SMT-based Counterexample Generation for Markov Chains

Bettina Braitling¹  Ralf Wimmer¹  Bernd Becker¹  
Nils Jansen²  Erika Ábrahám²

¹Computer Architecture  
Albert-Ludwigs-University Freiburg, Germany

²Software Modeling and Verification  
RWTH Aachen, Germany

ROCKS Workshop  
March 27th, 2011
Motivation

- Complex (embedded) systems everywhere:

- Correct behaviour has to be ensured.
- Verification is needed.
  - Counterexamples
  - Bounded Model Checking (BMC)
- Some systems have probabilistic elements.
  - Stochastic Bounded Model Checking
# Table of Contents

1. Motivation

2. Foundations
   - Stochastic Models
   - Counterexample

3. Counterexample Generation
   - Previous Approaches
   - SMT-based Stochastic BMC (SSBMC)
   - Counterexamples for Markov Reward Models

4. Experimental Results

5. Conclusion & Outlook
A Discrete-Time Markov Chain (DTMC)

$M = (S, s_I, P, L)$ consists of

- $S$: finite set of states with initial state $s_I$,
- $P : S \times S \rightarrow [0, 1]$: matrix of transition probabilities,
- $L : S \rightarrow 2^{AP}$: labeling function.
A Markov Reward Model (MRM) \((M, R)\) consists of a DTMC \(M = (S, s_I, P, L)\) and a reward function \(R : S \rightarrow \mathbb{R}\).
Critical state $s_3$.
Should be reached with a probability of at most 0.08:

$$\mathcal{P}_{\leq 0.08}(\top \mathcal{U} s_3)$$
Critical state $s_3$.
Should be reached with a probability of at most 0.08:

\[ P_{\leq 0.08}(\top U s_3) \]

Does this property hold?
Given: A DTMC $M$ and a PCTL-property $\varphi = \mathcal{P}_{\leq p}(aUb)$.
Given: A DTMC $M$ and a PCTL-property $\varphi = \Pr_{\leq p}(aUb)$.

- An evidence is a finite path $\pi = s_0, s_1, \ldots, s_n$ with $s_0 = s_i$ and $\pi \models aUb$. $\pi$ is not a prefix of another evidence.
  
  $\pi$ has the probability $\Pr(\pi) = \prod_{i=0}^{n-1} P(s_i, s_{i+1})$. 
Given: A DTMC $M$ and a PCTL-property $\varphi = P_{\leq p}(aUb)$.

- An **evidence** is a finite path $\pi = s_0, s_1, \ldots, s_n$ with $s_0 = s_i$ and $\pi \models aUb$. $\pi$ is not a prefix of another evidence.
  
  $\pi$ has the probability $\Pr(\pi) = \prod_{i=0}^{n-1} P(s_i, s_{i+1})$.

- A **counterexample** is a set $C$ of evidences such that $\Pr(C) > p$. 

---

B. Braitling  (University of Freiburg)  SMT-based Counterexample Generation  03/27/2011  5 / 17
Given: An MRM $M$ and a PCTL-property $\varphi = \mathcal{P}_\leq_p (aU^I b)$, $I \subseteq \mathbb{R}$.
Counterexample for MRMs (1)

**Given:** An MRM $M$ and a PCTL-property $\varphi = \mathcal{P}_{\leq \rho}(aU^J b)$, $J \subseteq \mathbb{R}$.

- An **evidence** is a finite path $\pi = s_0, s_1, \ldots, s_n$ with $s_0 = s_I$ and $\pi \models aUb$. $\pi$ is not a prefix of another evidence.
  
  $\pi$ has the probability $\Pr(\pi) = \prod_{i=0}^{n-1} P(s_i, s_{i+1})$.

- A **counterexample** is a set $C$ of evidences such that $\Pr(C) > \rho$. 

---

B. Braitling  (University of Freiburg)
SMT-based Counterexample Generation
Counterexample for MRMs (1)

Given: An MRM $M$ and a PCTL-property $\varphi = \mathcal{P}_{\leq p}(aUb)$, $I \subseteq \mathbb{R}$.

- An evidence is a finite path $\pi = s_0, s_1, \ldots, s_n$ with $s_0 = s_I$ and $\pi \models aUb$. $\pi$ is not a prefix of another evidence.
  
  $\pi$ has the probability $\Pr(\pi) = \prod_{i=0}^{n-1} P(s_i, s_{i+1})$.

  $\pi$ has the reward $\Re(\pi) = \sum_{i=0}^{n-1} R(s_i)$ and $\Re(\pi) \in I$.

