Continuous Stochastic Logic

Lecture #22 of Advanced Model Checking

Joost-Pieter Katoen

Lehrstuhl 2: Software Modeling & Verification

E-mail: `katoen@cs.rwth-aachen.de`

February 5, 2007

Exponential distribution

Continuous r.v. X is *exponential* with parameter $\lambda > 0$ if its density is

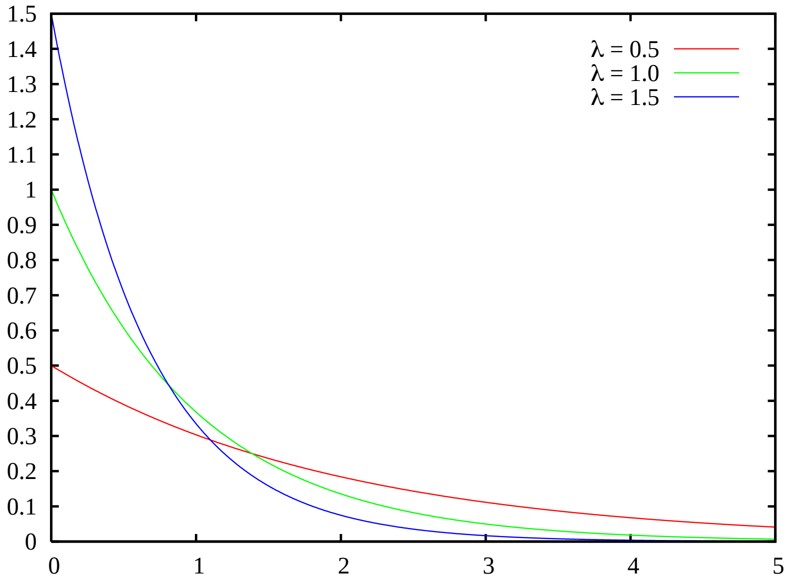
$$f(x) = \lambda{\cdot}e^{-\lambda \cdot x} \quad \text{for } x > 0 \quad \text{and } 0 \text{ otherwise}$$

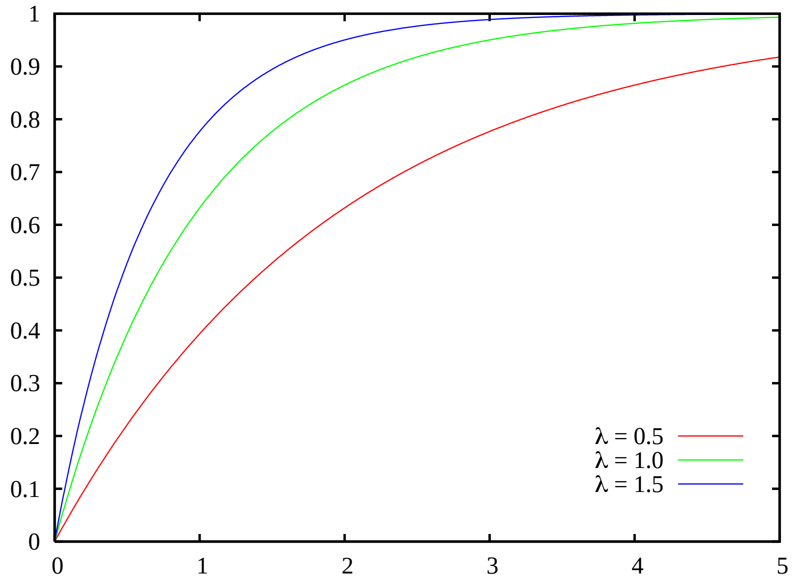
Cumulative distribution of X:

$$F_X(d) \;=\; \int_0^d \lambda{\cdot}e^{-\lambda\cdot x}\, dx \;=\; [-e^{-\lambda\cdot x}]_0^d \;=\; 1 - e^{-\lambda\cdot d}$$

- $\Pr\{X > d\} = e^{-\lambda\cdot d}$
- expectation $E[X] = \int_0^\infty x{\cdot}\lambda{\cdot}e^{-\lambda\cdot x}\, dx = \frac{1}{\lambda}$
- variance *Var*$[X] = \frac{1}{\lambda^2}$

