PRISM: A Tool for Automatic Verification of Probabilistic Systems

Andrew Hinton, Marta Kwiatkowska,
Gethin Norman, Dave Parker

University of Birmingham

TACAS'06, March 2006

www.cs.bham.ac.uk/~dxp/prism

Overview

- Probabilistic model checking
- The PRISM tool
  - functionality + features
  - modelling language
  - property specification
  - brief demo
  - case studies
  - resources

Probabilistic Model Checking

- Formal methods for analysing probabilistic systems
  - randomisation, e.g. Bluetooth, FireWire, ...
  - uncertainty, e.g. message losses/delays, failures, ...
- Exhaustive exploration/analysis of probabilistic models
  - discrete/continuous-time Markov chains (DTMCs/CTMCs)
  - Markov decision processes (MDPs) (nondeterminism + prob.)
- Analysis of quantitative (or qualitative) properties
  - probabilistic extensions of temporal logic (e.g. PCTL)
  - graph-based algorithms + numerical computation

Probabilistic model checking

+ wide range of quantitative measures
+ exact answers computed
+ fully automatic process
  - model construction + numerical solution
– no counterexamples
  - but can identify patterns, trends, anomalies in quantitative results
+ exhaustive analysis, good for 'corner cases', e.g.
  - all possible initial configurations/model parameter values
  - all possible process schedulings (nondeterminism)
– state space explosion: time and memory constraints
+ efficient algorithms and implementations

PRISM: Probabilistic Model Checker

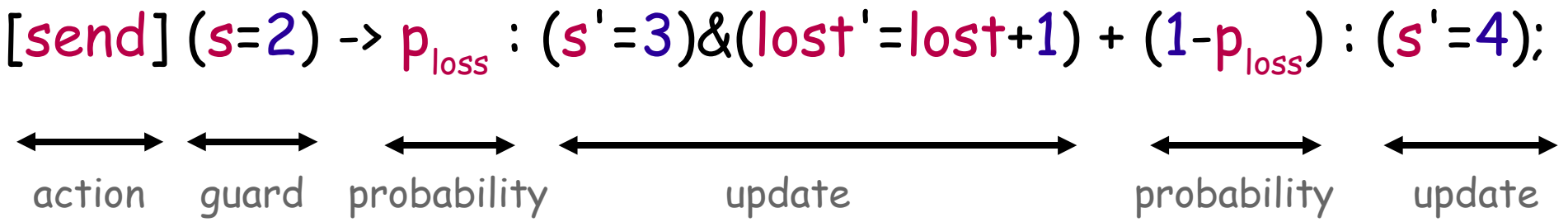
- Constructs three types of of probabilistic models:
  - DTMCs, CTMCs, MDPs
  - augmented with costs/rewards (new)
  - PRISM modelling language
- Variety of import/export functionality
  - model output: text files, Dot graphs, Matlab, ETMCC/MRMC
  - model import: text files (new)
  - other input formalisms via language translation
    - PEPA (now), CSP, PROBMELA, pi-calculus (in progress)
  - connections to other tools:
    - APMC, ProVer, Ymer, CASPA, KRONOS

PRISM: Probabilistic Model Checker

- Supports verification of:
  - PCTL, CSL + extensions
  - cost/reward-based properties (new)
- Multiple computation engines:
  - efficient symbolic (BDD-based) implementations
  - sampling-based computation (discrete-event simulation) (new)
    - distributed implementation under development
- Graphical user interface
  - model/property editor
  - easy automation + graphical visualisation of results
  - debugging tool: simulation engine (new)
    - manual/automatic generation of model traces

PRISM modelling language

- Simple, state-based language for DTMCs/CTMCs/MDPs
  - based on Reactive Modules [Alur/Henzinger]
- Modules (system components, composed in parallel)
- Variables (finite-valued, local or global)
- Guarded commands (labelled with probabilities/rates)
- Synchronisation (CSP-style) + process-algebraic operators (parallel composition, action hiding/renaming)



