Performance analysis of Stochastic Process Algebra models using Stochastic Simulation

Jeremy Bradley | Stephen Gilmore | Nigel Thomas

Email: jb@doc.ic.ac.uk   stephen.gilmore@ed.ac.uk   nigel.thomas@ncl.ac.uk

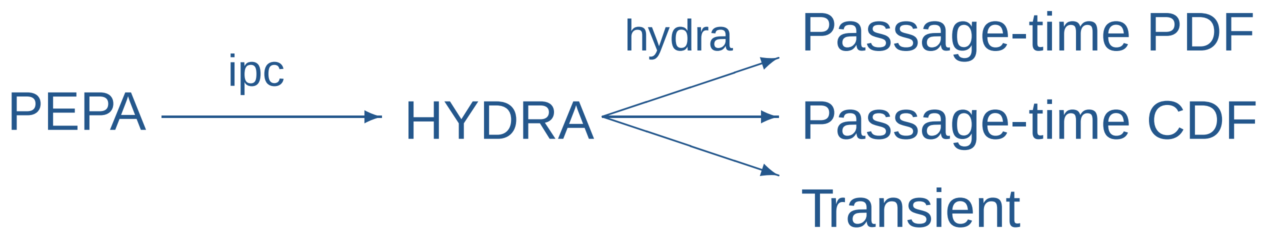
Department of Computing, Imperial College London

LFCS, University of Edinburgh

School of Computing Science, Univerity of Newcastle

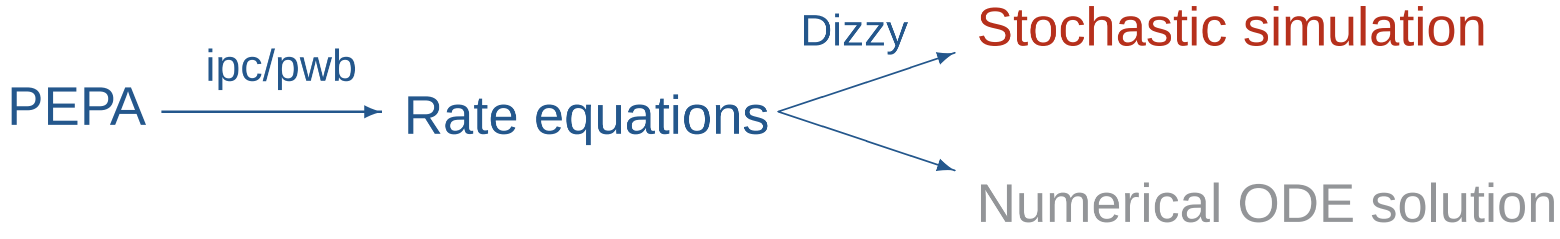
Produced with prosper and LaTeX

The story used to be...



- For state spaces of less than $O(10^9)$
- Very precise probabilistic results

Now the story is...



- For very large state spaces, e.g. $10^{1000}+$ states
- Aggregate deterministic results

Stochastic Process Algebra

PEPA syntax:

$$P \; ::= \; (a, \lambda).P \;\big|\; P + P \;\big|\; P \bowtie_{L} P \;\big|\; P/L \;\big|\; A$$

Stochastic Process Algebra

PEPA syntax:

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

- Action prefix: $(a, \lambda).P$

Stochastic Process Algebra

PEPA syntax:

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

- Action prefix: $(a, \lambda).P$
- Competitive choice: $P_1 + P_2$

Stochastic Process Algebra

PEPA syntax:

$$P \quad ::= \quad (a, \lambda).P \;\Big|\; P + P \;\Big|\; P \bowtie_L P \;\Big|\; P/L \;\Big|\; A$$

- Action prefix: $(a, \lambda).P$
- Competitive choice: $P_1 + P_2$
- Cooperation: $P_1 \bowtie_L P_2$

Stochastic Process Algebra

PEPA syntax:

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

- Action prefix: $(a, \lambda).P$
- Competitive choice: $P_1 + P_2$
- Cooperation: $P_1 \underset{L}{\bowtie} P_2$
- Action hiding: P/L

Stochastic Process Algebra

PEPA syntax:

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

- Action prefix: $(a, \lambda).P$
- Competitive choice: $P_1 + P_2$
- Cooperation: $P_1 \bowtie_{L} P_2$
- Action hiding: P/L
- Constant label: A

PEPA: Example

$$\mathrm{Sys} \stackrel{\mathrm{def}}{=} (\mathrm{AA} \underset{\{\mathrm{run}\}}{\bowtie} \mathrm{A1}) \underset{\{\mathrm{alert}\}}{\bowtie} (\mathrm{BB} \underset{\{\mathrm{run}\}}{\bowtie} \mathrm{B1})$$

PEPA: Example

$$\text{Sys} \stackrel{\text{def}}{=} (\text{AA} \underset{\{\text{run}\}}{\bowtie} \text{A1}) \underset{\{\text{alert}\}}{\bowtie} (\text{BB} \underset{\{\text{run}\}}{\bowtie} \text{B1})$$

$$\text{AA} \stackrel{\text{def}}{=} (\text{run}, \top).(\text{alert}, r_5).\text{AA}$$

