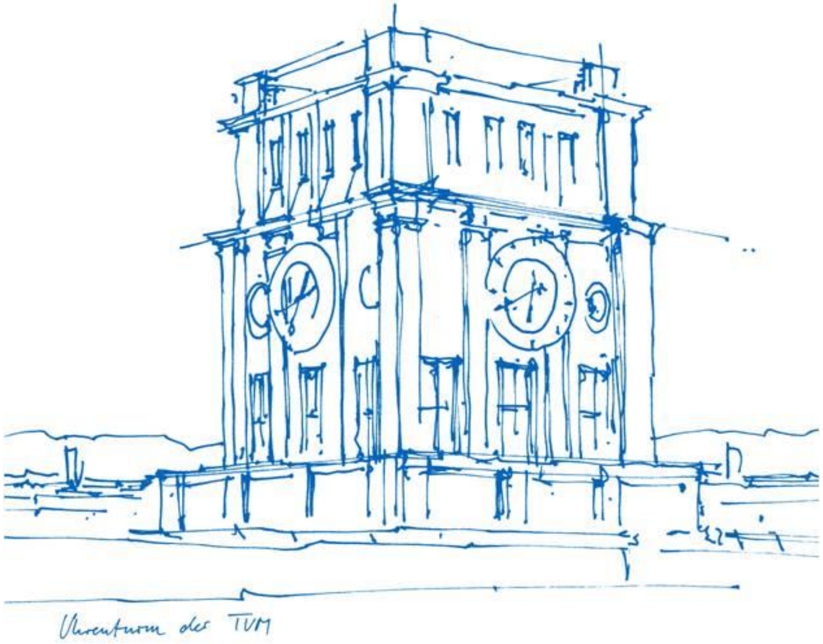
Recent Advances in Model Cheking

Technische Universität München

Lehrstuhl für Theoretische Informatik

Ort, Datum: Garching, 05. July 2022



Recent Advances in Model Checking

Paper: Correct Probabilistic Model Checking with Floating-Point Arithmetic - Arnd Hartmanns

Key points:

- Value Iteration can lead to wrong results in model checking
  - Examples given in the Paper
- Solved by using Interval Iteration and controlling the Rounding mode
  - Algorithm with rounding mode shown in the paper
- Experiments showing that the given algorithm works were done in paper