Exponential pdf and cdf





the higher λ, the faster the cdf approaches 1

Exponential distributions

- have *nice mathematical* properties (cf. next slide)
- are *adequate* for many real-life phenomena
  - describes the time for a continuous process to change state
  - the time until you have your next car accident (failure rates)
  - the inter-arrival times (i.e., the times between customers entering a shop)
- combinations can *approximate* general distributions arbitrarily closely
- maximal *entropy* probability distribution if just the mean is known

CTMCs

A *continuous-time Markov chain* (CTMC) is a tuple $(S, \mathbf{R}, L)$ where:

- S is a finite set of states and L the state-labelling (as before)
- $\mathbf{R} : S \times S \rightarrow \mathbb{R}_{\geqslant 0}$, a *rate matrix*
  - $\mathbf{R}(s, s') = \lambda$ means that the average speed of going from s to s' is $\frac{1}{\lambda}$
- $E(s) = \sum_{s' \in S} \mathbf{R}(s, s') = \mathbf{R}(s, S)$ is the *exit rate* of state s
  - s is called absorbing whenever $E(s) = 0$

$\Rightarrow$ a CTMC is a Kripke structure with probabilistically timed transitions

Interpretation

- The probability that transition $s \longrightarrow s'$ is *enabled* in $[0, t]$:

$$1 - e^{-\mathbf{R}(s,s') \cdot t}$$

- The probability to *move* from non-absorbing s to s' in $[0, t]$ is:

$$\frac{\mathbf{R}(s, s')}{E(s)} \cdot \left(1 - e^{-E(s) \cdot t}\right)$$

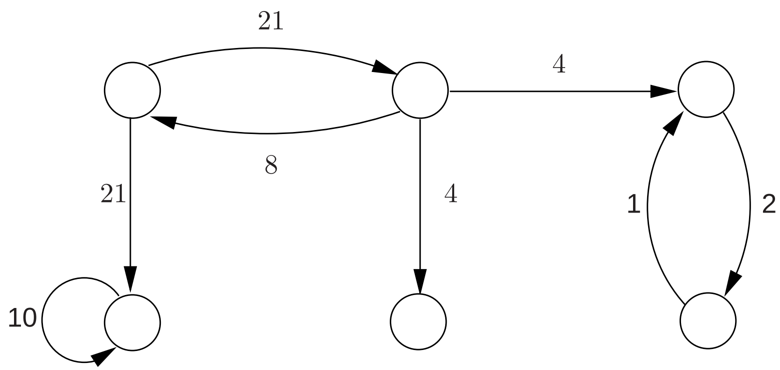
- The probability to take an outgoing transition from s within $[0, t]$ is:

$$1 - e^{-E(s) \cdot t}$$

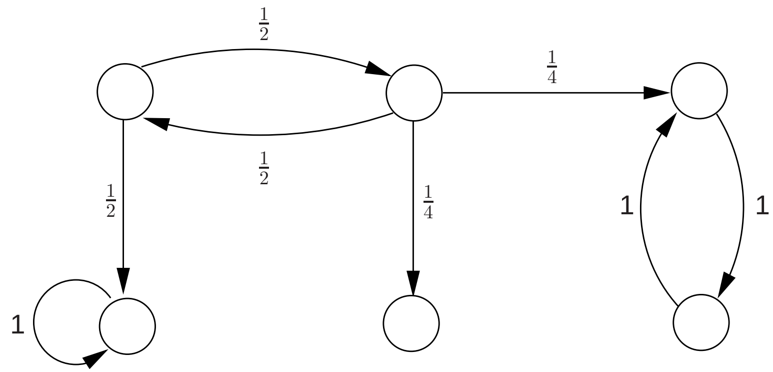
Embedded DTMC

The *embedded* DTMC of the CTMC $(S, \mathbf{R})$ is $(S, \mathbf{P})$ where

$$\mathbf{P}(s, s') = \begin{cases} \frac{\mathbf{R}(s,s')}{E(s)} & \text{if } E(s) > 0 \\ 0 & \text{otherwise} \end{cases}$$



a CTMC



iits embedded DTMC

Elementary probabilities for CTMCs

- *Transient* probability vector $\underline{\pi}(t) = (\cdots, \pi_i(t), \cdots)$ for $t \geqslant 0$
  - where $\pi_i(t)$ is the probability to be in state s_i after t time units (given $\underline{\pi}(0)$)
  - $\underline{\pi}(t)$ is computed by solving a linear differential equations

$$\underline{\pi}'(t) \;=\; \underline{\pi}(t) \cdot \mathbf{Q} \quad \text{given} \quad \underline{\pi}(0) \quad \text{where} \quad \mathbf{Q} = \mathbf{R} - \mathit{diag}(E)$$

