Analysis of UML Models with SPT Annotations

John Håkansson
Leonid Mokrushin
Paul Pettersson
Wang Yi
Analysis of UML Models with SPT Annotations

- UML Models
  - Statecharts
  - Object diagrams
- SPT Annotations
  - UML profile for schedulability, performance and time
  - Assumptions and requirements
- Extract timing information
  - For schedulability analysis
UML Models with SPT Annotations

- Statechart describe behaviour
- SPT: UML profile for scheduling, performance and time
UML Models with SPT Annotations

- Components with ports and links
  - Links are used to connect ports
  - Structure diagrams: hierarchical

```
+ x : int
+ run() : void
```

```
+ trigger()
```

```
+ get() : int
+ set(int value) : void
```
Motivation

- Impact of UML
  - In industry and academia
- SPT
  - Relatively new UML profile (2002)
  - Commercial tools emerging
- Schedulability analysis
  - Event-driven systems (non-RMA)
Timed Model

- Timed automata extended with Tasks

- Events

- Timing constraints
  - Specifying task releases and event arrivals
  - E.g. periodic or sporadic

- Tasks
  - Computation time, deadline
Extracting a Timed Model

```
+ x : int
+ run() : void
<<component>>
```

```
trigger[IS_PORT(tick)]/run();
get_output()->set(1);
```

```
tm == 20
port_output:=0
```

```
port_tick > 0
go?
port_tick:=port_tick-1,
REF_run:=1
```

```
RUN
```

```
REF_run == 0
go?
port_output:=0
tm == 20
x:=0
port_output:=0
```

```
DeadlineComputation
Time
PriorityTask
name
tm == 20
x:=0
port_output:=0
```

<table>
<thead>
<tr>
<th>Task name</th>
<th>Priority</th>
<th>Computation Time</th>
<th>Deadline</th>
</tr>
</thead>
<tbody>
<tr>
<td>run</td>
<td>1</td>
<td>12</td>
<td>42</td>
</tr>
</tbody>
</table>
Extracting a Timed Model

- Statecharts → automata
  - Timeouts
- Operations → tasks
  - SPT annotations
- Ports → shared variables
  - DataPort: data item in port
  - TriggerPort: number of events queued
Tool Overview

- **Rhapsody UML tool**
  - Ports and components
  - Statecharts
  - SPT annotations

- **Analysis using TIMES tool**
  - Schedulability analysis using timed model
  - Scheduling strategy (e.g. FPS, EDF)
Case Study: Adaptive Cruise Control

- Extension of traditional cruise control
- Adapts to traffic conditions
  - Regulates speed and distance to vehicle in front
- Schedulability analysis
  - Model is abstracted to timing behaviour
  - Timed model has 5 asynchronous tasks
  - Events trigger task releases

\[ v_{lead} \quad d_{des} \quad v \quad d_{act} \]
Case Study: Adaptive Cruise Control
Tool Demonstration
Ongoing work

- Extend case-study
- Handle richer classes of UML with SPT annotations
- Predicate abstraction
Thank You