Recent Advances in Model Checking

- Value Iteration can lead to wrong results in model checking

–Examples given in the Paper

–→ Verify Examples

- Solved by using Interval Iteration and controlling the Rounding mode

–Algorithm with rounding mode shown in the paper

- Experiments showing that the given algorithm works were done in paper

Recent Advances in Model Checking

- Value Iteration can lead to wrong results in model checking

–Examples given in the Paper

–→ Verify Examples

- Solved by using Interval Iteration and controlling the Rounding mode

–Algorithm with rounding mode shown in the paper

–→ Figure out how to control Rounding modes in C

–→ Implement Algorithm

- Experiments showing that the given algorithm works were done in paper

Recent Advances in Model Checking

● Value Iteration can lead to wrong results in model checking

–Examples given in the Paper

–→ Verify Examples

● Solved by using Interval Iteration and controlling the Rounding mode

–Algorithm with rounding mode shown in the paper

–→ Figure out how to control Rounding modes in C

–→ Implement Algorithm

● Experiments showing that the given algorithm works were done in paper

→ Confirm Experiments*

*or maybe do something else like finding counterexamples

Recent Advances in Model Checking

→ Verify Examples in the paper in PRISM

→ Verify Examples in the paper in STORM

→ Figure out how to control Rounding modes in C

→ Implement Algorithm

→ Confirm Experiments*

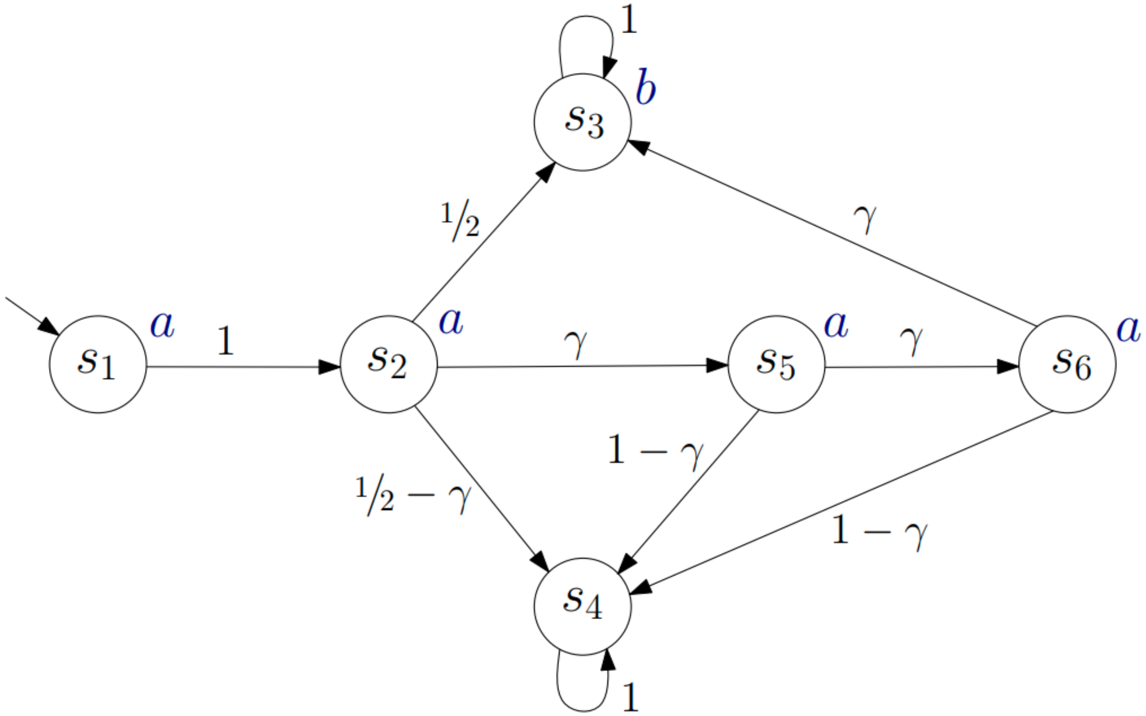
→ Own potential ideas (more research needed first):

- When is controlled rounding is NOT needed ?
- Are there cases where controlled rounding is worse than normal ? (maybe higher run time in some cases)

