Semi-Markov PEPA

Jeremy Bradley

NeSC, Edinburgh – 12 June 2003

What is PEPA?

- a stochastic process algebra
- Markovian or exponential distributions
- fast, sleek, no reward cards, spreads straight from the fridge

PEPA: Process Algebra

- Syntax:

$$P \quad ::= \quad (a,\lambda).P \;\Big|\; P + P \;\Big|\; P \underset{S}{\bowtie} P \;\Big|\; P/L \;\Big|\; A$$

- $(a,\lambda).P$: prefix operation $\qquad P + P$: competitive choice

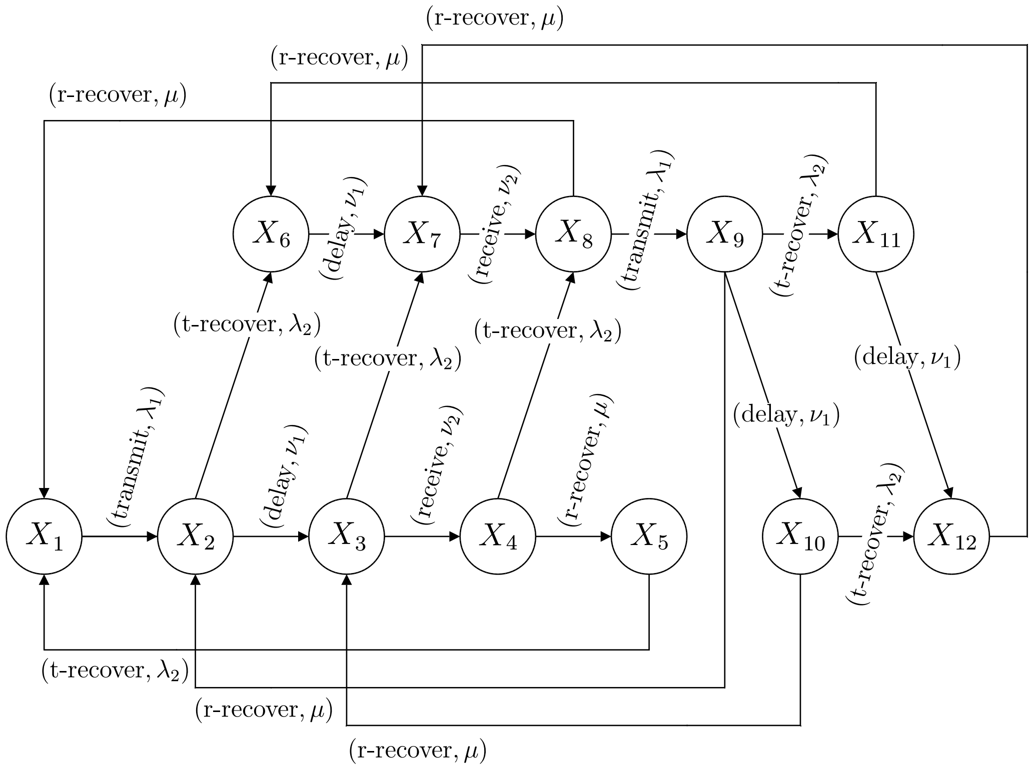
- $P \underset{S}{\bowtie} P$: component cooperation $\qquad P/L$: action hiding

PEPA: Example

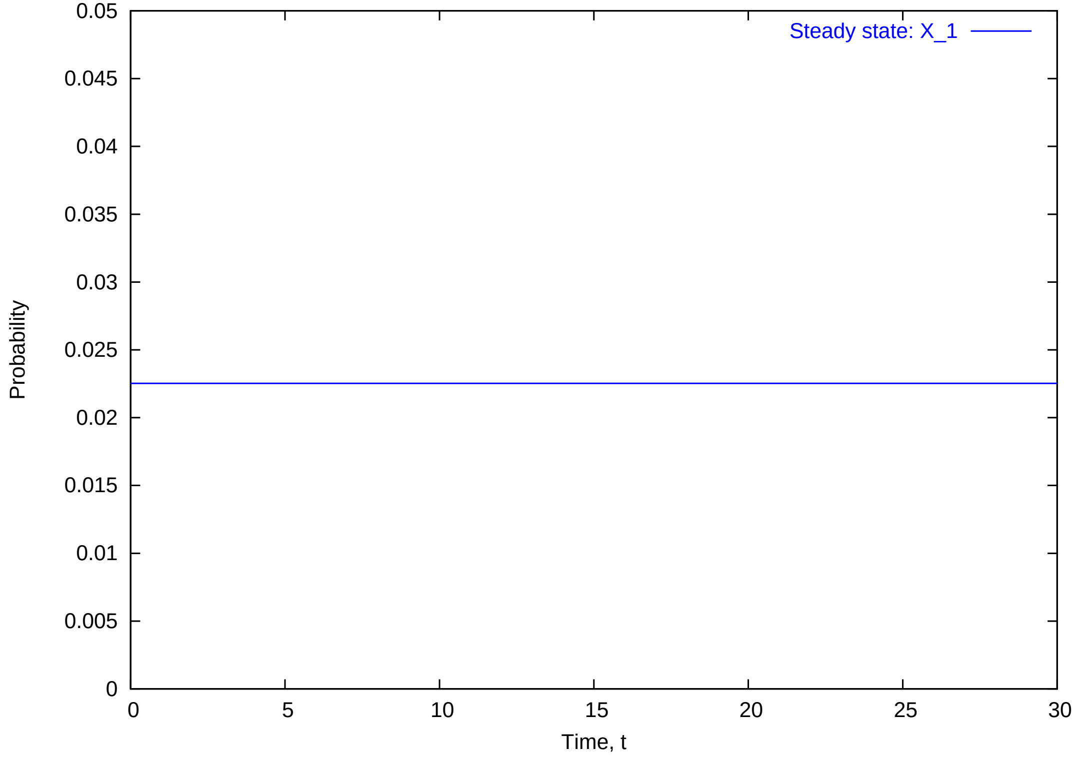
- A simple transmitter-receiver network:

$$
\begin{aligned}
Transmitter &\stackrel{\mathsf{def}}{=} (\mathsf{transmit}, \lambda_1).(\mathsf{t\text{-}recover}, \lambda_2).Transmitter \\
Receiver &\stackrel{\mathsf{def}}{=} (\mathsf{receive}, \top).(\mathsf{r\text{-}recover}, \mu).Receiver \\
Network &\stackrel{\mathsf{def}}{=} (\mathsf{transmit}, \top).(\mathsf{delay}, \nu_1).(\mathsf{receive}, \nu_2).Network \\
System &\stackrel{\mathsf{def}}{=} (Transmitter \underset{\emptyset}{\bowtie} Receiver) \underset{\{\mathsf{transmit},\mathsf{receive}\}}{\bowtie} Network
\end{aligned}
$$

