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:

\[ P ::= (a, \lambda).P \mid P + P \mid P \boxdot P \mid P/L \mid A \]

• \((a, \lambda).P\): prefix operation \hspace{1cm} \(P + P\): competitive choice

• \(P \boxdot P\): component cooperation \hspace{1cm} P/L: action hiding
PEPA: Example

- A simple transmitter-receiver network:

\[
\begin{align*}
Transmitter & \overset{\text{def}}{=} (\text{transmit}, \lambda_1).(t\text{-recover}, \lambda_2).\text{Transmitter} \\
Receiver & \overset{\text{def}}{=} (\text{receive}, T).(r\text{-recover}, \mu).\text{Receiver} \\
Network & \overset{\text{def}}{=} (\text{transmit}, T).(\text{delay}, \nu_1).(\text{receive}, \nu_2).\text{Network} \\
System & \overset{\text{def}}{=} (Transmitter \downarrow Receiver) \downarrow Network
\end{align*}
\]
Global State Space
Transient and Steady State

![Graph showing probability vs time with a horizontal line at steady state]
Transient and Steady State

PEPA model: transient \( X_1 \rightarrow X_1 \)
Steady state: \( X_1 \)
Transient and Steady State

PEPA model: transient $X_1 \rightarrow X_1$

Steady state: $X_1$

![Graph showing transient and steady state probabilities over time](image-url)
Transient and Steady State

PEPA model: transient $X_1 \rightarrow X_1$

Steady state: $X_1$
Transient and Steady State

PEPA model: transient $X_1 \rightarrow X_1$
Steady state: $X_1$
Concurrent Interleaving

\[ \begin{array}{c|c}
A \text{ and } B \text{ are started together} & \\
\hline
\begin{array}{c}
B \text{ completes before } A \\

\begin{array}{c}
\hline
A \\
\hline
B
\end{array}
\end{array}
\end{array} \]
Concurrent Interleaving

\[ A \text{ and } B \text{ are started together} \]

\[ \text{B completes before A} \]

\[ A \]

\[ B \]
Concurrent Interleaving

A and B are started together

B completes before A

A

B

A_1
Concurrent Interleaving

\[ A \text{ and } B \text{ are started together} \]

\[ B \text{ completes before } A \]

\[ A_1 = (A|B < A) \oplus (B|B < A) \]
Concurrent Interleaving

\[ A \text{ and } B \text{ are started together} \]

\[ B \text{ completes before } A \quad \text{and} \quad A \text{ completes before } B \]

\[ A_1 = (A | B < A) \cap (B | B < A) \]
Concurrent Interleaving

$A$ and $B$ are started together

$B$ completes before $A$  

$A$ completes before $B$

$A_1 = (A \| B < A) \odot (B \| B < A)$
Concurrent Interleaving

\[ A \text{ and } B \text{ are started together} \]

\[ \begin{align*}
A & \text{ completes before } A \\
B & \text{ completes before } A \\
A_1 & = (A | B < A) \ominus (B | B < A) \\
B_1 & = (B | A < B) \ominus (A | A < B)
\end{align*} \]
Exponential memorylessness

$X \sim \text{exp}(1.25)$
Exponential memorylessness

\[ X \sim \text{exp}(1.25) \]
Exponential memorylessness

\[ X \sim \text{exp}(1.25) \]
Exponential memorylessness

\[ X \sim \text{exp}(1.25) \]
Exponential memorylessness

\( X \sim \text{exp}(1.25) \)
Exponential memorylessness

\[ X \sim \text{exp}(1.25) \]
Exponential memorylessness

![Graph showing exponential distribution]

$X \sim \text{exp}(1.25)$
Exponential memorylessness

\[ X \sim \text{exp}(1.25) \]
Exponential memorylessness

\[ X \sim \text{exp}(1.25) \]
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:

\[ P ::= (a, D).P \mid P + P \mid P \Join_{S} P \mid P/L \mid A \]
\[ D ::= \lambda \mid \omega : L(s) \]

• new prefix operator: \( (a, D).P \)

  – \( \lambda \): normal exponential rate parameter

  – \( \omega : L(s) \): a selection weight, \( \omega \), and a general distribution description, \( L(s) \)
Semi-Markov Example I

- Mutual exclusion modelling:

\[ A \overset{\text{def}}{=} (\text{think}, \lambda_1).(\text{recover}, \lambda_2).A \]
\[ + (\text{error}, \lambda_3).(\text{mutex}, 1 : L_1(s)).A \]

\[ S_n \overset{\text{def}}{=} \underbrace{A \otimes A \otimes \cdots \otimes A}_{n} \]

- Areas of Markovian concurrency interspersed with semi-Markov sequential behaviour
Semi-Markov Example II

- Web-server/database model:

\[ \begin{align*}
  Server & \overset{\text{def}}{=} (\text{get, } 5 : \exp(1.5, s))).Server_1 \\
  Server_1 & \overset{\text{def}}{=} (\text{static_page, } 1 : \det(3, s)).Server \\
 & \hspace{1em} + (\text{dbase\_fetch, } 2 : \gamma(2.2, 3.2, s)).Server_2 \\
  Server_2 & \overset{\text{def}}{=} (\text{dbase\_rtn, } \top).(\text{dynamic\_page, } 1 : \text{uniform}(2, 5, s)).Server \\
  Dbase & \overset{\text{def}}{=} (\text{dbase\_fetch, } \top).(\text{dbase\_rtn, } 4 : \exp(2.3, s)).Dbase \\
  Sys & \overset{\text{def}}{=} Server \parallel \bigcirc Dbase \\
\end{align*} \]

- 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

Cumulative passage time distribution: 10.9 million state voting model

Cumulative probability

Time
Semi-Markov Passage-time

Cumulative passage time distribution: 10.9 million state voting model