

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/4341826>

# Probabilistic Model Checking and Reliability of Results

Conference Paper · May 2008

DOI: 10.1109/DDECS.2008.4538787 · Source: IEEE Xplore

CITATIONS

9

READS

39

4 authors, including:



**Ralf Wimmer**

Concept Engineering GmbH

104 PUBLICATIONS 1,058 CITATIONS

[SEE PROFILE](#)



**Marc Herbstritt**

34 PUBLICATIONS 276 CITATIONS

[SEE PROFILE](#)



**Bernd Becker**

University of Freiburg

601 PUBLICATIONS 7,011 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



SC-square: Symbolic Computation and Satisfiability Checking [View project](#)



Automatic Verification and Analysis of complex Systems [View project](#)

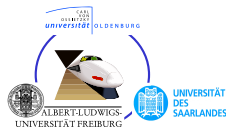
# Probabilistic Model Checking and Reliability of Results



Ralf Wimmer, Alexander Kortus, Marc Herbstritt, Bernd Becker

Bratislava, Slovakia

April 16–18, 2008



# Outline

- 1 Introduction
  - Foundations of Probabilistic Model Checking
  - Motivational Example
- 2 Sources of Inaccuracy
- 3 Ideas for Obtaining Reliable Results
  - Degree of Belief
  - Exact Arithmetic
  - Interval Arithmetic
  - Certificates
- 4 Conclusion

# Introduction

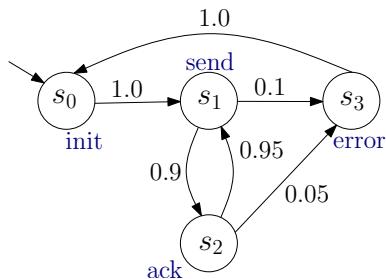
## Why do we need probabilistic systems?

- Our world is inherently stochastic:
  - Component failures
  - Inputs arrive with a probability distribution
  - Randomized algorithms / protocols
- Properties cannot be guaranteed to hold under all circumstances, but only with a certain probability.
  - The probability that component failures cause a safety-critical system state within 1000 steps is at most  $10^{-6}$ .

# Probabilistic Models

Very simple model:

## Discrete-time Markov Chains



# Describing Probabilistic Properties: PCTL

## Extension to Computation Tree Logic (CTL)

- Path formulae as in CTL:
  - $X \varphi$
  - $\varphi_1 U \varphi_2$
  - $\varphi_1 U^{\leq k} \varphi_2$
- No  $A$  and  $E$  path quantifiers
- Instead: Probabilistic  $\mathcal{P}_{\bowtie p}$  quantifier ( $\bowtie \in \{\leq, \geq, <, >\}$ )

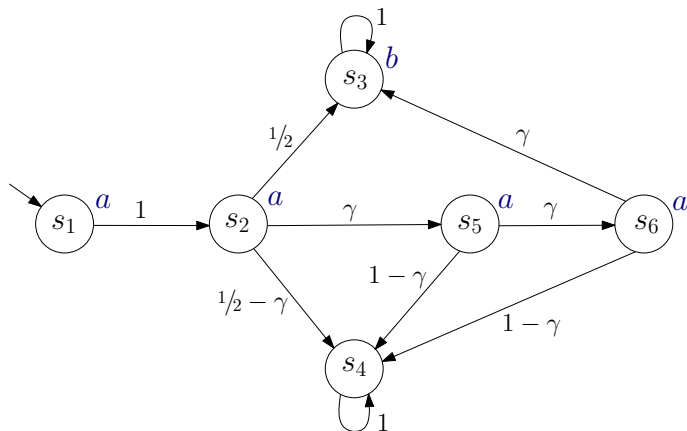
$\mathcal{P}_{\bowtie p}(\psi)$ : the probability to walk along a path satisfying  $\psi$  has to be within the bound  $\bowtie p$

**Model Checking for PCTL:** Traverse the syntax tree of the formula and mark the states with the subformulae that are satisfied there.