- A counterexample is a set $C$ of evidences such that $\Pr(C) > p$. 
Critical state $s_3$.
Should be reached with a probability of at most $0.08$, rewards $\leq 2$:

$$P_{\leq 0.08}(\top U^{[0,2]} s_3)$$

Does this property hold?
1 Motivation

2 Foundations
   • Stochastic Models
   • Counterexample

3 Counterexample Generation
   • Previous Approaches
   • SMT-based Stochastic BMC (SSBMC)
   • Counterexamples for Markov Reward Models

4 Experimental Results

5 Conclusion & Outlook
Previous Approaches

Explicit:

Symbolic:
Previous Approaches

Explicit:
- Shortest path:
  - Aljazzar & Leue, 2010
  - Han, Katoen & Damman, 2009
- Regular expressions:
  - Han, Katoen, & Damman, 2009
- Strongly Connected Components (SCCs):
  - Andrés, D’Argenio & van Rossum, 2008
  - Ábrahám, Jansen, Wimmer, Katoen & Becker, 2010

Symbolic:
Previous Approaches

Explicit:

- Shortest path:
  - Aljazzar & Leue, 2010
  - Han, Katoen & Damman, 2009

- Regular expressions:
  - Han, Katoen, & Damman, 2009

- Strongly Connected Components (SCCs):
  - Andrés, D’Argenio & van Rossum, 2008
  - Ábrahám, Jansen, Wimmer, Katoen & Becker, 2010

Symbolic:

- Shortest path:
  - Günther, Schuster & Siegle, 2010

- Stochastic BMC (SBMC):
  - Wimmer, Braitling & Becker, 2009
# Table of Contents

1. **Motivation**

2. **Foundations**
   - Stochastic Models
   - Counterexample

3. **Counterexample Generation**
   - Previous Approaches
   - SMT-based Stochastic BMC (SSBMC)
   - Counterexamples for Markov Reward Models

4. **Experimental Results**

5. **Conclusion & Outlook**
Consider transition probabilities during search:

$$I(s_0) \land \bigwedge_{i=0}^{k-1} T_{\text{SMT}}(s_i, s_{i+1}, \hat{p}_i) \land L_b(s_k) \land \left( \sum_{i=0}^{k-1} \hat{p}_i \geq \log p_t \right)$$

- Solved by an SMT-solver.
- Solution corresponds to an evidence $\pi$ of length $k$, $\Pr(\pi) \geq p_t$.
- Binary search by re-adjusting $p_t$. 
\[ n \leftrightarrow ((x \wedge \hat{p} = \log p_2) \lor (\neg x \wedge \hat{p} = \log p_1)) \]
SMT-based Stochastic BMC (SSBMC) (3)

- Search for new path
  - \( p_i := \frac{1}{2} p_t \) or \( k := k + 1 \)
  - Path found?
    - Yes
      - Probability
      - Mass
    - No
      - Big enough?
        - Yes
        - Finished
        - No
SMT allows us to consider rewards:

\[
I(s_0) \land \bigwedge_{i=0}^{k-1} T_{\text{SMT}}(s_i, s_{i+1}, \hat{p}_i) \land L_b(s_k) \land \left( \sum_{i=0}^{k-1} \hat{p}_i \geq \log p_t \right) \\
\land \bigwedge_{i=0}^{k-1} R(s_i, \hat{r}_i) \land \left( \min(J) \leq \sum_{i=0}^{k-1} \hat{r}_i \leq \max(J) \right)
\]

Only paths with rewards within interval $J$ are regarded.
Experimental Results

Comparison between SSBMC and SBMC.

**Benchmarks:**
- Contract signing protocol
- Crowds protocol
- Leader election protocol
- Self-stabilizing minimal spanning tree algorithm

**Setup:**
- Underlying solvers: Yices (SMT-solver), Minisat (SAT-solver).
- Dual Core AMD Opteron with 2.4 GHz per core, 4 GB RAM.
- Time limit: 2 h, memory limit: 2 GB
Computation Time for Contract, Crowds & Leader

- SSBMC
- SBMC

models:
- contract05_03
- contract05_08
- contract06_03
- contract06_06
- contract07_03
- crowdso2_05
- crowdso2_06
- crowdso2_07
- crowdso15_05
- crowdso15_06
- leadero3_04
- leadero3_08
- leadero4_04
- leadero4_06
- leadero5_04

