Fluid models in performance analysis

Miklós Telek

Dept. of Telecom., Technical University of Budapest

SFM-07:PE,

May 31, 2007

Bertinoro, Italy

Joint work with

Marco Gribaudo

Dip. di Informatica, Università di Torino,

Outline

Outline

1. Motivations

- Conventional modelling techniques have some limitations due to:

  - State space explosion,
  - Granularity and size,
  - Modelling power limitations,
  - Inaccurate distribution approximation.

- Continuous modelling techniques can help (in some cases) to overcome these limitations!

1. Motivations: State space explosion

- The size of the state space of a model generally grows exponentially by increasing its parameters (i.e. increasing the number of costumer in a non-product form queuing network).

- This size can reach very quickly the storage and processing capacity of a machine.

- Fluid techniques use additional continuous variables which are not part of the conventional state space, leading to smaller sets.

1. Motivations: Granularity and Size

- Many systems are characterized by huge amount of very small elements (i.e. the packets in a broadband router, raw parts in a flexible manufactory system).

- Continuous variables may very naturally approximate these large discrete numbers.

1. Motivations: Modelling power

- In some cases, physical quantities like time, temperature, or speed must be modelled explicitly.

- Conventional modelling technique "discretize" those quantities by choosing a finite set of possible values.

- Continuous variables can instead exactly model these quantities.

1. Motivations: Inaccurate approximations

- Many conventional modelling techniques relay only on Exponential distributions and homogeneous Poission processes (i.e. Markov Chains, GSPNs).

- Fluid model can directly embed more complex distributions and non-homogenous Poission process, without the need of using approximate techniques (like Phase-Type expansion).

2. Formalisms

- Continuous quantities have been introduced in performance models in many flavors.

- Many high-level and low-level performance evaluation formalisms have been developed to deal with continuous quantities. Here we will consider:

• Reward Models,

• Fluid Models,

• Fluid Stochastic Petri Nets.

2.1. Reward Models

- A **Reward Model** is a Markov chain in which each state has associated a positive quantity called *Reward Rate*.

- Reward is accumulated proportionally to the time spent in a state and to the corresponding reward rate.

- The accumulated reward is unbounded.

2.1. Reward Models

- The Markov Chain that governs the reward is called the *underlaying Markov Chain*.

- It is described by a generator matrix $\mathbf{Q}$, whose element q_{ij} defines the transition from state i to state j:

$$
\begin{aligned}
q_{ij} &= \lim_{\Delta t \to 0} \frac{P\{S(t+\Delta t) = j | S(t) = i\}}{\Delta t}, \text{ for } i \neq j \\
q_{ij} &= -\sum_{k \neq i} q_{ik}, \text{ for } i = j
\end{aligned}
$$

- $S(t)$ represents the state of the underlaying Markov chain at time t.

2.1. Reward Models

- The *reward rate* of the state i is denoted by r_i, $r_i \geq 0$.

- They are collected in a diagonal Matrix R, whose elements $[R]_{ij}$ are such that:

$$\begin{aligned} [R]_{ij} &= 0, && \text{for } i \neq j, \\ [R]_{ij} &= r_i, && \text{for } i = j. \end{aligned}$$

2.1. Reward Models

- We denote with $X(t)$ the total reward accumulated until time t.

- We set $X(0) = 0$.

- In this case

$$X(t) \quad = \quad \int_0^t r_{S(u)}\, du.$$

2.2. Fluid Models

- **Fluid Models** are an extension of Reward Models.

- The rate associated to each state (called in this case *flow rate or drift*) can be positive, negative or zero.

- The accumulated reward is called *Fluid Level*.

- The Fluid level has at least a lower bound at zero.

- The analysis techniques for Reward and Fluid Models will be presented in Part 3 and 4 of this tutorial.

2.2. Fluid Models

- Fluid Models are described by the same parameters used for Reward Models:

- The Transition Rate Matrix **Q**.

- The Flow Rate Matrix **R**.

- The Flow Rate Matrix is equivalent to the Reward Rate Matrix, without the restriction that its elements must be positive.

2.3. Fluid Stochastic Petri Nets

- A **Fluid Stochastic Petri Net** (**FSPN**) is an extension of an ordinary Stochastic Petri Net, capable of incorporating continuous quantities.

- Other similar extensions with minor differences are: *Continuous Petri Nets* and *Hybrid Petri Nets*. In this tutorial we will not consider such formalisms.

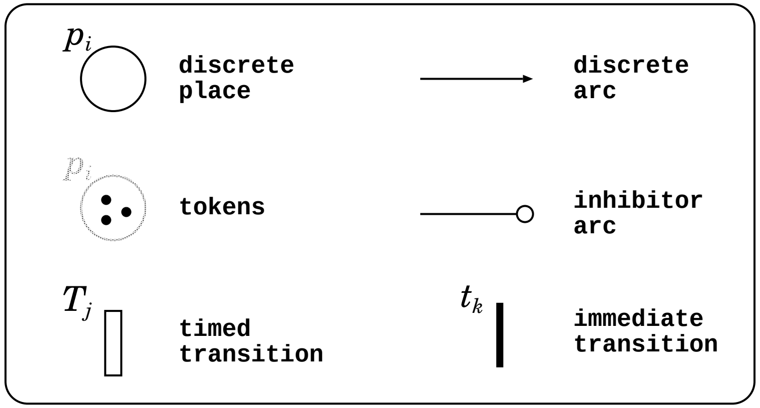
- We will present the basic formalism, intended for stochastic analysis (not simulation).

2.3. Fluid Stochastic Petri Nets

- The modelling primitives that can be used in a FSPN model are divided into two categories:

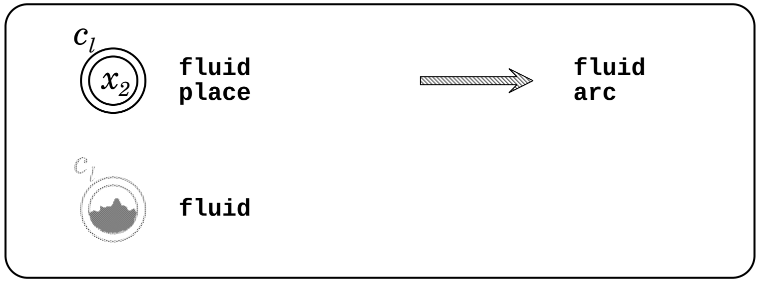
- Discrete primitives,
- Fluid (Continuous) primitives.

2.3. Fluid Stochastic Petri Nets



- *Discrete primitives* are identical to the equivalent primitives of a Generalized Stochastic Petri Net (GSPN).

2.3. Fluid Stochastic Petri Nets



- *Fluid primitives* are instead specific for FSPN.

2.3. Fluid Stochastic Petri Nets

- The *Discrete Part of a model* is the subset of the model by all and only its discrete primitives.

- The *Fluid Part of a model* is the subset of the model composed by all and only its fluid primitives.

- It can be easily shown that the Discrete Part of a FSPN is a GSPN.

2.3. Fluid Stochastic Petri Nets

- *Discrete Places* contain *Tokens*.

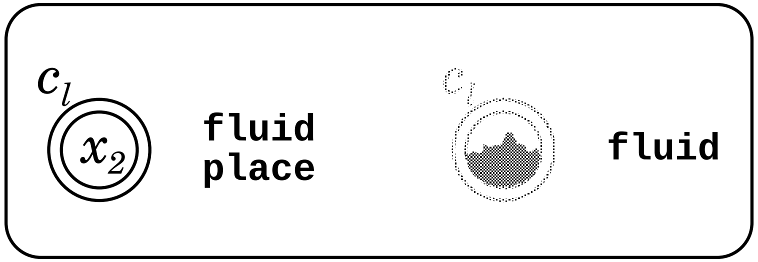
- The number of tokens contained in a Discrete Place represent its *Marking*.

- The *Discrete Marking* of a discrete place is a natural number.

2.3. Fluid Stochastic Petri Nets

- We call P_d the set of discrete places.
- We indicate with $p_i \in P_d$ an element of this set.
- We denote with m_i the discrete marking of place p_i.

2.3. Fluid Stochastic Petri Nets

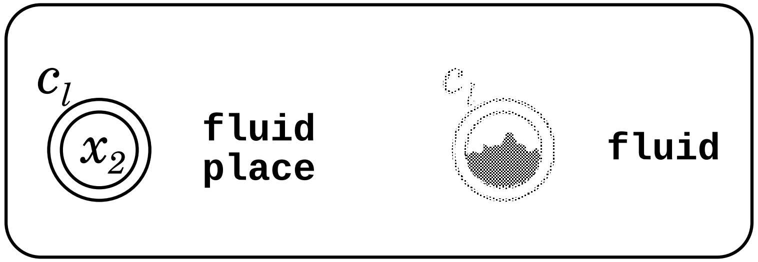


- *Fluid (or Continuous) Places* contain a continuous quantity called *Fluid*.

- This corresponds to the *Marking* for a Fluid Place.

- The *Fluid Marking (Fluid Level)* is a non-negative real number.

2.3. Fluid Stochastic Petri Nets



- We call P_c the set of fluid places.
- We indicate with $c_l \in P_c$ an element of this set.
- We denote with x_l the fluid level of place c_l.

2.3. Fluid Stochastic Petri Nets

- Markings of Discrete Places are collected in a vector of $|P_d|$ natural numbers, $\mathbf{m} = (\mathbf{m_1}, \ldots, \mathbf{m_{|P_d|}})$.

- Markings of Fluid Places are collected in a vector of $|P_c|$ real numbers, $\mathbf{x} = (\mathbf{x_1}, \ldots, \mathbf{x_{|P_c|}})$.

- The *Complete Marking* $M = (\mathbf{m}, \mathbf{x})$ of the model is the set of both the Discrete and Fluid Markings.

2.3. Fluid Stochastic Petri Nets

- The Marking $M = (\mathbf{m}, \mathbf{x})$ evolves in time.

- We denote with $m_i(t)$ and $x_l(t)$ respectively the discrete marking of place p_i at time t, and the fluid level of place c_l at time t.

- We call $M(t) = \{(m_i(t), x_l(t)), t \leq 0\}$ the stochastic process that defines the model evolution (the *Marking Process*).

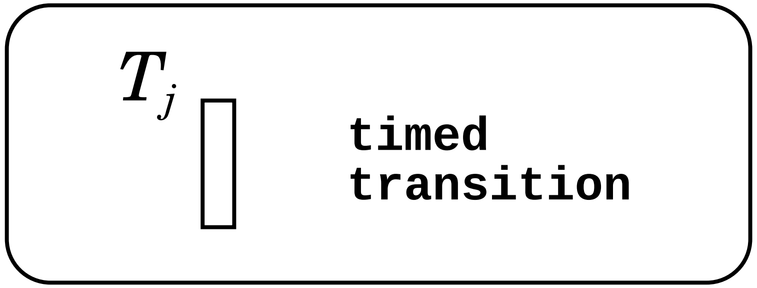
2.3. Fluid Stochastic Petri Nets

- Places (both Fluid and Discrete) are characterized by an *Initial Marking*.

- The *Initial Marking* of a place represents its marking at time $t = 0$.

- We call it $M_0 = (\mathbf{m_0}, \mathbf{x_0})$.

2.3. Fluid Stochastic Petri Nets

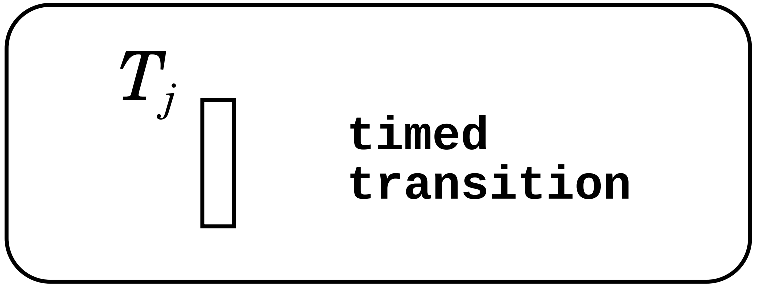


- *Timed Transitions* represents events that happens with time.

- They move tokens among the discrete places.

- They move fluid along the fluid places.

2.3. Fluid Stochastic Petri Nets



- We call $\mathcal{T}_e$ the set of the timed transitions.

- We address with $T_j \in \mathcal{T}_e$ a transition of this set.

2.3. Fluid Stochastic Petri Nets

- *Timed Transitions* can be *enabled*, depending on the marking of the places and on the weights of the discrete and inhibitor arcs that are connected to it.

- When a Timed Transition T_j is enabled, it *fires* after an exponentially distributed time.

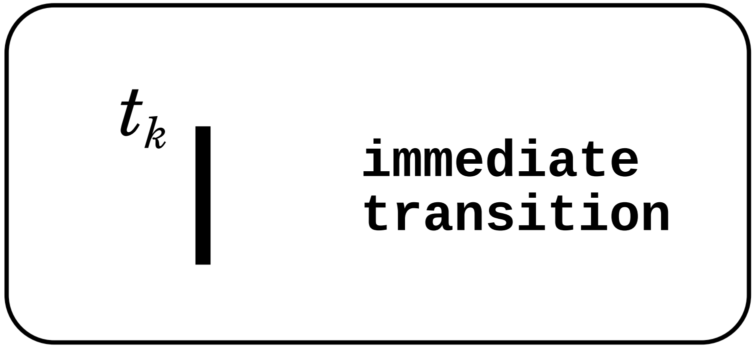
- We denote with $F(T_j, M)$ the *Instantaneous Firing Rate* of transition T_j in marking M (that is the rate parameter of the exponential distribution of the transition firing time).

2.3. Fluid Stochastic Petri Nets

- An enabled timed transition T_j changes the marking of the discrete places to which it is connected with discrete arcs *when it fires*.

- An enabled timed transition T_j continuously changes the marking of the fluid places to which it is connected with fluid arcs *as long as it is enabled*.

2.3. Fluid Stochastic Petri Nets

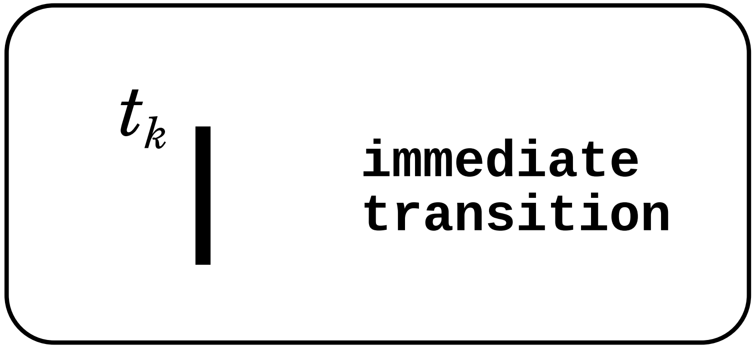


- *Immediate Transitions* represents events that happens in zero time.

- They move tokens among discrete places.

- They cannot change the fluid level in continuous places.

2.3. Fluid Stochastic Petri Nets



- We call $\mathcal{T}_i$ the set of the immediate transitions.

- We address with $t_k \in \mathcal{T}_i$ a transition of this set.

2.3. Fluid Stochastic Petri Nets

- *Immediate Transitions* can be *enabled*, following the same rules as timed transitions.

- Immediate transitions are characterized by their *Weight*, which is used to determine which transition will fire when more than one are enabled at the same time.

- We denote with $W(t_k, M)$ the *Weight* of immediate transition t_k in marking M.

2.3. Fluid Stochastic Petri Nets

- When more than one transition (timed or immediate) are enabled in a marking, a *conflict* arises.

- The conflict resolution algorithm determines which transition actually fires.

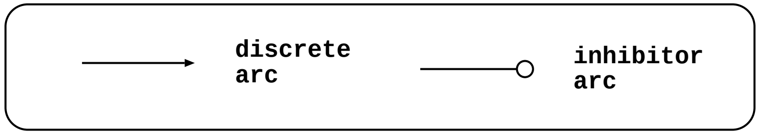
2.3. Fluid Stochastic Petri Nets

- If both timed and immediate transitions are enabled in a marking, immediate transitions have priority over the timed ones (i.e. timed transitions can be ignored).

- *Race policy* solves conflict among timed transition (whichever fires first).

- *Probabilistic* decision, based on the transition weights, determines which fires among several immediate transition concurrently enabled.