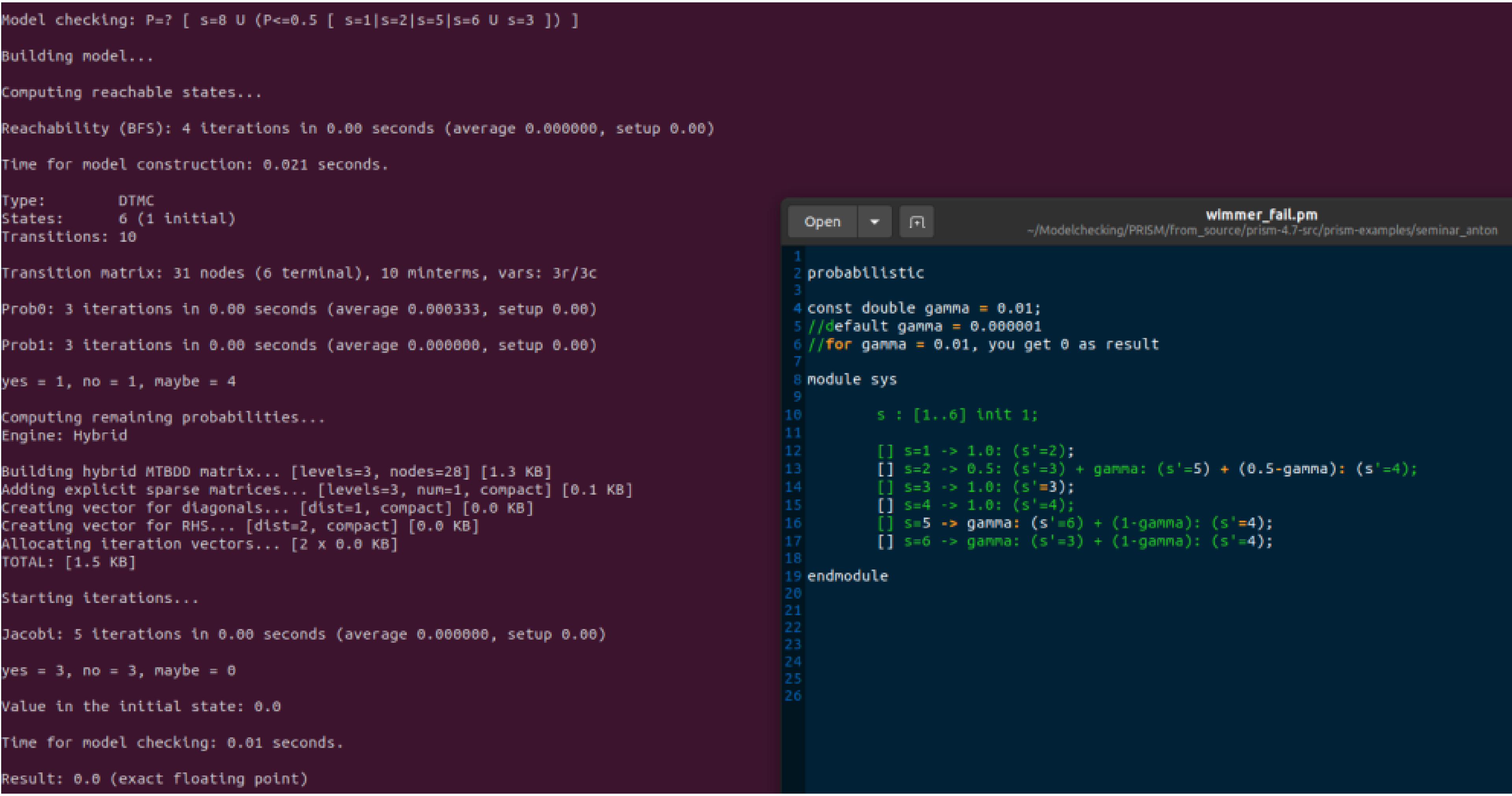
PRISM - Example

Example from „Probabilistic Model Checking and Reliability of Results“ (Wimmer et al)

- for small values of γ, the model checkers give wrong results



PRISM Example – big gamma, no issues (correct result 0.0)

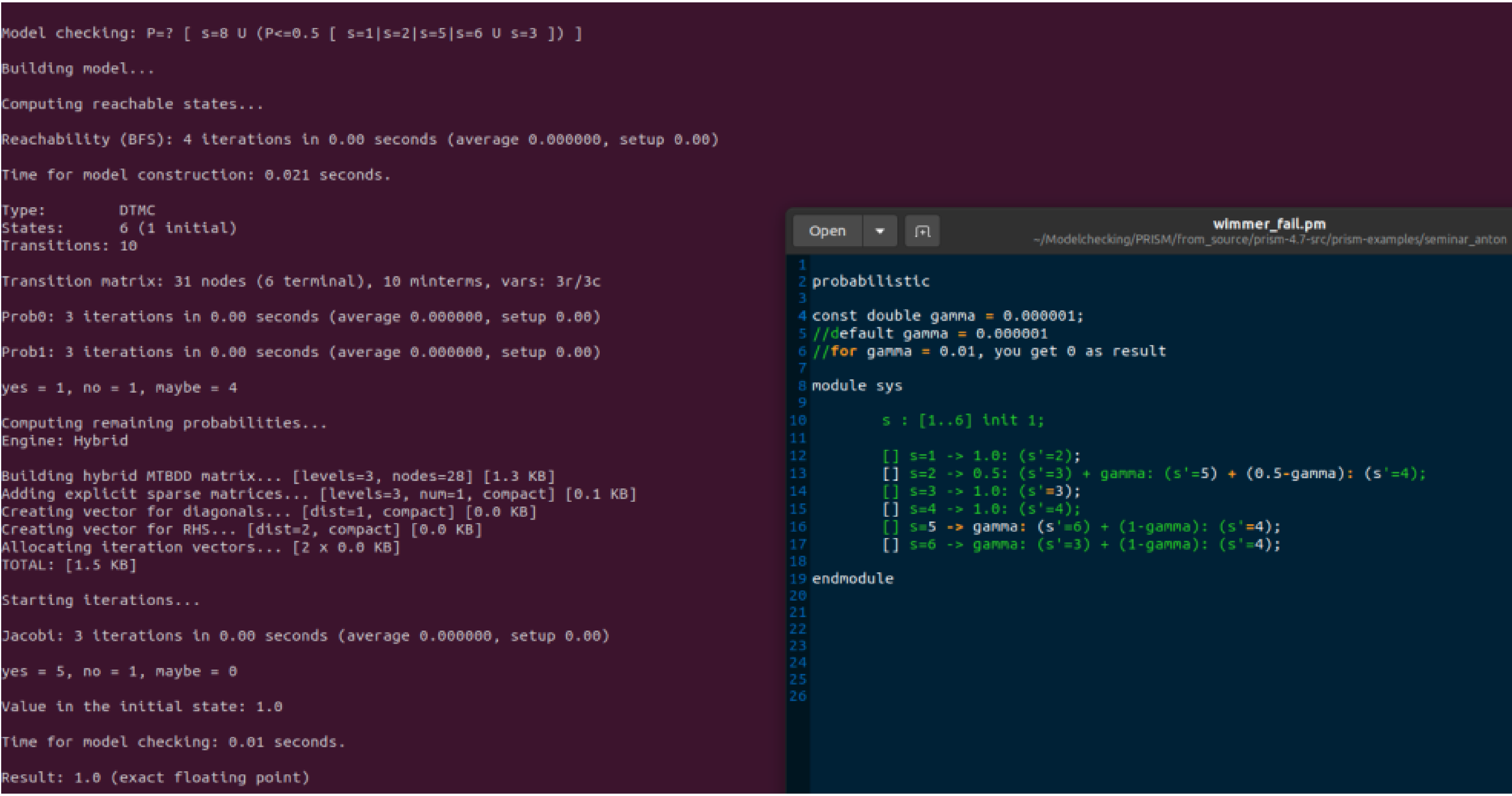


```
Model checking: P=? [ s=8 U (P<=0.5 [ s=1|s=2|s=5|s=6 U s=3 ]) ]

Building model...

Computing reachable states...

Reachability (BFS): 4 iterations in 0.00 seconds (average 0.000000, setup 0.00)

Time for model construction: 0.021 seconds.

Type:        DTMC
States:      6 (1 initial)
Transitions: 10

Transition matrix: 31 nodes (6 terminal), 10 minterms, vars: 3r/3c

Prob0: 3 iterations in 0.00 seconds (average 0.000333, setup 0.00)

Prob1: 3 iterations in 0.00 seconds (average 0.000000, setup 0.00)

yes = 1, no = 1, maybe = 4

Computing remaining probabilities...
Engine: Hybrid

Building hybrid MTBDD matrix... [levels=3, nodes=28] [1.3 KB]
Adding explicit sparse matrices... [levels=3, num=1, compact] [0.1 KB]
Creating vector for diagonals... [dist=1, compact] [0.0 KB]
Creating vector for RHS... [dist=2, compact] [0.0 KB]
Allocating iteration vectors... [2 x 0.0 KB]
TOTAL: [1.5 KB]

Starting iterations...

Jacobi: 5 iterations in 0.00 seconds (average 0.000000, setup 0.00)

yes = 3, no = 3, maybe = 0

Value in the initial state: 0.0

Time for model checking: 0.01 seconds.

Result: 0.0 (exact floating point)
```