PRISM Language Example

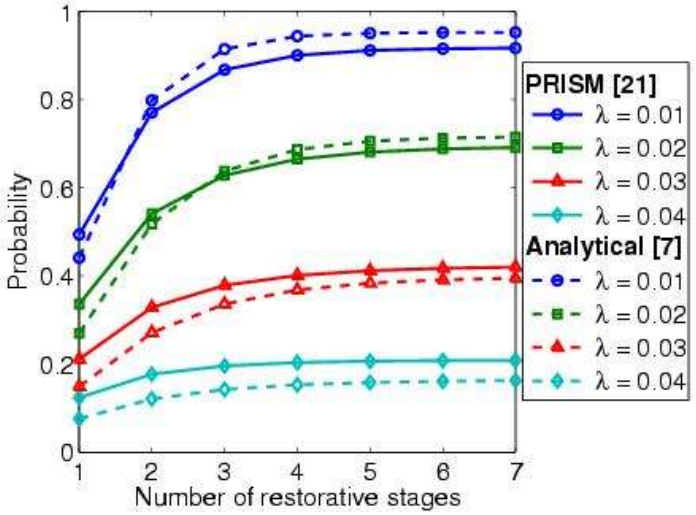
```
// hermans self-stabilisation algorithm [Her90]

dtmc // algorithm is synchronous

module process1 // first of N=5 symmetric processes

    x1 : [0..1];  // one bit per process; xi=x(i-1) means process i has a token

    [step] (x1=x5) -> 0.5 : (x1'=0) + 0.5 : (x1'=1);

    [step] !x1=x5 -> (x1'=x5);

endmodule

// add further processes through renaming
module process2 = process1[ x1=x2, x5=x1 ] endmodule
module process3 = process1[ x1=x3, x5=x2 ] endmodule
module process4 = process1[ x1=x4, x5=x3 ] endmodule
module process5 = process1[ x1=x5, x5=x4 ] endmodule

// can start in any possible configuration
init true endinit

// cost - 1 in each state (expected number of steps)
rewards true : 1; endrewards
```

PRISM – Property specifications

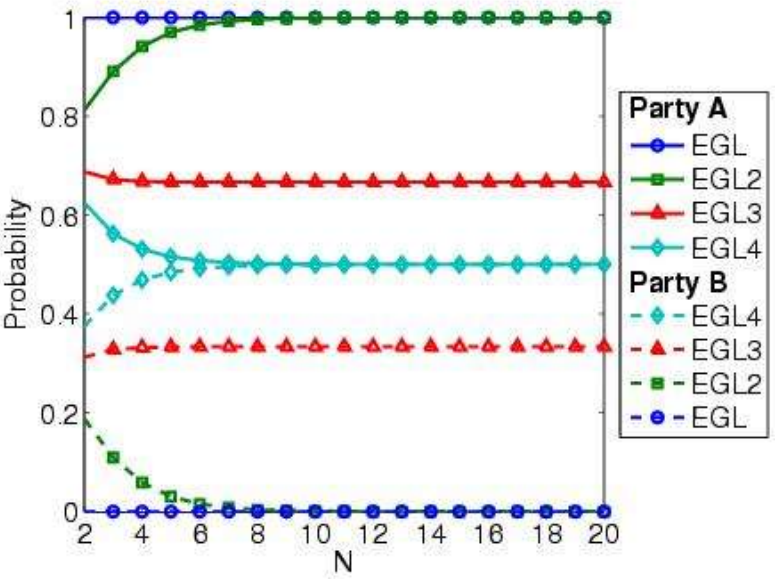
- Formulae in probabilistic extensions of temporal logic
  - PCTL for DTMCs/MDPs, CSL for CTMCs
- Examples:
  - $P<0.001$ [F shutdown] - "shutdown eventually occurs with probability at most 0.001"
  - $P<0.2$ [F[t,t] (deliv_rate < min)] "the probability that the current packet delivery rate has dropped below minimum at time t is less than 0.2"
  - $P\geq 0.95$ [!repair U$\leq$200 done] - "with probability 0.95 or greater, the process will successfully complete within 200 hours and without requiring any repairs"
  - $S>0.75$ [num_sensors $\geq$ min] - "in the long-run, the probability that an adequate number of sensors are operational is greater than 0.75"