PEPA: Example

$$\mathrm{Sys} \stackrel{\mathrm{def}}{=} (\mathrm{AA} \underset{\{\mathrm{run}\}}{\bowtie} \mathrm{A1}) \underset{\{\mathrm{alert}\}}{\bowtie} (\mathrm{BB} \underset{\{\mathrm{run}\}}{\bowtie} \mathrm{B1})$$

$$\mathrm{AA} \stackrel{\mathrm{def}}{=} (\mathrm{run}, \top).(\mathrm{alert}, r_5).\mathrm{AA}$$

$$\mathrm{A1} \stackrel{\mathrm{def}}{=} (\mathrm{start}, r_1).\mathrm{A2} + (\mathrm{pause}, r_2).\mathrm{A3}$$

$$\mathrm{A2} \stackrel{\mathrm{def}}{=} (\mathrm{run}, r_3).\mathrm{A1} + (\mathrm{fail}, r_4).\mathrm{A3}$$

$$\mathrm{A3} \stackrel{\mathrm{def}}{=} (\mathrm{recover}, r_1).\mathrm{A1}$$

PEPA: Example

$$\mathrm{Sys} \stackrel{\mathrm{def}}{=} (\mathrm{AA} \underset{\{\mathrm{run}\}}{\bowtie} \mathrm{A1}) \underset{\{\mathrm{alert}\}}{\bowtie} (\mathrm{BB} \underset{\{\mathrm{run}\}}{\bowtie} \mathrm{B1})$$

$$\mathrm{AA} \stackrel{\mathrm{def}}{=} (\mathrm{run}, \top).(\mathrm{alert}, r_5).\mathrm{AA}$$

$$\mathrm{A1} \stackrel{\mathrm{def}}{=} (\mathrm{start}, r_1).\mathrm{A2} + (\mathrm{pause}, r_2).\mathrm{A3}$$

$$\mathrm{A2} \stackrel{\mathrm{def}}{=} (\mathrm{run}, r_3).\mathrm{A1} + (\mathrm{fail}, r_4).\mathrm{A3}$$

$$\mathrm{A3} \stackrel{\mathrm{def}}{=} (\mathrm{recover}, r_1).\mathrm{A1}$$

$$\mathrm{BB} \stackrel{\mathrm{def}}{=} (\mathrm{run}, \top).(\mathrm{alert}, r_5).\mathrm{BB}$$

PEPA: Example

$$\mathrm{Sys} \stackrel{\mathrm{def}}{=} (\mathrm{AA} \bowtie_{\{\mathrm{run}\}} \mathrm{A1}) \bowtie_{\{\mathrm{alert}\}} (\mathrm{BB} \bowtie_{\{\mathrm{run}\}} \mathrm{B1})$$

$$\mathrm{AA} \stackrel{\mathrm{def}}{=} (\mathrm{run}, \top).(\mathrm{alert}, r_5).\mathrm{AA}$$

$$\mathrm{A1} \stackrel{\mathrm{def}}{=} (\mathrm{start}, r_1).\mathrm{A2} + (\mathrm{pause}, r_2).\mathrm{A3}$$

$$\mathrm{A2} \stackrel{\mathrm{def}}{=} (\mathrm{run}, r_3).\mathrm{A1} + (\mathrm{fail}, r_4).\mathrm{A3}$$

$$\mathrm{A3} \stackrel{\mathrm{def}}{=} (\mathrm{recover}, r_1).\mathrm{A1}$$

$$\mathrm{BB} \stackrel{\mathrm{def}}{=} (\mathrm{run}, \top).(\mathrm{alert}, r_5).\mathrm{BB}$$

$$\mathrm{B1} \stackrel{\mathrm{def}}{=} (\mathrm{start}, r_1).\mathrm{B2} + (\mathrm{pause}, r_2).\mathrm{B1}$$

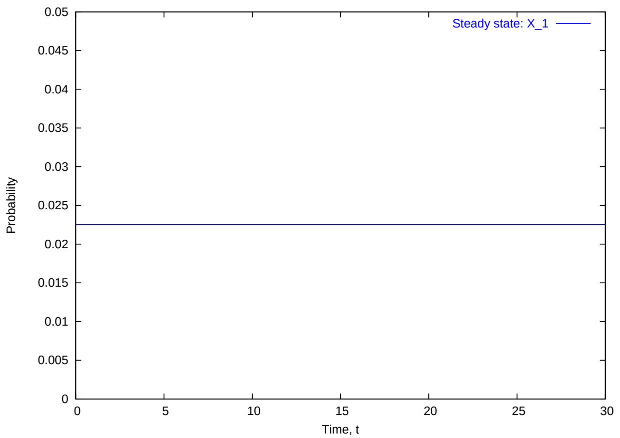
$$\mathrm{B2} \stackrel{\mathrm{def}}{=} (\mathrm{run}, r_3).\mathrm{B1}$$

Types of Analysis

Steady-state and transient analysis in PEPA:

$$
\begin{aligned}
A1 &\stackrel{\text{def}}{=} (start, r_1).A2 + (pause, r_2).A3 \\
A2 &\stackrel{\text{def}}{=} (run, r_3).A1 + (fail, r_4).A3 \\
A3 &\stackrel{\text{def}}{=} (recover, r_1).A1 \\
AA &\stackrel{\text{def}}{=} (run, \top).(alert, r_5).AA \\
Sys &\stackrel{\text{def}}{=} AA \underset{\{run\}}{\bowtie} A1
\end{aligned}
$$

$\Rightarrow$

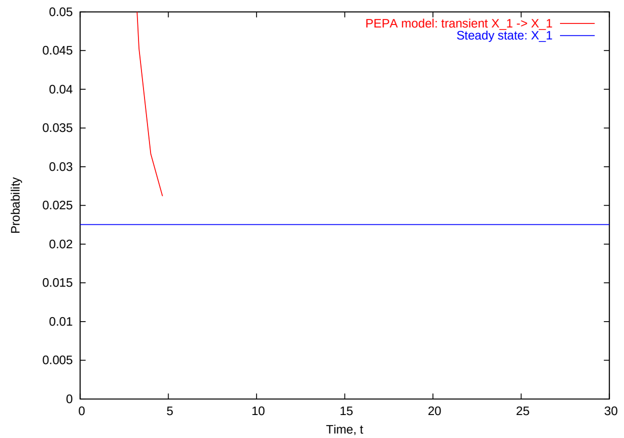


Types of Analysis

Steady-state and transient analysis in PEPA:

$$
\begin{aligned}
A1 &\overset{\text{def}}{=} (start, r_1).A2 + (pause, r_2).A3 \\
A2 &\overset{\text{def}}{=} (run, r_3).A1 + (fail, r_4).A3 \\
A3 &\overset{\text{def}}{=} (recover, r_1).A1 \\
AA &\overset{\text{def}}{=} (run, \top).(alert, r_5).AA \\
Sys &\overset{\text{def}}{=} AA \underset{\{run\}}{\bowtie} A1
\end{aligned}
$$