- $X$  1 matrix-vector multiplication
- $U^{\leq k}$   $k$  matrix-vector multiplications
- $U$  Solution of a system of linear equations  
Reduction to matrix-vector multiplication by using iterative methods (e.g. Jacobi method)



# A Discrete-time Markov Chain (DTMC)

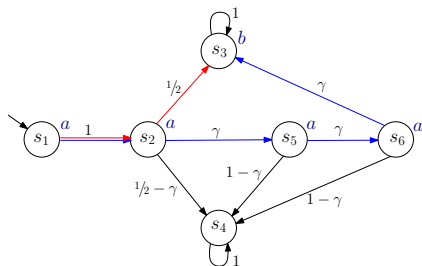


Probability to compute:

$$P^?(c \cup P_{\leq 1/2}(a \cup b))$$

# Computing probabilities

Let  $\gamma$  be a small constant  $< 1/2$ .



State	$P^?(a U b)$	$P_{\leq 1/2}(a U b)$ satisfied?
$s_1$	$1/2 + \gamma^3$	no
$s_2$	$1/2 + \gamma^3$	no
$s_3$	1	no
$s_4$	0	yes
$s_5$	$\gamma^2$	yes
$s_6$	$\gamma$	yes

# PRISM

Let's see, what PRISM says for  $\gamma = 10^{-6}$ :

```
probabilistic
const double gamma = 0.000001;

module sys
  s: [1..6] init 1;

  [] s=1 -> 1.0: (s'=2);
  [] s=2 -> 0.5: (s'=3) + gamma: (s'=5) + (0.5-gamma): (s'=4);
  [] s=3 -> 1.0: (s'=3);
  [] s=4 -> 1.0: (s'=4);
  [] s=5 -> gamma: (s'=6) + (1-gamma): (s'=4);
  [] s=6 -> gamma: (s'=3) + (1-gamma): (s'=4);
endmodule
```

Result:

```
yes = 5, no = 1, maybe = 0
Time for model checking: 0.022 seconds.
Result: 1.0
```

# MRMC

What does MRMC say (for  $\gamma = 10^{-6}$ )?

Transitions:

```
STATES 6
TRANSITIONS 10
1 2 1.0
2 3 0.5
2 4 0.499999
2 5 0.000001
3 3 1.0
4 4 1.0
5 4 0.999999
5 6 0.000001
6 3 0.000001
6 4 0.999999
```

Labels:

```
#DECLARATION
a b c
#END
1 a
2 a
3 b
5 a
6 a
```

Result:

```
$RESULT: ( 1.0000000, 1.0000000, 0.0000000, 1.0000000, 1.0000000, 1.0000000 )
$STATE: { 1, 2, 4, 5, 6 }
```

# The origin of the problem

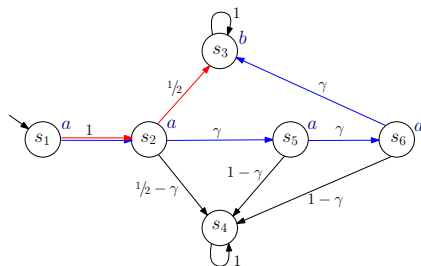
- The model checker has to represent the value  $\frac{1}{2} + \gamma^3$  such that it is larger than 0.5.
- For  $\gamma = 10^{-6}$  this value is

$$\frac{1}{2} + 10^{-18} = 0.500000000000000001$$

- Floating-point arithmetic with 64 bits can represent numbers with an accuracy of about 15 decimal digits.
- The number is rounded downwards to 0.5.
- Changing the value of  $\frac{1}{2} + \gamma^3$  to 0.5 flips the truth value of  $P_{\leq 0.5}(a \cup b)$  in  $s_1$  and  $s_2$  from “no” to “yes”.

# Computing probabilities – inexact arithmetic

Let  $\gamma$  be a small constant  $< 1/2$ .



State	$P^?(a \cup b)$	$P_{\leq 1/2}(a \cup b)$ satisfied?
$s_1$	$1/2 + \gamma^3$	yes
$s_2$	$1/2 + \gamma^3$	yes
$s_3$	1	no
$s_4$	0	yes
$s_5$	$\gamma^2$	yes
$s_6$	$\gamma$	yes

## Sources of Inaccuracy

# Sources of Inaccuracy

- **Inexact arithmetic** for additions and multiplications (floating-point arithmetic according to IEEE 754 standard)
- Termination criterion for the **iterative solution of linear equation systems** (Jacobi / Gauss-Seidel method)
- Symbolic Model Checking: **MTBDD-packages** only create a new leaf with value  $v$  if there is no other leaf with value  $v'$  and  $|v - v'| < \varepsilon$  for some constant  $\varepsilon > 0$ .
- Continuous-time Markov chains: model checking for the time-bounded until operator requires **uniformization of the CTMC**, i. e. the evaluation of an infinite sum, which has to be truncated after a finite number of summands.