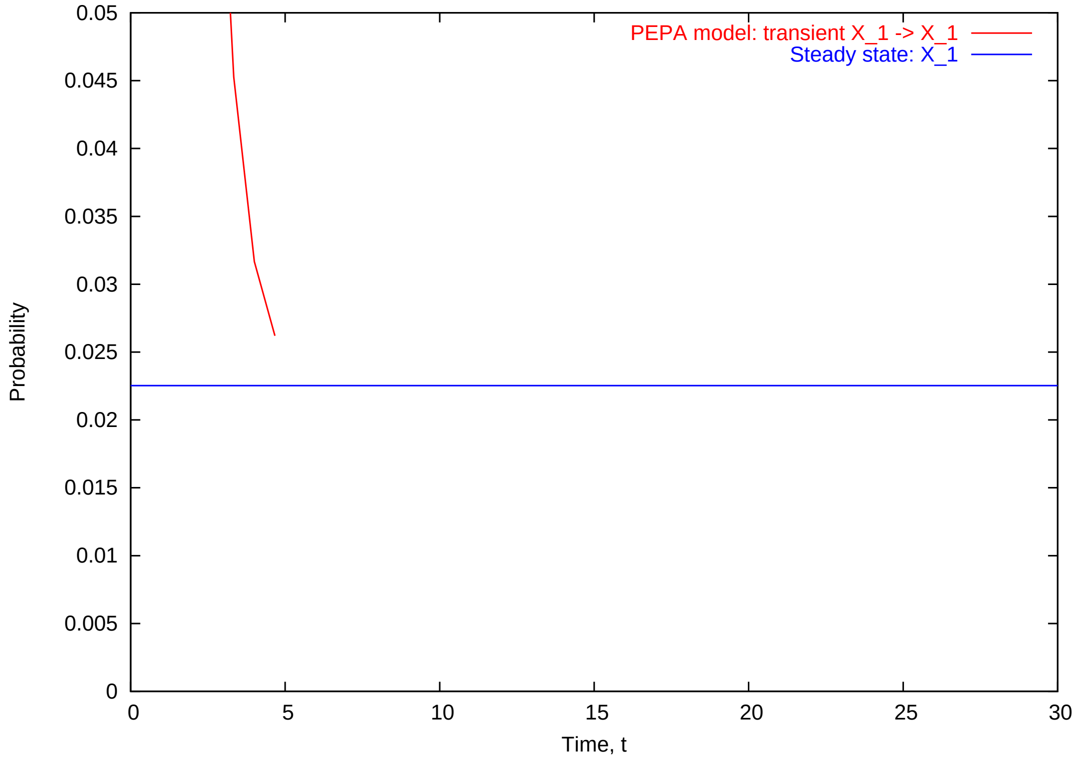
Global State Space



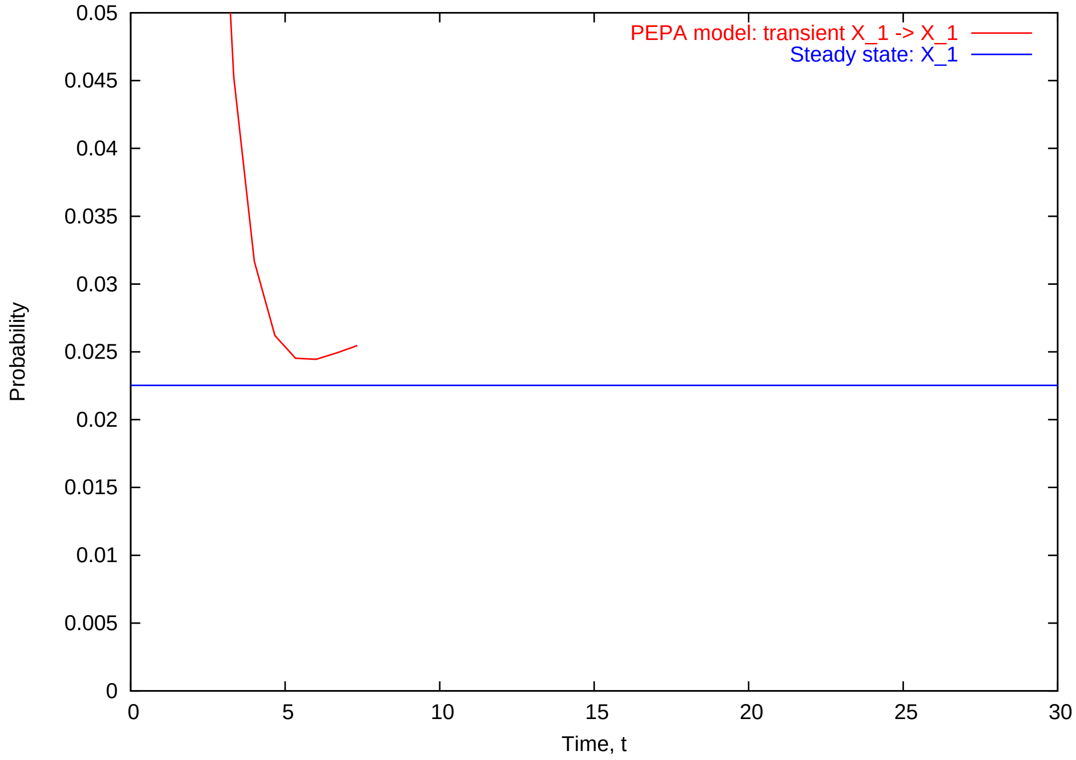
Transient and Steady State



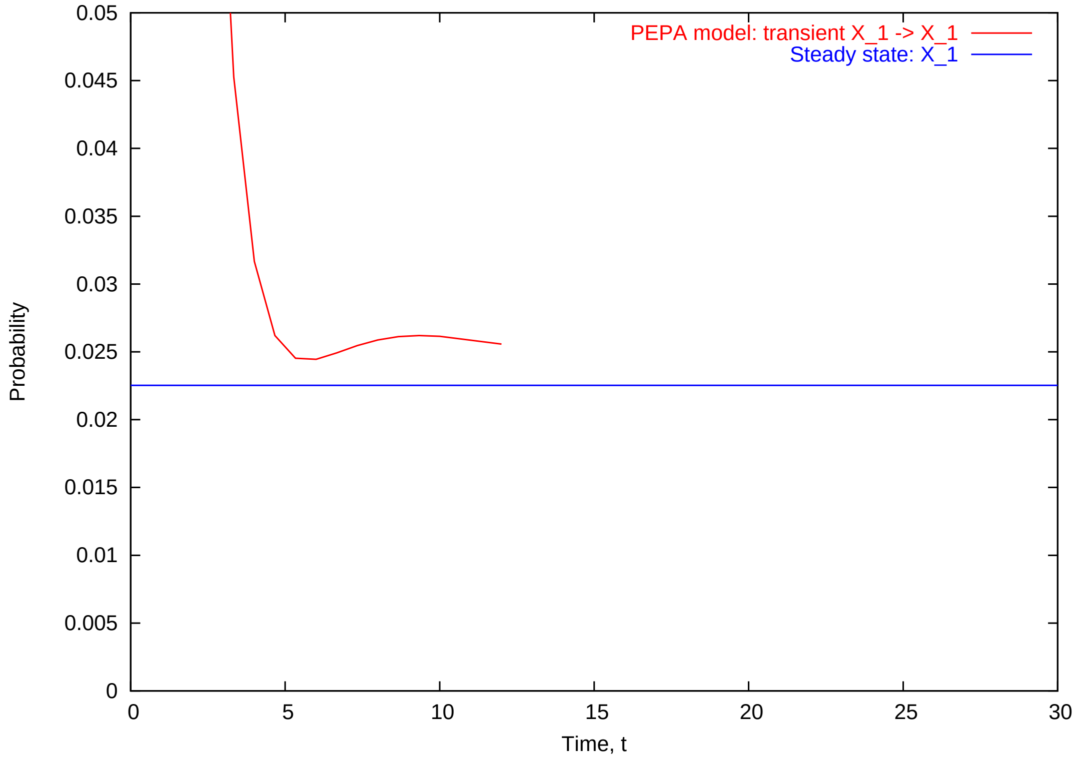
Transient and Steady State



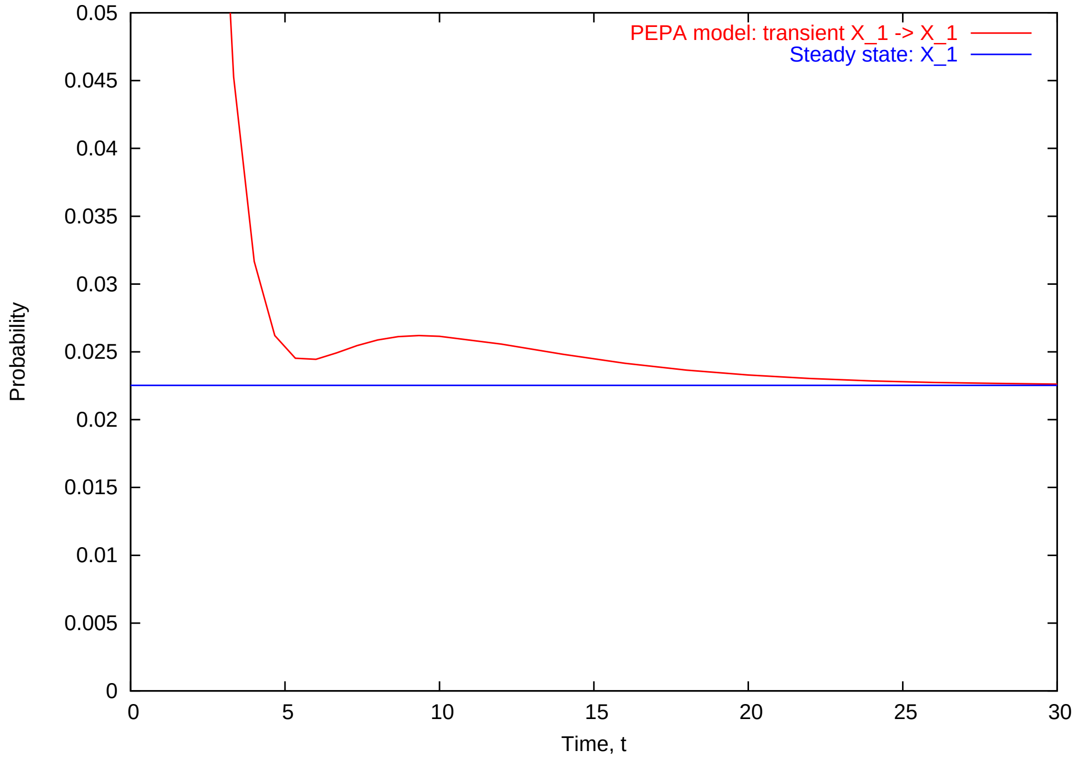
Transient and Steady State



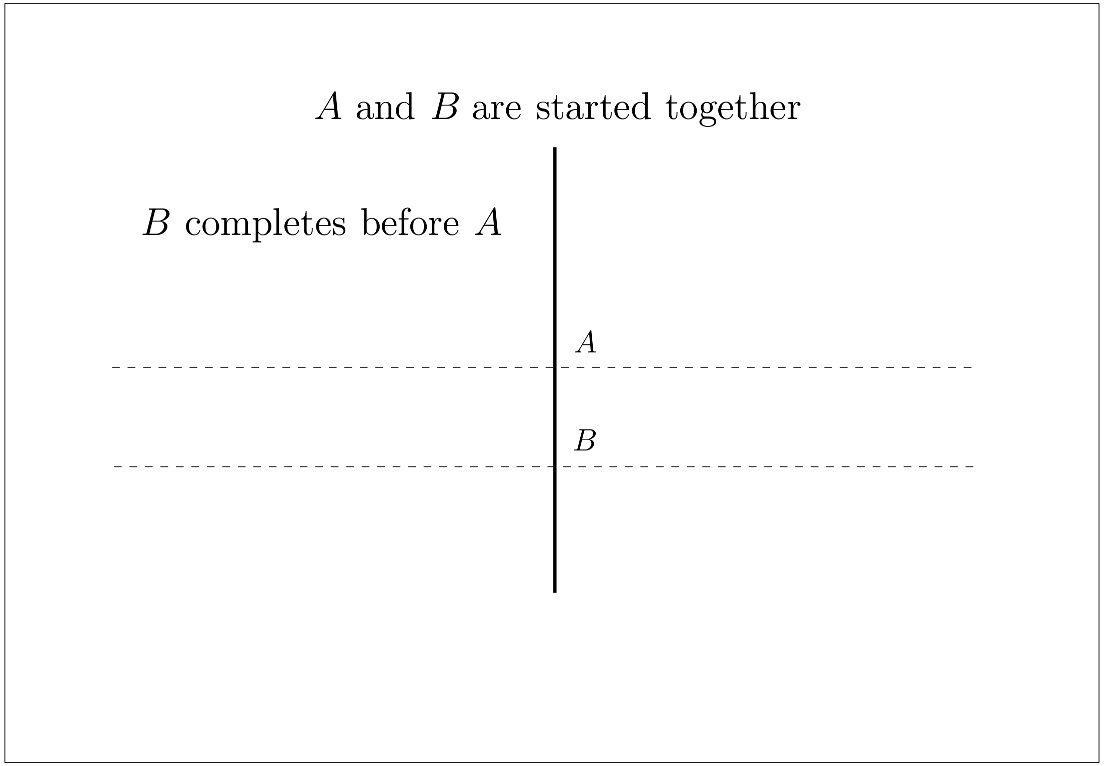
Transient and Steady State



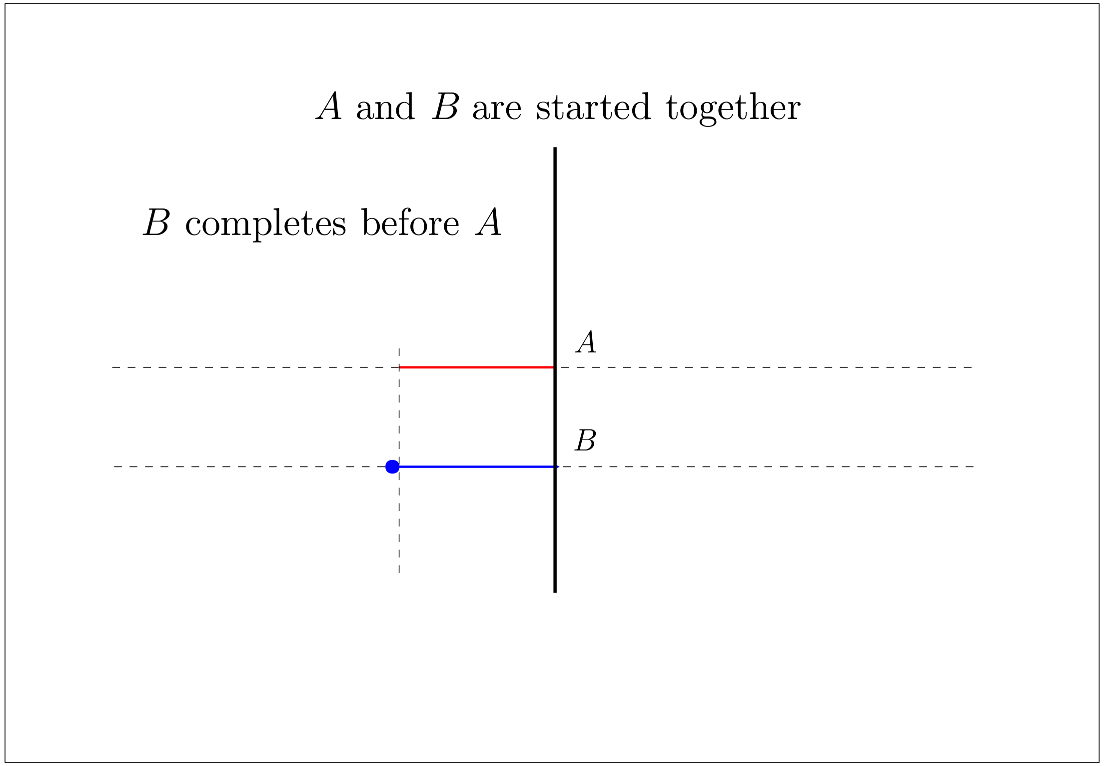
Transient and Steady State



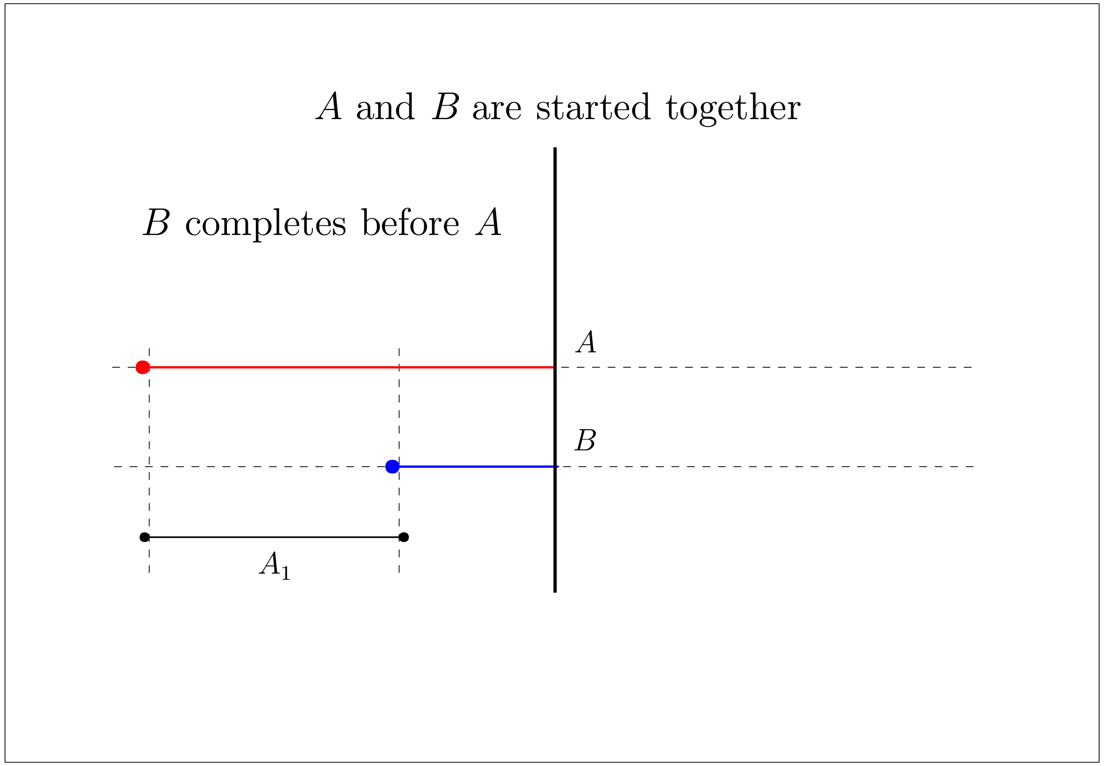
Concurrent Interleaving



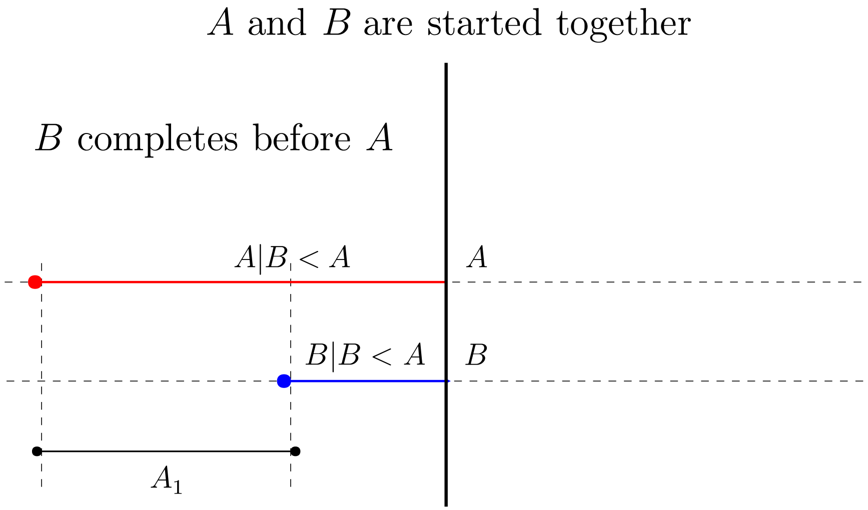
Concurrent Interleaving



