Counterexamples for Stochastic Model Checking

Husain Aljazzar

Chair for Software Engineering
University of Constance
Joint work with

- Holger Hermanns, University of Saarland
- Stefan Leue, University of Constance

„Counterexamples for Timed Probabilistic Reachability“

FORMATS 2005
Overview

- Introduction
- (Directed) Explicit-State Reachability Analysis
- Directed Probabilistic Reachability Analysis
- Case Study and Experimental Results
- Future Work & Conclusion
Overview

- Introduction
- (Directed) Explicit-State Reachability Analysis
- Directed Probabilistic Reachability Analysis
- Case Study and Experimental Results
- Future Work & Conclusion
Motivation

- Stochastic models, e.g. *DTMC* and *CTMC*: performance and dependability analysis.

- A few model checking approaches for stochastic models have been presented.

- **Common weakness**: Inability to give detailed debugging information (Counterexamples).

- **Approach**: Use (Directed) Explicit-State Model Checking (ESMC/DESMC) in the reachability analysis of stochastic models to deliver counterexamples.
A DTMC is a quadruple \((S, s_0, P, L)\), where

- \(S\) is a finite set of states, and
- \(s_0 \in S\) is an initial state
- \(P : S \times S \to \mathbb{R}\) is the transition probability matrix,
- \(L : S \to 2^{AP}\) is labeling function.

An finite/ infinite run:

\[
s_0 \rightarrow s_1 \rightarrow s_2 \rightarrow \ldots \rightarrow s_n, \\
\]

\[
s_0 \rightarrow s_1 \rightarrow s_2 \rightarrow \ldots ,
\]
Explicit-State Model Checking (ESMC) -- Transition Systems

- Explicit-State model checking (ESMC): exploring the state space using graph search algorithms like DFS and BFS.
- If an error is found, an offending system run is returned (Counterexample)
- What constitutes a good counterexample?
  - In typical non-stochastic transition systems: good = short
- How to obtain good (short) counterexamples?
  - Optimizing Search (Best First)
    - BFS
    - Directed Explicit-State Model Checking (DESMC), i.e., Heuristic Search, e.g. Greedy Best First (GBestFS) or A*
Directed Explicit-State Model-Checking (DESMC) -- Transition Systems

- Directed search algorithms use knowledge about
  - the state space or/and
  - the specification of the goal state

- A heuristic function $h$ is used in the state evaluation.

- Advantages of DESMC: Improving the performance
  - Memory effort
  - Runtime
Overview

- Introduction
- (Directed) Explicit-State Reachability Analysis
- Directed Probabilistic Reachability Analysis
- Case Study and Experimental Results
- Future Work & Conclusion

Software Engineering
Counterexamples for Stochastic Models

- Use ESMC or DESMC on stochastic models

- What is a good counterexample in stochastic models?
  - A counterexample which carries a high probability mass (more informative).
  - The length of a run is not indicative of its probability mass.
  - Timed run probability
Timed Run Probability $\gamma$

- Let $r = s_0 \rightarrow s_1 \rightarrow s_2 \rightarrow ... \rightarrow s_n$ be a run.

- The timed run probability of $r$, $\gamma(r, k)$, is the probability to execute $r$ within at most $k$ time units.

$\gamma(r, k) = P(s_{n-1}, s_n) \cdot \sum_{i=0}^{k-1} \pi(s_{n-1}, i)$

Note: For CTMCs it is more complicated
   The determination of the timed run probability is computationally very expensive.
   $\rightarrow$ An approximation based on Uniformisation of the model.
ESMC and DESMC for Stochastic Models

Idea: Use of optimizing algorithms with the timed run probability as optimization criterion!

- Dijkstra, (ESMC)
- GBestFS (DESMC)
- Z* (DESMC)
Overview

- Introduction
- (Directed) Explicit-State Reachability Analysis
- Directed Probabilistic Reachability Analysis
- Case Study and Experimental Results
- Future Work
- Conclusion
Case-Study: SCSI-2-Protocol

In our experiments:
- One Controller
- One main disk (frequently used)
- Two backup disks (rarely used)

The system was modeled in LOTOS and transformed into an interactive Markov chain (IMC) by the CADP toolbox.
SCSI-2-Protocol: A Timed Reachability Property

- **Main disk overload (MDOL):** The main disk is overloaded while the backup disks are not accessed.

- The probability to reach a MDOL state within $t$ time units does not exceed 0.3.
SCSI-2-Protocol: Counterexample

- The counterexample delivered by $Z^*$
SCSI-Protocol: Experimental Results

- For \( t \in \{1, 2, \ldots, 10\} \)

<table>
<thead>
<tr>
<th>Time bound</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>0.235</td>
<td>0.312</td>
<td>0.327</td>
<td>0.329</td>
<td>0.329</td>
<td>0.329</td>
<td>0.330</td>
<td>0.330</td>
<td>0.330</td>
<td>0.330</td>
</tr>
<tr>
<td>DFS</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>0.000</td>
<td>-</td>
<td>-</td>
<td>0.000</td>
</tr>
<tr>
<td>BFS</td>
<td>-</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
</tr>
<tr>
<td>Dijkstra</td>
<td>-</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
</tr>
<tr>
<td>GBestFS</td>
<td>-</td>
<td>0.012</td>
<td>0.012</td>
<td>0.012</td>
<td>0.012</td>
<td>0.012</td>
<td>0.012</td>
<td>0.012</td>
<td>0.012</td>
<td>0.012</td>
</tr>
<tr>
<td>Z*</td>
<td>-</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
<td>0.161</td>
</tr>
</tbody>
</table>
SCSI-Protocol: Experimental Results

Runtime

- BFS and DFS do not scale to large models.
- Good runtime behavior of Dijkstra, GBestFS, Z*
- Directed algorithms GBestFS and Z* have the best runtime performance.
Memory effort

- The behavior of DFS and BFS is unacceptable.
- Dijkstra does not scale to large models
- Z* and GBestFS bring significant improvement
- GBestFS has the best behavior.
Overview

- Introduction
- (Directed) Explicit-State Reachability Analysis
- Directed Probabilistic Reachability Analysis
- Case Study and Experimental Results
- Future Work & Conclusion
Future Work

- More case studies
- Finding more than one path (counterexample = offending tree)
- Visualization of counterexamples
- General heuristics
- Non-Determinism (CT Markov Decision Processes)
Conclusion

- Novel approach to generate counterexamples for timed probabilistic reachability analysis.
- Heuristic guided
- Good experimental results
- A good step in the right direction
Thanks for your attention!