PRISM – Property specifications

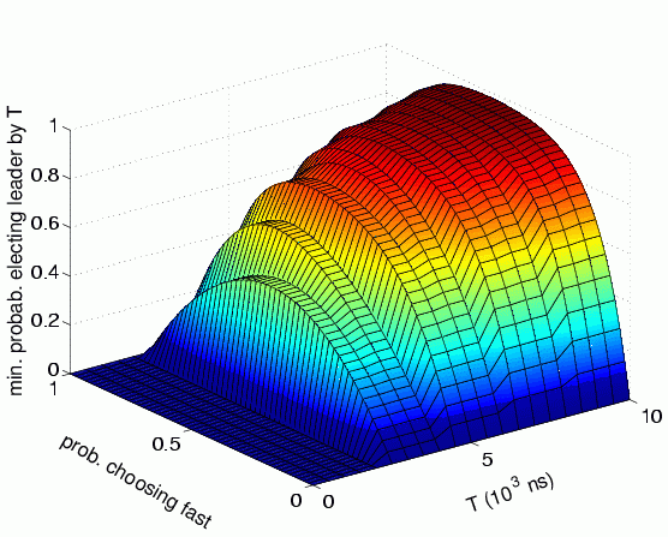
- Focus on quantitative properties, compute actual values
  - P=? [F≤T "shutdown"] - "what is the probability of shutdown occurring within T hours?"
  - S=? [num_sensors ≥ 4] - "what is the long-run probability that 4 or more sensors are operational?"
- Best/worst-case scenarios
  - P=? [F "error" {"init"}{max}] - "what is the worst-case error probability over all possible initial configurations?"
  - Pmin=? [!end2 U end1] - "what is the minimum probability of process 1 finishing before process 2 over all possible schedulings of the processes?"
- Experiments – ranges of model/property parameters
  - P=? [F≤T error] for N=1..5, T=1..100



Probability that 10% of gate outputs are erroneous for varying gate failure rates and numbers of stages



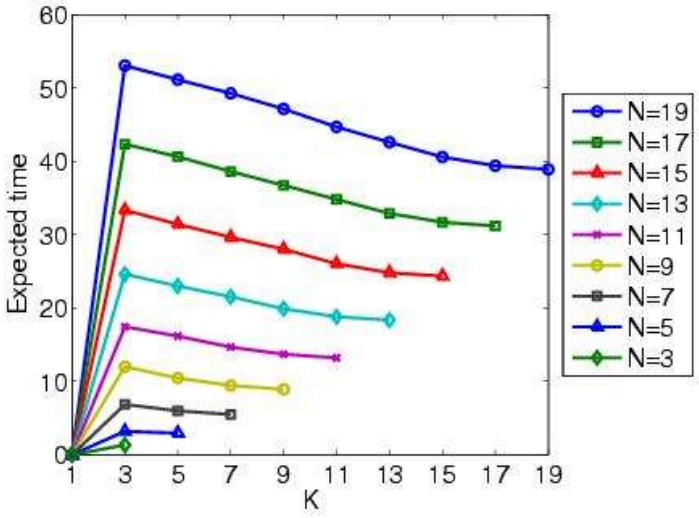
Probability that parties gain unfair advantage for varying numbers of secret packets sent