Concurrent Interleaving

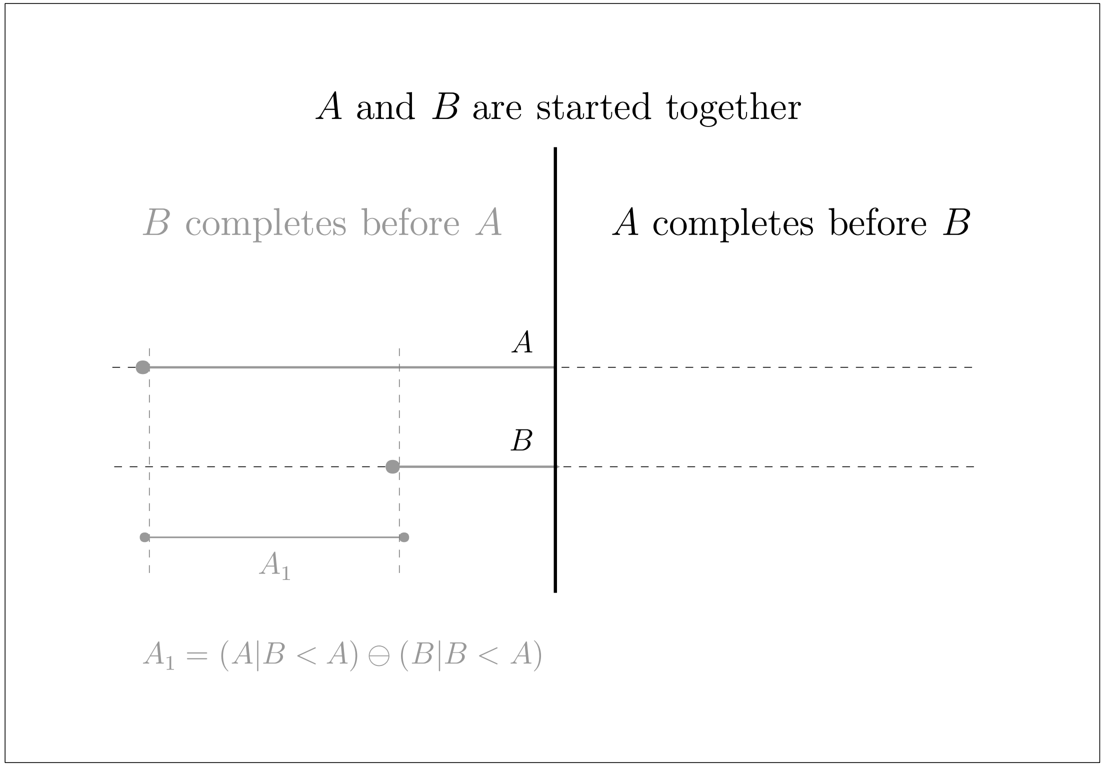


Concurrent Interleaving

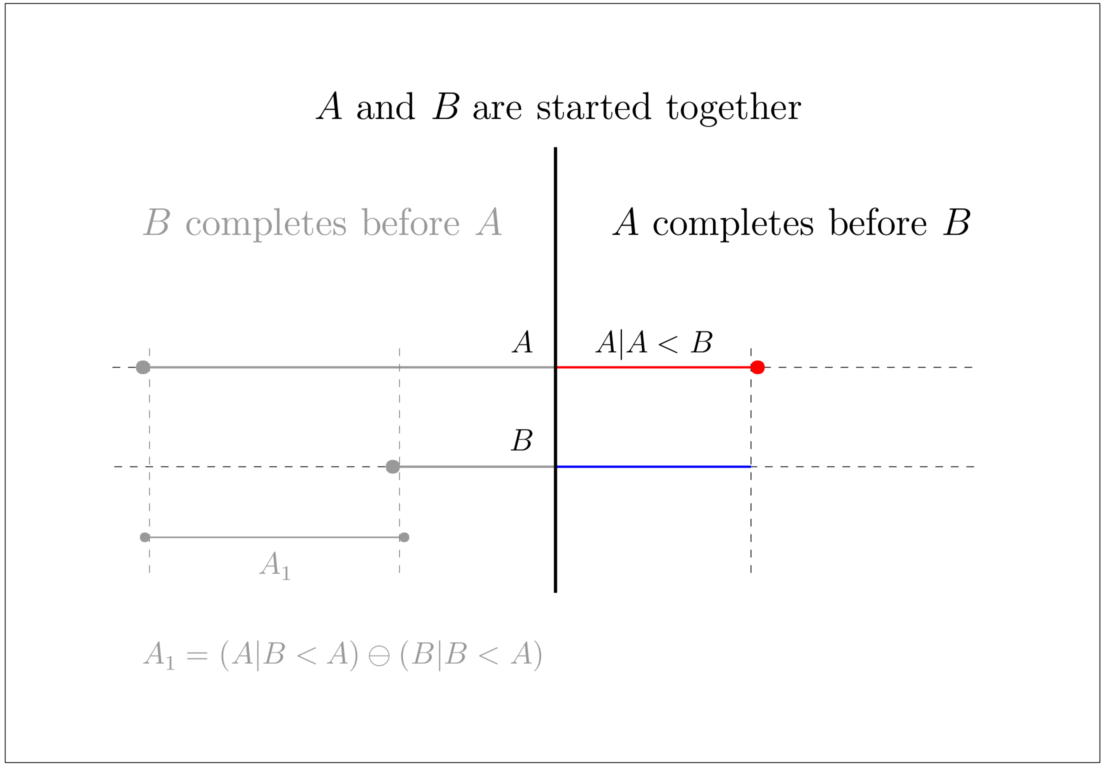


$A_1 = (A|B < A) \ominus (B|B < A)$

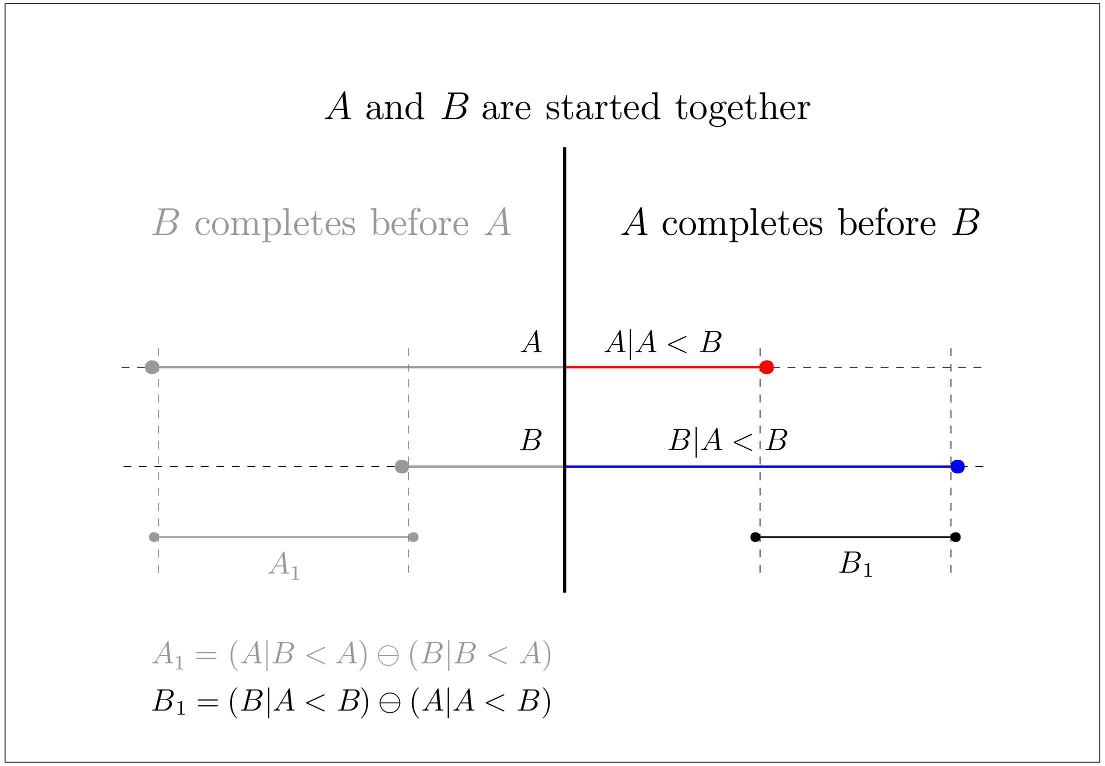
Concurrent Interleaving



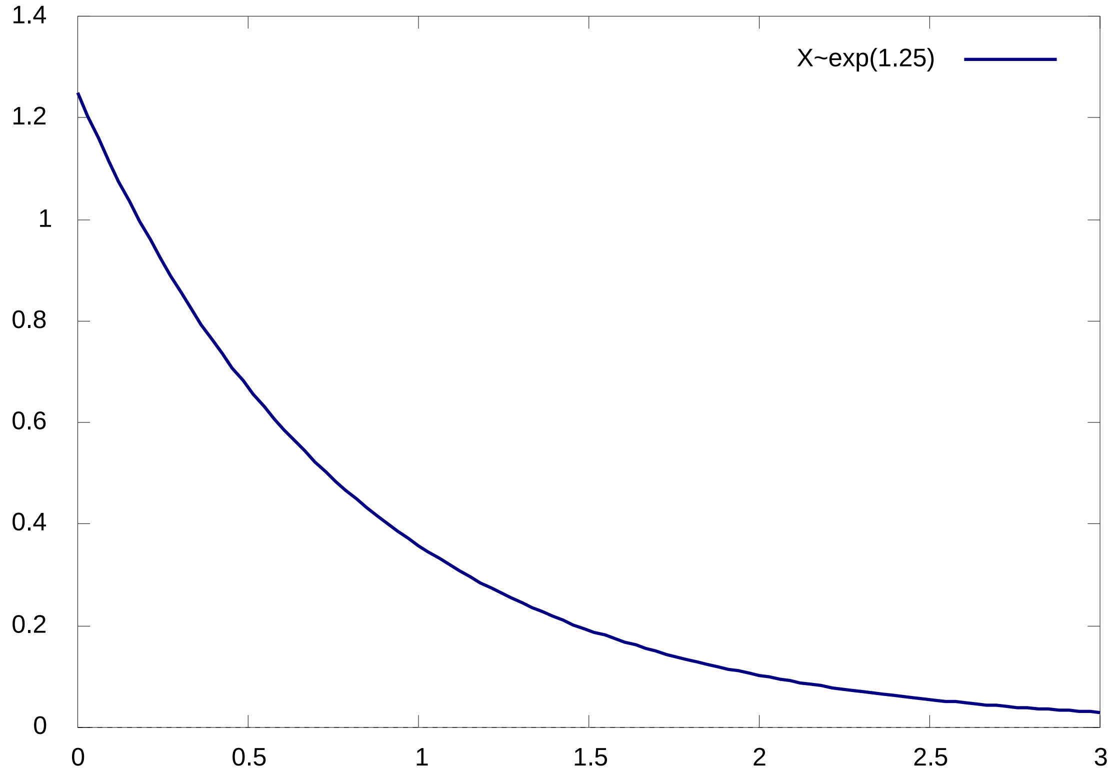
Concurrent Interleaving



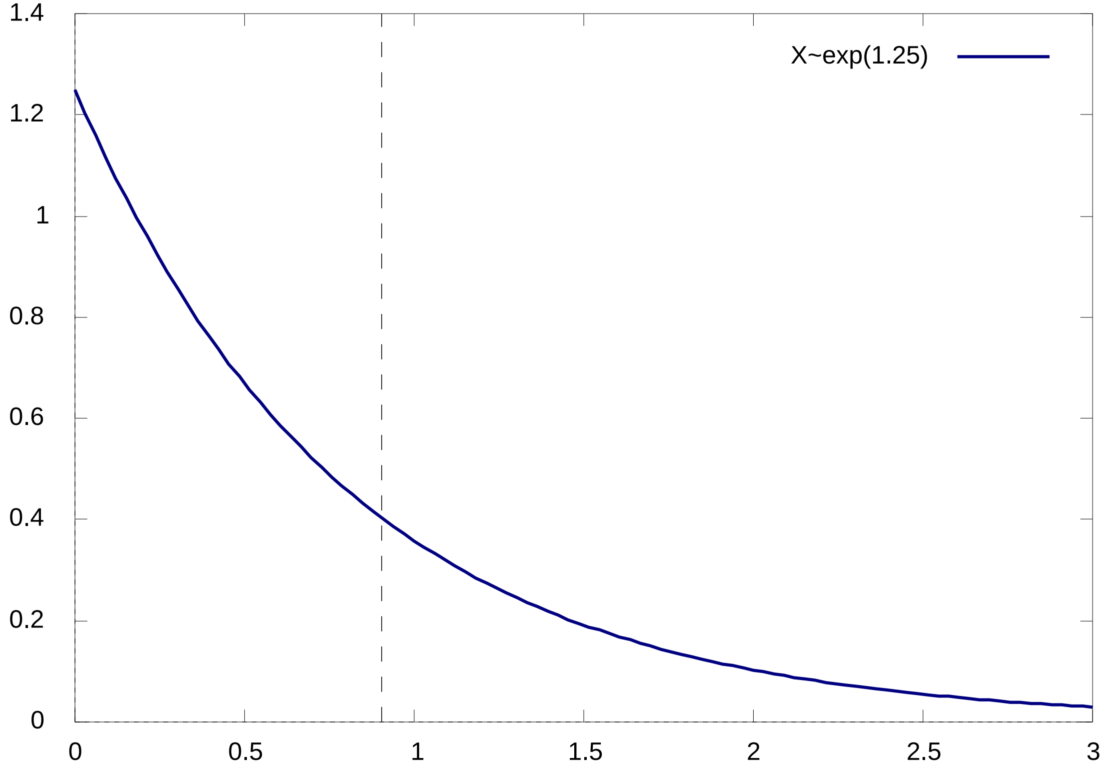
Concurrent Interleaving



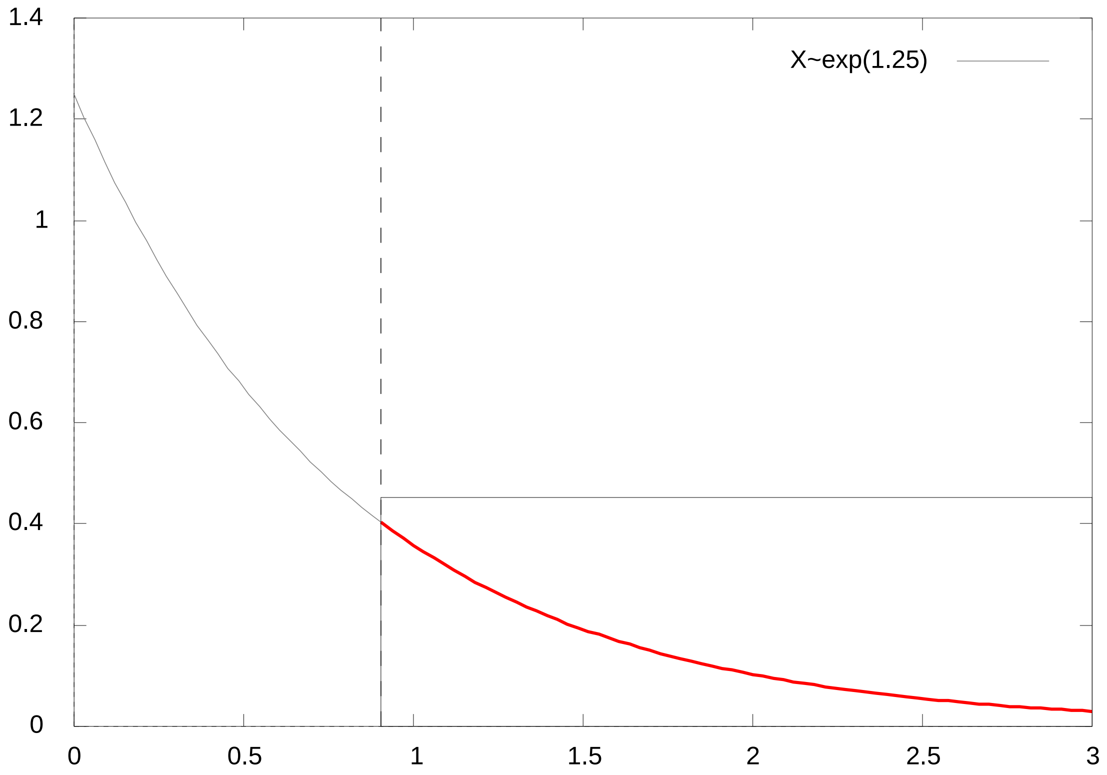
Exponential memorylessness



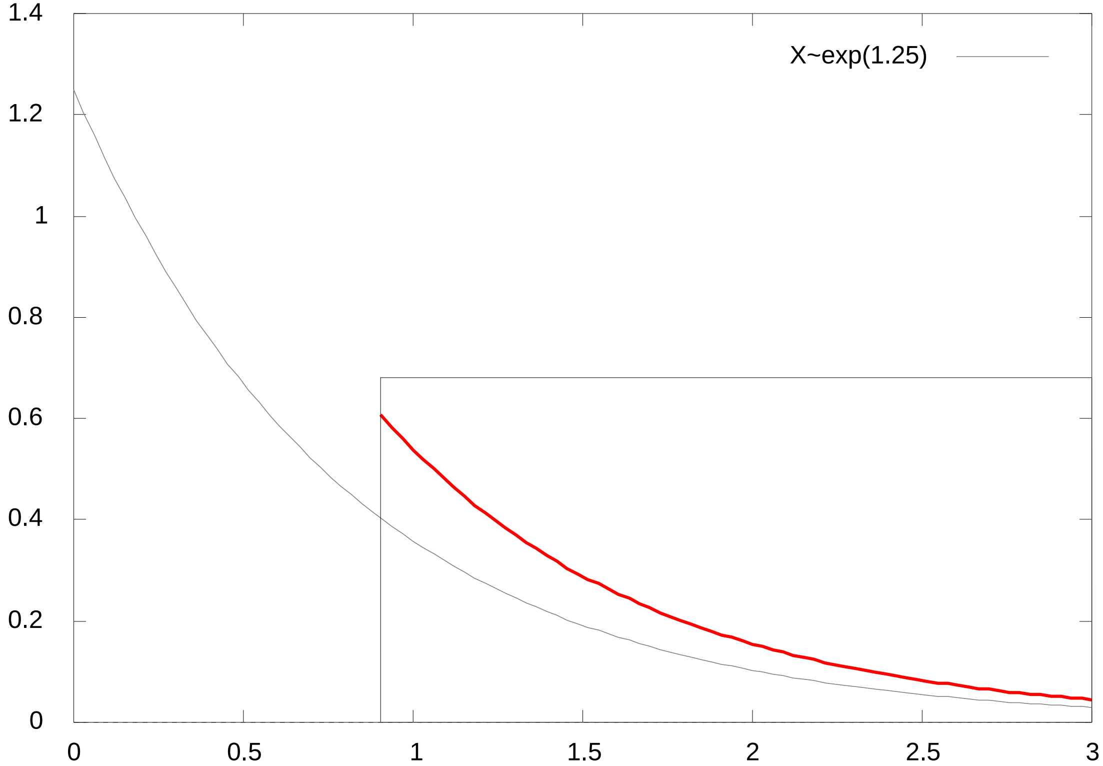
Exponential memorylessness