(time in s)
Memory Consumption for Contract, Crowds & Leader

SSBMC  SBMC

models

memory (MB)

contract05_03  contract05_08  contract06_03  contract06_06  contract07_03  crowds02_05  crowds02_06  crowds02_07  crowds15_05  crowds15_06  leader03_04  leader03_08  leader04_04  leader04_06  leader05_04
Further Results

Minimal Spanning Tree:

<table>
<thead>
<tr>
<th>Name</th>
<th>$p$</th>
<th>$k_{\text{max}}$</th>
<th>SSBMC</th>
<th>SBMC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#paths</td>
<td>time</td>
<td>mem.</td>
<td>#paths</td>
</tr>
<tr>
<td>mst15</td>
<td>0.049</td>
<td>15</td>
<td>4531</td>
<td>98.58</td>
</tr>
<tr>
<td>mst16</td>
<td>0.047</td>
<td>16</td>
<td>4648</td>
<td>107.27</td>
</tr>
<tr>
<td>mst18</td>
<td>0.036</td>
<td>18</td>
<td>4073</td>
<td>109.26</td>
</tr>
<tr>
<td>mst20</td>
<td>0.034</td>
<td>20</td>
<td>452</td>
<td>19.57</td>
</tr>
</tbody>
</table>
Further Results

- **Minimal Spanning Tree:**

<table>
<thead>
<tr>
<th>Name</th>
<th>$p$</th>
<th>$k_{\text{max}}$</th>
<th>#paths</th>
<th>time</th>
<th>mem.</th>
<th>#paths</th>
<th>time</th>
<th>mem.</th>
</tr>
</thead>
<tbody>
<tr>
<td>mst15</td>
<td>0.049</td>
<td>15</td>
<td>4531</td>
<td>98.58</td>
<td>148.82</td>
<td>&gt; 600000</td>
<td>– TO –</td>
<td></td>
</tr>
<tr>
<td>mst16</td>
<td>0.047</td>
<td>16</td>
<td>4648</td>
<td>107.27</td>
<td>158.25</td>
<td>&gt; 600000</td>
<td>– MO –</td>
<td></td>
</tr>
<tr>
<td>mst18</td>
<td>0.036</td>
<td>18</td>
<td>4073</td>
<td>109.26</td>
<td>164.24</td>
<td>&gt; 600000</td>
<td>– MO –</td>
<td></td>
</tr>
<tr>
<td>mst20</td>
<td>0.034</td>
<td>20</td>
<td>452</td>
<td>19.57</td>
<td>58.21</td>
<td>&gt; 500000</td>
<td>– TO –</td>
<td></td>
</tr>
</tbody>
</table>

- **SSBMC for MRMs:**

<table>
<thead>
<tr>
<th>Model</th>
<th>$k_{\text{max}}$</th>
<th>$p$</th>
<th>#paths</th>
<th>time</th>
<th>mem.</th>
</tr>
</thead>
<tbody>
<tr>
<td>leader03.02</td>
<td>25</td>
<td>0.06226</td>
<td>360</td>
<td>1.00</td>
<td>29.61</td>
</tr>
<tr>
<td>leader04.02</td>
<td>25</td>
<td>0.21875</td>
<td>37376</td>
<td>912.11</td>
<td>1110.54</td>
</tr>
<tr>
<td>leader05.02</td>
<td>23</td>
<td>0.14771</td>
<td>4840</td>
<td>40.16</td>
<td>163.06</td>
</tr>
<tr>
<td>leader06.02</td>
<td>25</td>
<td>0.12378</td>
<td>32448</td>
<td>907.33</td>
<td>1360.11</td>
</tr>
</tbody>
</table>
# Table of Contents

1 Motivation

2 Foundations
   - Stochastic Models
   - Counterexample

3 Counterexample Generation
   - Previous Approaches
   - SMT-based Stochastic BMC (SSBMC)
   - Counterexamples for Markov Reward Models

4 Experimental Results

5 Conclusion & Outlook
Conclusion & Outlook

Conclusion:
- SMT for Stochastic BMC.
- Find paths with higher probabilities sooner.
- Counterexamples for MRMs.
- Promising results.

Outlook:
- Optimize the search for paths with higher probabilities.
- Additional features: Loop detection, bisimulation minimization, transition rewards, etc.
- More benchmarks, especially MRMs.
Conclusion & Outlook

Conclusion:
- SMT for Stochastic BMC.
- Find paths with higher probabilities sooner.
- Counterexamples for MRMs.
- Promising results.

Outlook:
- Optimize the search for paths with higher probabilities.
- Additional features: Loop detection, bisimulation minimization, transition rewards, ...
- More benchmarks, especially MRMs.