$\Rightarrow$

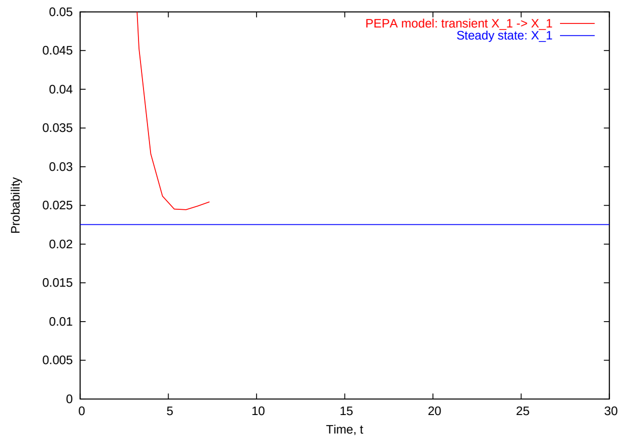


Types of Analysis

Steady-state and transient analysis in PEPA:

$$
\begin{aligned}
A1 &\stackrel{\text{def}}{=} (\mathrm{start}, r_1).A2 + (\mathrm{pause}, r_2).A3 \\
A2 &\stackrel{\text{def}}{=} (\mathrm{run}, r_3).A1 + (\mathrm{fail}, r_4).A3 \\
A3 &\stackrel{\text{def}}{=} (\mathrm{recover}, r_1).A1 \\
AA &\stackrel{\text{def}}{=} (\mathrm{run}, \top).(\mathrm{alert}, r_5).AA \\
Sys &\stackrel{\text{def}}{=} AA \underset{\{run\}}{\bowtie} A1
\end{aligned}
$$

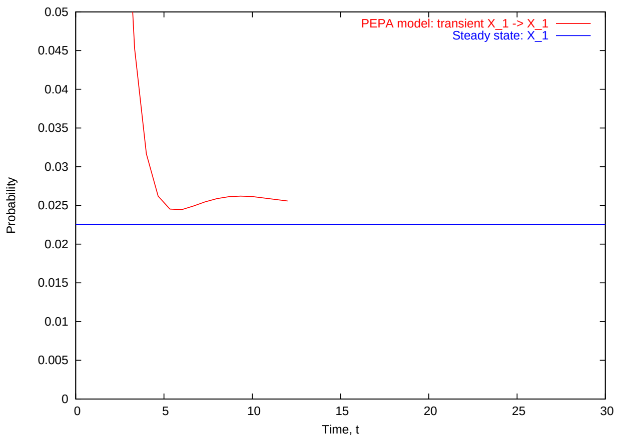
$\Rightarrow$



Types of Analysis

Steady-state and transient analysis in PEPA:

$$
\begin{aligned}
A1 &\stackrel{\text{def}}{=} (\text{start}, r_1).A2 + (\text{pause}, r_2).A3 \\
A2 &\stackrel{\text{def}}{=} (\text{run}, r_3).A1 + (\text{fail}, r_4).A3 \\
A3 &\stackrel{\text{def}}{=} (\text{recover}, r_1).A1 \\
AA &\stackrel{\text{def}}{=} (\text{run}, \top).(\text{alert}, r_5).AA \\
Sys &\stackrel{\text{def}}{=} AA \underset{\{run\}}{\bowtie} A1
\end{aligned}
\quad \Longrightarrow
$$

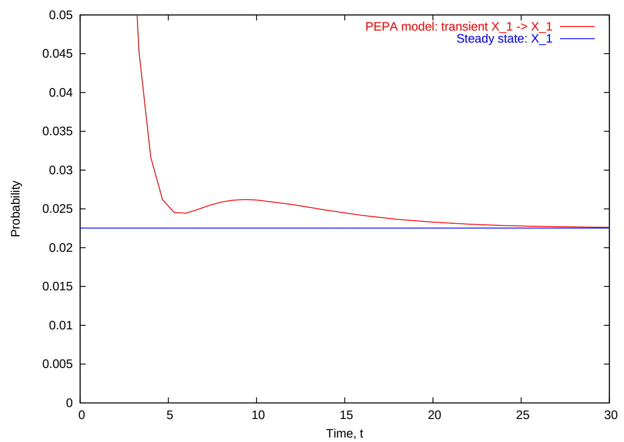


Types of Analysis

Steady-state and transient analysis in PEPA:

$$
\begin{aligned}
A1 &\stackrel{\text{def}}{=} (\text{start}, r_1).A2 + (\text{pause}, r_2).A3 \\
A2 &\stackrel{\text{def}}{=} (\text{run}, r_3).A1 + (\text{fail}, r_4).A3 \\
A3 &\stackrel{\text{def}}{=} (\text{recover}, r_1).A1 \\
AA &\stackrel{\text{def}}{=} (\text{run}, \top).(\text{alert}, r_5).AA \\
Sys &\stackrel{\text{def}}{=} AA \underset{\{run\}}{\bowtie} A1
\end{aligned}
$$

$\Rightarrow$

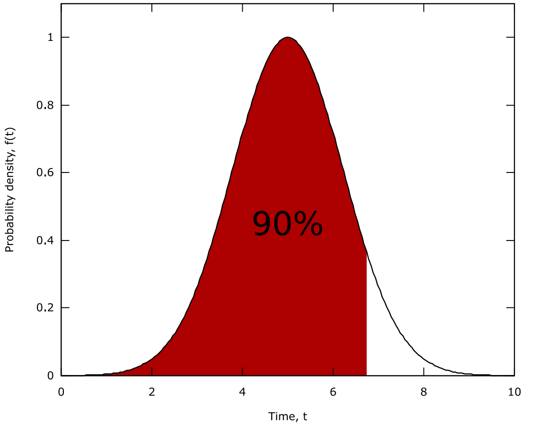


Passage-time Quantiles

Extract a passage-time density from a PEPA model:

$$\begin{aligned}
A1 &\stackrel{\text{def}}{=} (\text{start}, r_1).A2 + (\text{pause}, r_2).A3 \\
A2 &\stackrel{\text{def}}{=} (\text{run}, r_3).A1 + (\text{fail}, r_4).A3 \\
A3 &\stackrel{\text{def}}{=} (\text{recover}, r_1).A1 \\
AA &\stackrel{\text{def}}{=} (\text{run}, \top).(\text{alert}, r_5).AA \\
Sys &\stackrel{\text{def}}{=} AA \underset{\{run\}}{\bowtie} A1
\end{aligned}$$