2.3. Fluid Stochastic Petri Nets



- *Discrete Arcs* and *Inhibitor Arcs* connect discrete places to transitions.

- They determine when a transition is enabled.

- Discrete Arcs also determine what happens when a transition fires.

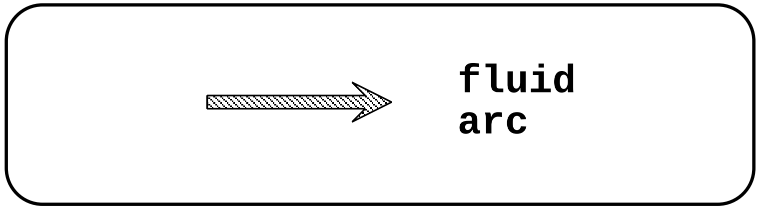
2.3. Fluid Stochastic Petri Nets

- Each *Discrete Arc* or *Inhibitor Arcs* has associated a weight.

- The standard GSPNs: firing rules apply to FSPNs:

  - A transition is enabled if all its input places have at least as many tokens as the weight of the corresponding arc.
  - A transition is enabled if all the places to which it is connected with inhibitor arcs have at most as many tokens as the weight of the connecting arc, minus one.
  - When a transition fires, it removes from its input places and it puts into the output places as many tokens as the corresponding connecting arc.

2.3. Fluid Stochastic Petri Nets

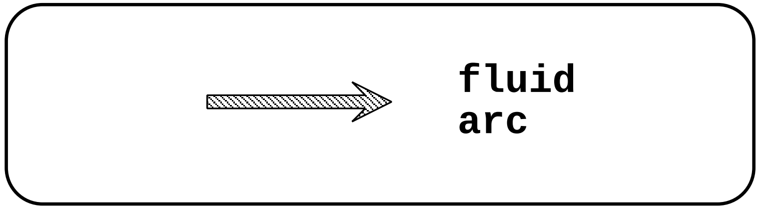


- *Fluid (continuous) Arcs* connect timed transitions to fluid places.

- Each Fluid arc has associated a *Flow Rate*.

- The Flow Rate is a non-negative real number.

2.3. Fluid Stochastic Petri Nets



- A fluid arc directed form a timed transition to a fluid place, pumps fluid into the place at a rate equal to the arc's Flow Rate.

- A fluid arc directed form a fluid place to a timed transition, removes fluid from the place at a rate equal to the arc's Flow Rate.

- Fluid flows only when the Timed Transition at beginning or at the end of the arc is enabled.

2.3. Fluid Stochastic Petri Nets

- When the fluid place becomes *empty* (its fluid marking reaches zero), the fluid flow stop.

- We denote with $R(T_j, c_l, M)$ the flow rate of a fluid arc from timed transition T_j to fluid place c_l in marking M.

- We use $R(c_l, T_j, M)$ when the arc is directed in the opposite direction (from the place to the transition).

2.3. Fluid Stochastic Petri Nets

- Fluid Stochastic Petri Nets are analyzed by transforming them into equivalent Fluid Models.

- If the FSPN has a single fluid place, then standard FM can be applied.

- If the FSPN has more than one fluid place, then special FM with multiple continuous variables must be used.

2.3. Fluid Stochastic Petri Nets

- The Transition Rate Matrix **Q** and the Flow Rate Matrix **R** can be automatically generated starting from the FSPN Model.

- The state space of the FM corresponds to the state space of the discrete part of the FSPN model.

- Both the state space and the Transition Rate Matrix **Q** can be calculated applying standard GSPN techniques to the discrete part of FSPN the model.

2.3. Fluid Stochastic Petri Nets

- The elements of the flow rate matrix can be computed from the flow rates of the fluid arcs. If we imagine to have only one single fluid place c_l, then we can define:

$$r_i = \sum_{T_j \in \mathcal{E}(\mathbf{m_i})} \left(R(T_j, c_l, \mathbf{m}_i) - R(T_j, c_l, \mathbf{m_i}) \right)$$

$\mathbf{m}_i$ is the discrete marking corresponding to state i, and $\mathcal{E}(\mathbf{m_i})$ is the set of timed transition enabled in $\mathbf{m}_i$.

2.3. Fluid Stochastic Petri Nets

- Some important extensions have been proposed in the literature. Two of them are:

- Fluid-dependent transition and flow rates,
- Flush-out arcs.

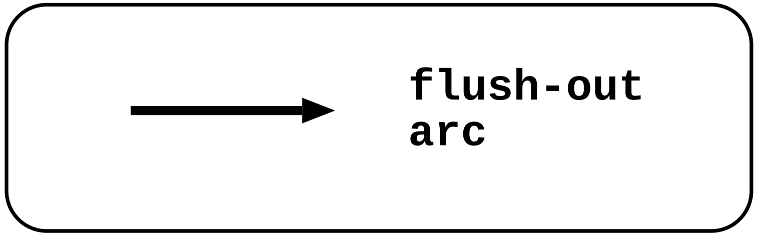
2.3. Fluid Stochastic Petri Nets

- Both the transition rates of timed transition, and the flow rates associated to the fluid arcs can depend on the complete (discrete and continuous) marking of the process.

- In this case the underlaying stochastic process should be analyzed using *non-homogenous Fluid Models*.

- Both the transition rate matrix and the flow rate matrix depend on the fluid part of the model: $\mathbf{Q}(\mathbf{x})$, $\mathbf{R}(\mathbf{x})$.

2.3. Fluid Stochastic Petri Nets

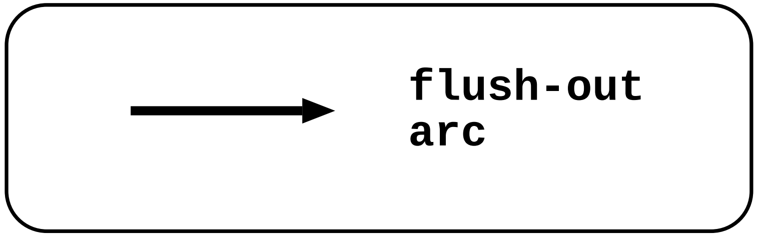


- Flush-out arcs are special arcs that connect fluid places to timed transition (but not timed transition to fluid places).

- They are drawn using thick lines.

- When a transition fires, the places connected with a flush-out arc are emptied in zero time.

2.3. Fluid Stochastic Petri Nets



- Despite their simplicity, Flush-out Arcs can be exploited to obtain many interesting effects, like dropping the content of the transmission buffer.

- The underlaying stochastic model is no longer a standard Fluid Model, but it can be analyzed similarly using appropriate boundary conditions.

3.1 Introduction to fluid models

Continuous time stochastic processes with

- discrete value (state),
  e.g. CTMC,
- continuous value,
  e.g. unfinished work in a queue,
- hybrid (continuous and discrete) value,
  e.g. unfinished work and the number of customers.

3.1 Introduction to fluid models

General continuous and hybrid valued stochastic processes are hard to analyze.

But, there are special cases:

- reward models,
- fluid models.

A simple function of a discrete state stochastic process governs the evolution of the continuous variable.

When the discrete state stochastic process is a CTMC

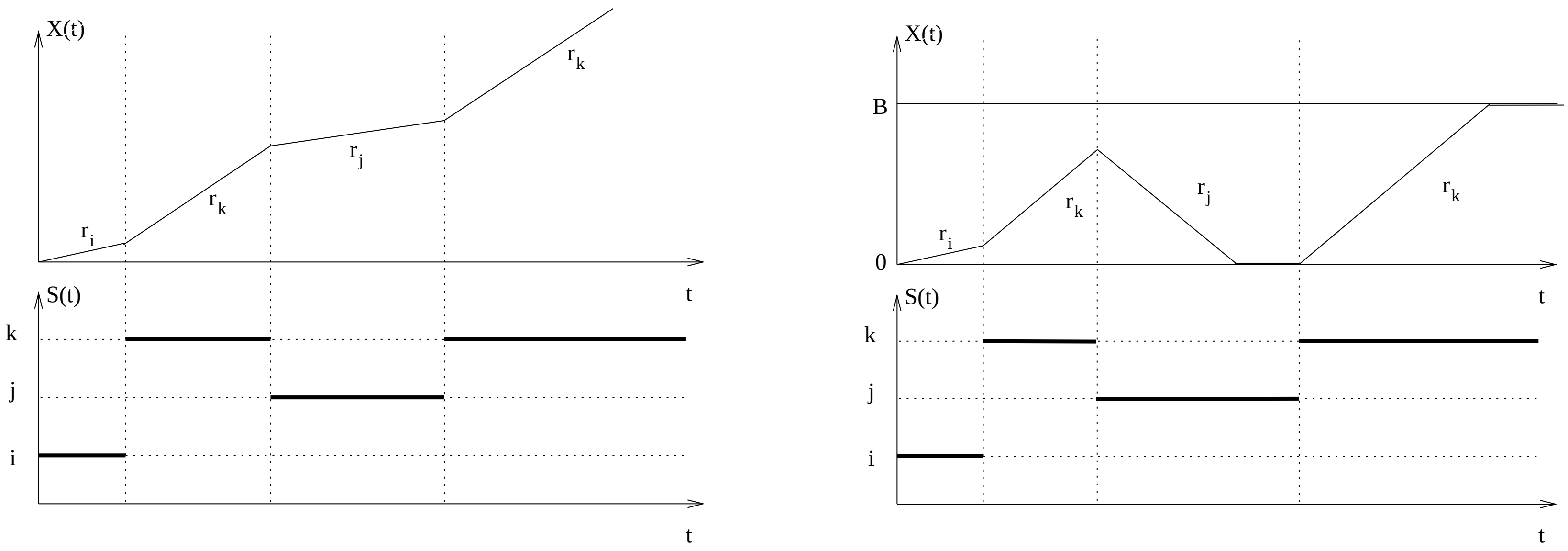
- Markov reward models,
- Markov fluid models.

We focus on this Markovian case.

3.1 Introduction to fluid models

<u>Reward models:</u> unbounded (non-decreasing) evolution,

<u>Fluid models:</u> bounded evolution.



3.1 Introduction to fluid models

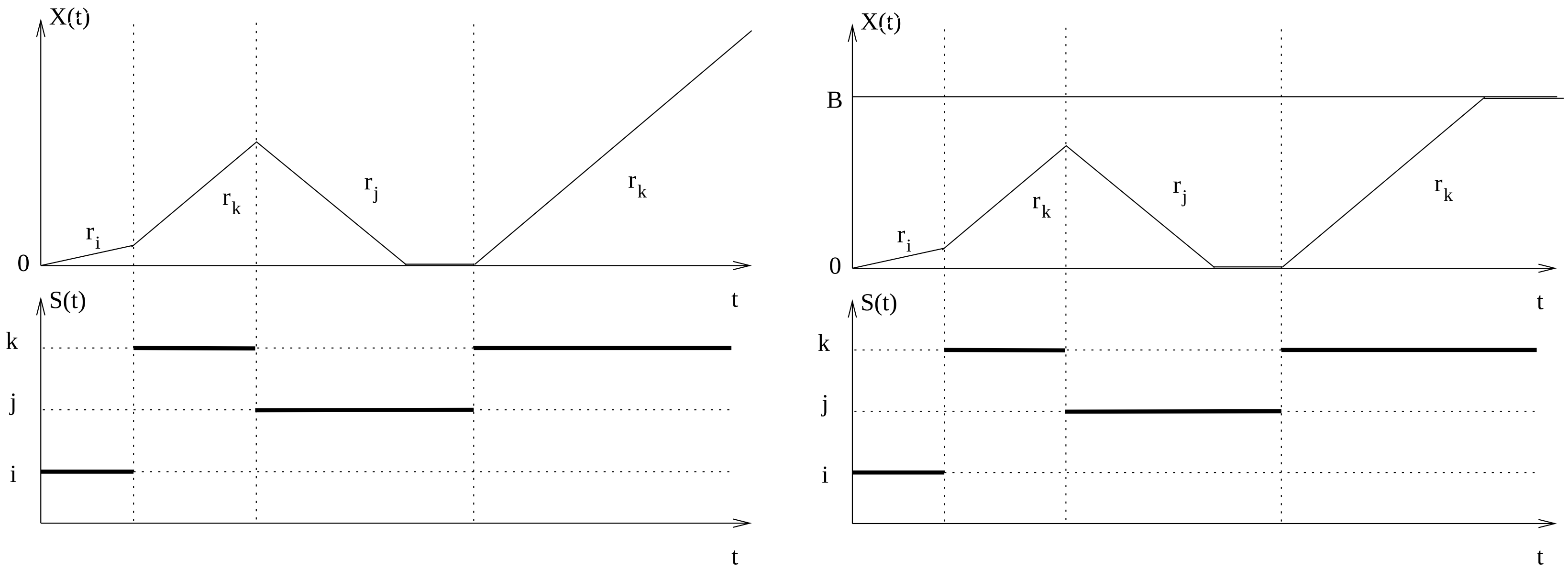
Classes of fluid models:

- finite buffer – infinite buffer,
- first order – second order,
- homogeneous – fluid level dependent,
- barrier behaviour in second order case
  - reflecting – absorbing.

3.1 Introduction to fluid models

Infinite buffer: the continuous quantity is only lower bounded at zero.

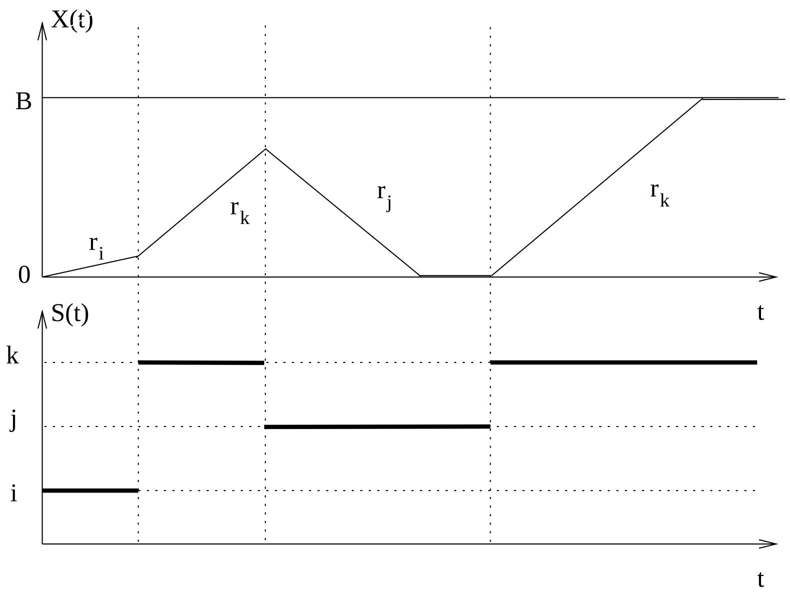
Finite buffer: the continuous quantity is lower bounded at zero and upper bounded at B.

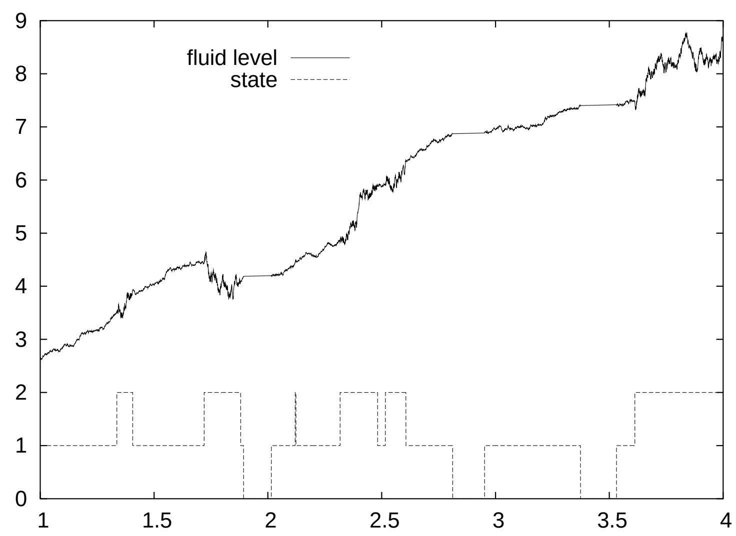


3.1 Introduction to fluid models

First order: the continuous quantity is a deterministic function of a CTMC.

Second order: the continuous quantity is a stochastic function of a CTMC.

