Model Checking for Probabilistic Hybrid Systems

Marta Kwiatkowska, Ernst Moritz Hahn
Oxford University Computing Laboratory

Holger Hermanns, Arnd Hartmanns
Saarland University, Dependable Systems and Software

CPSWeek’13, Philadelphia, April 2013
Part 3

Tools and case studies
Overview (Part 3)

• Tools and modelling languages
  – PRISM & guarded commands
  – Modest & the Modest Toolset

• Probabilistic timed BRP (PTA)
  – Case Study & Demo

• Temperature control (PHA)
  – Case Study & Demo

• ETCS level 3 train control (SHA)
  – Case Study & Demo
Tools for Quantitative Verification

- **PRISM**
  www.prismmodelchecker.org
  - developed at Birmingham/Oxford University, since 1999
  - modelling language: guarded commands
  - model checking for PTA, MDPs, DTMCs and CTMCs

- **The Modest Toolset**
  www.modestchecker.net
  - developed at Saarland University, since 2008
  - modelling language: Modest
  - other languages also supported: e.g. guarded commands
  - model checking and simulation for different subsets of SHA
Modelling Languages

• **Guarded Commands**
  – low-level language
  – few, but powerful concepts

```plaintext
module sender

s : [0..6] init 0;
srep : [0..3];
nrtr : [0..MAX];
...
ls : bool;

[NewFile] (s = 0) -> (s' = 1) & (i' = 1) & (srep' = 0);
[aF] (s = 1) -> (s' = 2) & (fs' = (i = 1)) & (ls' = (i = N)) & ...
[aB] (s = 2) -> (s' = 4) & (s_ab' = !s_ab);
...
[SyncWait] (s = 5) -> (s' = 6);
[SyncWait] (s = 6) -> (s' = 0) & (s_ab' = false);
endmodule
```
Modelling Languages

- Modest
  - high-level language
  - focus on readability, expressivity and conciseness

```plaintext
process Sender() {
    bool bit;
    int(0..MAX) rc;

    new_file {= i = 0, rc = 0 =};

    try {
        do {
            :: when(i < N) {= i = i + 1 =};
            do {
                :: put_k {= ff = (i == 1), lf = (i == N), ab = bit =}
                alt {
                    :: get_l {= bit = !bit, rc = 0 =};
                    break
                    :: when(rc == MAX && i < N)
                    s_nok {= rc = 0 =};
                    throw(error)
                }
            }
        }
    }
}
...
The Modest Toolset

- **mcpta**
  - model checking for PTA
  - using PRISM

- **mctau**
  - model checking for TA
  - using the UPPAAL model checker
  - more efficient than mcpta for TA models

- **modes**
  - statistical model checking (= simulation) for STA
  - sound treatment of nondeterminism (POR, confluence)

- **prohver**
  - safety verification for SHA
  - uses a modified version of PHAVer
The Modest Toolset

- Toolset overview

![Diagram of Modest Toolset]

- Modest
- Guarded Commands
- UPFAAL .xml
- Networks of Stochastic Hybrid Automata
  - SHA
  - STA
  - PTA
  - MDPs
  - prohver
  - modes
  - mcpta
  - mctau
- PHAVer
- mod.
- Results

- PHAVer
- ≥ 4.0
- ≥ 4.1
• The probabilistic model checking process in PRISM
  – two distinct phases: model construction, model checking

Model (Syntax) \rightarrow \text{Model construction} \rightarrow \text{Model}\rightarrow \text{Model checking} \rightarrow \text{Result}

guarded commands

DTMC, MDP, CTMC or PTA

PCTL/CSL/LTL/… formula
The probabilistic model checking process in PRISM:
- two distinct phases: model construction, model checking

Model construction:
- Translation from high-level language
- Reachability: building set of reachable states

High-level model
- PRISM language description
- Matrix manipulation
- Graph-based algorithm

Model
- DTMC, MDP, CTMC or PTA
The safety verification process for SHA in prohver

- SHA
- PHA
- HA + probs
- PHAVer
- LTS
- MDP

Overapproximation of continuous distributions
Decomposition
Reconstruction
Value iteration
Results
Overview (Part 3)

• Tools and modelling languages
  – PRISM & guarded commands
  – Modest & the Modest Toolset

• Probabilistic timed BRP (PTA)
  – Case Study & Demo

• Temperature control (PHA)
  – Case Study & Demo

• ETCS level 3 train control (SHA)
  – Case Study & Demo
Case Study: BRP

- **Bounded Retransmission Protocol**
  - timed model: timed automata, UPPAAL
  - probabilistic timed model: mcpta
  - allows new kinds of properties to be checked
Case Study: BRP

- **Results**

<table>
<thead>
<tr>
<th>property</th>
<th>mctau</th>
<th>mcpta</th>
<th>modes</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_{A1}$</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>$T_{A2}$</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>$P_A$</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$P_B$</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$P_1$</td>
<td>?</td>
<td>$[0, 1]$</td>
<td>$4.233 \cdot 10^{-4}$</td>
</tr>
<tr>
<td>$P_2$</td>
<td>?</td>
<td>$[0, 1]$</td>
<td>$2.645 \cdot 10^{-5}$</td>
</tr>
<tr>
<td>$D_{\text{max}}$</td>
<td>?</td>
<td>$[0, 1]$</td>
<td>$9.996 \cdot 10^{-1}$</td>
</tr>
<tr>
<td>$E_{\text{max}}$</td>
<td>?</td>
<td>n/a</td>
<td>33.473</td>
</tr>
</tbody>
</table>
Overview (Part 3)

• Tools and modelling languages
  – PRISM & guarded commands
  – Modest & the Modest Toolset

• Probabilistic timed BRP (PTA)
  – Case Study & Demo

• **Temperature control (PHA)**
  – Case Study & Demo

• ETCS level 3 train control (SHA)
  – Case Study & Demo
Case Study: Thermostat

- Simple temperature control model
  - PHA: finite-support probabilistic choice
  - nonlinear continuous dynamics
  - probability to reach error?