$\Rightarrow$



Example of aggregate states

$$
\begin{aligned}
Client &\stackrel{\text{def}}{=} (compute, \top).Client_1 \\
Client_1 &\stackrel{\text{def}}{=} (delay, \mu).Client \\
Server &\stackrel{\text{def}}{=} (compute, \lambda).Server_1 \\
Server_1 &\stackrel{\text{def}}{=} (recover, \nu).Server
\end{aligned}
$$

$$
Sys = (\underbrace{Client \parallel \cdots \parallel Client}_{N}) \underset{\{compute\}}{\bowtie} (\underbrace{Server \parallel \cdots \parallel Server}_{M})
$$

- Cooperating clusters can be represented as tuples

Rate Equation Translation

- Action: $delay$

Rate Equation Translation

- Action: $delay$

$$Client_1 \xrightarrow{n(Client_1)\mu} Client$$

Rate Equation Translation

- Action: $delay$

$$Client_1 \xrightarrow{n(Client_1)\mu} Client$$

- Action: $recover$

$$Server_1 \xrightarrow{n(Server_1)\nu} Server$$

Rate Equation Translation

- Action: $delay$

$$Client_1 \xrightarrow{n(Client_1)\mu} Client$$

- Action: $recover$

$$Server_1 \xrightarrow{n(Server_1)\nu} Server$$

- Action: $compute$

$$Client + Server \xrightarrow{\theta(n(Client))n(Server)\lambda} Client_1 + Server_1$$

where $\theta(x) = 1$ if $x > 0$, else 0.

Why the θ function?

- There are N client cpts enabling a $compute$ action
- There are M server cpts enabling a $compute$ action
- Overall $compute$ rate is:

$$r_{compute}(Sys) = \min(N\top, M\lambda)$$

Why the θ function?

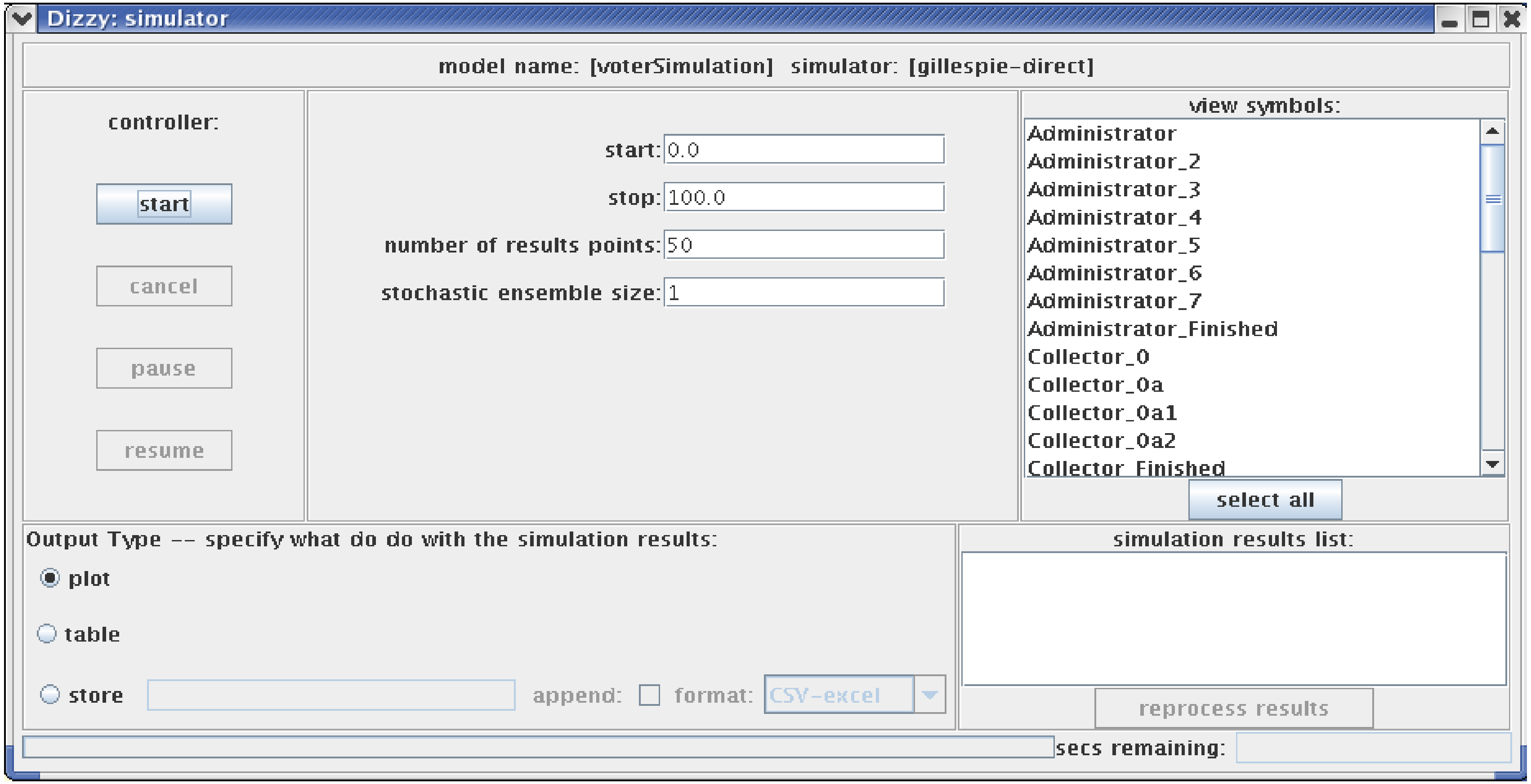
- There are N client cpts enabling a $compute$ action
- There are M server cpts enabling a $compute$ action
- Overall $compute$ rate is:

$$r_{compute}(Sys) = \min(N\top, M\lambda)$$

- If $N = 0$ then overall rate is 0, hence:

