transitions $t_1, \ldots, t_k$ serving $n_1, \ldots, n_k$ firings (depending on the service policy):

- The sojourn time in m is an exponential distribution with rate $\sum_i n_i \lambda_i$.
- The probability that t_i is the next transition to fire is $n_i \lambda_i / (\sum_j n_j \lambda_j)$.
- The residual time of a delay $d_j - d_i$ of transition t_j knowing that t_i has fired and that d_i is the shortest delay has for density function $e^{-\lambda_j \tau}$, the same as the initial delay. Thus the memory policy is irrelevant.
- The weights are not required since equality of two samples has a null probability (due to continuity of distributions).

The stochastic process is a CTMC whose states are markings and whose transitions are the transitions of the reachability graph allowing standard analysis methods.

Generalizing Distributions for Nets

Observations

Modelling delays with exponential distributions is **reasonable** when:

- Only mean value information is known about distributions.
- Exponential distributions (or combination of them) are enough to approximate the "real" distributions.

Modelling delays with exponential distributions is **not reasonable** when:

- The distribution of an event is known and is poorly approximable with exponential distributions like a time-out of 10 time units. (see phase-type SPNs in the third talk)
- The delays of the events have different magnitude orders like executing an instruction versus performing a database request. In this case, the 0-Dirac distribution is required.

Generalized Stochastic Petri Nets (GSPN) are nets whose *timed transitions* have exponential distributions and *immediate transitions* have 0-Dirac distributions.

Generalizing Distributions for Nets

Observations

Modelling delays with exponential distributions is **reasonable** when:

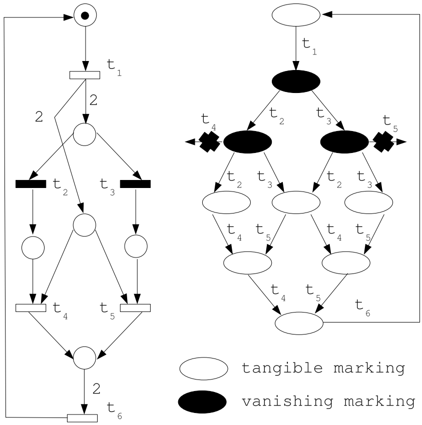
- Only mean value information is known about distributions.
- Exponential distributions (or combination of them) are enough to approximate the "real" distributions.

Modelling delays with exponential distributions is **not reasonable** when:

- The distribution of an event is known and is poorly approximable with exponential distributions like a time-out of 10 time units.
  (see phase-type SPNs in the third talk)
- The delays of the events have different magnitude orders like executing an instruction versus performing a database request.
  In this case, the 0-Dirac distribution is required.

Generalized Stochastic Petri Nets (GSPN) are nets whose *timed transitions* have exponential distributions and *immediate transitions* have 0-Dirac distributions.

A GSPN is a Markovian Renewal Process



Observations

- Weights are required for immediate transitions.
- The *restricted* reachability graph corresponds to the embedded DTMC.

Steady-State Analysis of a GSPN

Standard method for MRP

- Build the restricted reachability graph equivalent to the embedded DTMC and deduce the probability matrix P
- Compute π^* the steady-state distribution of the visits of markings: $\pi^* = \pi^* P$.
- Compute π the steady-state distribution of the sojourn in tangible markings: $\pi(m) = \pi^*(m)\texttt{Soj}(m) / \sum_{m' \text{ tangible}} \pi^*(m')\texttt{Soj}(m')$.

How to eliminate the vanishing markings sooner in the computation?

Alternative method for this particular case

- As before, compute the transition probability matrix P
- Compute the transition probability matrix P' between tangible markings.
- Compute π'^* the (relative) steady-state distribution of the visits of tangible markings: $\pi'^* = \pi'^* P'$.
- Compute π the steady-state distribution of the sojourn in tangible markings: $\pi(m) = \pi'^*(m)\texttt{Soj}(m) / \sum_{m' \text{ tangible}} \pi'^*(m')\texttt{Soj}(m')$.

Steady-State Analysis of a GSPN

Standard method for MRP

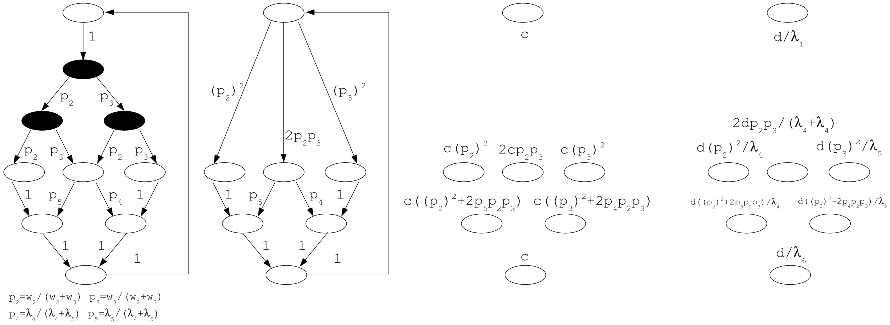
- Build the restricted reachability graph equivalent to the embedded DTMC and deduce the probability matrix P
- Compute π^* the steady-state distribution of the visits of markings: $\pi^* = \pi^* P$.
- Compute π the steady-state distribution of the sojourn in tangible markings: $\pi(m) = \pi^*(m)\texttt{Soj}(m) / \sum_{m' \text{ tangible}} \pi^*(m')\texttt{Soj}(m')$.

How to eliminate the vanishing markings sooner in the computation?

Alternative method for this particular case

- As before, compute the transition probability matrix P
- Compute the transition probability matrix P' between tangible markings.
- Compute π'^* the (relative) steady-state distribution of the visits of tangible markings: $\pi'^* = \pi'^* P'$.
- Compute π the steady-state distribution of the sojourn in tangible markings: $\pi(m) = \pi'^*(m)\texttt{Soj}(m) / \sum_{m' \text{ tangible}} \pi'^*(m')\texttt{Soj}(m')$.

Steady-State Analysis: Illustration



Computation of P'

- Let $P_{X,Y}$ the probability transition matrix from subset X to subset Y. Let V (resp. T) be the set of vanishing (resp. tangible) markings.
- $P' = P_{T,T} + P_{T,V}(\sum_{n\in\mathbb{N}} P^n_{V,V})P_{V,T} = P_{T,T} + P_{T,V}(Id - P_{V,V})^{-1}P_{V,T}$
- Iterative (resp. direct) computations uses the first (resp. second) expression.

Some References

M. Ajmone Marsan, G. Balbo, G. Conte, S. Donatelli, G. Franceschinis

Modelling with Generalized Stochastic Petri Nets
Wiley series in parallel computing, John Wiley & Sons, 1995
Freely available on the Web site of GreatSPN

S. Haddad, P. Moreaux

Chapter 7: Stochastic Petri Nets
Petri Nets: Fundamental Models and Applications Wiley pp. 269-302, 2009