Formal Verification of Real-time Systems

Tomohiro Yoned

National Institute of Informatics
Tokyo Institute of Technology
Personal History

- **Ph.D thesis**
  - 1985, Tokyo Institute of Technology
  - Adviser: Prof. Tohma

- **3 Papers in FTCS**
  - FTCS-15, 16, 19

- **Current interests**
  - Formal verification of real-time systems
  - Synthesis tools for Asynchronous circuits
  - Main Conferences: ASYNC, CAV
What are verified and how?

- **What?**
  - Time dependent systems
    - Circuits composed of gates with bounded delays
    - Protocols for real-time control

- **How?**
  - Conformance Checking
    - Impl. and Spec. are expressed by the same model
    - State space exploration checking correspondence
  - Contribution
    - Extension to real-time models, improvement in performance, development of a tool
**Approach**

- **Formal model used**
  - Time Petri nets
- **Improvement in performance**
  - Partial order reduction technique
  - Hierarchical verification technique
Verification method (1)
Verification method (1)

- Unexpected outputs must not be produced.
- Expected outputs must be produced within a given time frame.
Verification method (2)

Mirror of Spec.: Inputs and Outputs are swapped

$M_s$

$M_1$

$M_2$

$M_3$

$M_4$
Verification method (2)

M₅

M₁

M₂

M₃

M₄

Time
Verification method (2)
Verification method (2)

M₅

M₁

M₂

M₃

M₄

Time

in

out

in

in

in
Verification method (2)
Verification method (2)
Verification method (2)

\[ M_5 \]
\[ M_1 \]
\[ M_2 \]
\[ M_3 \]
\[ M_4 \]
Verification method (2)
Verification method (2)

Safety failure

M_5

M_1

M_2

M_3

M_4

in

out

Time
Verification method (2)
Verification method (2)
Verification method (2)
Verification method (3)

- **Advantages**
  - More intuitive than temporal logics
    - It is not easy to express complicated properties or expected behavior using temporal logics
  - Several methods for performance improvement
    - Partial order reduction
    - Hierarchical verification

- **Drawbacks**
  - Spec. must be deterministic
  - A little less expressive
Time Petri nets

- Token 1 arrives
- Token 2 arrives
- Enabled
- Firable
- Fire
- [4,7]
Timed state space exploration

- **State**
  - marking
  - a set of inequalities

Decision of existence of solutions

Floyd's shortest path algorithm

variables for times of transition firings

\[
\begin{align*}
1 \leq x - v &\leq 5 \\
2 \leq y - v &\leq 3 \\
2 \leq w - v &\leq 7
\end{align*}
\]
Partial Order Reduction (1)

Full state space exploration

State space exploration based on Partial Order Reduction
Partial Order Reduction (2)

Failure state

Failure state
Partial Order Reduction (3)

- **Basic idea**
  - Only one interleaving is considered, if possible

![Diagram showing partial order reduction with transitions t1, t2, and t3, and places p1, p2, p3, and p4.](image)
Partial Order Reduction (3)

Basic idea

- Only one interleaving is considered, if possible

```
p1
\[1, 5\]
t1
\[1, 7\]
p2

failure
```

```
t1
[0, 3]
t2
[1, 5]
t3
[1, 5]
t4

failure
```
Hierarchical verification

These two imply the above
Experimental results (1)

- Verification of STARI circuits

Spec. of STARI circuit

Two stage STARI circuit
Experimental results (1)

- Hierarchical verification
### Experimental results (1)

#### Total order vs Partial order

<table>
<thead>
<tr>
<th>no. of stage</th>
<th>CPU time (sec.)</th>
<th>Memory(MB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>total</td>
<td>partial</td>
</tr>
<tr>
<td>7</td>
<td>1057.93</td>
<td>0.20</td>
</tr>
<tr>
<td>8</td>
<td>2014.90</td>
<td>0.29</td>
</tr>
<tr>
<td>9</td>
<td>3650.61</td>
<td>0.35</td>
</tr>
<tr>
<td>10</td>
<td>4860.52</td>
<td>0.43</td>
</tr>
<tr>
<td>11</td>
<td>-</td>
<td>1.14</td>
</tr>
</tbody>
</table>

#### Flat vs Hierarchical

<table>
<thead>
<tr>
<th>no. of stage</th>
<th>CPU time (sec.)</th>
<th>Memory(MB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>flat</td>
<td>hierarchical</td>
</tr>
<tr>
<td>12</td>
<td>9.13</td>
<td>1.68</td>
</tr>
<tr>
<td>13</td>
<td>12.10</td>
<td>2.04</td>
</tr>
<tr>
<td>14</td>
<td>25.36</td>
<td>3.28</td>
</tr>
<tr>
<td>15</td>
<td>60.59</td>
<td>4.98</td>
</tr>
<tr>
<td>16</td>
<td>-</td>
<td>62.04</td>
</tr>
</tbody>
</table>

---

**flat verification**

**partial order method**
Experimental results (2)

Abstracted Circuit of ICACHE

Spec. of ICACHE
Experimental results (2)

- low level
  - Cache Memory Module
  - Exist Register

- upper level
  - Abstracted Circuit of ICACHE
  - Spec. of ICACHE

verification
Experimental results (2)

- low level
Experimental results (2)

- low level

Cache Memory Module
Experimental results (2)

<table>
<thead>
<tr>
<th>Method</th>
<th>Verification of ICACHE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CPU time (sec.)</td>
</tr>
<tr>
<td></td>
<td>non-op.</td>
</tr>
<tr>
<td>flat</td>
<td>12.50</td>
</tr>
<tr>
<td>Hierarchical</td>
<td>10.81</td>
</tr>
</tbody>
</table>
Experimental results (3)

- Distributed algorithm to initiate an air-conditioner network

No. of places: 354
No. of transitions: 569
No. of generated states: 361530
Verification time: 18 min.
Conclusion

- Formal verification of time dependent systems
- Approach
  - Timed extension of Conformance Checking
- Formal model
  - Time Petri nets
- Improvement
  - Partial Order Reduction
  - Hierarchical verification