wimmer_fall.pm
~/Modelchecking/PRISM/from_source/prism-4.7-src/prism-examples/seminar_anton

```

probabilistic

const double gamma = 0.01;
//default gamma = 0.000001
//for gamma = 0.01, you get 0 as result

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
```

PRISM Example – small gamma, rounding issues

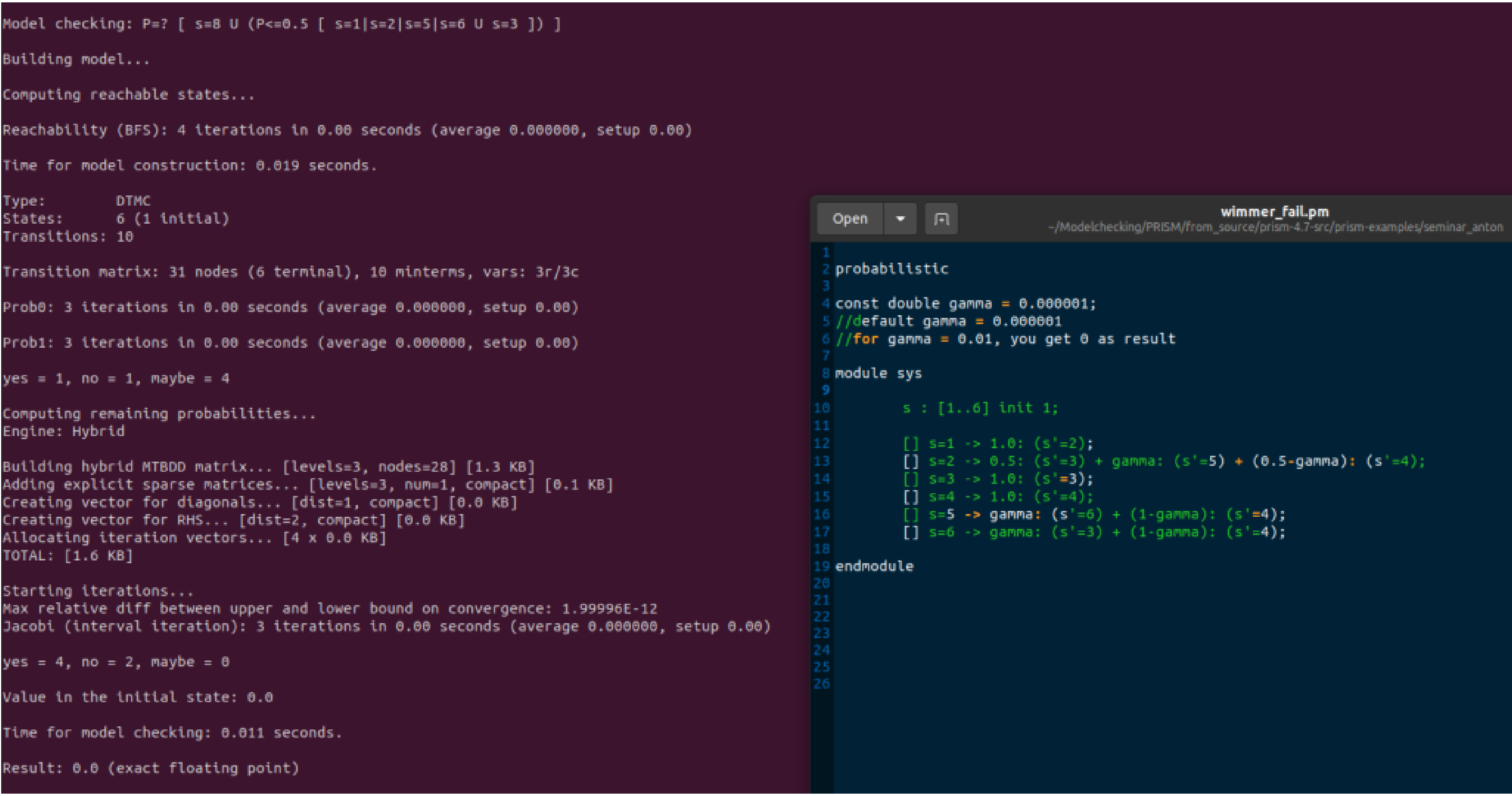


```
Model checking: P=? [ s=8 U (P<=0.5 [ s=1|s=2|s=5|s=6 U s=3 ]) ]

Building model...

Computing reachable states...

Reachability (BFS): 4 iterations in 0.00 seconds (average 0.000000, setup 0.00)

Time for model construction: 0.021 seconds.

Type:        DTMC
States:      6 (1 initial)
Transitions: 10

Transition matrix: 31 nodes (6 terminal), 10 minterms, vars: 3r/3c

Prob0: 3 iterations in 0.00 seconds (average 0.000000, setup 0.00)

Prob1: 3 iterations in 0.00 seconds (average 0.000000, setup 0.00)

yes = 1, no = 1, maybe = 4

Computing remaining probabilities...
Engine: Hybrid

Building hybrid MTBDD matrix... [levels=3, nodes=28] [1.3 KB]
Adding explicit sparse matrices... [levels=3, num=1, compact] [0.1 KB]
Creating vector for diagonals... [dist=1, compact] [0.0 KB]
Creating vector for RHS... [dist=2, compact] [0.0 KB]
Allocating iteration vectors... [2 x 0.0 KB]
TOTAL: [1.5 KB]

Starting iterations...

Jacobi: 3 iterations in 0.00 seconds (average 0.000000, setup 0.00)

yes = 5, no = 1, maybe = 0

Value in the initial state: 1.0

Time for model checking: 0.01 seconds.

Result: 1.0 (exact floating point)
```

Open
wimmer_fail.pm
~/Modelchecking/PRISM/from_source/prism-4.7-src/prism-examples/seminar_anton

```

probabilistic

const double gamma = 0.000001;
//default gamma = 0.000001
//for gamma = 0.01, you get 0 as result

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
```

