Specification and Efficient Computation of Passage-Time Distributions in GPA
QEST 2012

Matej Kohut, Anton Stefanek,
Richard A. Hayden, Jeremy T. Bradley

Department of Computing, Imperial College London

September 19, 2012
Overview

- Chemical equations
- Spatial algebra
- Large scale systems
- GPEPA
- iGPEPA
- Generates a new model measuring passage times
- Converts back to GPEPA
- Passage-time CDF
- Transforms results into a CDF
- Performance specification and evaluation with Unified Stochastic Probes
- Fluid analysis by Hayden, Bradley, Clark, IEEE TSE 2011
Overview

Chemical equations

Spatial algebra

Large scale systems

GPEPA

PCTMC

ODEs

Moments

Performance specification and evaluation with Unified Stochastic Probes and fluid analysis by Hayden, Bradley, Clark, IEEE TSE 2011
Overview

Chemical equations → GPEPA

Large scale systems → PCTMC

Spatial algebra

PCTMC → ODEs

Transforms results into a CDF

GPA Tool
Overview

Chemical equations → GPEPA → PCTMC → ODEs → Moments

Unified Stochastic Probes

Performance specification and evaluation with Unified Stochastic Probes and fluid analysis by Hayden, Bradley, Clark, IEEE TSE 2011
Overview

Chemical equations

Spatial algebra

Large scale systems

GPEPA

Probe specification

Unified Stochastic Probes

Generates a new model measuring passage times

iGPEPA

Performance specification and evaluation with Unified Stochastic Probes and fluid analysis by Hayden, Bradley, Clark, IEEE TSE 2011
Overview

Chemical equations

Large scale systems

Spatial algebra

GPEPA

PCTMC

ODEs

Moments

Unified Stochastic Probes

GPA Tool

Performance specification and evaluation with Unified Stochastic Probes and fluid analysis by Hayden, Bradley, Clark, IEEE TSE 2011
Overview

Performance specification and evaluation with Unified Stochastic Probes and fluid analysis by Hayden, Bradley, Clark, IEEE TSE 2011
Overview

Chemical equations

Spatial algebra

Large scale systems

GPEPA

Probe specification

iGPEPA

Generates a new model measuring passage times

Conforms back to GPEPA

Transforms results into a CDF

Passage-time CDF

PCTMC

ODEs

Moments

Unified Stochastic Probes

Performance specification and evaluation with Unified Stochastic Probes and fluid analysis by Hayden, Bradley, Clark, IEEE TSE 2011
Simple example

![Diagram showing client-server interactions]

Client = (data,rdata).Client_w;
Client_w = (think,rthink).Client;

Server = (data,rdata).Server_d;
Server_d = (reset,rreset).Server

Clients{Client[100]}<data>Servers{Server[50]}
Transient Individual Passage time probe

▶ How long until the first *think* of a client?
Transient Individual Passage time probe

- How long until the first $think$ of a client?
Transient Individual Passage time probe

How long until the first *think* of a client?
How long until the first *think* of a client?
Can attach a component that remembers the first *think* action
Can attach a component that remembers the first *think* action
How long until the first \textit{think} of a client?

- Can attach a component that remembers the first \textit{think} action
- This can be conveniently expressed by a local probe

\[ LProbe = \text{eE: begin, think: end} \]
How long until the first *think* of a client?
Can attach a component that remembers the first *think* action
This can be conveniently expressed by a local probe
\[
LProbe = \text{eE: begin, think: end}
\]
begin and end are signals sent to a global probe measuring time
\[
GProbe = \text{begin: start, end: stop}
\]
Transient Individual Passage time probe

- How long until the first \textit{think} of a client?
- Can attach a component that remembers the first \textit{think} action
- This can be conveniently expressed by a local probe
  \[ \text{LProbe} = eE: \text{begin}, \text{think}: \text{end} \]
- \text{begin} and \text{end} are signals sent to a global probe measuring time
  \[ \text{GProbe} = \text{begin}: \text{start}, \text{end}: \text{stop} \]
- Efficient fluid techniques to approximate the CDF of the time between \textit{start} and \textit{end}
**Individual Passage time probe CDF**

![Individual Passage time probe CDF](image)

---

**Probe** (stopTime = 40.0, stepSize = 0.1, density = 10)

```plaintext
transient 300 {
    GProbe = begin: start, end: stop
    observes {
        LProbe = eE: begin, think: end
    } where {
        Clients{Client[n]} =>
        Clients{Client<*>LProbe | Client[n-1]}
    }
}
```
How long until half of the clients do first \textit{think}?

\begin{verbatim}
Probes (stopTime = 40.0, stepSize = 0.1, density = 10) {
    GProbe = eE: start, end[n/2]:stop
    observes {
        LProbe = think: end
    } where {
        Clients{Client[n]} =>
        Clients{(Client <*> LProbe)[n]}})
\end{verbatim}
Global Passage time probe CDF
Larger model

- Wireless battery-powered clients
- Battery approximated by multiple levels
- Clients can hibernate
- Send control signals and data via Channels
Steady-state individual passage time

How long does a client take to discharge after its first transmission:

\[
\text{Probes}[^{\text{sipt.dat}}]\text{(stopTime=350, stepSize=1, density=10)}
\]

\[
\text{steady 500 } \{ \\
\text{SIPT} = \text{begin} : \text{start}, \text{end} : \text{stop} \leftarrow \\
\text{observes } \{ \\
\text{LProbe} = ((\text{data}; \text{control}) / \text{shutdown}) : \text{begin}, \\
\text{shutdown} : \text{end} \leftarrow \\
\} \\
\text{where } \{ \\
\text{Clients}^{\text{CB[n]}} \rightarrow \\
\text{Clients}^{\text{CB<*>LProbe}| \text{CB[n-1]}}
\}
\]
Steady-state individual passage time

\begin{align*}
\text{Time, } t &\quad 0 \quad 100 \quad 200 \quad 300 \\
\text{Probability} &\quad 0 \quad 0.2 \quad 0.4 \quad 0.6 \quad 0.8 \quad 1
\end{align*}

- $N_c = 10$, $N_h = 4$
- ODE approximation
Steady-state individual passage time
Steady-state individual passage time

![Graph showing the probability over time for different values of $N_c$ and $N_h$.]