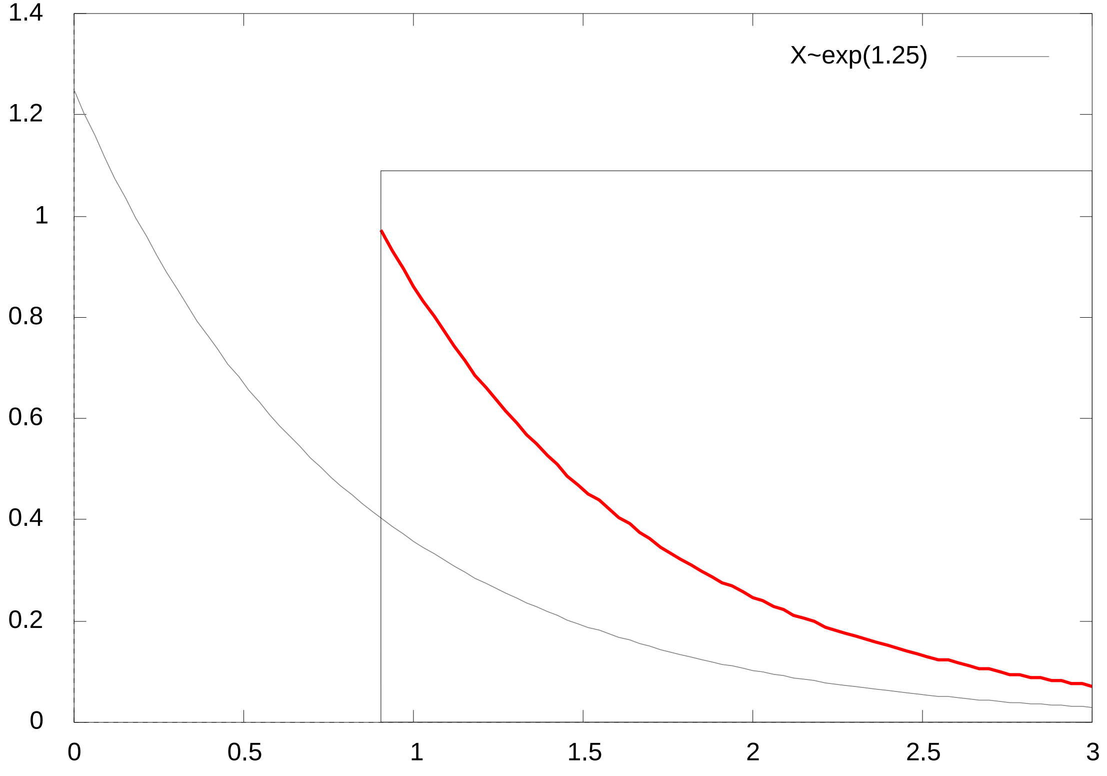
Exponential memorylessness



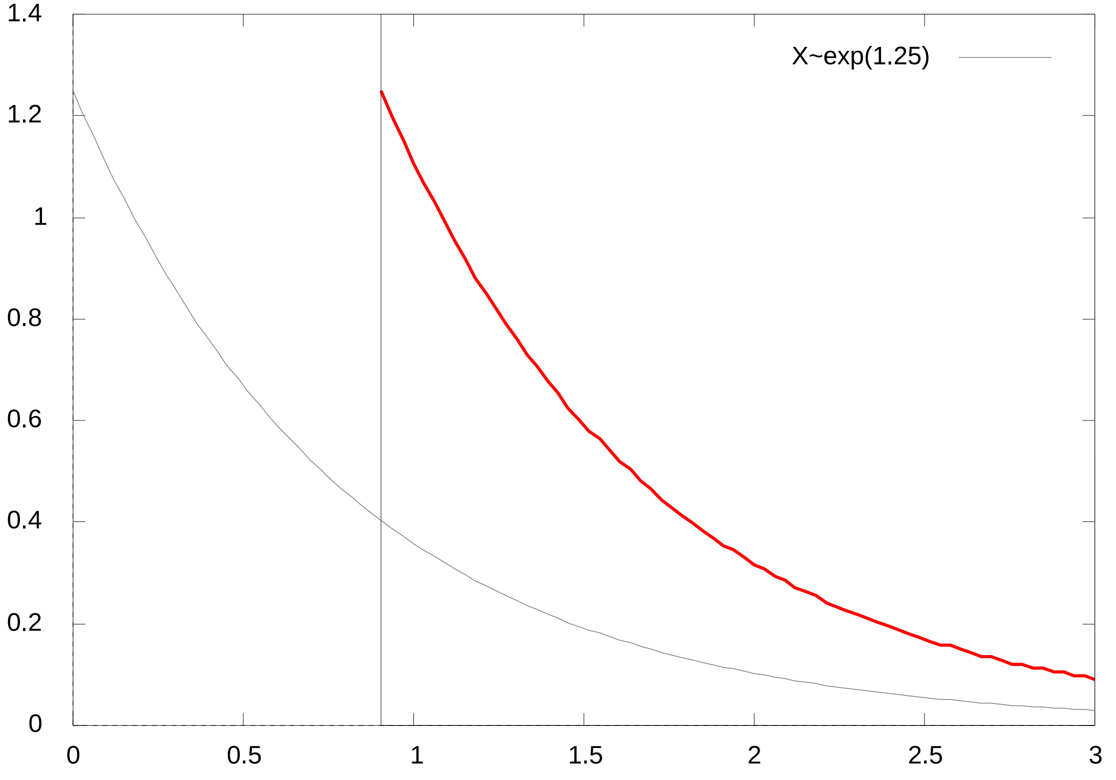
Exponential memorylessness



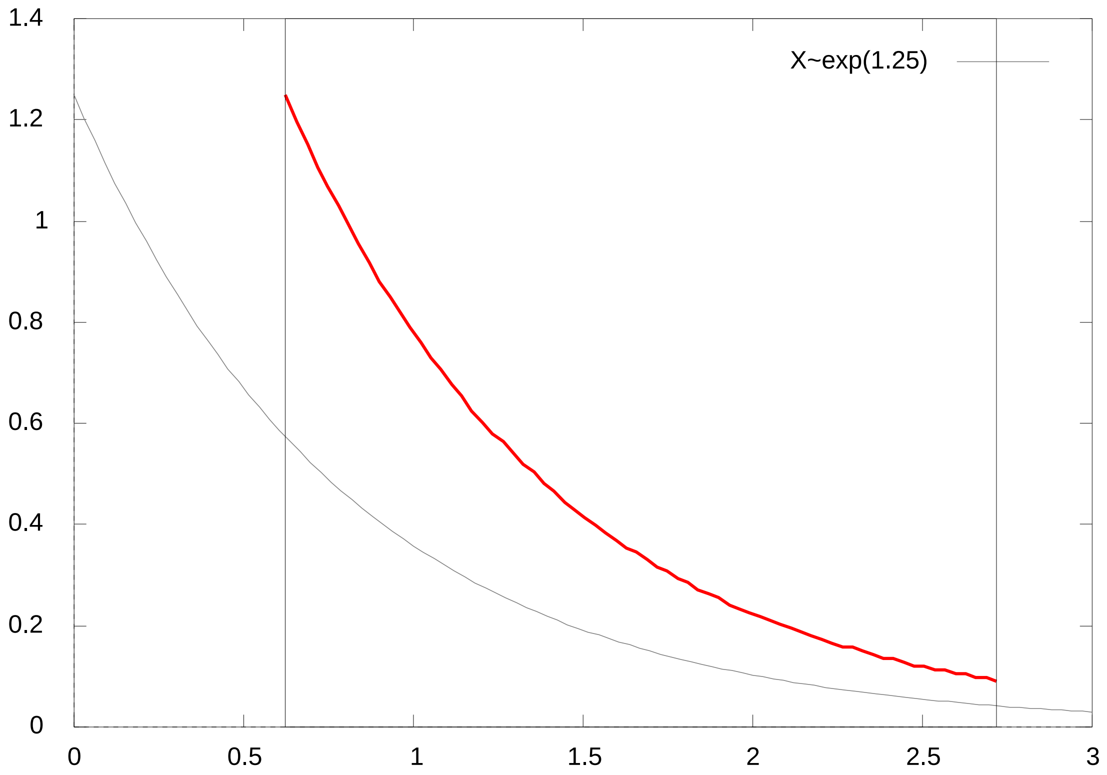
Exponential memorylessness



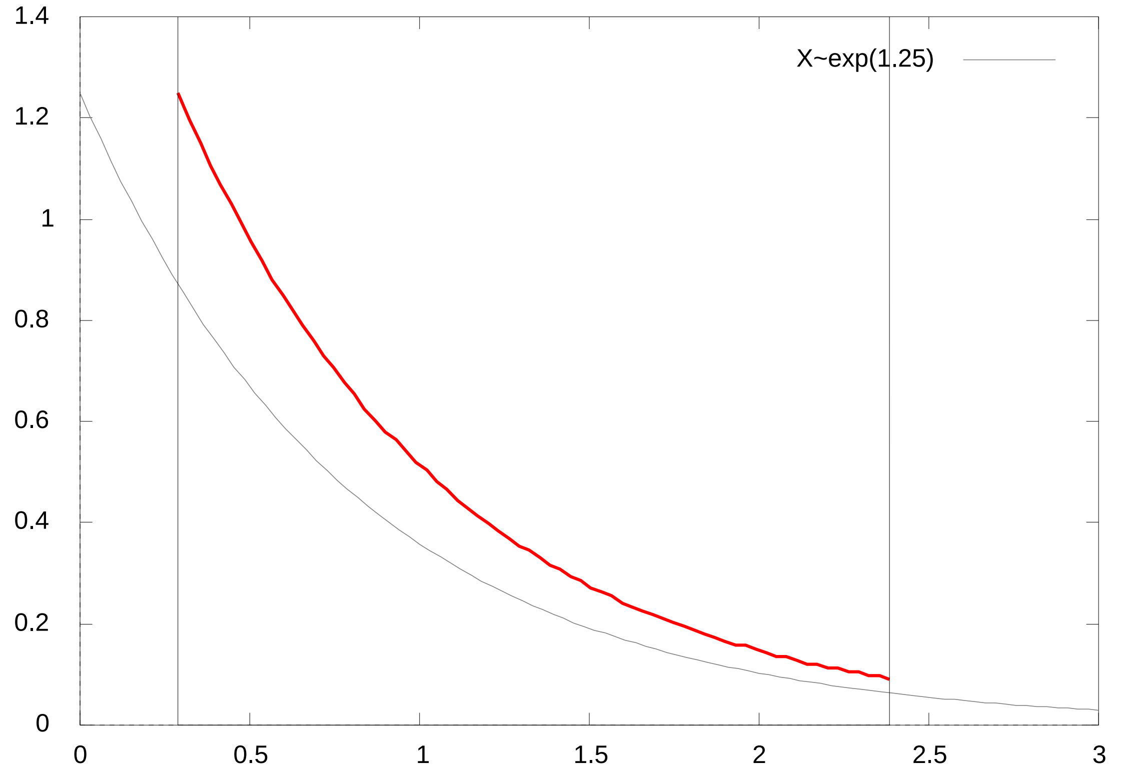
Exponential memorylessness



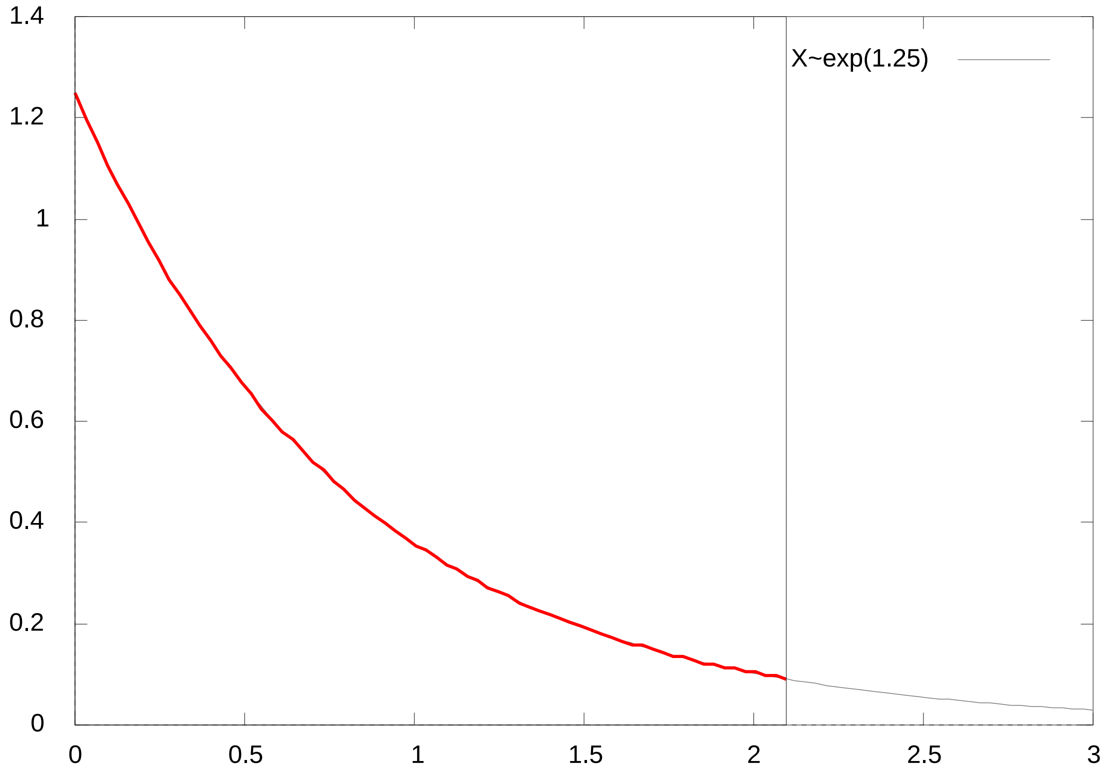
Exponential memorylessness



Exponential memorylessness



Exponential memorylessness



Semi-Markov Processes

- Semi-Markov processes use arbitrary distributions
- But no support for competitive choice or concurrent execution