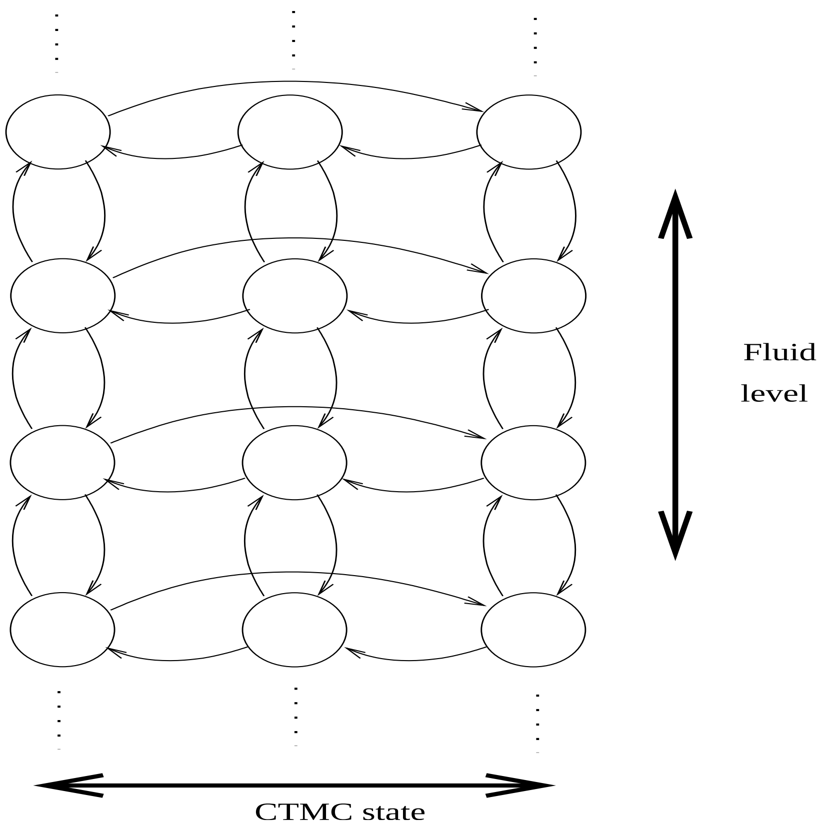




3.1 Introduction to fluid models

Interpretation of second order fluid models.

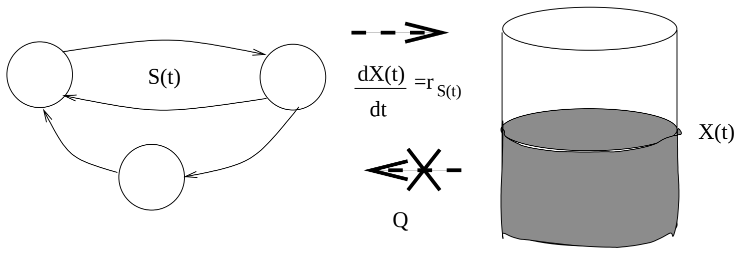
Random walk with decreasing time and fluid granularity.

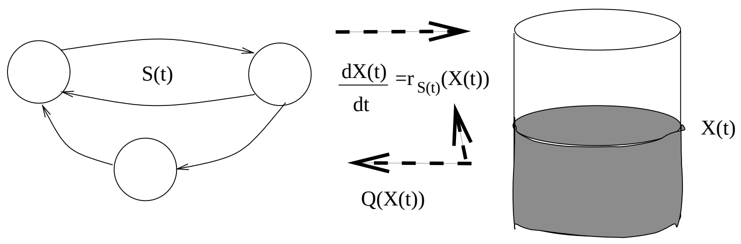


3.1 Introduction to fluid models

Homogeneous: the evolution of the CTMC is independent of the fluid level.

Fluid level dependent: the generator of the CTMC is a function of the fluid level.



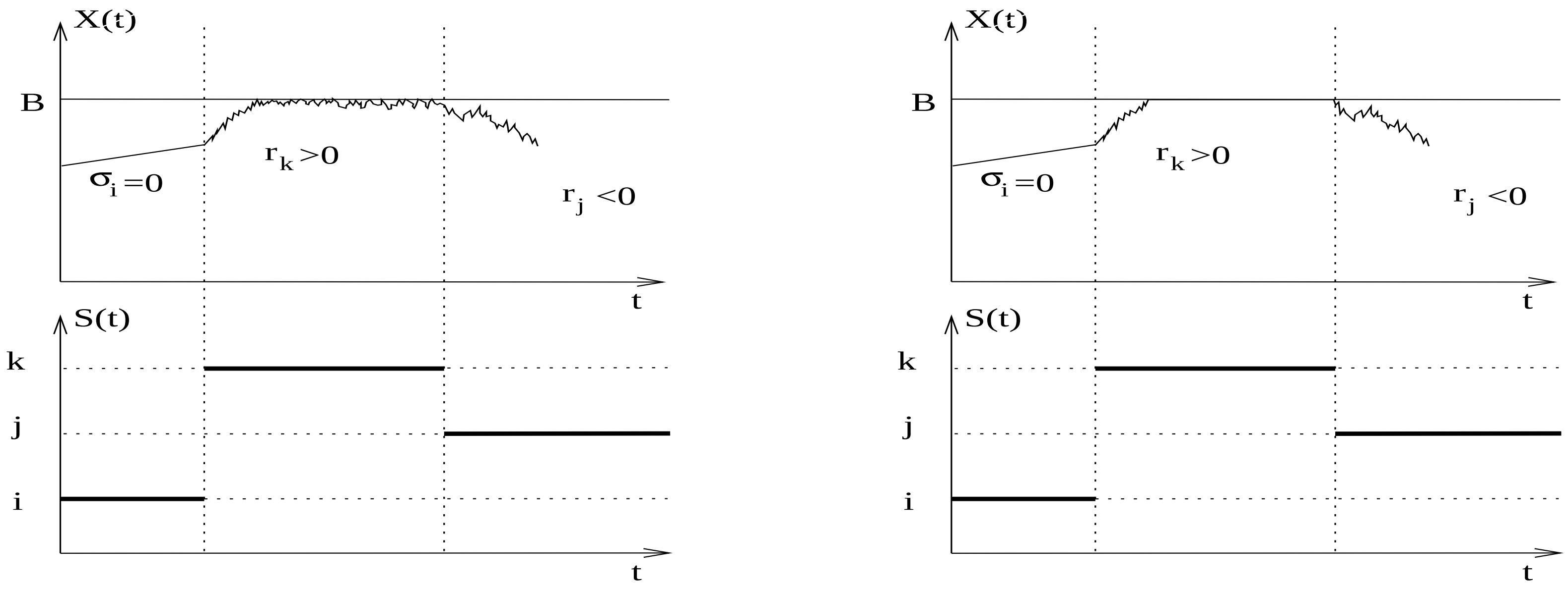


3.1 Introduction to fluid models

Boundary behaviour of second order fluid models.

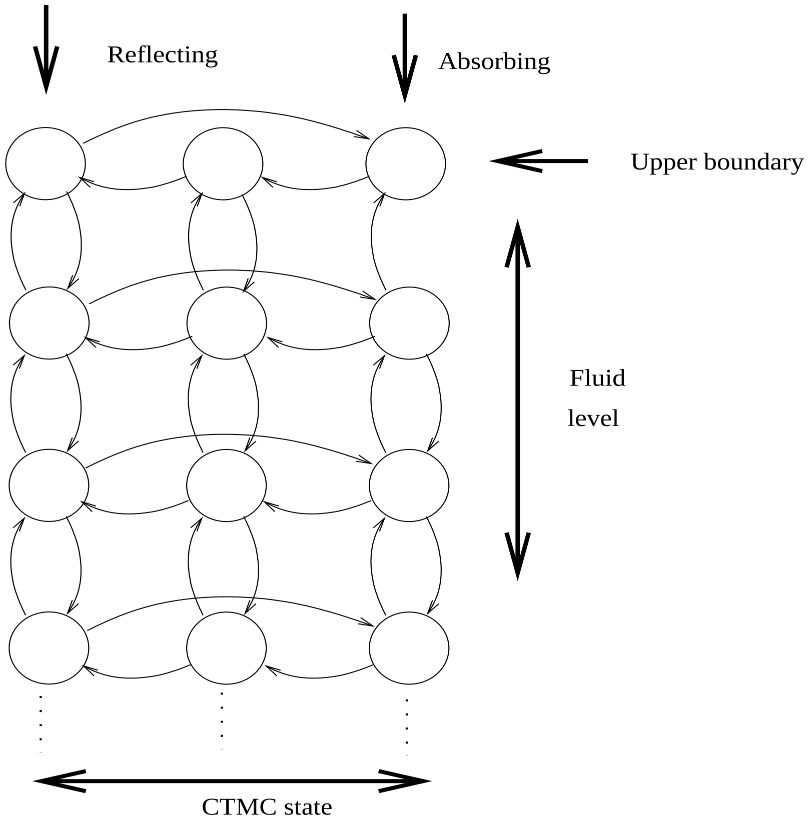
Reflecting: the fluid level is immediately reflected at the boundary.

Absorbing: the fluid level remains at the boundary up to a state transition of the Markov chain.



3.1 Introduction to fluid models

Interpretation of the boundary behaviours:



3.2 Transient behaviour of fluid models

- Transient behaviour of *first order infinite buffer homogeneous* Markov fluid models,

- Extensions:

  - finite buffer,
  - second order,
  - fluid level dependency.

3.2 Transient behaviour of fluid models

First order, infinite buffer, homogeneous Markov fluid models

During a sojourn of the CTMC in state i ($S(t) = i$) the fluid level ($X(t)$) increases at rate r_i when $X(t) > 0$:

$$X(t+\Delta) - X(t) = r_i \Delta \quad \rightarrow \quad \frac{d}{dt} X(t) = r_i \qquad \text{if } S(t) = i, X(t) > 0.$$

When $X(t) = 0$ the fluid level can not decrease:

$$\frac{d}{dt} X(t) = \max(r_i, 0) \qquad \text{if } S(t) = i, X(t) = 0.$$

That is

$$\frac{d}{dt} X(t) = \begin{cases} r_{S(t)} & \text{if } X(t) > 0, \\ \max(r_{S(t)}, 0) & \text{if } X(t) = 0. \end{cases}$$

3.2 Transient behaviour of fluid models

First order, *finite buffer*, *homogeneous* Markov fluid models

When $X(t) = B$ the fluid level can not increase:

$$\frac{d}{dt}X(t) = \min(r_i, 0), \qquad \text{if } S(t) = i, X(t) = B.$$

That is

$$\frac{d}{dt}X(t) = \begin{cases} r_{S(t)}, & \text{if } X(t) > 0, \\ \max(r_{S(t)}, 0), & \text{if } X(t) = 0, \\ \min(r_{S(t)}, 0), & \text{if } X(t) = B. \end{cases}$$

3.2 Transient behaviour of fluid models

Second order, *infinite buffer*, *homogeneous* Markov fluid models with reflecting barrier

During a sojourn of the CTMC in state i ($S(t) = i$) in the sufficiently small $(t, t + \Delta)$ interval the distribution of the fluid increment $(X(t + \Delta) - X(t))$ is normal distributed with mean $r_i\Delta$ and variance $\sigma_i^2\Delta$:

$$X(t + \Delta) - X(t) = \mathcal{N}(r_i\Delta, \sigma_i^2\Delta),$$

$$\text{if } S(u) = i, u \in (t, t + \Delta), X(t) > 0.$$

At $X(t) = 0$ the fluid process is reflected immediately,

$\longrightarrow Pr(X(t) = 0) = 0.$

3.2 Transient behaviour of fluid models

Second order, *infinite buffer*, *homogeneous* Markov fluid models with absorbing barrier

Between the boundaries the evolution of the process is the same as before.

First time when the fluid level decreases to zero the fluid process stops,

$\longrightarrow Pr(X(t) = 0) > 0.$

Due to the absorbing property of the boundary the probability that the fluid level is close to it is very low,

$\longrightarrow \lim_{\Delta \to 0} \frac{Pr(0<X(t)<\Delta)}{\Delta} = 0.$

3.2 Transient behaviour of fluid models

Inhomogeneous (fluid level dependent), first order, infinite buffer Markov fluid models

The evolution of the fluid level is the same:

$$\frac{d}{dt}X(t) = \begin{cases} r_{S(t)}(X(t)), & \text{if } X(t) > 0, \\ \max(r_{S(t)}(X(t)), 0), & \text{if } X(t) = 0. \end{cases}$$

But the evolution of the CTMC depends on the fluid level:

$$\lim_{\Delta \to 0} \frac{Pr(S(t+\Delta) = j | S(t) = i)}{\Delta} = q_{ij}(X(t)) \ .$$

The generator of the CTMC is $Q(X(t))$ and the rate matrix is $R(X(t))$.

3.3 Transient description of fluid models

Notations:

$\pi_i(t) = Pr(S(t) = i)$ – state probability,

$u_i(t) = Pr(X(t) = B, S(t) = i)$ – buffer full probability,

$\ell_i(t) = Pr(X(t) = 0, S(t) = i)$ – buffer empty probability,

$$p_i(t, x) = \lim_{\Delta \to 0} \frac{1}{\Delta} Pr(x < X(t) < x + \Delta, S(t) = i)$$

– fluid density.

$$\Longrightarrow \quad \pi_i(t) = \ell_i(t) + u_i(t) + \int_x p_i(t, x) dx.$$

3.3 Transient description of fluid models

First order, infinite buffer, homogeneous behaviour.

Forward argument:

If $S(t+\delta) = i$, then between t and $t+\Delta$ the CTMC

- stays in i with probability $1 + q_{ii}\Delta$,
- moves from k to i with probability $q_{ki}\Delta$,
- has more than 1 state transition with probability $\sigma(\Delta)$.

3.3 Transient description of fluid models

Fluid density:

$$\begin{aligned} p_i(t+\Delta, x) \; = \; & (1 + q_{ii}\Delta)\; p_i(t, x - r_i\Delta) + \\ & \sum_{k \in \mathcal{S}, k \neq i} q_{ki}\Delta\; p_k(t, x - \mathcal{O}(\Delta)) + \\ & \sigma(\Delta)\;, \end{aligned}$$

where $\lim_{\Delta \to 0} \sigma(\Delta)/\Delta = 0$ and $\lim_{\Delta \to 0} \mathcal{O}(\Delta) = 0$.