- $N_c = 10, N_h = 4$
- $N_c = 20, N_h = 8$
- $N_c = 50, N_h = 20$

- ODE approximation
Steady-state individual passage time

![Graph showing steady-state individual passage time with different configurations of $N_c$ and $N_h$.]
Global passage time

How long does it take for half of the clients to have discharged their battery?:

\[
\text{Probe["gpt.dat"] (stopTime=400, stepSize=1, density=10) \{ }
\text{GPT = eE : start, end [nc/2] : stop}
\text{observes \{} \\
\text{LProbe = clt_shutdown : end}
\text{\}} \text{ where \{} \\
\text{Clients\{CB[nc]\} => Clients(CB <> LProbe)[nc]}
\text{\}}
\]
Global passage time

$N_c = 10, N_h = 4$

ODE approx.
Global passage time

Time, $t$ vs. Probability

- $N_c = 10$, $N_h = 4$
- $N_c = 20$, $N_h = 8$
- ODE approx.

$N_c$ and $N_h$ are the number of critical and high nodes, respectively.
Global passage time

![Graph showing Global passage time with different values of $N_c$ and $N_h$.](image)

- $N_c = 10, N_h = 4$
- $N_c = 20, N_h = 8$
- $N_c = 50, N_h = 20$
- ODE approx.
Global passage time

![Graph showing global passage time with different curves for different values of $N_c$ and $N_h$.](image)

- $N_c = 10, N_h = 4$
- $N_c = 20, N_h = 8$
- $N_c = 50, N_h = 20$
- $N_c = 100, N_h = 40$

-- ODE approx.
Global passage time

![Graph showing the relationship between time and probability for different values of \( N_c \) and \( N_h \). The graph includes curves for different values of \( N_c \) and \( N_h \), and an ODE approximation is also shown. The legend indicates the curves correspond to \( N_c = 10, N_h = 4 \), \( N_c = 20, N_h = 8 \), \( N_c = 50, N_h = 20 \), \( N_c = 100, N_h = 40 \), and an ODE approximation.]
Summary & Future Work

- Extended GPA with **Unified Stochastic Probes** for GPEPA process algebra
Summary & Future Work

- Extended GPA with Unified Stochastic Probes for GPEPA process algebra
- Fully supported probes syntax
Summary & Future Work

- Extended GPA with Unified Stochastic Probes for GPEPA process algebra
- Fully supported probes syntax
- Implemented efficient fluid analysis techniques, simulation for comparison
- Code available on http://code.google.com/p/gpanalyser
- Soon full integration with other extensions of GPA (moment closures)
- Generalisation to other specification languages
- Embedding probes in optimisation problem constraints
Summary & Future Work

- Extended GPA with Unified Stochastic Probes for GPEPA process algebra
- Fully supported probes syntax
- Implemented efficient fluid analysis techniques, simulation for comparison
- **Further enhancements of GPA** – immediate transitions, passive cooperation, C++ implementation

Code available on [http://code.google.com/p/gpanalyser](http://code.google.com/p/gpanalyser)

Soon full integration with other extensions of GPA (moment closures)

Generalisation to other specification languages

Embedding probes in optimisation problem constraints
Summary & Future Work

- Extended GPA with Unified Stochastic Probes for GPEPA process algebra
- Fully supported probes syntax
- Implemented efficient fluid analysis techniques, simulation for comparison
- Further enhancements of GPA – immediate transitions, passive cooperation, C++ implementation
- Code available on http://code.google.com/p/gpanalyser
Summary & Future Work

- Extended GPA with Unified Stochastic Probes for GPEPA process algebra
- Fully supported probes syntax
- Implemented efficient fluid analysis techniques, simulation for comparison
- Further enhancements of GPA – immediate transitions, passive cooperation, C++ implementation
- Code available on http://code.google.com/p/gpanalyser
- Soon full integration with other extensions of GPA (moment closures)
Summary & Future Work

- Extended GPA with Unified Stochastic Probes for GPEPA process algebra
- Fully supported probes syntax
- Implemented efficient fluid analysis techniques, simulation for comparison
- Further enhancements of GPA – immediate transitions, passive cooperation, C++ implementation
- Code available on http://code.google.com/p/gpanalyser
- Soon full integration with other extensions of GPA (moment closures)
- **Generalisation** to other specification languages
Summary & Future Work

- Extended GPA with Unified Stochastic Probes for GPEPA process algebra
- Fully supported probes syntax
- Implemented efficient fluid analysis techniques, simulation for comparison
- Further enhancements of GPA – immediate transitions, passive cooperation, C++ implementation
- Code available on http://code.google.com/p/gpanalyser
- Soon full integration with other extensions of GPA (moment closures)
- Generalisation to other specification languages
- Embedding probes in optimisation problem constraints
Thank you!