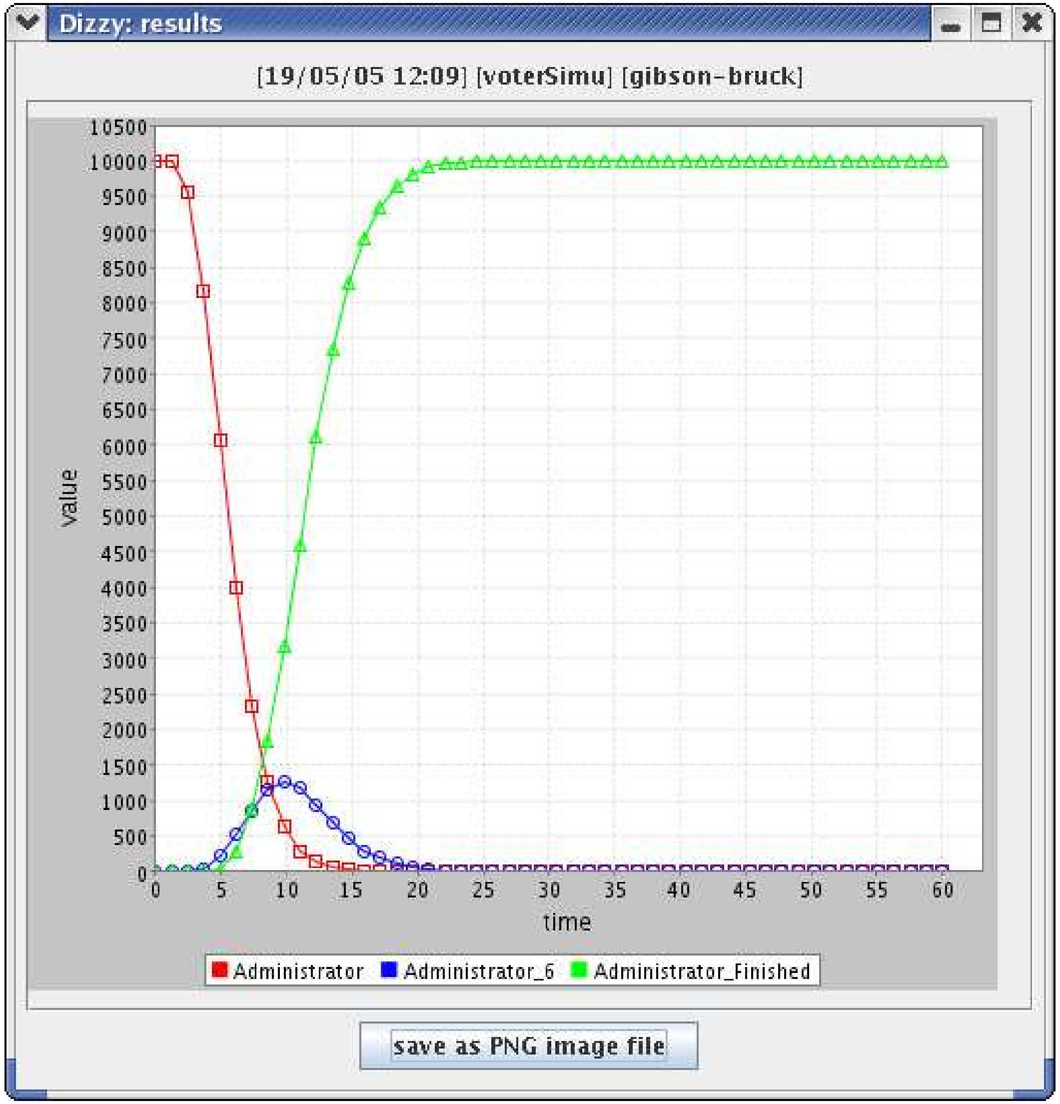
$$r_{compute}(Sys) = \theta(N)\, M\lambda$$

Dizzy setup



Dizzy: simulator
model name: [voterSimulation] simulator: [gillespie-direct]
controller:
start
cancel
pause
resume
start: 0.0
stop: 100.0
number of results points: 50
stochastic ensemble size: 1
view symbols:
Administrator
Administrator_2
Administrator_3
Administrator_4
Administrator_5
Administrator_6
Administrator_7
Administrator_Finished
Collector_0
Collector_0a
Collector_0a1
Collector_0a2
Collector_Finished
select all
Output Type -- specify what do do with the simulation results:
plot
table
store
append:
format:
CSV-excel
simulation results list:
reprocess results
secs remaining:

Dizzy simulation



Voter example

$$Election\_Preparation \underset{\mathcal{L}}{\bowtie} Electoral\_Personae$$

$$Electoral\_Personae \stackrel{\text{def}}{=} Voter0[N] \underset{\mathcal{M}}{\bowtie} Electoral\_App$$

$$Electoral\_App \stackrel{\text{def}}{=} Collector\_0[N] \parallel Counter\_1[N] \parallel Administrator[N]$$