PRISM Example – improvement with Interval Iteration

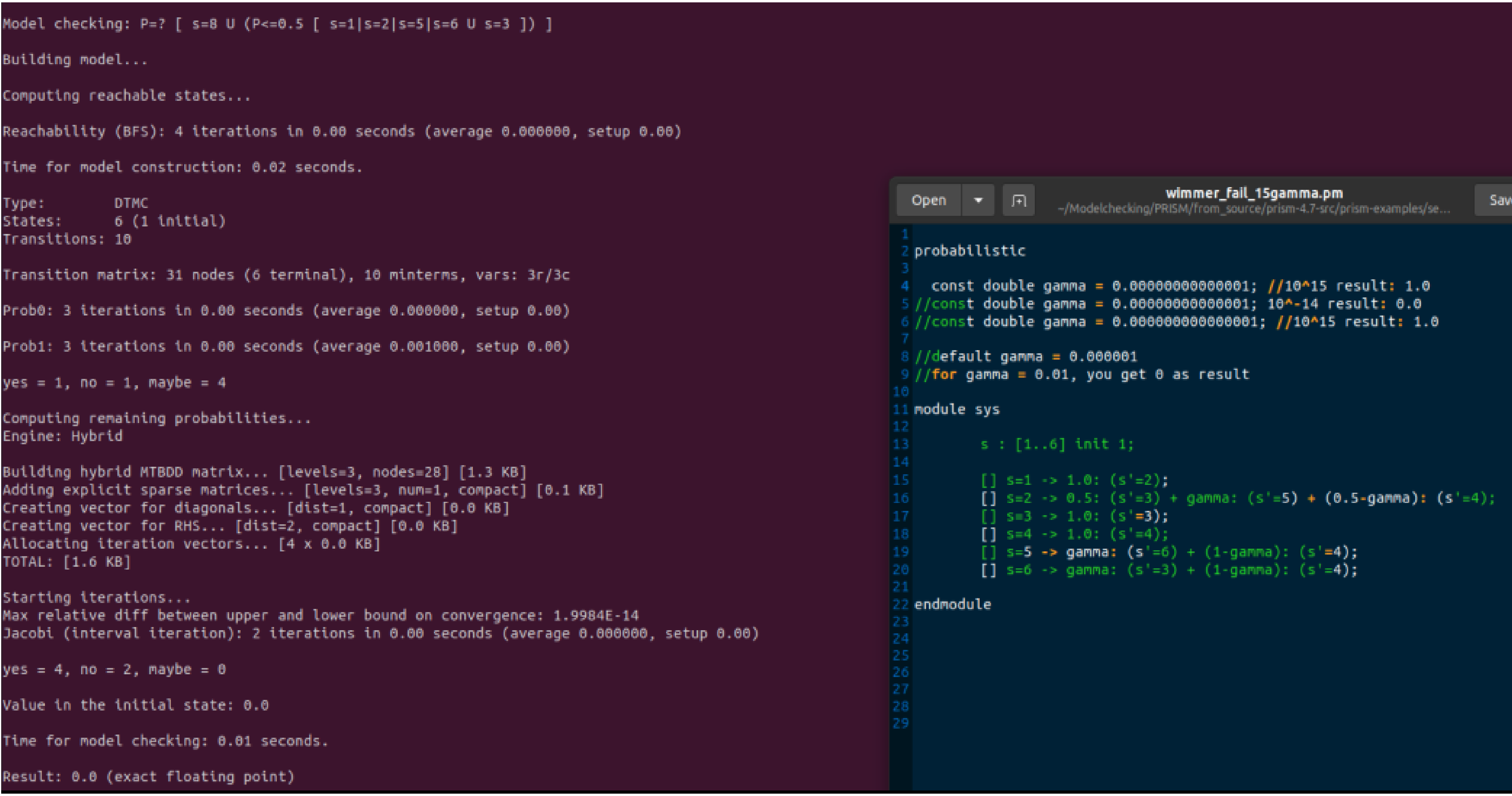


```
Model checking: P=? [ s=8 U (P<=0.5 [ s=1|s=2|s=5|s=6 U s=3 ]) ]

Building model...

Computing reachable states...

Reachability (BFS): 4 iterations in 0.00 seconds (average 0.000000, setup 0.00)

Time for model construction: 0.019 seconds.

Type:        DTMC
States:      6 (1 initial)
Transitions: 10

Transition matrix: 31 nodes (6 terminal), 10 minterms, vars: 3r/3c

Prob0: 3 iterations in 0.00 seconds (average 0.000000, setup 0.00)

Prob1: 3 iterations in 0.00 seconds (average 0.000000, setup 0.00)

yes = 1, no = 1, maybe = 4

Computing remaining probabilities...
Engine: Hybrid

Building hybrid MTBDD matrix... [levels=3, nodes=28] [1.3 KB]
Adding explicit sparse matrices... [levels=3, num=1, compact] [0.1 KB]
Creating vector for diagonals... [dist=1, compact] [0.0 KB]
Creating vector for RHS... [dist=2, compact] [0.0 KB]
Allocating iteration vectors... [4 x 0.0 KB]
TOTAL: [1.6 KB]

Starting iterations...
Max relative diff between upper and lower bound on convergence: 1.99996E-12
Jacobi (interval iteration): 3 iterations in 0.00 seconds (average 0.000000, setup 0.00)

yes = 4, no = 2, maybe = 0

Value in the initial state: 0.0

Time for model checking: 0.011 seconds.

Result: 0.0 (exact floating point)
```