3.3 Transient description of fluid models

$$p_i(t+\Delta, x) - p_i(t, x - r_i\Delta) = \sum_{k\in\mathcal{S}} q_{ki}\Delta\; p_k(t, x - \mathcal{O}(\Delta)) + \sigma(\Delta)\ ,$$

$$\frac{p_i(t+\Delta, x) - p_i(t,x)}{\Delta} + r_i\,\frac{p_i(t,x) - p_i(t, x - r_i\Delta)}{r_i\Delta} = \sum_{k\in\mathcal{S}} q_{ki}\; p_k(t, x - \mathcal{O}(\Delta)) + \frac{\sigma(\Delta)}{\Delta}\ ,$$

$$\boxed{\frac{\partial}{\partial t}p_i(t,x) + r_i\,\frac{\partial}{\partial x}p_i(t,x) = \sum_{k\in\mathcal{S}} q_{ki}\; p_k(t,x)\ .}$$

3.3 Transient description of fluid models

Empty buffer probability:

If $r_i > 0$,

$\longrightarrow$ the fluid level increases in state i,

$\longrightarrow$ $\ell_i(t) = Pr(X(t) = 0, S(t) = i) = 0.$

3.3 Transient description of fluid models

If $r_i \leq 0$:

$$
\begin{aligned}
\ell_i(t+\Delta) \;=\; & \\
& (1+q_{ii}\Delta)\left(\ell_i(t) + \underbrace{\int_0^{-r_i\Delta} p_i(t,x)dx}_{*}\right) + \\
& \sum_{k\in\mathcal{S},k\neq i} q_{ki}\Delta \left(\ell_k(t) + \underbrace{\int_0^{\mathcal{O}(\Delta)} p_k(t,x)dx}_{\mathcal{O}(\Delta)}\right) + \\
& \sigma(\Delta)\ .
\end{aligned}
$$

3.3 Transient description of fluid models

When $x \leq -r_i\Delta$, then

$$p_i(t,x) = p_i(t,0) + x p_i'(t,0) + \sigma(\Delta) \ ,$$

and

$$\boxed{*} = \int_0^{-r_i\Delta} p_i(t,x)dx$$

$$= \int_0^{-r_i\Delta} p_i(t,0)dx + \int_0^{-r_i\Delta} x p_i'(t,0)dx + \int_0^{-r_i\Delta} \sigma(\Delta)dx$$

$$= -r_i\Delta \ p_i(t,0) \quad + \underbrace{\frac{(-r_i\Delta)^2}{2} \ p_i'(t,0)}_{\sigma(\Delta)} \quad + \underbrace{(-r_i\Delta) \ \sigma(\Delta)}_{\sigma(\Delta)} \ .$$

3.3 Transient description of fluid models

From which the empty buffer probability:

$$\ell_i(t+\Delta) \;=\; (1+q_{ii}\Delta) \quad \ell_i(t) \; -r_i \Delta p_i(t,0) + \sigma(\Delta) \quad + \quad \sum_{k\in\mathcal{S},k\neq i} q_{ki}\Delta \;\; (\ell_k(t)+\mathcal{O}(\Delta)) + \sigma(\Delta) \;,$$

$$\ell_i(t+\Delta)-\ell_i(t) \;=\; q_{ii}\Delta \; \ell_i(t) - r_i \Delta p_i(t,0) + \quad \sum_{k\in\mathcal{S},k\neq i} q_{ki}\Delta \;\; (\ell_k(t)+\mathcal{O}(\Delta)) + \sigma(\Delta) \;,$$

3.3 Transient description of fluid models

and

$$\frac{\ell_i(t+\Delta) - \ell_i(t)}{\Delta} = $$
$$- r_i \; p_i(t,0) + \sum_{k \in \mathcal{S}} q_{ki} \; \left(\ell_k(t) + \mathcal{O}(\Delta)\right) + \frac{\sigma(\Delta)}{\Delta} \;,$$

$$\boxed{\frac{d}{dt}\ell_i(t) \;=\; -r_i \; p_i(t,0) + \sum_{k \in \mathcal{S}} q_{ki} \; \ell_k(t) \;.}$$

3.3 Transient description of fluid models

Set of governing equations:

Fluid density:

$$\frac{\partial}{\partial t} p_i(t,x) + r_i \frac{\partial}{\partial x} p_i(t,x) = \sum_{k \in \mathcal{S}} q_{ki}\ p_k(t,x)\ ,$$

Empty buffer probability:

if $r_i <= 0$:

$$\frac{d}{dt}\ell_i(t) \ = \ -r_i\ p_i(t,0) + \sum_{k \in \mathcal{S}} q_{ki}\ \ell_k(t),$$

if $r_i > 0$:

$$\ell_i(t) \ = 0.$$

3.3 Transient description of fluid models

By the definition of fluid density and empty buffer probability:

$$\int_0^\infty p_i(t,x)dx + \ell_i(t) = \pi_i(t) \ .$$

In the homogeneous case:

$$\frac{d}{dt}\pi_i(t) \ = \sum_{k \in \mathcal{S}} q_{ki} \ \pi_k(t), \quad \longrightarrow \quad \pi_i(t) = \pi_i(0)e^{Qt}.$$

3.3 Transient description of fluid models

First order, finite buffer, homogeneous behaviour.

If there is also an upper boundary:

if $r_i < 0$:

$$u_i(t) = 0,$$

if $r_i \geq 0$:

$$\frac{d}{dt} u_i(t) \;=\; r_i \; p_i(t, B) + \sum_{k \in \mathcal{S}} q_{ki} \; u_k(t).$$

3.3 Transient description of fluid models

Second order, infinite buffer, homogeneous behaviour.

Fluid density:

$$p_i(t+\Delta, x) \;=\;$$

$$(1+q_{ii}\Delta)\;\underbrace{\int_{-\infty}^{\infty} p_i(t, x-u) f_{\mathcal{N}(\Delta r_i, \Delta\sigma_i^2)}(u)du}_{**} +$$

$$\sum_{k\in\mathcal{S}, k\neq i} q_{ki}\Delta\; p_k(t, x-\mathcal{O}(\Delta)) +$$

$$\sigma(\Delta)$$

3.3 Transient description of fluid models

Using

$$p_i(t, x-u) = p_i(t, x) - u p_i'(t, x) + \frac{u^2}{2} p_i''(t, x) + \mathcal{O}(u)^3$$

we have:

$$** =$$

$$p_i(t, x) \underbrace{\int_{-\infty}^{\infty} f_{\mathcal{N}(\Delta r_i, \Delta\sigma_i^2)}(u) du}_{1} - p_i'(t, x) \underbrace{\int_{-\infty}^{\infty} u f_{\mathcal{N}(\Delta r_i, \Delta\sigma_i^2)}(u) du}_{\Delta r_i} +$$

$$p_i''(t, x) \underbrace{\int_{-\infty}^{\infty} \frac{u^2}{2} f_{\mathcal{N}(\Delta r_i, \Delta\sigma_i^2)}(u) du}_{\Delta^2 r_i^2 + \Delta\sigma_i^2/2 = \Delta\sigma_i^2/2 + \sigma(\Delta)} + \underbrace{\int_{-\infty}^{\infty} \mathcal{O}(u)^3 f_{\mathcal{N}(\Delta r_i, \Delta\sigma_i^2)}(u) du}_{\mathcal{O}(\Delta)^2 = \sigma(\Delta)} .$$

3.3 Transient description of fluid models

From which:

$$p_i(t+\Delta, x) \; = \; (1+q_{ii}\Delta) \; p_i(t,x) - p_i'(t,x)\Delta r_i + p_i''(t,x)\Delta \sigma_i^2/2 \; + \sum_{k\in\mathcal{S}, k\neq i} q_{ki}\Delta \; p_k(t, x - \mathcal{O}(\Delta)) + \;\; \sigma(\Delta) \; ,$$

$$p_i(t+\Delta, x) - p_i(t,x) \; = \; q_{ii}\Delta p_i(t,x) - p_i'(t,x)\Delta r_i + p_i''(t,x)\Delta \sigma_i^2/2 + \sum_{k\in\mathcal{S}, k\neq i} q_{ki}\Delta \; p_k(t, x - \mathcal{O}(\Delta)) + \;\; \sigma(\Delta) \; ,$$

$$\boxed{\frac{\partial}{\partial t} p_i(t,x) + \frac{\partial}{\partial x} p_i(t,x) r_i - \frac{\partial^2}{\partial x^2} p_i(t,x)\frac{\sigma_i^2}{2} \; = \sum_{k\in\mathcal{S}} q_{ki} \; p_k(t,x).}$$

3.3 Transient description of fluid models

Second order, *infinite buffer*, *reflecting barrier*, *homogeneous behaviour.*

Boundary condition:

Reflecting barrier $\longrightarrow \ell_i(t) = 0.$

Fluid density at 0:

$$\int_0^\infty p_i(t,x)dx = \pi_i(t) \qquad \Big/ \; \frac{\partial}{\partial t}$$

3.3 Transient description of fluid models

$$\int_{x=0}^{\infty} \overbrace{\underbrace{\frac{\partial}{\partial t} p_i(t,x)}}^{-\frac{\partial p_i(t,x)}{\partial x} r_i + \frac{\partial^2 p_i(t,x)}{\partial x^2}\frac{\sigma_i^2}{2} + \sum_{k\in\mathcal{S}} q_{ki}\ p_k(t,x)} dx = \overbrace{\underbrace{\frac{\partial}{\partial t}\pi_i(t)}}^{\sum_{k\in\mathcal{S}} q_{ki}\pi_i(t)}$$

$$-r_i \underbrace{\Big[p_i(t,x) \Big]_{x=0}^{\infty}}_{-p_i(t,0)} + \frac{\sigma_i^2}{2} \underbrace{\Big[p_i'(t,x) \Big]_{x=0}^{\infty}}_{-p_i'(t,0)} + \sum_{k\in\mathcal{S}} q_{ki} \underbrace{\int_{x=0}^{\infty} p_k(t,x)dx}_{\pi_i(t)} = \sum_{k\in\mathcal{S}} q_{ki}\pi_i(t)$$

$$\boxed{r_i p_i(t,0) - \frac{\sigma_i^2}{2} p_i'(t,0) = 0}$$

3.3 Transient description of fluid models

First order, infinite buffer, inhomogeneous behaviour.

Fluid density:

$$\frac{\partial}{\partial t} p_i(t,x) + r_i(x) \frac{\partial}{\partial x} p_i(t,x) = \sum_{k \in \mathcal{S}} q_{ki}(x)\, p_k(t,x)$$

Empty buffer probability:

if $r_i(0) < 0$ (and $r_i(x)$ is continuous):

$$\frac{d}{dt}\ell_i(t) \;=\; -r_i(0)\, p_i(t,0) + \sum_{k \in \mathcal{S}} q_{ki}(0)\, \ell_k(t),$$

if $r_i(0) > 0$ (and $r_i(x)$ is continuous):

$$\ell_i(t) \;= 0.$$

3.3 Transient description of fluid models

General case:

Second order, *finite buffer*, *inhomogeneous behaviour*.

Differential equations:

$$\frac{\partial p(t,x)}{\partial t} + \frac{\partial p(t,x)}{\partial x}\mathbf{R}(x) - \frac{\partial^2 p(t,x)}{\partial x^2}\mathbf{S}(x) = p(t,x)\,\mathbf{Q}(x),$$

$$p(t,0)\,\mathbf{R}(0) - p'(t,0)\,\mathbf{S}(0) = \ell(t)\,\mathbf{Q}(0),$$

$$-p(t,B)\,\mathbf{R}(B) + p'(t,B)\,\mathbf{S}(B) = u(t)\,\mathbf{Q}(B),$$

where $\mathbf{R}(x) = \text{Diag}\langle r_i(x)\rangle$ and $\mathbf{S}(x) = \text{Diag}\langle\frac{\sigma_i^2(x)}{2}\rangle$.

3.3 Transient description of fluid models

General case:

Second order, *finite buffer*, *inhomogeneous behaviour*.

Bounding behaviour:

$\sigma_i = 0$ and positive/negative drift: $\ell_i(t) = 0 / u_i(t) = 0$.

$\sigma_i > 0$, reflecting lower/upper barrier: $\ell_i(t) = 0 / u_i(t) = 0$.

$\sigma_i > 0$, absor. lower/upper barrier: $p_i(t, 0) = 0 / p_i(t, B) = 0$.

Normalizing condition:

$$\int_0^B p(t, x)\, dx \mathbb{1} + \ell(t) \mathbb{1} + u(t) \mathbb{1} = 1.$$

3.4 Stationary description of fluid models

Condition of ergodicity:

For $\forall x, y \in \mathbb{R}^+, \forall i, j \in \mathcal{S}$ the transition time

$$T = \min_{t>0}(X(t) = y, S(t) = j | X(0) = x, S(0) = i)$$

has a finite mean (i.e., $E(T) < \infty$).

3.4 Stationary description of fluid models

Notations:

$\pi_i = \lim_{t\to\infty} Pr(S(t) = i)$ – state probability,

$u_i = \lim_{t\to\infty} Pr(X(t) = B, S(t) = i)$ – buffer full probability,

$\ell_i = \lim_{t\to\infty} Pr(X(t) = 0, S(t) = i)$ – buffer empty probability,

$p_i(x) = \lim_{t\to\infty} \lim_{\Delta\to 0} \frac{1}{\Delta} Pr(x < X(t) < x + \Delta, S(t) = i)$

– fluid density,

$F_i(x) = \lim_{t\to\infty} Pr(X(t) < x, S(t) = i)$

– fluid distribution.

3.4 Stationary description of fluid models

First order, infinite buffer, homogeneous behaviour.

Fluid density:

$$r_i \frac{\partial}{\partial x} p_i(x) = \sum_{k \in \mathcal{S}} q_{ki} \; p_k(x) \ .$$

Empty buffer probability:

if $r_i <= 0$:

$$0 \ = -r_i \; p_i(0) + \sum_{k \in \mathcal{S}} q_{ki} \; \ell_k ,$$

if $r_i > 0$:

$$\ell_i \ = 0.$$

3.4 Stationary description of fluid models

First order, finite buffer, homogeneous behaviour.

Fluid density:

$$r_i \frac{\partial}{\partial x} p_i(x) = \sum_{k \in \mathcal{S}} q_{ki} \; p_k(x) \; .$$

Boundary equations:

$$\begin{cases} r_i \; p_i(0) = \sum_{k \in \mathcal{S}} q_{ki} \; \ell_k, & \text{if } r_i \leq 0, \\ \ell_i \; = 0, & \text{if } r_i > 0. \end{cases}$$

$$\begin{cases} -r_i \; p_i(B) = \sum_{k \in \mathcal{S}} q_{ki} \; u_k, & \text{if } r_i \geq 0, \\ u_i \; = 0, & \text{if } r_i < 0. \end{cases}$$

3.4 Stationary description of fluid models

Second order, *infinite buffer*, *reflecting boundary*, *homogeneous behaviour.*

Fluid density:

$$r_i \frac{\partial}{\partial x} p_i(x) - \frac{\partial^2}{\partial x^2} p_i(x) \frac{\sigma_i^2}{2} = \sum_{k \in \mathcal{S}} q_{ki} \; p_k(x) \; .$$

Empty buffer probability:

$$\ell_i \; = 0.$$

Boundary equation:

$$r_i p_i(0) - \frac{\sigma_i^2}{2} p_i'(0) = \sum_{k \in \mathcal{S}} q_{ki} \; \ell_k = 0.$$

3.4 Stationary description of fluid models

Second order, *infinite buffer*, *absorbing boundary*, *homogeneous behaviour.*

Fluid density:

$$r_i \frac{\partial}{\partial x} p_i(x) - \frac{\partial^2}{\partial x^2} p_i(x) \frac{\sigma_i^2}{2} = \sum_{k \in \mathcal{S}} q_{ki} \; p_k(x).$$

Empty buffer probability:

$$p_i(0) = 0.$$

Boundary equation:

$$-\frac{\sigma_i^2}{2} p_i'(0) = \sum_{k \in \mathcal{S}} q_{ki} \; \ell_k.$$

3.4 Stationary description of fluid models

General case:

Second order, *finite buffer*, *inhomogeneous behaviour*.

$$p'(x)\,\mathbf{R}(x) - p''(x)\,\mathbf{S}(x) = p(x)\,\mathbf{Q}(x),$$

$$p(0)\,\mathbf{R}(0) - p'(0)\,\mathbf{S}(0) = \ell\,\mathbf{Q}(0),$$

$$-p(B)\,\mathbf{R}(B) + p'(B)\,\mathbf{S}(B) = u\,\mathbf{Q}(B),$$

$\sigma_i = 0$ and positive/negative drift: $\ell_i = 0/u_i = 0$.

$\sigma_i > 0$, reflecting lower/upper barrier: $\ell_i = 0/u_i = 0$.

$\sigma_i > 0$, absorbing lower/upper barrier: $p_i(0) = 0/p_i(B) = 0$.

4 Solution methods

Numerical techniques:

	reward	fluid
differential equations	(+)	+
spectral decomposition	(+)	+
randomization	+	+
transform domain	+	+
matrix exponent	+	+
moments	+	-

4 Solution methods

Transient analysis:

- initial condition,
- set of differential equations,
- bounding behaviour.

Stationary analysis:

- set of differential equations,
- bounding behaviour,
- normalizing condition.

4.1 Transient solution methods

- Numerical solution of differential equations,
- Randomization,
- Markov regenerative approach,
- Transform domain.

4.1 Transient solution methods

Numerical solution of differential equations (Chen et al.)

All cases.

The approach

- starts from the initial condition, and
- follows the evolution of the fluid distribution in the $(t, t + \Delta)$ interval at some fluid levels based on the differential equations and the boundary condition.

This is the only approach for inhomogeneous models.

4.1 Transient solution methods

Randomization (Sericola)

First order, infinite buffer, homogeneous behaviour.

$$F_i^c(t,x) = \sum_{n=0}^{\infty} e^{-\lambda t} \frac{(\lambda t)^n}{n!} \sum_{k=0}^{n} \binom{n}{k} x_j^k (1-x_j)^{n-k} b_i^{(j)}(n,k),$$

where $F_i^c(t,x) = Pr(X(t) > x, S(t) = i)$,

$x_j = \frac{x - r_{j-1}^+ t}{r_j t - r_{j-1}^+ t}$ if $x \in [r_{j-1}^+ t, r_j t)$, and

$b_i^{(j)}(n,k)$ is defined by initial value and a simple recursion.