Early voter description

$$Voter0 \stackrel{\text{def}}{=} (choose, c_1).Voter0\_1$$

$$Voter0\_1 \stackrel{\text{def}}{=} (bitcommit, b_1).Voter0\_2$$

$$Voter0\_2 \stackrel{\text{def}}{=} (blind_1, b_2).Voter0\_3$$

$$Voter0\_3 \stackrel{\text{def}}{=} (blind_2, b_3).Voter0\_4$$

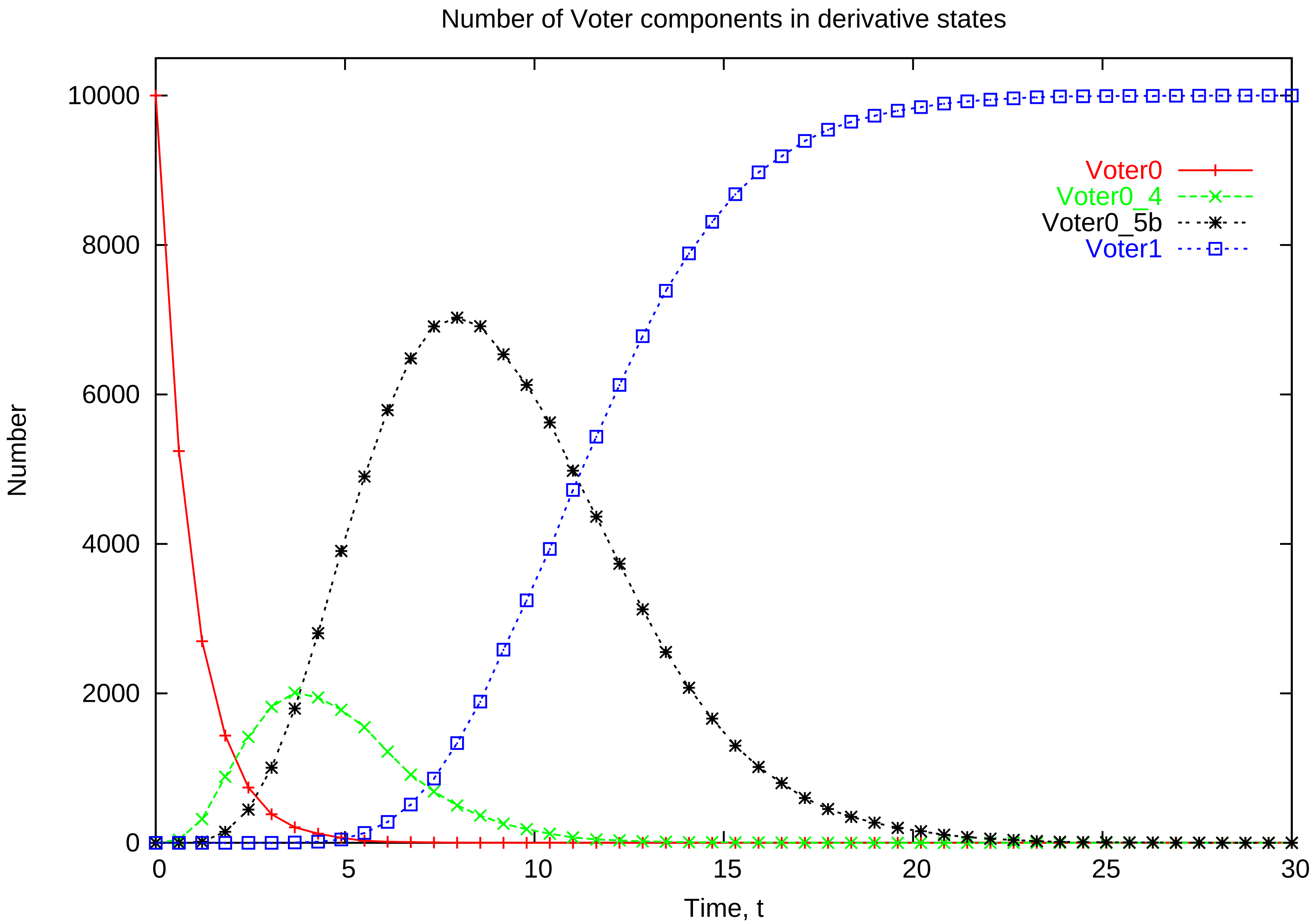
$$Voter0\_4 \stackrel{\text{def}}{=} (voter\_sign, s_1).Voter0\_5$$