Open
wimmer_fail.pm
~/Modelchecking/PRISM/from_source/prism-4.7-src/prism-examples/seminar_anton

```

probabilistic

const double gamma = 0.000001;
//default gamma = 0.000001
//for gamma = 0.01, you get 0 as result

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
```

PRISM Example – Interval Iteration correct until 10^14

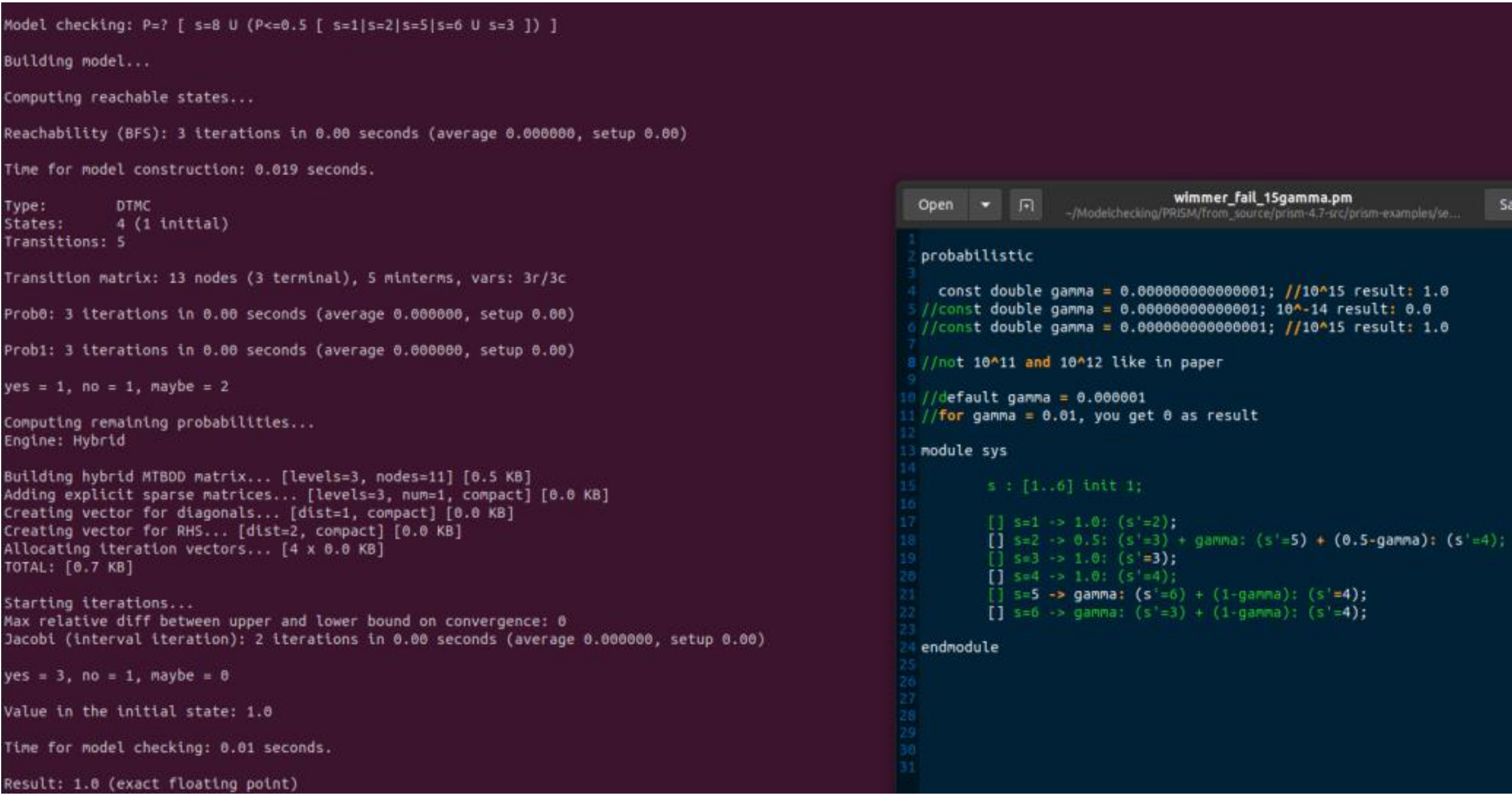


```
Model checking: P=? [ s=8 U (P<=0.5 [ s=1|s=2|s=5|s=6 U s=3 ]) ]

Building model...

Computing reachable states...

Reachability (BFS): 4 iterations in 0.00 seconds (average 0.000000, setup 0.00)

Time for model construction: 0.02 seconds.

Type:        DTMC
States:      6 (1 initial)
Transitions: 10

Transition matrix: 31 nodes (6 terminal), 10 minterms, vars: 3r/3c

Prob0: 3 iterations in 0.00 seconds (average 0.000000, setup 0.00)

Prob1: 3 iterations in 0.00 seconds (average 0.001000, setup 0.00)

yes = 1, no = 1, maybe = 4

Computing remaining probabilities...
Engine: Hybrid

Building hybrid MTBDD matrix... [levels=3, nodes=28] [1.3 KB]
Adding explicit sparse matrices... [levels=3, num=1, compact] [0.1 KB]
Creating vector for diagonals... [dist=1, compact] [0.0 KB]
Creating vector for RHS... [dist=2, compact] [0.0 KB]
Allocating iteration vectors... [4 x 0.0 KB]
TOTAL: [1.6 KB]

Starting iterations...
Max relative diff between upper and lower bound on convergence: 1.9984E-14
Jacobi (interval iteration): 2 iterations in 0.00 seconds (average 0.000000, setup 0.00)

yes = 4, no = 2, maybe = 0

Value in the initial state: 0.0

Time for model checking: 0.01 seconds.

Result: 0.0 (exact floating point)
```