4.1 Transient solution methods

Properties of the randomization based solution method:

- the expression with the given recursive formulas is a solution of the differential equation,
  the initial value of $b_i^{(j)}(n,k)$ is set to fulfill the boundary condition,
- $0 \leq x_j \leq 1$
  $\longrightarrow$ convex combination of non-negative numbers
  $\longrightarrow$ numerical stability,
- the initial fluid level is $X(0) = 0$.
  (extension to $X(0) > 0$ and to finite buffer is not available.)

4.1 Transient solution methods

First order, infinite buffer, homogeneous case.

Markov regenerative approach (Ahn-Ramaswami)

Busy/idle period:
interval when the buffer is non-empty/empty.

T_i : the beginning of the ith busy period.

$\Longrightarrow (S(t_i), T_i)$ is a Markov renewal sequence.

The idle period is PH distributed.

Analysis of a single busy period:
similar analysis as in Matrix geometric models.

4.1 Transient solution methods

First order, infinite/finite buffer, homogeneous case.

Transform domain description (Ren-Kobayashi)

The Laplace transform of

$$\frac{\partial p(t,x)}{\partial t} + \frac{\partial p(t,x)}{\partial x}\ \mathbf{R}\ - \ \frac{\partial^2 p(t,x)}{\partial x^2}\ \mathbf{S} \ = p(t,x)\ \mathbf{Q}\ ,$$

is

$$p^{**}(s,v) = (\quad \underbrace{p^*(0,v)}_{\text{initial condition}} \quad + \ \underbrace{p^*(s,0)}_{\text{unknown}}\ \mathbf{R})(s\mathbf{I} + v\mathbf{R} - \mathbf{Q})^{-1}.$$

$p^*(s,0)$ eliminates the roots of $\det(s\mathbf{I} + v\mathbf{R} - \mathbf{Q})$.

4.2 Stationary solution methods

Condition of stability of infinite buffer first/second order homogeneous fluid models.

Suppose $S(t)$ is a finite state irreducible CTMC with stationary distribution π.

The fluid model is stable if the overall drift is negative:

$$d = \sum_{i \in \mathcal{S}} \pi_i r_i < 0.$$

$\longrightarrow$ the variance does not play role.

4.2 Stationary solution methods

- Spectral decomposition,
- Matrix exponent,
- Numerical solution of differential equations,
- Randomization.

4.2 Stationary solution methods

State space partitioning:

- $\mathcal{S}^+$: $i \in \mathcal{S}^+$ iff $\sigma_i > 0$,
  second order states,
- $\mathcal{S}^0$: $i \in \mathcal{S}^0$ iff $r_i = 0$ and $\sigma_i = 0$,
  zero states,
- $\mathcal{S}^{0+}$: $i \in \mathcal{S}^{0+}$ iff $r_i > 0$ and $\sigma_i = 0$,
  positive first order states,
- $\mathcal{S}^{0-}$: $i \in \mathcal{S}^{0-}$ iff $r_i < 0$ and $\sigma_i = 0$,
  negative first order states,
- $\mathcal{S}^* = \mathcal{S}^{0-} \bigcup \mathcal{S}^{0+}$,
  first order states.

4.2 Stationary solution methods

First order, infinite/finite buffer, homogeneous case.

Spectral decomposition (Kulkarni)

Differential equation: $p'(x)\,\mathbf{R} = p(x)\,\mathbf{Q}$,

Form of the solution vector: $p(x) = e^{\lambda x}\phi$,

Substituting this solution we get the characteristic equation:

$$\phi(\lambda\mathbf{R} - \mathbf{Q}) = 0,$$

whose solutions are obtained at $\det(\lambda\mathbf{R} - \mathbf{Q}) = 0$.

4.2 Stationary solution methods

Spectral decomposition

The characteristic equation has $|\mathcal{S}^{0+}| + |\mathcal{S}^{0-}|$ solutions, with

$$\begin{cases} |\mathcal{S}^{0+}| & \text{negative eigenvalue,} \\ 1 & \text{zero eigenvalue,} \\ |\mathcal{S}^{0-}| - 1 & \text{positive eigenvalue.} \end{cases}$$

From which the solution is: $$p(x) = \sum_{j=1}^{|\mathcal{S}^{0+}|+|\mathcal{S}^{0-}|} a_j e^{\lambda_j x} \phi_j,$$

and the a_j coefficients are set to fulfill the boundary and normalizing conditions.

4.2 Stationary solution methods

Spectral decomposition

In the *infinite buffer* case these conditions are:

- $p(0)\ \mathbf{R}\ = \ell\ \mathbf{Q}$,
- $\ell_i = 0$ if $r_i > 0$, and
- $\int_0^\infty p_i(x) + \ell_i = \pi_i$.

From which $a_j = 0$ for $\lambda_j > 0$
and the rest of the coefficients are obtained from a linear system of equations.

4.2 Stationary solution methods

Spectral decomposition

In the *finite buffer* case these conditions are:

- $p(0)\,\mathbf{R} = \ell\,\mathbf{Q}$, $p(B)\,\mathbf{R} = u\,\mathbf{Q}$,
- $\ell_i = 0$ if $r_i > 0$, $u_i = 0$ if $r_i < 0$, and
- $\int_0^\infty p_i(x) + \ell_i + u_i = \pi_i$.

From which the a_j coefficients are obtained from a linear system of equations.

4.2 Stationary solution methods

Consequences:

- If $|\mathcal{S}^{0-}| = 1$
  $\longrightarrow$ all eigenvalues are non-positive.
- If $|\mathcal{S}^{0-}| > 1$ and the buffer is infinite
  $\longrightarrow$ special treatment of the positive eigenvalues
  $\longrightarrow$ spectral decomposition is necessary.
- If the buffer is finite
  $\longrightarrow$ no need for special treatment of the positive eigenvalues.

4.2 Stationary solution methods

First order, finite buffer, homogeneous case.

Matrix exponent: (Gribaudo)

Assume that $|\mathcal{S}^0| = 0$ and $\mathcal{S} = \mathcal{S}^*$.

Introduce $v = \ell + u$, $\mathbf{Q}^-$, $\mathbf{Q}^+$,

where $q^-_{ij} = q_{ij}$ if $i \in \mathcal{S}^-$ and otherwise $q^-_{ij} = 0$.

The set of equations becomes:

$$\frac{\partial p(x)}{\partial x}\mathbf{R} = p(x)\mathbf{Q} \quad \longrightarrow \quad p(B) = p(0)\ e^{\mathbf{Q}\mathbf{R}^{-1}B} = p(0)\ \mathbf{\Phi},$$

$$p(0)\mathbf{R} = v\mathbf{Q}^- \quad \longrightarrow \quad p(0) = v\mathbf{Q}^-\mathbf{R}^{-1},$$

$$-p(B)\mathbf{R} = v\mathbf{Q}^+ \quad \longrightarrow \quad \boxed{v(\mathbf{Q}^-\mathbf{R}^{-1}\mathbf{\Phi}\mathbf{R} + \mathbf{Q}^+) = 0},$$

4.2 Stationary solution methods

Matrix exponent:

And the normalizing condition is

$$\ell \mathbb{1} + u\mathbb{1} + p(0) \underbrace{\int_0^B e^{\mathbf{Q}\mathbf{R}^{-1}x} dx}_{\mathbf{\Psi}} \mathbb{1} =$$

$$\boxed{v(\mathbf{I} + \mathbf{Q}^{-}\mathbf{R}^{-1}\mathbf{\Psi})\mathbb{1} = 1}.$$

4.2 Stationary solution methods

Relation of spectral decomposition and matrix exponent:

Assume that $|\mathcal{S}^0| = 0$ and $\mathcal{S} = \mathcal{S}^*$.

The characteristic equation is: $\phi(\lambda\mathbf{I} - \mathbf{Q}\mathbf{R}^{-1}) = 0$,

The spectral solution is: $$p(x) = \sum_{j=1}^{|\mathcal{S}|} a_j e^{\lambda_j x} \phi_j,$$

where λ_j and ϕ_j are the eigenvalues and the left eigenvector of matrix $\mathbf{Q}\mathbf{R}^{-1}$.

4.2 Stationary solution methods

Relation of spectral decomposition and matrix exponent:

Introducing $a = \{a_j\}$ and $\mathbf{B} = \begin{pmatrix} \hline \phi_1 \\ \hline \phi_2 \\ \hline \vdots \\ \hline \phi_{|\mathcal{S}^*|} \\ \hline \end{pmatrix}$,

the spectral solution can be rewritten as:

$$p(x) = \sum_{j=1}^{|\mathcal{S}|} a_j e^{\lambda_j x} \phi_j = a \ \mathrm{Diag}\langle e^{\lambda_i x} \rangle \ \mathbf{B}$$

$$= \underbrace{a \ \mathbf{B}}_{= p(0)} \ \underbrace{\mathbf{B}^{-1} \ \mathrm{Diag}\langle e^{\lambda_i x} \rangle \ \mathbf{B}}_{e^{\mathbf{Q}\mathbf{R}^{-1} x}},$$

4.2 Stationary solution methods

Second order, infinite/finite buffer, homogeneous case.

Spectral decomposition (Karandikar-Kulkarni)

Differential equation: $p'(x)\ \mathbf{R}\ -\ p''(x)\ \mathbf{S}\ \ = p(x)\ \mathbf{Q}\ ,$

Form of the solution vector: $p(x) = e^{\lambda x}\phi,$

Substituting this solution we get the characteristic equation:

$$\phi(\lambda\mathbf{R} - \lambda^2\mathbf{S} - \mathbf{Q}) = 0,$$

whose solutions are obtained at $\det(\lambda\mathbf{R} - \lambda^2\mathbf{S} - \mathbf{Q}) = 0.$

4.2 Stationary solution methods

Spectral decomposition

The characteristic equation has $2|\mathcal{S}^+| + |\mathcal{S}^*|$ solutions, with

$$\begin{cases} |\mathcal{S}^+| + |\mathcal{S}^{0+}| & \text{negative eigenvalue,} \\ 1 & \text{zero eigenvalue,} \\ |\mathcal{S}^+| + |\mathcal{S}^{0-}| - 1 & \text{positive eigenvalue.} \end{cases}$$

From which the solution is: $\quad p(x) = \displaystyle\sum_{j=1}^{2|\mathcal{S}^+|+|\mathcal{S}^*|} a_j e^{\lambda_j x} \phi_j ,$

and the a_j coefficients are set to fulfill the boundary and normalizing conditions.

4.2 Stationary solution methods

Second order, infinite/infinite buffer, homogeneous case.

A transformation of the quadratic equation to a linear one

Assume that $|\mathcal{S}^0| = |\mathcal{S}^*| = 0$ and $\mathcal{S} = \mathcal{S}^+$.

$$\begin{aligned} \frac{d}{dx} p(x)\ \mathbf{R} \ - \ \frac{d}{dx} p'(x)\ \mathbf{S} &= p(x)\ \mathbf{Q}\ , \\ \frac{d}{dx} p(x)\ \mathbf{I} &= p'(x)\ \mathbf{I}\ , \end{aligned}$$

$$\frac{d}{dx} \begin{array}{|c|c|} \hline p(x) & p'(x) \\ \hline \end{array} \begin{array}{|c|c|} \hline \mathbf{R} & \mathbf{I} \\ \hline -\mathbf{S} & \mathbf{0} \\ \hline \end{array} = \begin{array}{|c|c|} \hline p(x) & p'(x) \\ \hline \end{array} \begin{array}{|c|c|} \hline \mathbf{Q} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{I} \\ \hline \end{array}$$

$$\Longrightarrow \quad \frac{d}{dx}\hat{p}(x)\ \hat{\mathbf{R}} = \hat{p}(x)\ \hat{\mathbf{Q}} \quad \longrightarrow \quad \hat{p}(B) = \hat{p}(0)\ e^{\hat{\mathbf{Q}}\hat{\mathbf{R}}^{-1}B}.$$

4.2 Stationary solution methods

Numerical solution of differential equations (Gribaudo et al.)

All cases with finite buffer.

Numerically solve the matrix function $\mathbf{M}(x)$ with initial condition $\mathbf{M}(0) = \mathbf{I}$ based on

$$\mathbf{M}'(x)\,\mathbf{R}(x) - \mathbf{M}''(x)\,\mathbf{S}(x) = \mathbf{M}(x)\,\mathbf{Q}(x)$$

and calculate the unknown boundary conditions based on

$$p(B) = p(0)\,\mathbf{M}(B)$$

This is the only approach for inhomogeneous models.

4.2 Stationary solution methods

First order, infinite/finite buffer, homogeneous case.

Randomization (Sericola)

Randomization with simple coefficients:

$$F_i(x) = \sum_{n=0}^{\infty} e^{-\lambda t/r} \frac{(\lambda t/r)^n}{n!} \ b_i(n)$$

where $r = \min(r_i | r_i > 0)$ and
$b_i(n)$ is defined by initial value and a simple recursion.

Applicable only when $|\mathcal{S}^{0-}| = 1$.

5. Applications

- Fluid Models and FSPNs have been successfully used in the literature to study several interesting systems.

- Here we present three examples:

- Computation of transfer time distribution in P2P file sharing applications
- Model of a pharmaceutical production system
- Analysis of software systems with checkpointing and rejuvenation

5.1. Transfer Time in P2P

- Peer-to-Peer has recently emerged has a new paradigm for building network applications.

- In the last few year, P2P file-sharing applications (like Kazaa, eDonkey, Gnutella) are generating an increasing fraction on today's Internet.

5.1. Transfer Time in P2P

- In P2P applications, each peer can act both as a client and as a server.

- In many P2P protocols, a client can be served in parallel by more than one peer.

- The overall application performance is determined by the number of requests being served by each peer (both as a client and as a server).

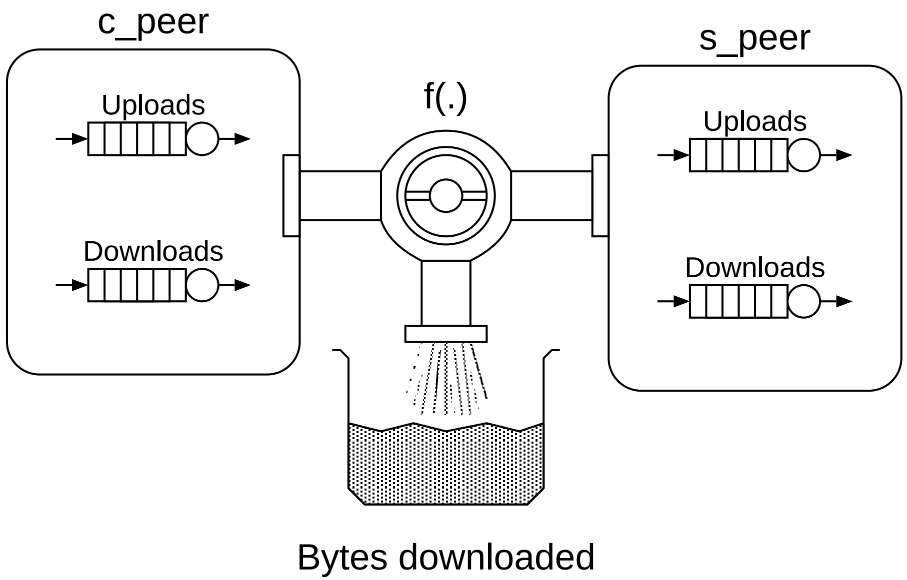
5.1. Transfer Time in P2P

- We design a fluid model to compute the transfer time distribution of P2P file sharing protocol.