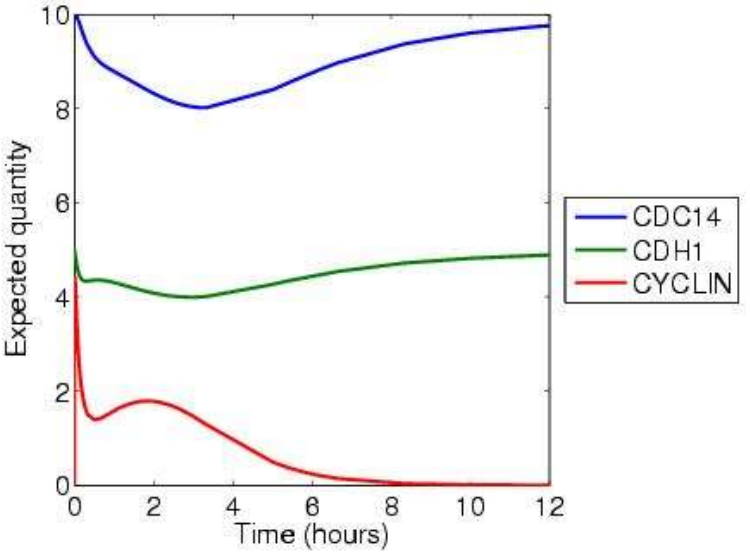
Optimum probability of leader election by time T for various coin biases

Cost- and reward-based properties

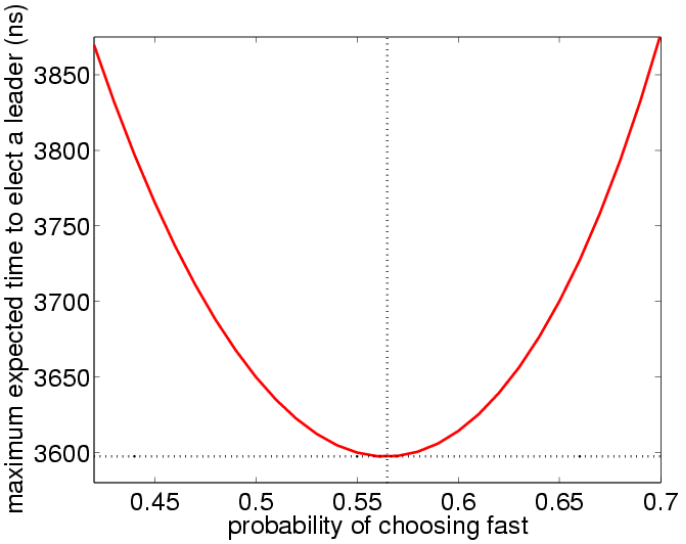
- Costs and rewards
  - real-valued quantities assigned to states/transitions
- Instantaneous – state-based measures
  - current queue size, number of operational channels, ...
  - "what is the expected size of the message queue at time t?"
  - "what is the long-run expected size of the queue?"
- Cumulative – state or transition (impulse) costs/rewards
  - time, power consumption, messages lost, ...
  - "what is the expected power consumption during the first 2 hours of operation?"
  - "what is the worst-case expected time taken for the protocol to terminate?"



Worst-case expected number of steps to stabilise for initial configurations with K tokens amongst N processes