```
invariant
  (mode = m_cool =>
    T >= 0 & x <= TIME_BOUND
    & der(T) = -T & der(x) = 1 & der(t) = 1)

& (mode = m_heat =>
    T <= 10 & t <= 3 & x <= TIME_BOUND
    & der(T) = 2 & der(x) = 1 & der(t) = 1)

& (mode = m_check =>
    t <= 1 & x <= TIME_BOUND
    & der(T) = -0.5 * T & der(x) = 1 & der(t) = 1)

& (mode = m_error =>
    x <= TIME_BOUND
    & der(T) = 0 & der(x) = 0 & der(t) = 0)
```

DEMO
Overview (Part 3)

• Tools and modelling languages
  – PRISM & guarded commands
  – Modest & the Modest Toolset

• Probabilistic timed BRP (PTA)
  – Case Study & Demo

• Temperature control (PHA)
  – Case Study & Demo

• ETCS level 3 train control (SHA)
  – Case Study & Demo
Case Study: ETCS

- **ETCS Level 3**
  - next-generation European train control system
  - moving block train control to increase capacity
  - trains measure and report position to RBC
  - radio block controller (RBC) assigns movement authority
  - communication is wireless
Case Study: ETCS

- **SHA model**
  - two trains – leader and follower – and Comm+RBC

- **Continuous aspects**
  - acceleration, deceleration, speed
  - acceleration of leader nondeterministic (within train limits)

- **Stochastic aspects**
  - position measurements scattered with normal distribution
  - message loss probability during communication
Case Study: ETCS

**FreeRun**

\[
\begin{align*}
    v & \in [0_{\frac{s}{v}}, v] \\
    a & \in [a, a] \\
    \frac{v}{t} &= a \\
    \frac{s}{t} &= v \\
    v \cdot v &\leq 2b (s - auth)
\end{align*}
\]

\[
\begin{align*}
    v \cdot v &\geq 2b (s - auth) \\
    v \cdot v &\leq 2b (s - auth)
\end{align*}
\]

**AutoBrake**

\[
\begin{align*}
    v & \in [0_{\frac{s}{v}}, v] \\
    a &= a \\
    \frac{v}{t} &= a \\
    \frac{s}{t} &= v \\
    v \cdot v &\geq 2b (s - auth)
\end{align*}
\]

**Leader**

**FreeRun**

\[
\begin{align*}
    v_l & \in [0_{\frac{s}{v}}, v_l] \\
    a_l & \in [a, a_l] \\
    \frac{v_l}{t} &= a_l \\
    \frac{s_l}{t} &= v_l
\end{align*}
\]

**Moving Block**

**Idle**

\[
\begin{align*}
    \frac{c}{t} &= 1 \\
    c &\leq 8
\end{align*}
\]

\[
\begin{align*}
    m' &= N(s_l, \sigma) \\
    c' &= 0
\end{align*}
\]

**Send**

\[
\begin{align*}
    \frac{m}{t} &= 0 \\
    \frac{c}{t} &= 1 \\
    c &\geq 8
\end{align*}
\]

\[
\begin{align*}
    \text{auth} &= 800 \\
    \text{auth'} &= m - len - sd
\end{align*}
\]

\[
\begin{align*}
    v &= 83.4_{\frac{s}{v}}, \quad len = 200, \quad sd = 400, \quad a = -1.4_{\frac{s}{v}}, \quad a = 0.7_{\frac{s}{v}}, \quad b = -0.7_{\frac{s}{v}}, \quad b = -0.3_{\frac{s}{v}}
\end{align*}
\]

[DEMO]
Case Study: ETCS

- **Results**
  - probability depending on message loss probability (p) and magnitude of measurement error (σ)
Tools – Summary

- **PRISM**
  www.prismmodelchecker.org
  - modelling language: guarded commands
  - model checking for PTA & MDP

- **The Modest Toolset**
  www.modestchecker.net
  - modelling language: Modest + guarded commands
  - prohver for STA (using PHAVer)
  - mcpta for PTA/MDP (using PRISM)
  - mctau for TA (using UPPAAL)
  - modes for statistical model checking

  Modest Toolset demo at poster & demo session tomorrow!