Performance Evaluation of Workflows
Using Continuous Approximation of
Discrete Sets and Probability Distributions

Kunihiko Hiraishi
School of Information Science
Japan Advanced Institute of Science and Technology
To make information systems dependable,

• **Qualitative** ⇒ Verification of logical correctness
  – Formal verification (e.g., Model checking, Theorem proving)

• **Quantitative** ⇒ Guaranteeing performance index
  – Performance modeling (e.g., queuing theory)
Workflow: The automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules. (Def. by WFMC)
Performance Evaluation of Workflows

• Each workflow is a template of a business process.
• Many instances of workflows are running simultaneously in the information system.
• Optimal resource (people, machines, time, ...) assignment is a crucial issue.
## Example:
Review Process of an Academic Journal

<table>
<thead>
<tr>
<th>Author</th>
<th>Editor</th>
<th>Assoc. Editor</th>
<th>Reviewer#1</th>
<th>Reviewer#2</th>
</tr>
</thead>
<tbody>
<tr>
<td>paper submission</td>
<td>accept</td>
<td>assign an assoc. editor</td>
<td>accept</td>
<td>accept review</td>
</tr>
<tr>
<td>receive receipt</td>
<td>send receipt</td>
<td>assign reviewers</td>
<td>accept review</td>
<td>accept review</td>
</tr>
</tbody>
</table>

- Author submits a paper for review.
- Editor accepts the paper for review.
- An associate editor is assigned to handle the review.
- Reviewers are assigned to review the paper.
- Reviewers submit their reviews.
- The paper goes through the review process.
- The paper is either accepted or rejected based on the reviews.
- The final decision is made by the editor.
Example: Review Process of an Academic Journal

• **Given:**
  – Workflow
  – Statistical data on paper submission
  – An upper bound of the number of papers each associate editor can handle

• **Find:**
  – The optimal number of associate editors

• **Method:**
  – Generalized Stochastic Petri net
  – Approximation by Extended Continuous Petri Nets
Analysis of Stochastic Petri Nets

$[\lambda_i]$ firing rate.

pdf of firing delay: $\lambda e^{-\lambda it}$
(exponentially distributed)

Stochastic Petri Net

Continuous Time Markov Chain
Statistics

• Duration between submission and final judgment
  – Accept at 1st review: 2.4 month
  – Reject at 1st review: 3.9 month
  – Accept at 2nd review: 5.9 month
  – Reject at 2nd review: 6.8 month

• Ratio of acceptance and rejection
  – Accept at 1st review: 0.065
  – Reject at 1st review: 0.687
  – Accept at 2nd review: 0.238
  – Reject at 2nd review: 0.010

• Average number of paper submissions: 16.9 / month
GSPN Model

PAPER POOL is necessary for the model to have finite state space.
Tool (DSPNexpress)
## Result

<table>
<thead>
<tr>
<th>N</th>
<th>#states</th>
<th>CPU Time (sec.)</th>
<th>#Waiting papers</th>
<th>p(#paper pool = 0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>2926</td>
<td>0.3</td>
<td>10.18</td>
<td>0.30</td>
</tr>
<tr>
<td>4</td>
<td>8866</td>
<td>0.7</td>
<td>5.94</td>
<td>0.094</td>
</tr>
<tr>
<td>5</td>
<td>23023</td>
<td>2.3</td>
<td>1.99</td>
<td>0.013</td>
</tr>
<tr>
<td>6</td>
<td>53053</td>
<td>6.2</td>
<td>0.63</td>
<td>0.0021</td>
</tr>
<tr>
<td>7</td>
<td>110968</td>
<td>15</td>
<td>0.21</td>
<td>0.00049</td>
</tr>
<tr>
<td>8</td>
<td>213928</td>
<td>29</td>
<td>0.08</td>
<td>0.00020</td>
</tr>
<tr>
<td>9</td>
<td>384098</td>
<td>58</td>
<td>0.03</td>
<td>0.00010</td>
</tr>
</tbody>
</table>

Itanium2 1.6GHz/9MBCache, 16GB Memory

1 token = 16.9 papers

N = 6 ⇒ processing power : P = 6×16.9 = 101.4 papers simultaneously

5 papers / person ⇒ 101.4/5 = 20.28 associate editors are necessary
Continuous Approximation

- Analysis by GSPN is costly, because of state space explosion.
- A large number of discrete resources can be approximated by a continuous quantity.
- We first make a hybrid Petri net model from the GSPN model as follows:
  - tokens in a place $\Rightarrow$ a continuous variable
  - state space $\Rightarrow$ polyhedral approximation
  - firing delay of a timed transition $\Rightarrow$ firing speed of a continuous transition
  - probability distribution of firing delay $\Rightarrow$ interval of firing speed
- Then we derive differential equation from the HPN model.
- Finally, we compute an approximated state space by symbolic computation.
Continuous Approximation

\[ l_x \leq x \leq h_x \]

\[ l_x \leq x \leq h_x \]
Firing Delay $\Rightarrow$ Firing Speed

$$\dot{y} = \frac{1}{\tau_1} x - \frac{1}{\tau_2}$$

($\tau_1 \in [L_1, U_1], \tau_2 \in [L_2, U_2], y \geq 0$)
Extended Continuous Petri Net Model