- So what use are semi-Markov processes as an underlying formalism for PEPA?

PEPA and SMPs?

Two motivations:

1. Systems with Markovian concurrency that have areas of mutual exclusion
2. Fully generally distributed concurrent design, but run on a single threaded architecture

Semi-Markov PEPA

- Syntax:

$$P \ ::= \ (a, D).P \,\big|\, P + P \,\big|\, P \underset{S}{\bowtie} P \,\big|\, P/L \,\big|\, A$$

$$D \ ::= \ \lambda \,\big|\, \omega : L(s)$$

- new prefix operator: $(a, D).P$
  - λ: normal exponential rate parameter
  - $\omega : L(s)$: a selection weight, ω, and a general distribution description, $L(s)$

Semi-Markov Example I

- Mutual exclusion modelling:

$$\begin{aligned} A &\stackrel{\text{def}}{=} (\text{think}, \lambda_1).(\text{recover}, \lambda_2).A \\ &\quad + (\text{error}, \lambda_3).(\text{mutex}, 1 : L_1(s)).A \end{aligned}$$

$$S_n \stackrel{\text{def}}{=} \underbrace{A \underset{\emptyset}{\bowtie} A \underset{\emptyset}{\bowtie} \cdots \underset{\emptyset}{\bowtie} A}_{n}$$

- Areas of Markovian concurrency interspersed with semi-Markov sequential behaviour