$$Voter0\_5 \stackrel{\text{def}}{=} (sendA, s_2).Voter0\_5b$$

$$Voter0\_5b \stackrel{\text{def}}{=} (sendV, \top).Voter1$$

$$Voter1 \stackrel{\text{def}}{=} (unblind_1, u_1).Voter1\_1$$

Voter: early stage



High-level voter description

$$Voter0 \stackrel{\text{def}}{=} (choose, c_1) \ldots (sendV, \top).Voter1$$

High-level voter description

$$Voter0 \stackrel{\text{def}}{=} (choose, c_1) \ldots (sendV, \top).Voter1$$

$$Voter1 \stackrel{\text{def}}{=} (unblind, u_1) \ldots (sendC, s_6).Voter2$$

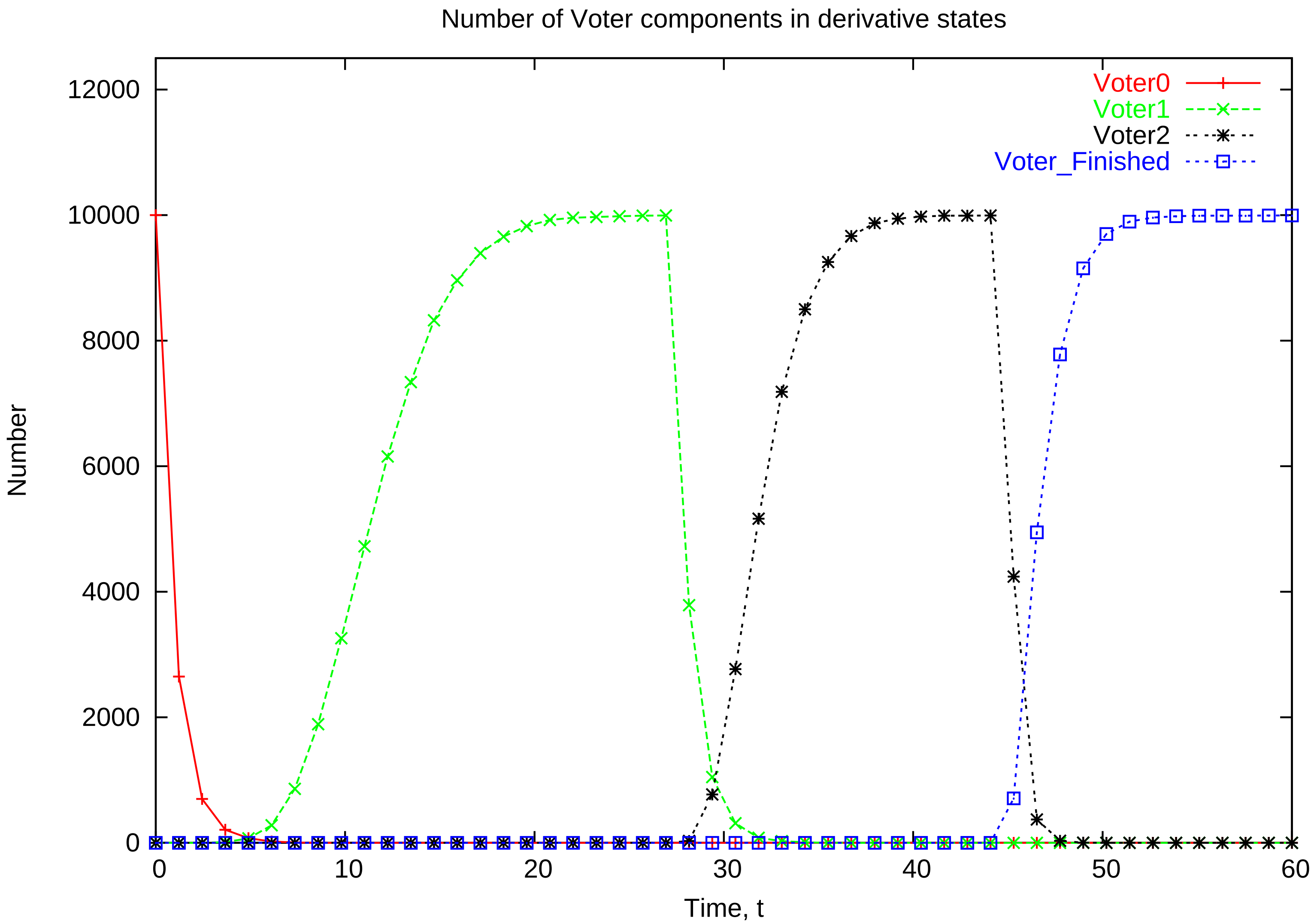
High-level voter description

$$Voter0 \stackrel{\text{def}}{=} (choose, c_1) \ldots (sendV, \top).Voter1$$

$$Voter1 \stackrel{\text{def}}{=} (unblind, u_1) \ldots (sendC, s_6).Voter2$$

$$Voter2 \stackrel{\text{def}}{=} (check, p \times c_4) \ldots (sendCo, s_7).Voter\_Finished$$

Voter: lifecycle



High-level Election description

$$\mathit{Election\_Preparation}$$

$$\stackrel{\text{def}}{=} (sendV, \top).\mathit{Election\_Preparation} + \cdots$$

$$+ (publishA, er).\mathit{Election\_Voting}$$

High-level Election description

$$Election\_Preparation$$

$$\stackrel{\text{def}}{=} (sendV, \top).Election\_Preparation + \cdots$$

$$+ (publishA, er).Election\_Voting$$

$$Election\_Voting$$

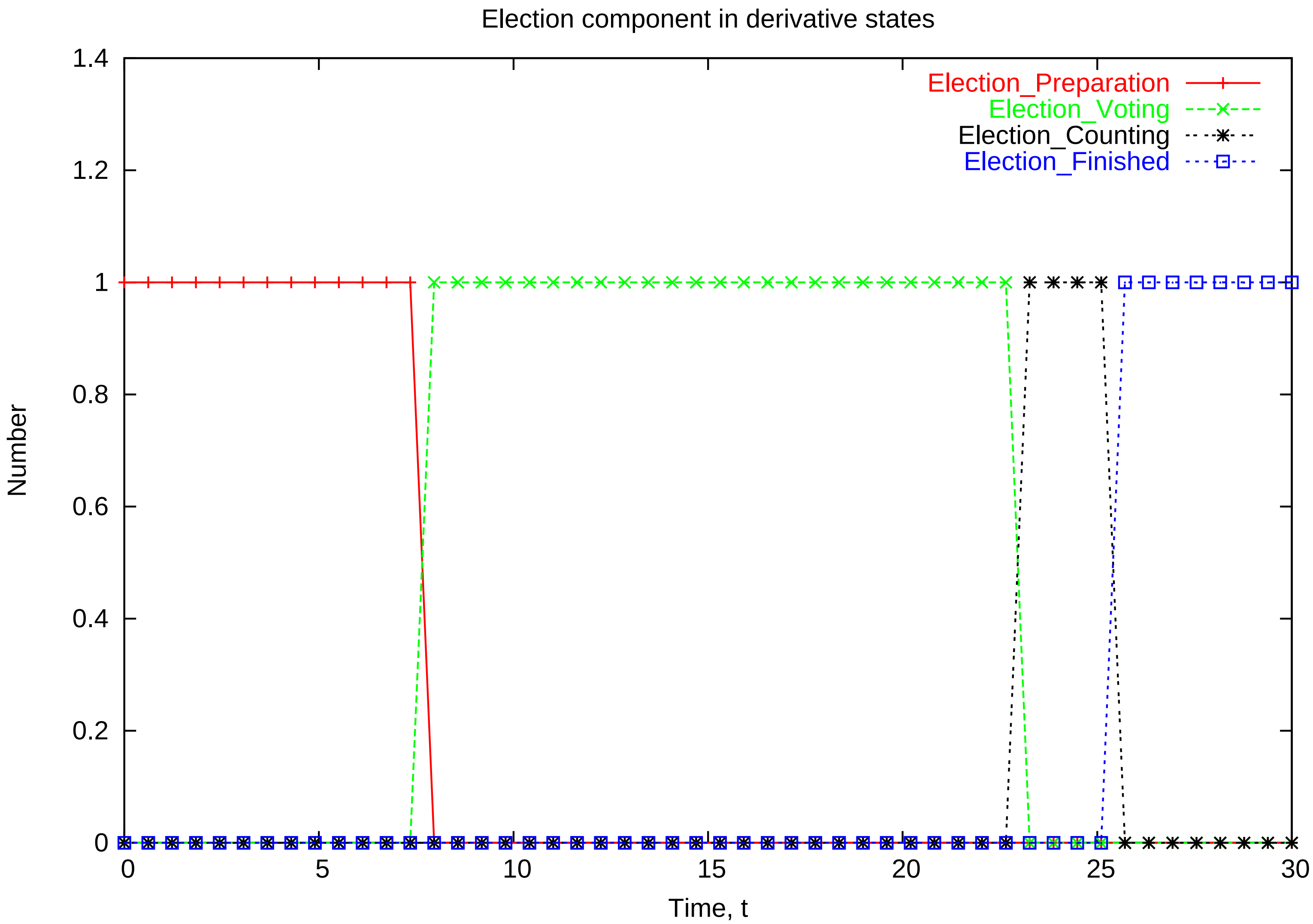
$$\stackrel{\text{def}}{=} (sendC, \top).Election\_Voting + \cdots$$