- We make several simplifying assumptions:

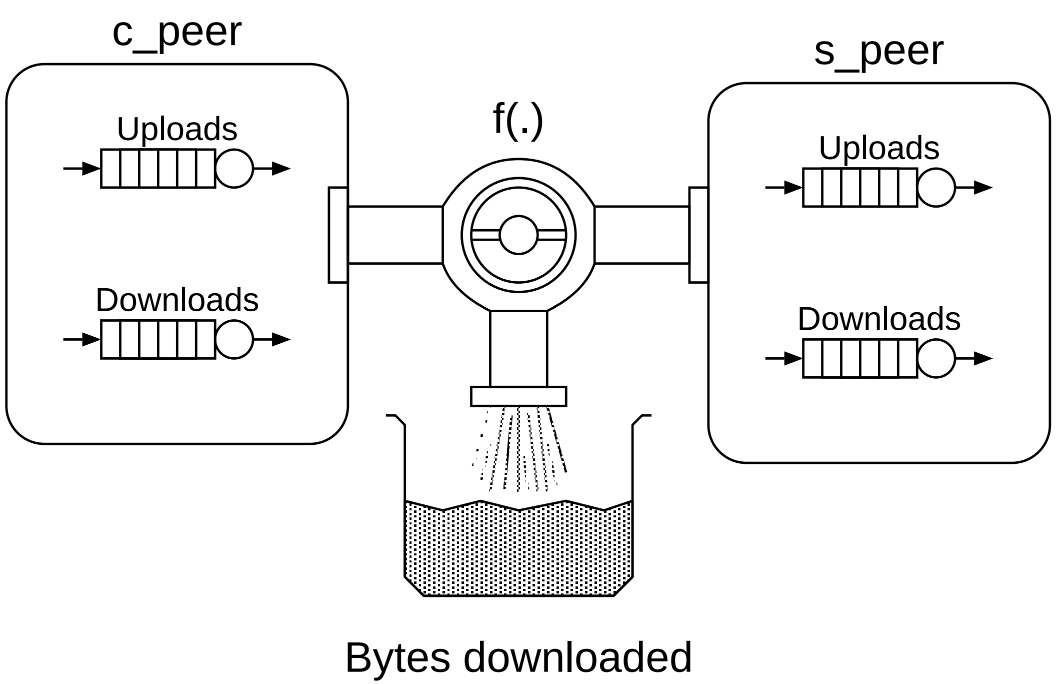
  - We neglect the search and queueing phase
  - We consider only one single source for download
  - We imagine that the overall bandwidth depends only on band and on the load at both the client and the server

5.1. Transfer Time in P2P



- We model both the server and the client with two independent service queues: one for the uploads, and another for the downloads. The number of costumers in a queue represents the load of that particular component.

5.1. Transfer Time in P2P



- The fluid buffer represents the quantity of byte received by the client for the request.

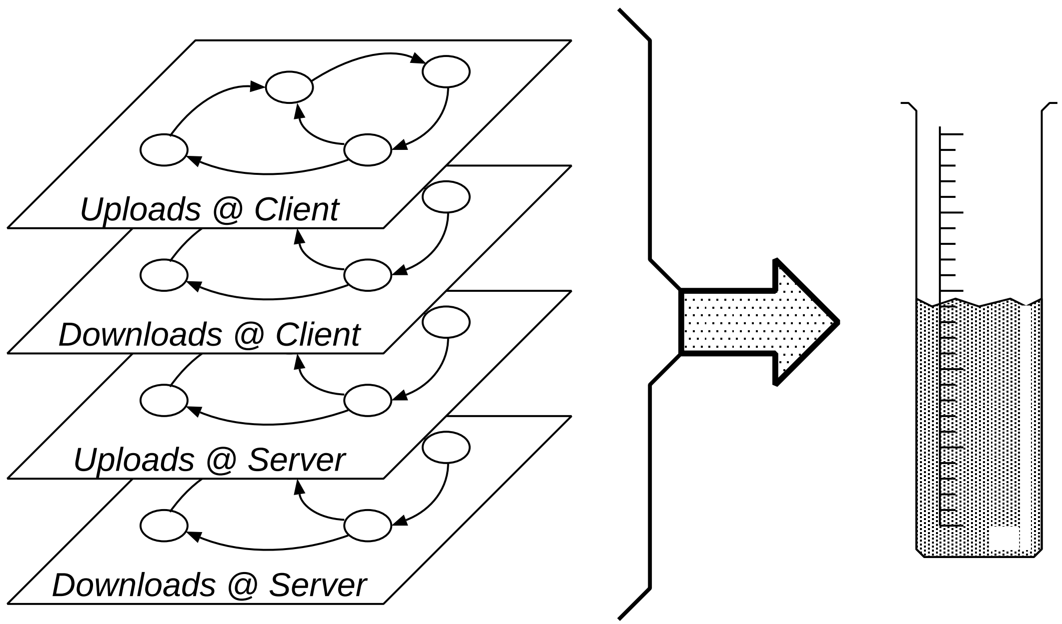
5.1. Transfer Time in P2P

- The flow rate in a state depends on the load of the four independent components in that state. A possible definition could be:

$$
\begin{aligned}
f(s) &= \min\left\{\frac{cb}{\#cu+\#cd+1}, \frac{sb}{\#su+\#sd+1}\right\} \\
s &= (\#cu, \#cd, \#su, \#sd)
\end{aligned}
$$

where s represents the discrete state of the model, cb the client bandwidth and sb the server bandwidth.

5.1. Transfer Time in P2P



- The resulting model is a fluid model (a reward model) whose underlaying Markov chain is the superposition of the Markov chains of the four queues.

5.1. Transfer Time in P2P

- The model can be solved using transient analysis.

- The obtained solution can be integrated to compute the transfer time distribution.

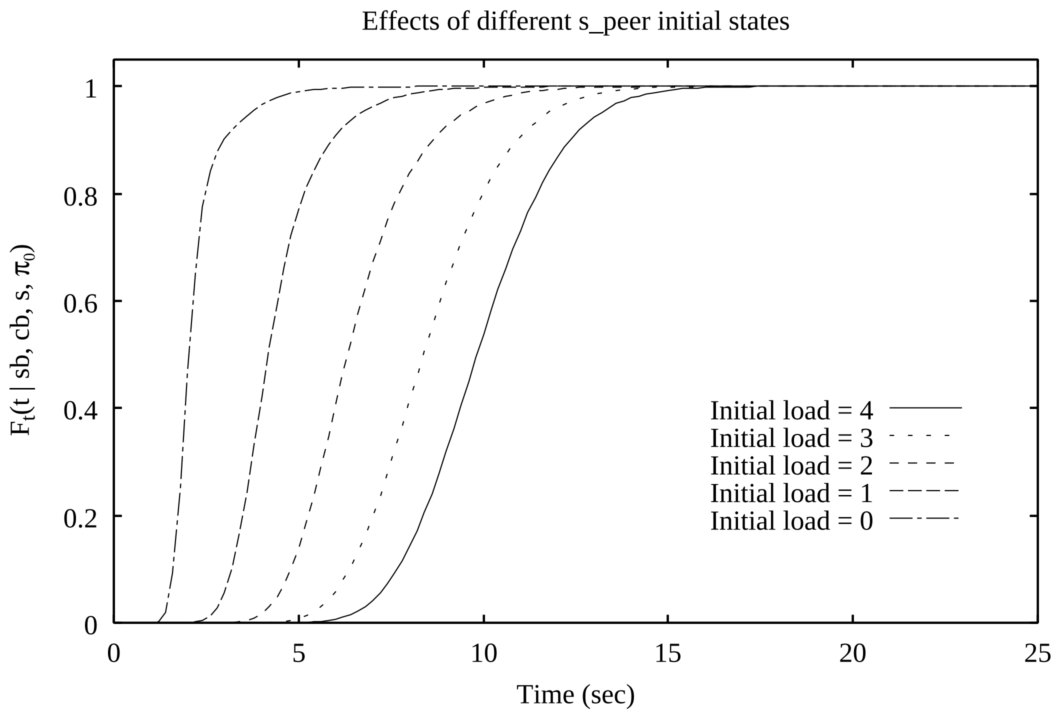
$$F(s,t) \;=\; P(T(s) < t) = P(F(t) > s) = \int_s^\infty \pi(t,x)dx$$

$F(s,t)$ is the probability that the application successfully downloads a file of length s in t time units.

$T(s)$ is the download time of a file of length s.

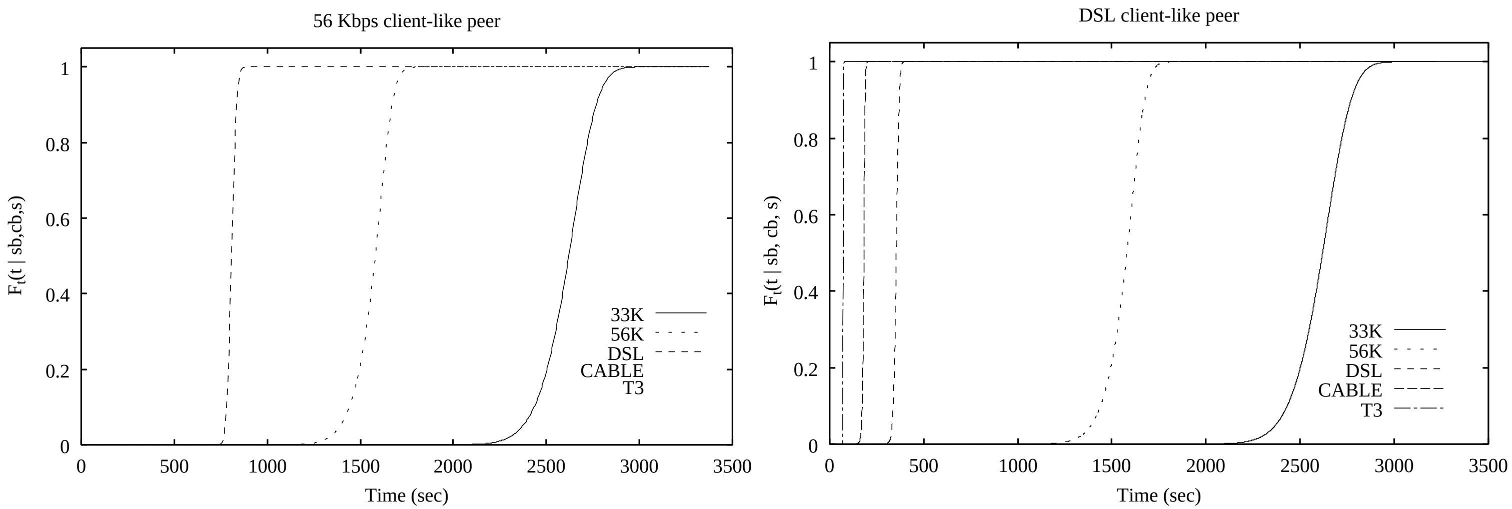
$F(t)$ is the amount of downloaded data in t time units.

5.1. Transfer Time in P2P



- The model can then be exploited, for example, to show the dependency of the transfer time on the initial load of the server (for short files).

5.1. Transfer Time in P2P



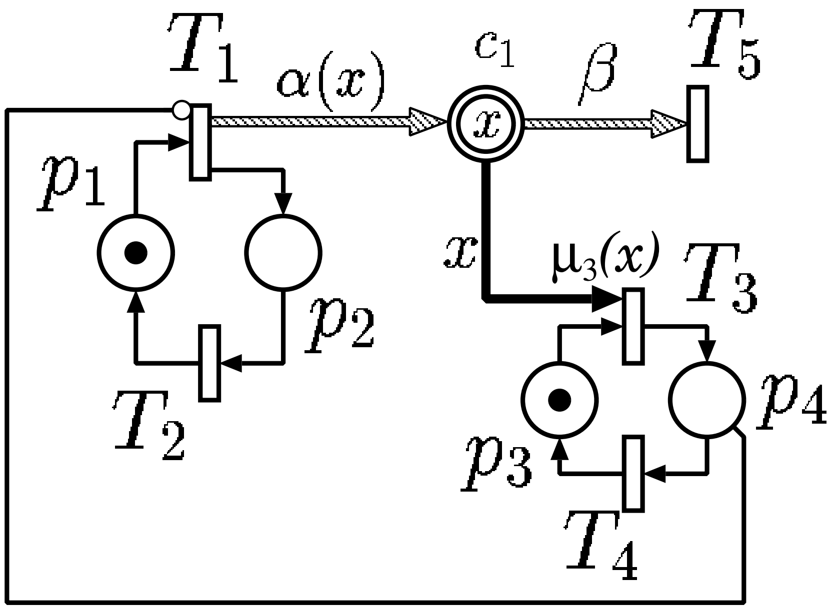
- Or to show that the speed and the state of the server are not influent if the client has a very low bandwidth.

5.2. Pharmaceutical production system

- We consider a pharmaceutical production system.

- If the equipment fails during the sterilization process, all the product contained in the buffer must be discarded.

5.2. Pharmaceutical production system



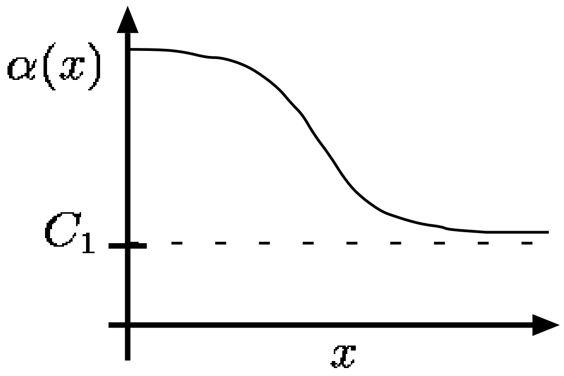
- We model the system with an FSPN with flush-out and fluid dependent transition and flow rates.

5.2. Pharmaceutical production system

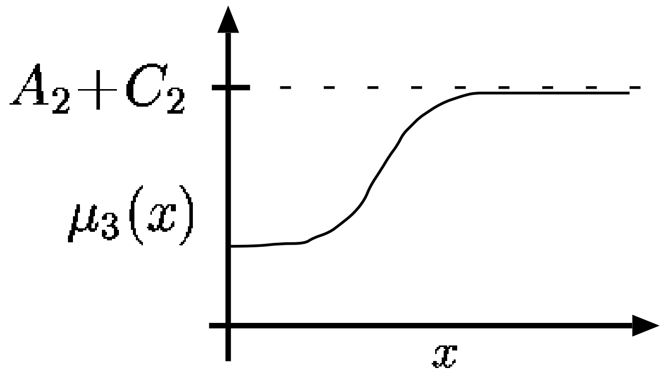
- The production slows down when the buffer becomes full (fluid dependent flow rate $\alpha(x)$)

- The probability that the sterilization process fails increases when the buffer becomes full (fluid dependent transition rate $\mu_3(x)$).

5.2. Pharmaceutical production system



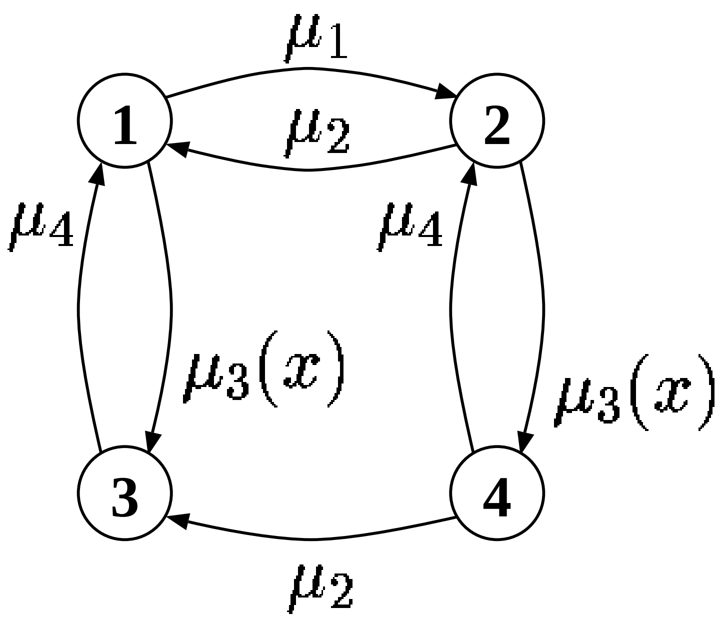
Flow rate



Transition rate

$$\alpha(x) = A_1 \left(1 - \frac{1}{1 + e^{B_1 - x}}\right) + C_1 \qquad \mu_3(x) = A_2 \frac{1}{1 + e^{B_2 - x}} + C_2$$

5.2. Pharmaceutical production system



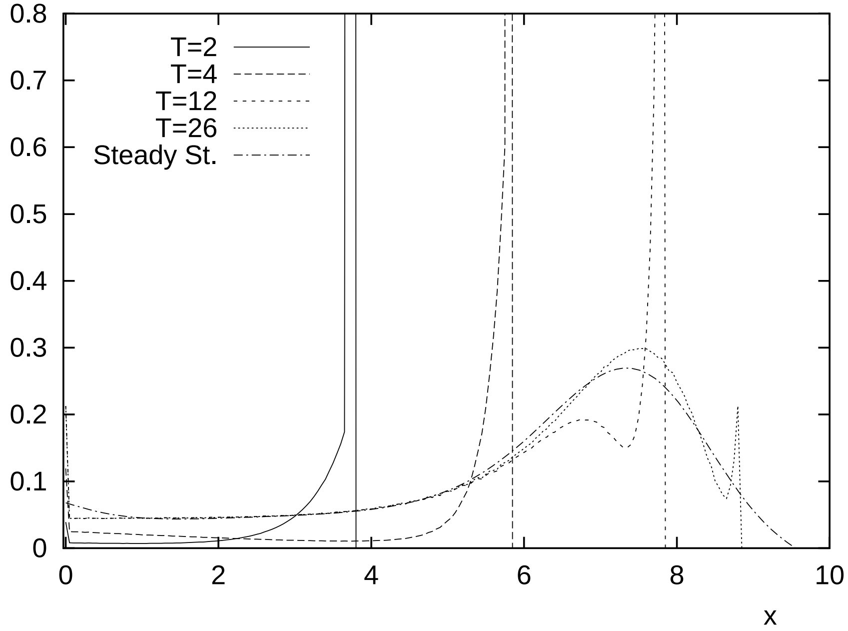
- The underlaying fluid model has only four states, but is non-homogenous.