## Ideas for Obtaining Reliable Results

# Degree of Belief (1)

Given the PCTL formula

$$\varphi = \mathcal{P}_{\bowtie p}(\psi)$$

we define the **degree of believe**

$$db(\varphi) = \min_{s \in S} |\Pr(s, \psi) - p|$$

If  $db(\varphi)$  is close to 0 for some sub-formula  $\varphi$  of our formula under consideration, the risk is high that wrong results are produced due to inexact computations.

## Degree of Belief (2)

### **Advantage:**

- Easy to integrate into existing tools.

### **Disadvantage:**

- Does not provide any guarantees for the correctness.

# Exact Arithmetic (1)

**Pro** solves the problem of rounding when using floating-point arithmetic.

**Pro** solves the problem of the restriction in the BDD packages

**Contra** inexact solutions to linear equations systems remain due to the iterative solution methods.

**Contra** the problem of uniformization of CTMCs also remains.

**Contra** the evaluation of the time-bounded next operator on CTMCs requires the computation of  $e^{-\lambda t}$  for a  $\lambda > 0$  which cannot be done exactly using rational arithmetic.

## Exact Arithmetic (2)

### **Disadvantages:**

- very expensive w. r. t. runtime and memory consumption
- does not solve most of the problems.

# Interval Arithmetic (1)

- Compute **safe intervals** which contain the probability with which a path formula holds in a state, i. e.

$$I_\psi(s) = [a, b] \text{ such that } a \leq \Pr(s, \psi) \leq b$$

- **Three-valued interpretation** of PCTL:

$$s \models \mathcal{P}_{\bowtie p}(\psi) \quad \Leftrightarrow \quad \forall x \in I_\psi(s) : x \bowtie p$$

$$s \not\models \mathcal{P}_{\bowtie p}(\psi) \quad \Leftrightarrow \quad \exists x \in I_\psi(s) : x \not\bowtie p$$

result unknown otherwise.

There are papers about obtaining such intervals for the solutions of linear equation systems and for the uniformization of CTMCs (Fox-Glynn approximation).

## Interval Arithmetic (2)

### **Advantage:**

- Intervals can be derived efficiently for all necessary operations.
- Results are really correct.

### **Disadvantage:**

- Overhead of a (probably) small factor
- We have to live with “unknowns”.

# Certificates (1)

A certificate is an output besides the yes/no answer which makes it easy to check that the decision of the tool is correct.

## Example: Linear Programming

**Question:** Has  $Ax = b$ ,  $x \geq 0$  a satisfying solution?

**Yes:** Give a satisfying assignment.

**No:** Give a satisfying assignment for  $y^T A \geq 0$ ,  $y^T b < 0$ , since exactly one of the two systems is satisfiable.



# Certificates for DTMCs (1)

Property:

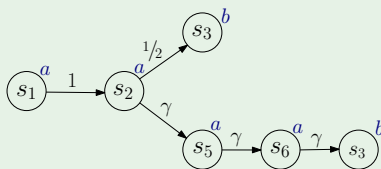
$$\mathcal{P}_{>p}(\varphi_1 U \varphi_2)$$

Certificate:

A finite set of finite paths with probability mass  $> p$ .

## Example

Property  $\mathcal{P}_{>0.5}(aUb)$ :



# Certificates for DTMCs (2)

## Advantages:

- easily verifiable using exact arithmetic.
- can guide the designer to find bugs in the design

## Disadvantages:

- only applicable for formulae of the form  $\mathcal{P}_{>p}(aUb)$  without nested sub-formulae
- certificates may consist of a large number of paths.

## Paper:

Han, Katoen – Counterexamples in Probabilistic Model Checking, TACAS 2007

## Conclusion

# Conclusion

- Reliable results are critical in probabilistic model checking.
- State-of-the-art model checkers use inexact computations. This may lead to wrong results.
- Interval arithmetic seems promising, but further research is necessary.