$$+ (publishC, er).Election\_Counting$$

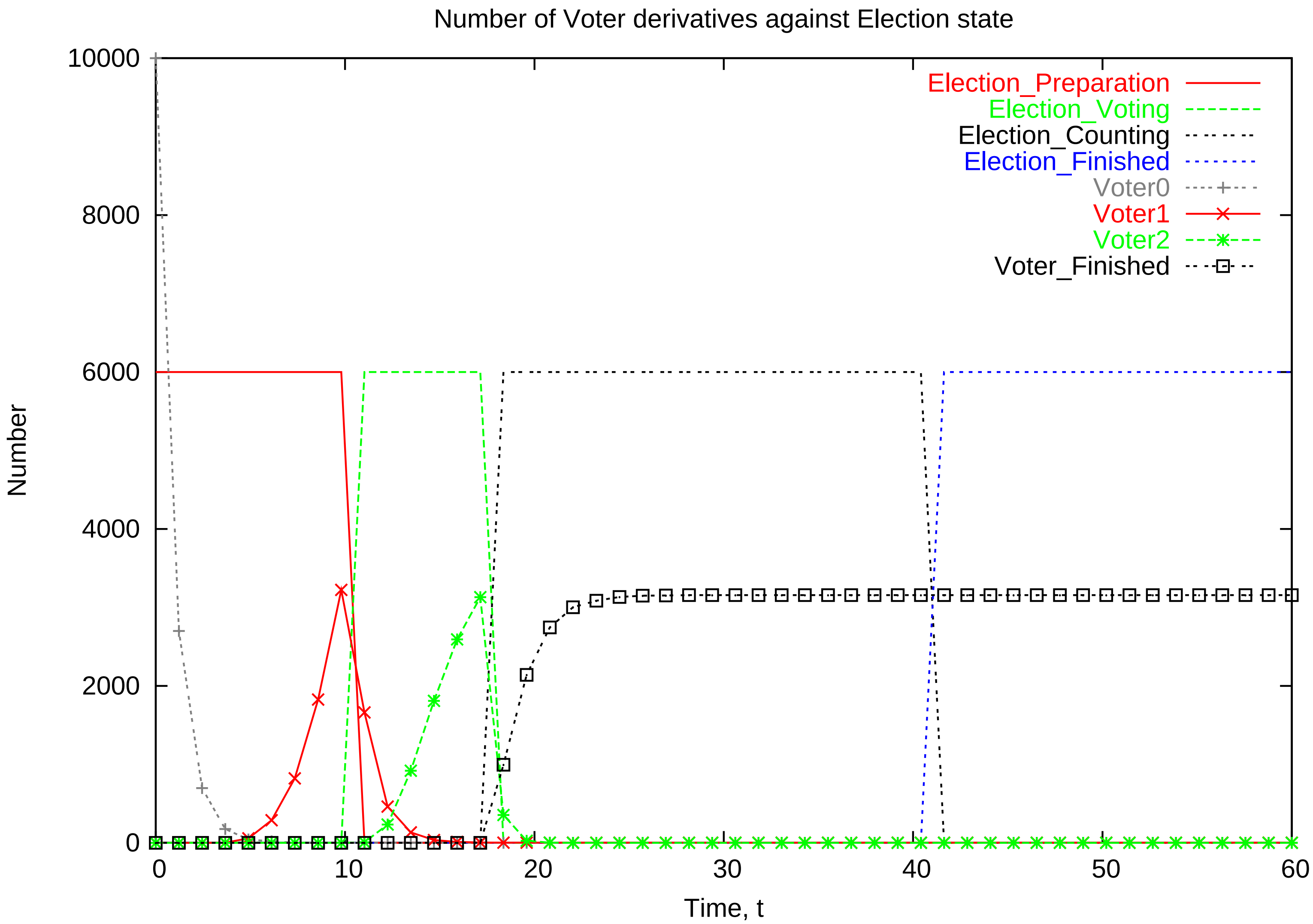
High-level Election description

$$
\begin{aligned}
&\mathit{Election\_Preparation} \\
&\quad \stackrel{\text{def}}{=} \quad (sendV, \top).\mathit{Election\_Preparation} + \cdots \\
&\qquad\quad + (publishA, er).\mathit{Election\_Voting} \\
&\mathit{Election\_Voting} \\
&\quad \stackrel{\text{def}}{=} \quad (sendC, \top).\mathit{Election\_Voting} + \cdots \\
&\qquad\quad + (publishC, er).\mathit{Election\_Counting} \\
&\mathit{Election\_Counting} \\
&\quad \stackrel{\text{def}}{=} \quad (sendCo, \top).\mathit{Election\_Counting} + \cdots \\
&\qquad\quad + (final\_publish, er).\mathit{Election\_Finished}
\end{aligned}
$$

Election: population of 1



Election + Voter: interaction



Conlcusion

- Novel simulation techniques based on chemical rate equations
- Orders of magnitude larger state spaces can be analysed
- Complexity of simulation method (Gibson-Bruck) is $O(\log n)$ where n is number of rate equations