5.2. Pharmaceutical production system

- We can solve the model using transient analysis techniques.

- Then we can integrate the solution in various ways to obtain interesting performance indices.

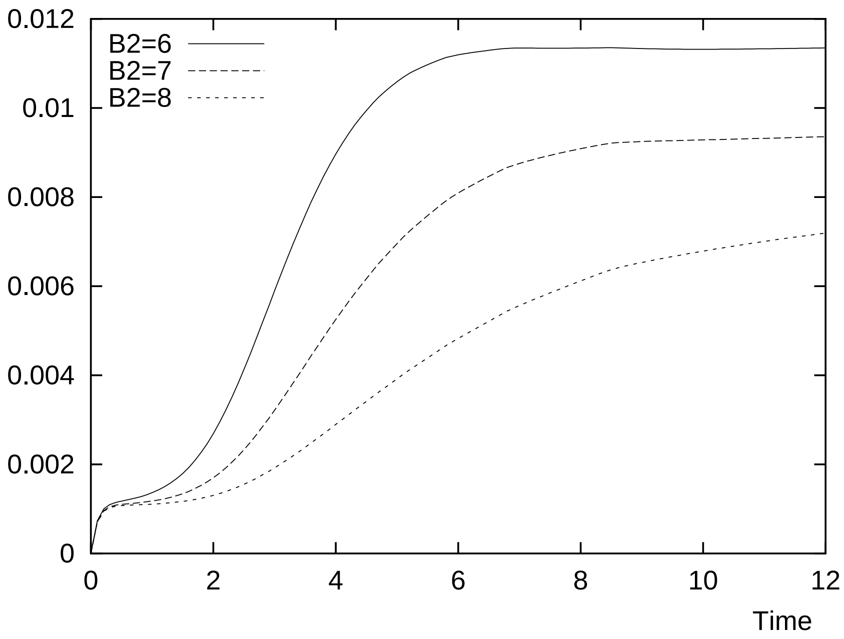
5.2. Pharmaceutical production system



- Buffer distribution:

$$\sum_{\mathbf{m_i} \in \mathcal{S}_\mathbf{d}} \pi_i(\tau, x_1).$$

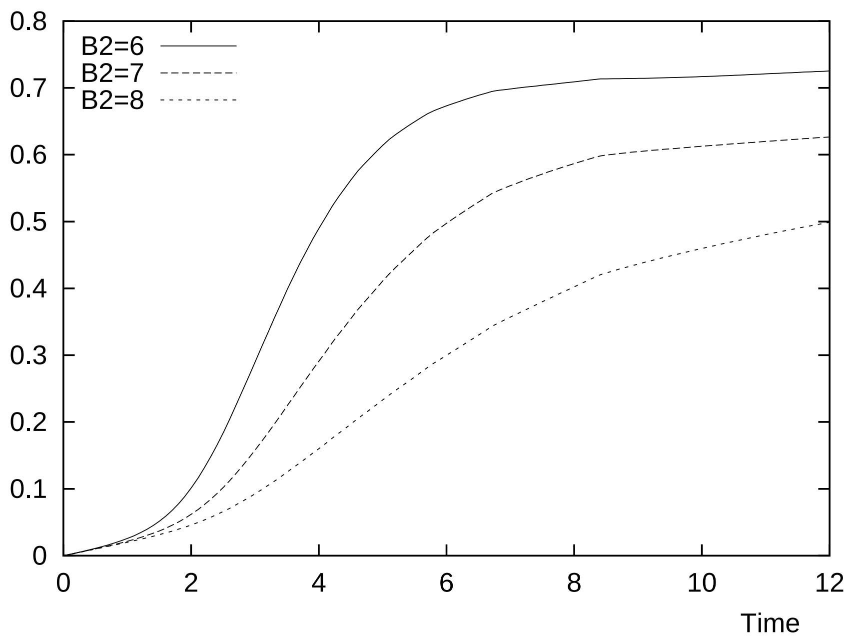
5.2. Pharmaceutical production system



- Crash probability:

$$P(\text{crash at } \tau) = \sum_{\boldsymbol{m_i}:m_4=1} \int_0^\infty \pi_i(\tau, x_1) dx_1.$$

5.2. Pharmaceutical production system



- Mean quantity of product wasted:

$$\Psi(c_1, T_3) = \sum_{\mathbf{m_i}:\mathcal{E}(\mathbf{m_i})\supseteq\mathbf{T_3}} \int_0^\infty x_1 \; \pi_i(\tau, x_1) \; \mu_3(x_1) \; dx_1$$

5.3. Software system with Rejuvenation

- It is now well established that outages in computer systems are caused more due to software faults.

- Cost-effective fault-tolerance techniques are an attractive way to try to cope with the problem.

- *Software rejuvenation, self-restoration and checkpointing* are some of such techniques.

5.3. Software system with Rejuvenation

- Rejuvenation restarts the system, making it experience a downtime equal to the time it takes to clean up the resources.

- Self-restoration does not block completely the system, but it only degrades its performance. However it is less effective than rejuvenation.

- Checkpointing saves the state of the system at predefined interval, in order to reduce the system recovery time.

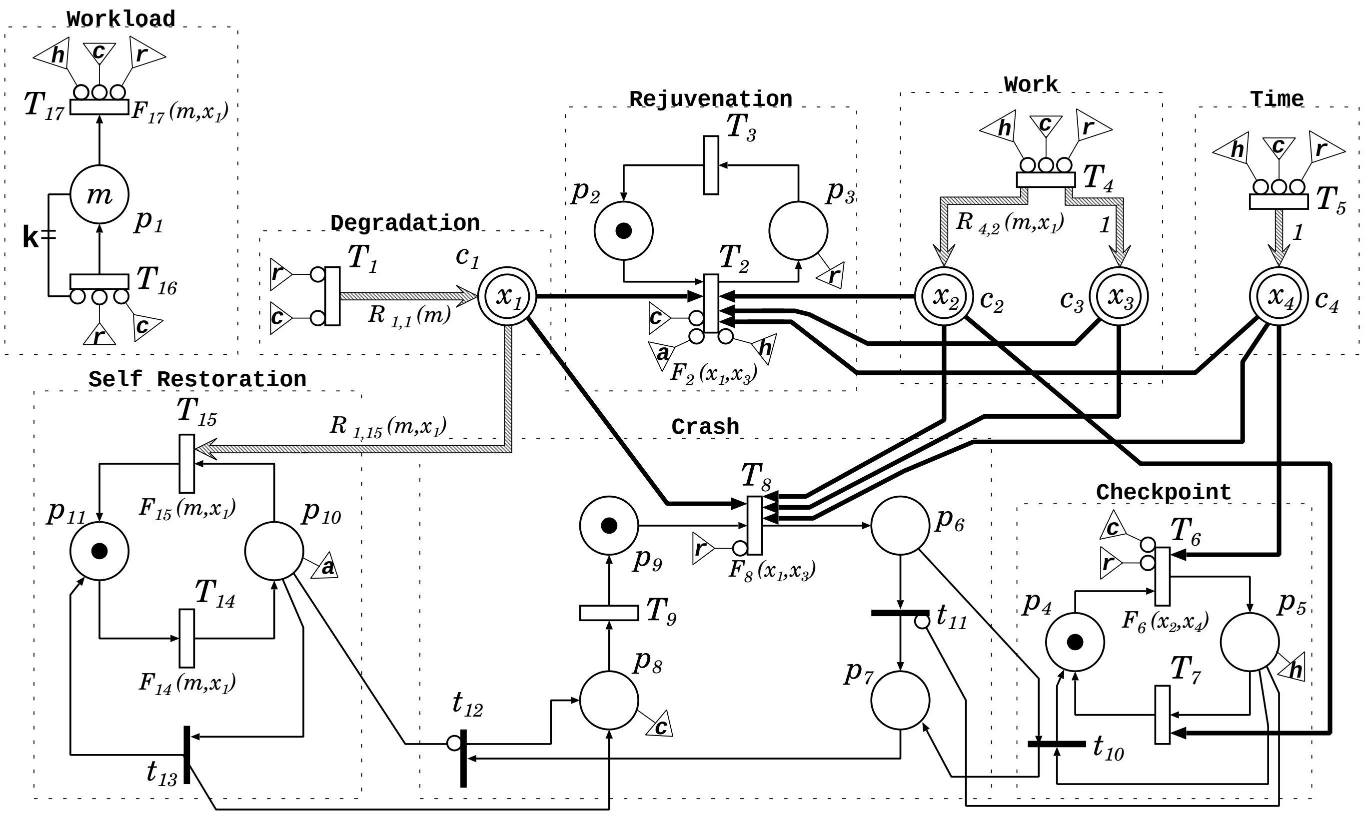
5.3. Software system with Rejuvenation

- In some cases, like Data-base systems, these three technique are used together.

- In the literature, most of the models deals only with some of these techniques - they do not consider all the features together.

- Using FSPN it is possible to design a model capable of considering all these aspects together.

5.3. Software system with Rejuvenation



- The *degradation* x_1, the work x_2 and the time (up-time x_3, time since last checkpoint x_4) can be represented using fluid places.

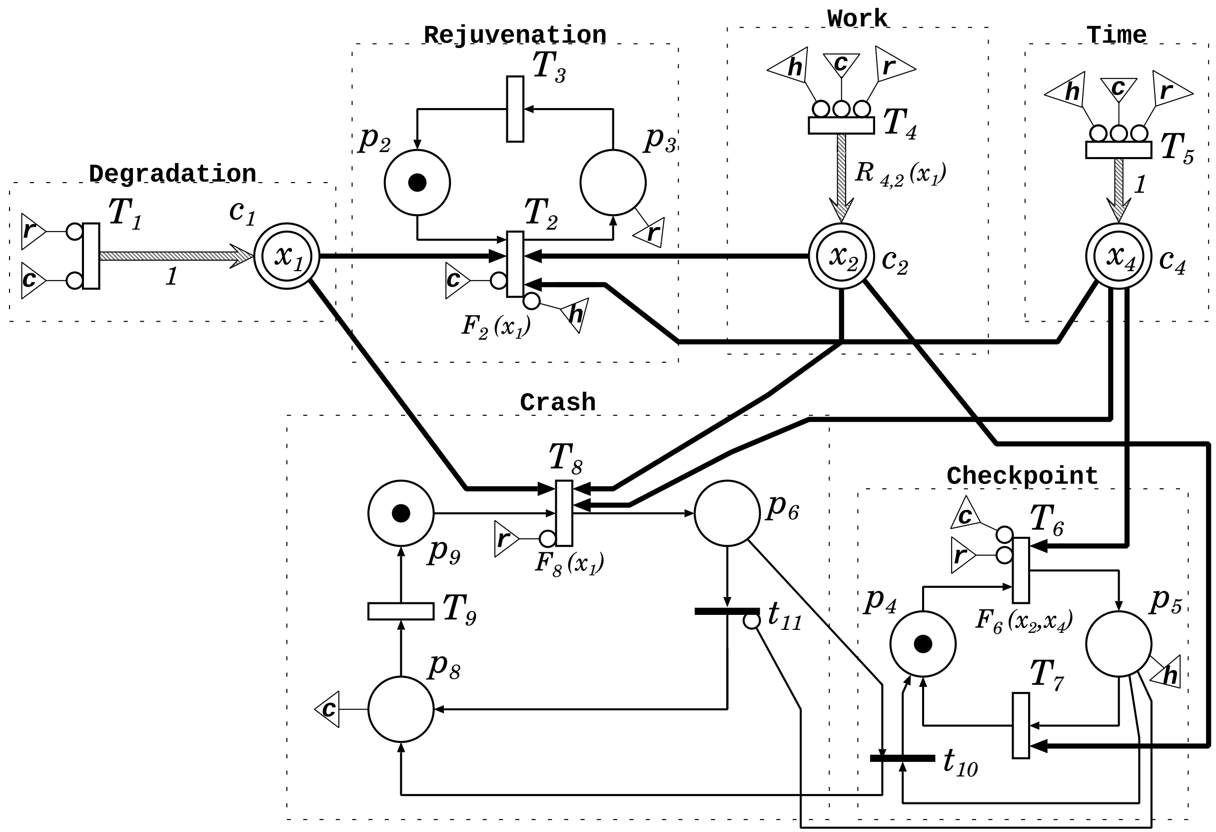
5.3. Software system with Rejuvenation

- Models with four fluid places can only be solved using simulative techniques (with current technologies).

- Fortunately, for most performance indices, the total up-time can be ignored, reducing the number of fluid places to three.

- By some deeper analysis, it can be shown that two of the three remaining fluid places (the work and the time since last checkpoint) are dependent, so one of them can be computed as a function of the other.

5.3. Software system with Rejuvenation



- The figure show a model obtained ignoring the external load and the self-restoration.

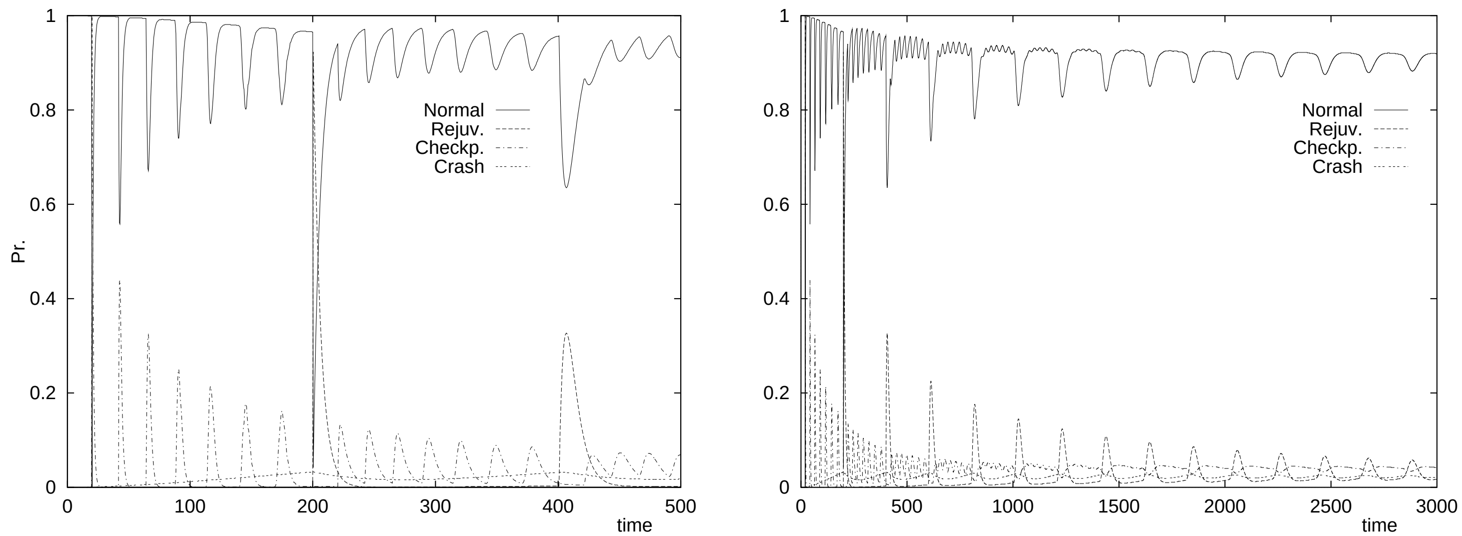
5.3. Software system with Rejuvenation

- A model with two fluid places can be studied using transient analysis techniques.