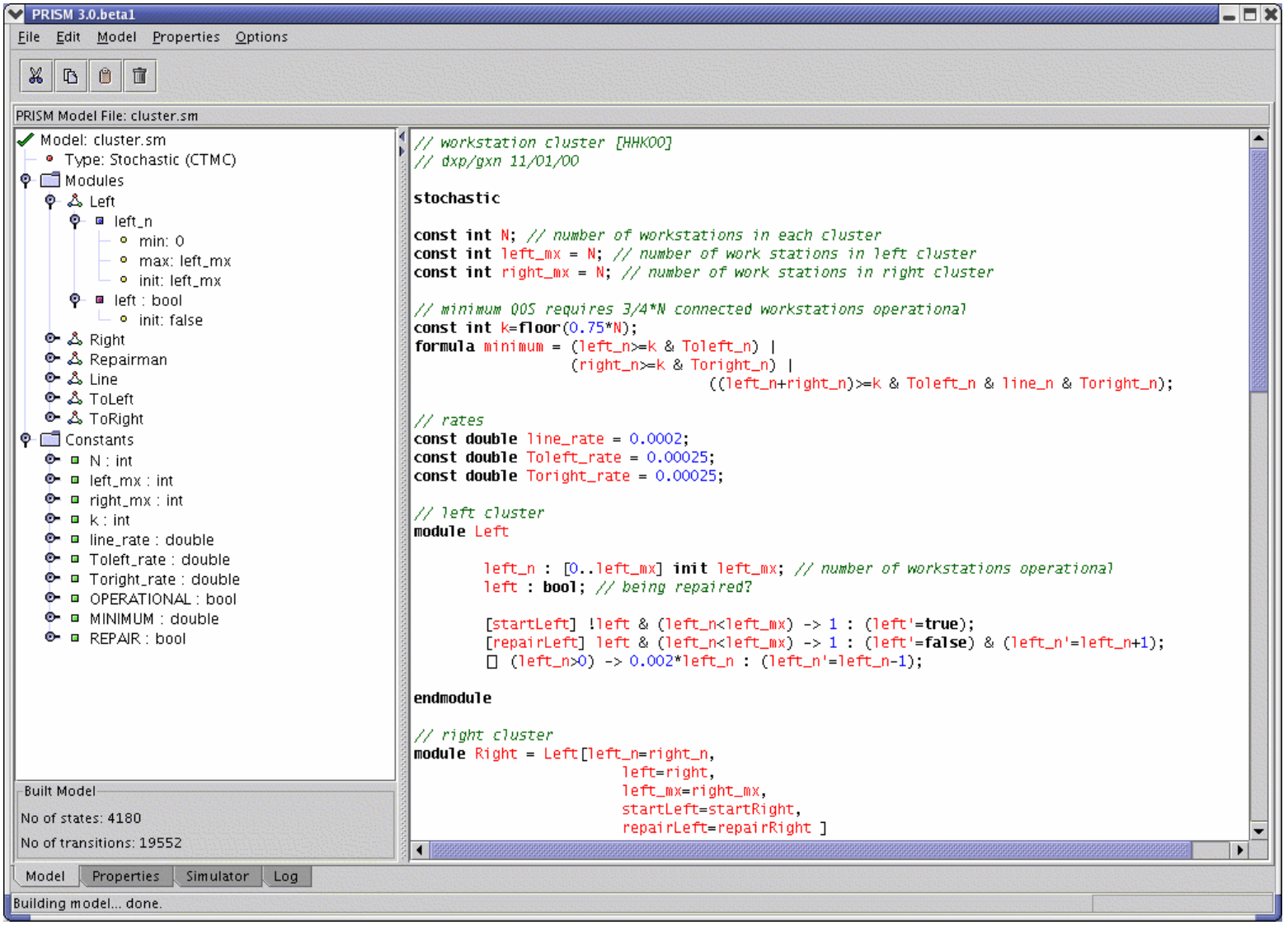
Expected reactant concentrations over the first 12 hours