Diagram showing the petri net model with transitions labeled as timed and instantaneous transitions.
Computation of the State Space

1. Differential equations (continuous time) ⇒ Difference equations (discrete time).
2. Rectangular approximation of reachable regions at each step. We use place invariants for avoiding divergence of intervals.
3. Symbolic computation by KCLP-HS (a rapid prototyping tool for algorithms on hybrid systems).
1-Step State Transition

\[ x_p \leq x_p \leq x_s \leq x_s, x_{a1} \leq x_{a1}, x_{r1} \leq x_{r1}, \]
\[ x_{ca} \leq x_{ca}, x_{a2} \leq x_{a2}, x_{r2} \leq x_{r2}, \]

\text{if } x_p \geq x_s, \text{ then } x_{tmp} := x_p - x_s, \text{ in } := x_s, x_s' := r_s;
\text{else } x_{tmp} := 0, \text{ in } := x_p, x_s' := x_s - x_p + r_s;

\text{if } x_{a1} \geq r_{a1}, \text{ then } x_{a1}' := x_{a1} - r_{a1} + \text{ in } \cdot p_{a1}, \text{ rel}_0 := r_{a1};
\text{else } x_{a1}' := \text{ in } \cdot p_{a1}, \text{ rel}_0 := x_{a1};

\text{if } x_{r1} \geq r_{r1}, \text{ then } x_{r1}' := x_{r1} - r_{r1} + \text{ in } \cdot p_{r1}, \text{ rel}_1 := \text{ rel}_0 + r_{r1};
\text{else } x_{r1}' := \text{ in } \cdot p_{r1}, \text{ rel}_1 := \text{ rel}_0 + x_{r1};

\text{if } x_{ca} \geq r_{ca}, \text{ then } x_{ca}' := x_{ca} - r_{ca} + \text{ in } \cdot p_{ca}, \text{ in}_2 := r_{ca};
\text{else } x_{ca}' := \text{ in } \cdot p_{ca}, \text{ in}_2 := x_{ca};

\text{if } x_{a2} \geq r_{a2}, \text{ then } x_{a2}' := x_{a2} - r_{a2} + \text{ in}_2 \cdot p_{a2}, \text{ rel}_2 := \text{ rel}_1 + r_{a1};
\text{else } x_{a2}' := \text{ in}_2 \cdot p_{a2}, \text{ rel}_2 := \text{ rel}_1 + x_{a2};

\text{if } x_{r2} \geq r_{r2}, \text{ then } x_{r2}' := x_{r2} - r_{r2} + \text{ in}_2 \cdot p_{r2}, \text{ rel}_3 := \text{ rel}_2 + r_{r2};
\text{else } x_{r2}' := \text{ in}_2 \cdot p_{r2}, \text{ rel}_3 := \text{ rel}_2 + x_{r2};

x_p' := x_{tmp} - \text{ rel}_3;

x_p + x_{a1} + x_{r1} + x_{ca} + x_{a2} + x_{r2} = N

\text{Approximation by Rectangular Sets}

\text{Compute Min and Max of each } x_i'
Result

\[ \theta = 0.1 \]

The number of waiting papers at 6 month later (sampling interval = 0.5 month).
Firing speeds may change \( \pm 10\% \). CPU time < 0.1 sec. for each P.
Result

$\theta = 0.2$

The number of waiting papers at 6 month later (sampling interval = 0.5 month).

Firing speeds may change $\pm 20\%$. CPU time $< 0.1$ sec. for each P.
## Comparison

### CPU Time (sec.)

<table>
<thead>
<tr>
<th>Duration</th>
<th>(1) Exact</th>
<th>(2) Approx</th>
<th>(3) Approx with inv.</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>0.02</td>
<td>0.01</td>
<td>0.02</td>
</tr>
<tr>
<td>4</td>
<td>0.09</td>
<td>0.02</td>
<td>0.03</td>
</tr>
<tr>
<td>5</td>
<td>0.27</td>
<td>0.03</td>
<td>0.03</td>
</tr>
<tr>
<td>6</td>
<td>3.3</td>
<td>0.04</td>
<td>0.05</td>
</tr>
<tr>
<td>7</td>
<td>17</td>
<td>0.05</td>
<td>0.07</td>
</tr>
<tr>
<td>8</td>
<td>48</td>
<td>0.06</td>
<td>0.08</td>
</tr>
<tr>
<td>9</td>
<td>106</td>
<td>0.08</td>
<td>0.09</td>
</tr>
<tr>
<td>10</td>
<td>212</td>
<td>0.09</td>
<td>0.10</td>
</tr>
</tbody>
</table>

\[ N = 60, \pm 20\%. \]
Comparison

$N = 60, \pm 20\%.$
Comparison

Waiting Papers

Month

Exact
Approx
Approx Inv.

$N = 60, \pm 20\%.$
Conclusion

• For performance evaluation of workflows, we have tried to methods, GSPN and continuous approximation by hybrid systems.
• The later method derives a similar result in a much shorter time.
• We expect that the continuous approximation by hybrid systems is applicable to larger workflows for which GSPN is infeasible to compute the solution.