- From the solution, some interesting performance indices can be integrated.

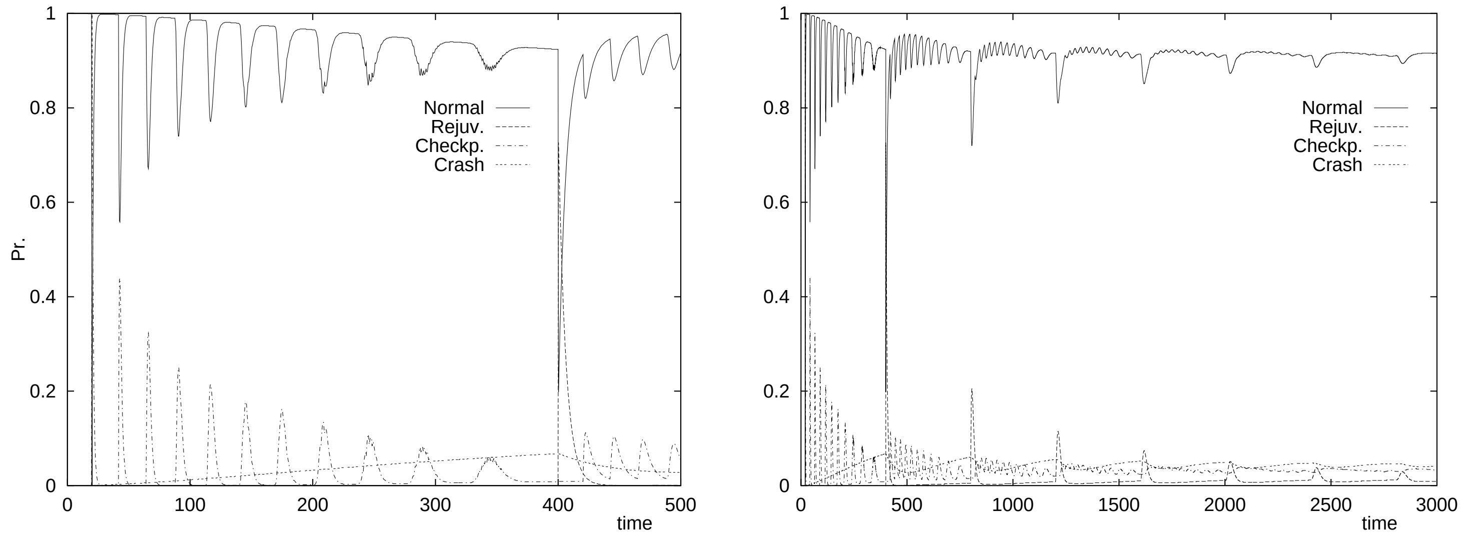
- For example the probability of the various discrete state can be evaluated for different values of the parameters.

5.3. Software system with Rejuvenation



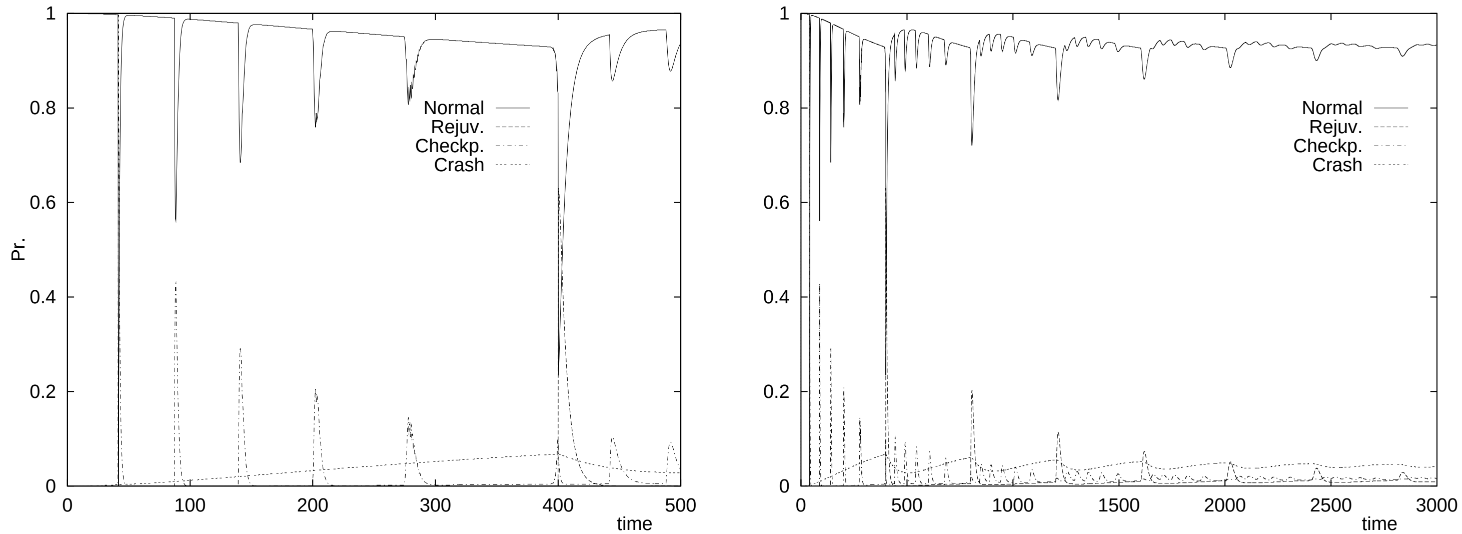
- Case where the mean working time is $\tau_{work} = 200$, and the mean rejuvenation time is $\tau_{rej} = 200$.

5.3. Software system with Rejuvenation



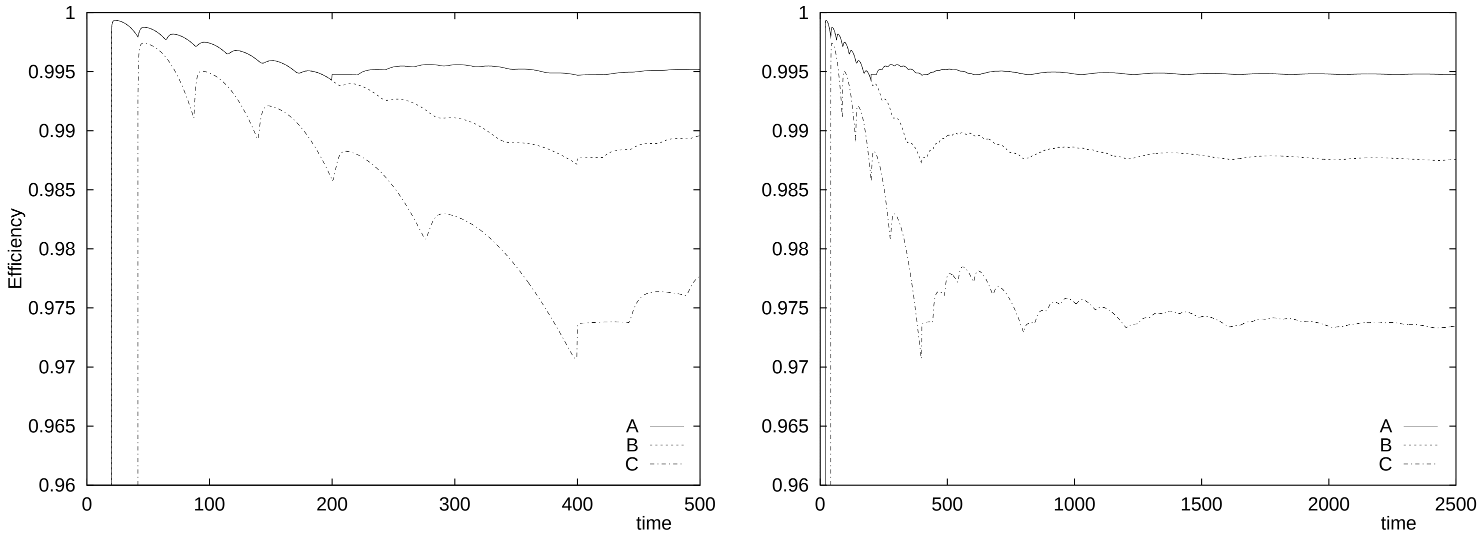
- Case with $\tau_{work} = 200$, and $\tau_{rej} = 400$.

5.3. Software system with Rejuvenation



- Case with $\tau_{work} = 400$, and $\tau_{rej} = 400$.

5.3. Software system with Rejuvenation



$$\text{The efficiency } E(t) = \frac{W_c(t)}{W_c(t) + W_l(t)}$$

$W_c(t)$ represents the average work checkpointed up time t, and $W_l(t)$ the average work lost.

6. Conclusions

- Stochastic models with continuous variables (Reward models, Fluid models and FSPNs) often allows proper modeling of real systems.

- Their analysis is a more complex than the ones of only discrete variables, but feasible for a wide class of models.

- The analytical description of these models and a set of solution techniques have been introduced.

- Some examples of applications demonstrate the potential use of fluid models in performance analysis.

Additional references

B. Sericola, B. Tuffin: A fluid queue driven by a Markovian queue. Queueing Syst. 31(3-4): 253-264 (1999)

B. Sericola: A Finite Buffer Fluid Queue Driven by a Markovian Queue. Queueing Syst. 38(2): 213-220 (2001)

S. Ahn and V. Ramaswami. Fluid Flow Models and Queues - A Connection by Stochastic Coupling. Stochastic Models, 19(3):325–348, 2003.

Q. Ren and H. Kobayashi, Transient solutions for the buffer behavior in statistical multiplexing, Performance Evaluation 23 (1995), 6587.

V. G. Kulkarni, Fluid models for single buffer systems, Frontiers in Queueing (J. H. Dshalalow, ed.), Probab. Stochastics Ser., CRC, Florida, 1997, pp. 321338.

R. German, M. Gribaudo, G. Horváth, and M. Telek, “Stationary analysis of FSPNs with mutually dependent discrete and continuous parts,” in *International Conference on Petri Net Performance Models – PNPM 2003*, (Urbana, IL, USA), pp. 30–39, IEEE CS Press, Sept 2003.

A. da Silva Soares and G. Latouche. Matrix analytic methods for fluid queues with finite buffers. Performance Evaluation. 2005

M. Ajmone Marsan, G. Balbo, G. Conte, S. Donatelli, and G. Franceschinis. *Modelling with Generalized Stochastic Petri Nets*. 1995.

M. Ajmone Marsan, M. Gribaudo, M. Meo, and M. Sereno. On Petri Net-Based Modeling Paradigms for the Performance Analysis of Wireless Internet Accesses.

In *Proceeding of Nineth International Workshop on Petri Nets and Performa nce Models 2001, PNPM'01*, pages 19–28, Aachen, Germany, Sep 2001.

H. Alla and R. David. Continuous and Hybrid Petri Nets. *Journal of Systems Circuits and Computers*, 8(1), Feb 1998.

D Anick, D. Mitra, and M. M. Sondhi. Stochastic Theory of a Data-Handling System. *Bell Sys. Thech. J*, 61(8):1871–1894, Oct 1982.

A. Bobbio, S. Garg, M. Gribaudo, A. Horváth, M. Sereno, and M. Telek. Modeling Software Systems with Rejuvenation, Restoration and Checkpointing through Fluid Stochastic Petri Nets. In *Proc.* 8^{th} *Intern. Workshop on Petri Nets and Performance Models*, Zaragoza, Spain, Sep 1999. IEEE-CS Press.

G. Ciardo, J. K. Muppala, and K. S. Trivedi. On the solution of GSPN reward models. *Performance Evaluation*, 12(4):237–253, 1991.

T. Czachorski and F. Pekergin. Diffusion approximation as a modelling tool. In *Tutorial Papers of the ATM & IP 2000*, pages 18/1 – 18/40, Ilkley, UK, Jul 2000.

A. I. Elwalid and D. Mitra. Statistical Multiplexing with Loss Priorities in Rate-Based Congestion Control of High-Speed Networks. *IEEE Transaction on Communications*, 42(11):2989–3002, November 1994.

S. Garg, A. Puliafito, M. Telek, and K. S. Trivedi. Analysis of preventive maintenance in transaction based software systems. *IEEE Trans. on Computers*, 47(1):96–107, 1998. Special issue on Dependability of Computing Systems.

R. German. Markov Regenerative Stochastic Petri Nets with general execution policies: Supplementary variable analysis and a prototype tool. In *In Proc.* 10^{th} *Int. Conf. on Modelling Techniques and Tools for Computer Performance Evaluation*, pages

255–266. Springer Verlag - LNCS, Vol. 1469, 1998.

M. Gribaudo and R. German. Numerical Analysis of Bounded Fluid Models using Matrix Exponentiation. In *Proceeding of Eleventh GI/ITG Conference on Measuring, Modeling and Evaluation of Computer and Communication Systems, MMB'01*, pages 41–57, Aachen, Germany, Sep 2001.

M. Gribaudo and A. Horvath. Fluid Stochastic Petri Nets Augmented with Flush-out Arcs: a Transient Analysis Technique. In *Proceeding of Nineth International Workshop on Petri Nets and Performa nce Models 2001, PNPM'01*, pages 145–154, Aachen, Germany, Sep 2001.

M. Gribaudo and M. Sereno. Simulation of Fluid Stochastic Petri Nets. In *Proc. of MASCOTS'2000*, pages 231–239, San Francisco, CA, Aug 2000.

M. Gribaudo, M. Sereno, and A. Bobbio. Fluid Stochastic Petri Nets: An Extended Formalism to Include non-Markovian Models. In *Proc.* 8^{th} *Intern. Workshop on Petri Nets and Performance Models*, Zaragoza, Spain, Sep 1999.

M. Gribaudo, M. Sereno, A. Bobbio, and A. Horvath. Fluid Stochastic Petri Nets augmented with Flush-out arcs: Modelling and Analysis. *Discrete Event Dynamic Systems*, 11(1 & 2), 2001.

G. Horton, V. G. Kulkarni, D. M. Nicol, and K. S. Trivedi. Fluid stochastic Petri Nets: Theory, Application, and Solution Techniques. *European Journal of Operations Research*, 105(1):184–201, Feb 1998.

R. J. Karandikar and V. G. Kulkarni. Second-Order Fluid Flow Models: Reflected Brownian Motion in a Random Environment. *Operations Research*, 43(1):77–88, 1995.

G. Latouche and V. Ramaswami. *Introduction to Matrix Geometric Methods in Stochastic Modeling*. ASA-SIAM Series on Statistics and Applied Probability. SIAM, Philadelphia PA, 1999.

P. A. W. Lewis and G. S. Shedler. Simulation of nonhomogeneous Poisson processes by thinning. *Naval Research Logistic Quarterly*, 26:403–414, 1979.

D. Mitra. Stochastic Theory of a Fluid Model of Producers and Consumers. *Adv. Appl. Prob.*, 20:646–676, 1988.

C. Moller and C. Van Loan. Nineteen Dubious Ways to Compute the Exponential of a Matrix. *SIAM Review*, 20(4):801–836, Oct 1978.

B. Sericola. Transient analysis of stochastic fluid models. *Performance Evaluation*, 1(32):245–263, May 1998.

K. Trivedi and V. Kulkarni. FSPNs: Fluid Stochastic Petri nets. In *Application and Theory of Petri Nets 1993, Proc.* 14^{th} *Intern. Conference*, LNCS, Chicago, USA, June 1993. Springer Verlag.

K. Wolter. Second order fluid stochastic petri nets: an extension of gspns for approximate and continuous modelling. In *Proc. of World Congress on System Simulation*, pages 328–332, Singapore, Sep 1997.

K. Wolter. Jump Transitions in Second Order FSPNs. In *Proc. of MASCOTS'99*, Washington, DC, Oct 1999.

R. Gaeta, M. Gribaudo, D. Manini, and M. Sereno. Analysis of Resource Transfers in Peer-to-Peer File Sharing Applications using Fluid Models, to appear in Performance Evaluation, 2005