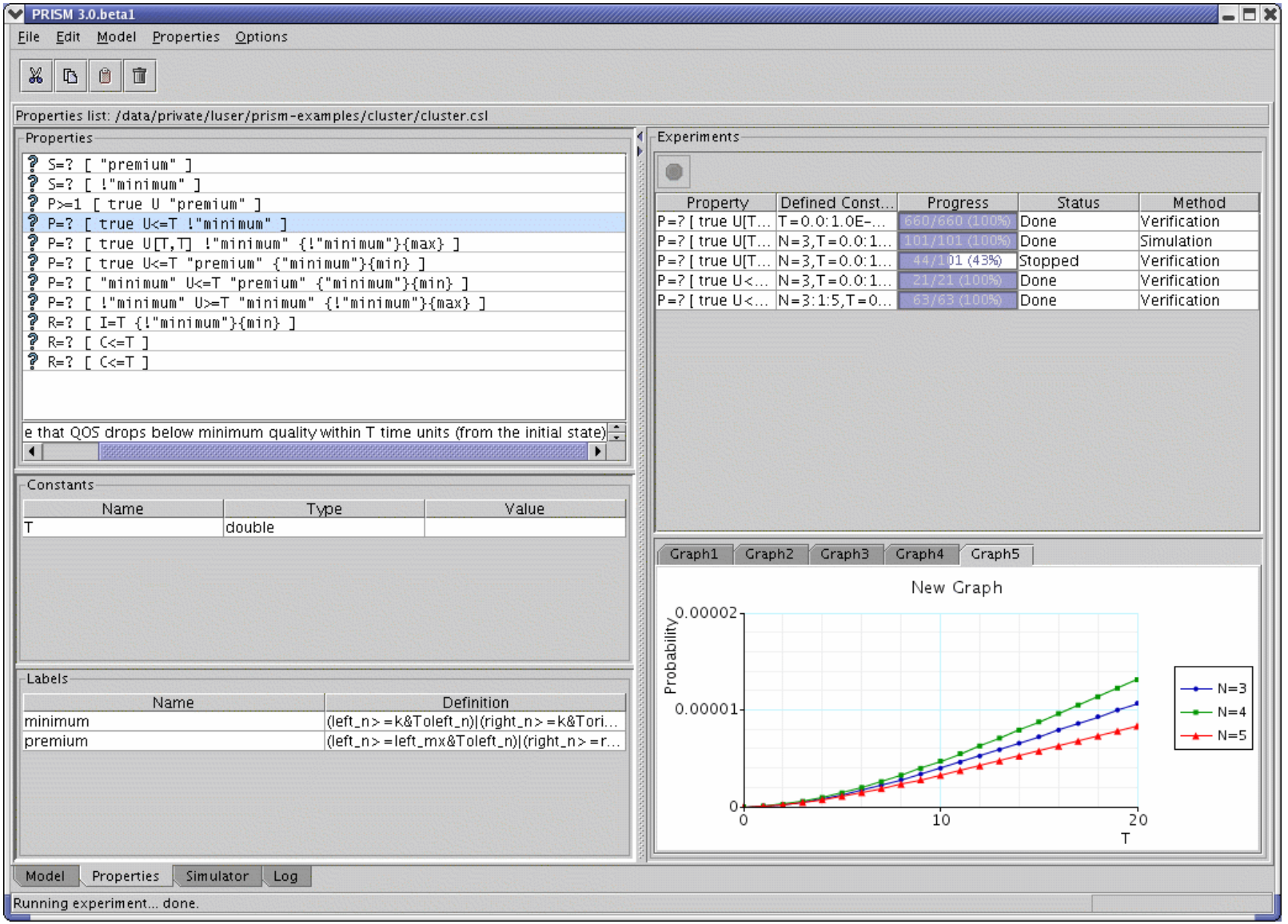
Maximum expected time for leader election for various coin biases