~/Modelchecking/PRISM/from_source/prism-4.7-src/prism-examples/se...

```

probabilistic

  const double gamma = 0.00000000000001; //10^15 result: 1.0
//const double gamma = 0.00000000000001; 10^-14 result: 0.0
//const double gamma = 0.000000000000001; //10^15 result: 1.0

//default gamma = 0.000001
//for gamma = 0.01, you get 0 as result

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
```

PRISM Example – rounding issues with <=10^15



Open
wimmer_fail_15gamma.pm
~/Modelchecking/PRISM/from_source/prism-4.7-src/prism-examples/se...
Sa

```
Model checking: P=? [ s=8 U (P<=0.5 [ s=1|s=2|s=5|s=6 U s=3 ]) ]

Building model...

Computing reachable states...

Reachability (BFS): 3 iterations in 0.00 seconds (average 0.000000, setup 0.00)

Time for model construction: 0.019 seconds.

Type:        DTMC
States:      4 (1 initial)
Transitions: 5

Transition matrix: 13 nodes (3 terminal), 5 minterms, vars: 3r/3c

Prob0: 3 iterations in 0.00 seconds (average 0.000000, setup 0.00)

Prob1: 3 iterations in 0.00 seconds (average 0.000000, setup 0.00)

yes = 1, no = 1, maybe = 2

Computing remaining probabilities...
Engine: Hybrid

Building hybrid MTBDD matrix... [levels=3, nodes=11] [0.5 KB]
Adding explicit sparse matrices... [levels=3, num=1, compact] [0.0 KB]
Creating vector for diagonals... [dist=1, compact] [0.0 KB]
Creating vector for RHS... [dist=2, compact] [0.0 KB]
Allocating iteration vectors... [4 x 0.0 KB]
TOTAL: [0.7 KB]

Starting iterations...
Max relative diff between upper and lower bound on convergence: 0
Jacobi (interval iteration): 2 iterations in 0.00 seconds (average 0.000000, setup 0.00)

yes = 3, no = 1, maybe = 0

Value in the initial state: 1.0

Time for model checking: 0.01 seconds.

Result: 1.0 (exact floating point)
```

```

probabilistic

  const double gamma = 0.000000000000001; //10^15 result: 1.0
//const double gamma = 0.00000000000001; 10^-14 result: 0.0
//const double gamma = 0.000000000000001; //10^15 result: 1.0

//not 10^11 and 10^12 like in paper

//default gamma = 0.000001
//for gamma = 0.01, you get 0 as result

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
```

Open Tasks and Timeline

→ Verify Examples in the paper in PRISM Done

→ Verify Examples in the paper in STORM

→ Figure out how to control Rounding modes in C

→ Implement Algorithm until 15.07

→ Confirm Experiments until 01.08

→ Own potential ideas (more research needed first):

- When is controlled rounding is NOT needed ?
- Are there cases where controlled rounding is worse than normal ? (maybe higher run time in some cases)