Semi-Markov Example II

- Web-server/database model:

$$
\begin{aligned}
Server &\stackrel{\text{def}}{=} (\mathsf{get}, 5 : exp(1.5, s))).Server_1 \\
Server_1 &\stackrel{\text{def}}{=} (\mathsf{static\_page}, 1 : det(3, s)).Server \\
&\quad + (\mathsf{dbase\_fetch}, 2 : gamma(2.2, 3.2, s)).Server_2 \\
Server_2 &\stackrel{\text{def}}{=} (\mathsf{dbase\_rtn}, \top).(\mathsf{dynamic\_page}, 1 : uniform(2, 5, s)).Server \\
Dbase &\stackrel{\text{def}}{=} (\mathsf{dbase\_fetch}, \top).(\mathsf{dbase\_rtn}, 4 : exp(2.3, s)).Dbase \\
Sys &\stackrel{\text{def}}{=} Server \underset{\{\mathsf{dbase\_fetch},\mathsf{dbase\_rtn}\}}{\bowtie} Dbase
\end{aligned}
$$

- Concurrent design. Single-threaded architecture with weighted process selection

Some Conclusions

- Proposal for semi-Markov PEPA

- Incorporates PEPA functionality as a subset

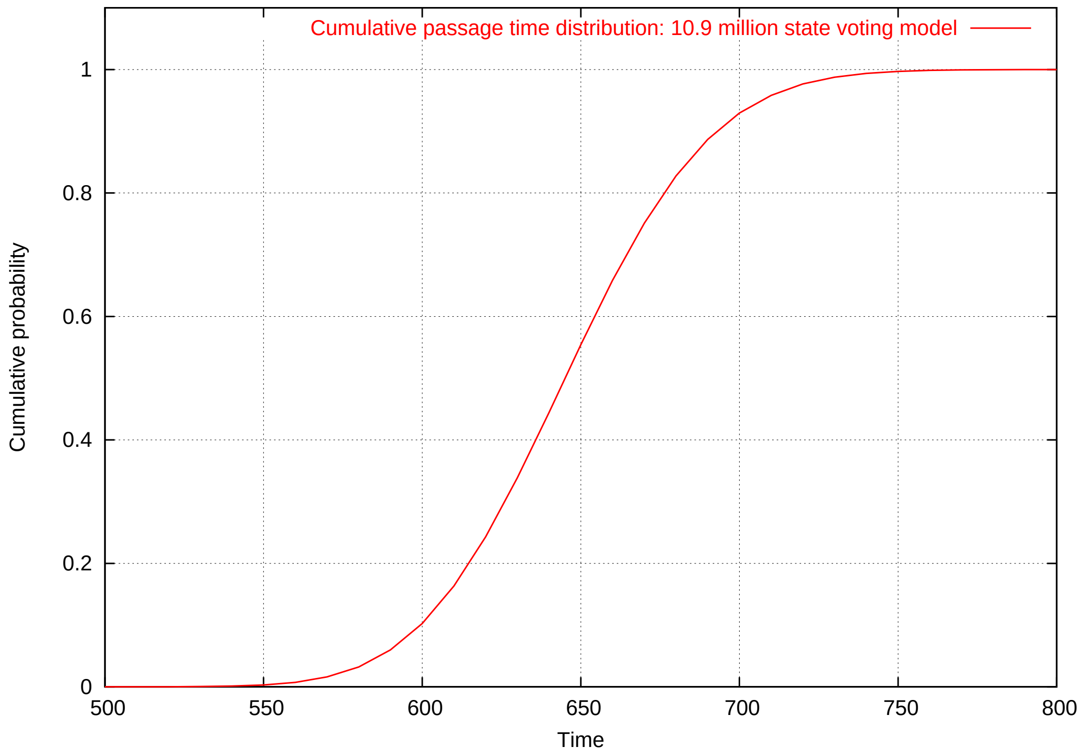
- Has 2 genuine application areas

Tool Support

- ipc: PEPA to DNAmaca

- SM-SPN DNAmaca
  - Transient distributions
  - Passage-time distributions

Semi-Markov Passage-time



Semi-Markov Passage-time