PRISM Demo

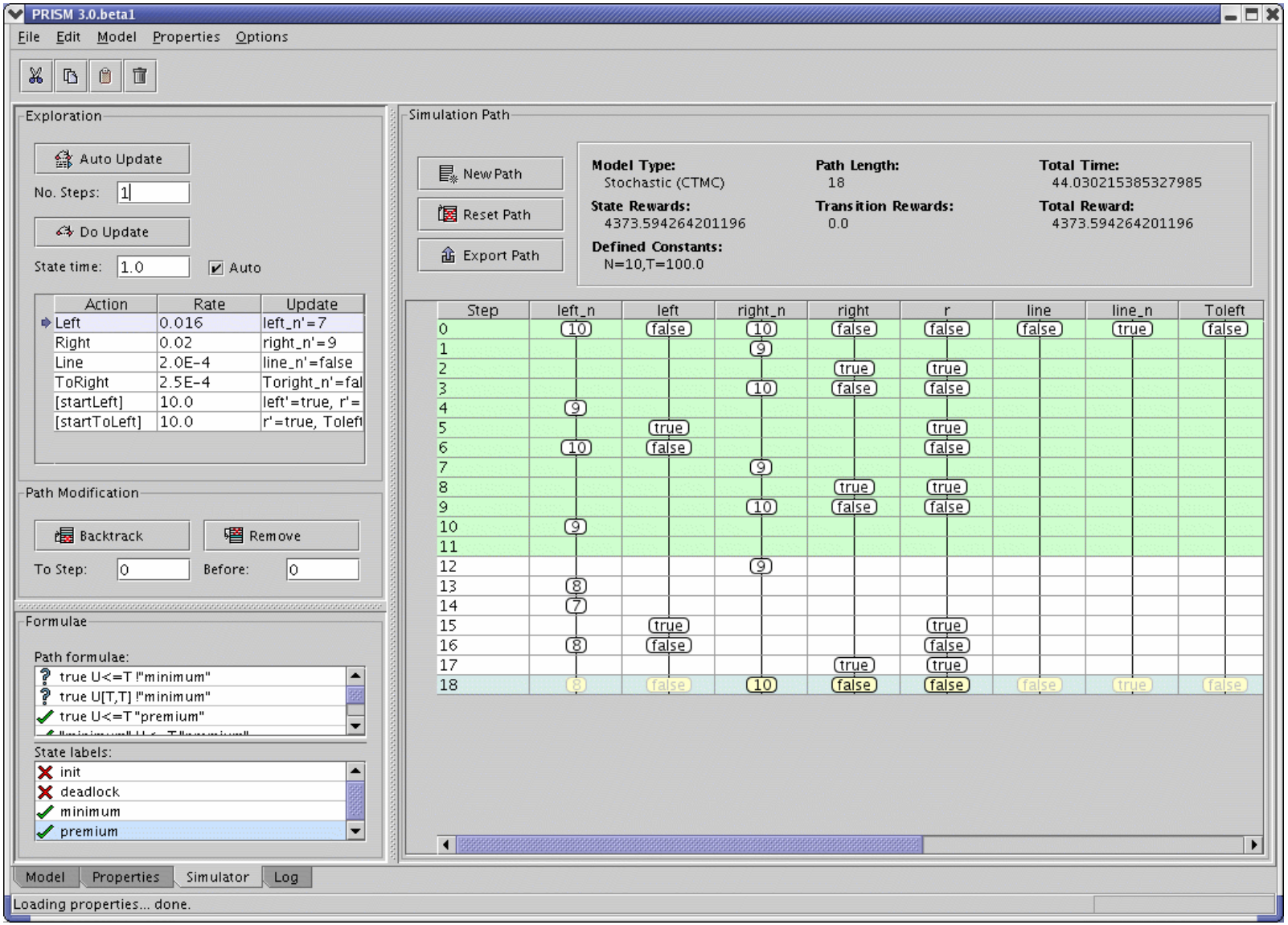
PRISM Screenshots



PRISM Screenshots



PRISM Screenshots



PRISM Case studies

- Wide range of application domains
  - communication and multimedia protocols – Bluetooth, FireWire, Zeroconf, CSMA/CD, WiFi, ...
  - security protocols – anonymity, contract signing, fair exchange, negotation frameworks, PIN cracking, ...
  - randomised distributed algorithms – self-stabilisation, consensus, leader election, mutual exclusion, ...
  - biological processes – signalling/cell cycle pathways, ..
  - performance/reliability of manufacturing systems, computer networks, NAND multiplexing, dynamic power management, embedded systems, quantum cryptography, thinkteam, ...

- See: www.cs.bham.ac.uk/~dxp/prism/casestudies

Getting PRISM + Other Resources

- Free and open source (GPL licensed)
  - Linux, Mac OS X (new), Windows, Unix
- Also available:
  - online example repository (~40 case studies)
  - documentation (wiki-based manual online soon)
  - support: help forum, bug tracking, feature requests
    - hosted on Sourceforge
  - related publications, links
- See: www.cs.bham.ac.uk/~dxp/prism