- *Steady-state* probability vector $\underline{\pi} = (\cdots, \pi_i, \cdots)$
  - π_i is mostly *in*dependent from the starting distribution
  - $\underline{\pi}$ is computed from a system of linear equations:

$$\underline{\pi} \cdot \mathbf{Q} \;=\; 0 \quad \text{where} \quad \sum_i \pi_i = 1$$

Continuous Stochastic Logic

State-formulas $\Phi ::= a \mid \neg \Phi \mid \Phi \vee \Phi \mid \mathbb{S}_{\trianglelefteq p}(\Phi) \mid \mathbb{P}_{\trianglelefteq p}(\varphi)$

with probability p and comparison operator $\trianglelefteq$

$\mathbb{S}_{\trianglelefteq p}(\Phi)$ probability that Φ holds in steady state is $\trianglelefteq p$

$\mathbb{P}_{\trianglelefteq p}(\varphi)$ probability that paths fulfill φ is $\trianglelefteq p$

Path-formulas $\varphi ::= \bigcirc^I \Phi \mid \Phi \, \mathsf{U}^I \, \Phi$ with interval I

$\bigcirc^I \Phi$ next state is reached at time $t \in I$ and fulfills Φ

$\Phi \, \mathsf{U}^I \, \Psi$ Φ holds along the path until Ψ holds at time $t \in I$

CTL operators $\bigcirc$ and U are special cases

Example properties

- In $\geqslant$ 92% of the cases, a goal state is legally reached within 3.1 sec:

$$\mathcal{P}_{\geqslant 0.92}\left(\neg\, \textit{illegal}\ \ \mathsf{U}^{\leqslant 3.1}\ \textit{goal}\right)$$

- ... a state is soon reached guaranteeing 0.9999 long-run availability:

$$\mathcal{P}_{\geqslant 0.92}\left(\neg\, \textit{illegal}\ \ \mathsf{U}^{\leqslant 0.7}\ \mathcal{S}_{\geqslant 0.9999}\left(\textit{goal}\right)\right)$$

- On the long run, illegal states can (almost surely) not be reached in the next 7.2 time units:

$$\mathcal{S}_{\geqslant 0.9999}\left(\mathcal{P}_{\geqslant 1}\left(\Box^{\leqslant 7.2} \neg\, \textit{illegal}\right)\right)$$

Semantics of CSL: state-formulas

$\mathcal{C}, s \models \Phi$ if and only if formula Φ holds in state s of CTMC $\mathcal{C}$

Relation $\models$ is defined by:

$s \models a$ iff $a \in L(s)$

$s \models \neg\,\Phi$ iff not $(s \models \Phi)$

$s \models \Phi \,\vee\, \Psi$ iff $(s \models \Phi)$ or $(s \models \Psi)$

$s \models \mathbb{S}_{\unlhd p}(\Phi)$ iff $\lim_{t\to\infty} \Pr\{\, \sigma \in \textit{Paths}(s) \mid \sigma@t \models \Phi \,\} \unlhd p$

$s \models \mathbb{P}_{\unlhd p}(\varphi)$ iff $\Pr\{\, \sigma \in \textit{Paths}(s) \mid \sigma \models \varphi \,\} \unlhd p$

$\Pr\{\ldots\}$ is measurable by a (i.e., cone) Borel space construction on paths in a CTMC

Semantics of CSL: path-formulas

A *path* in CTMC $\mathcal{C}$ is an infinite alternating sequence

$$s_0\, t_0\, s_1\, t_1 \ldots \quad \text{with} \quad \mathbf{R}(s_i, s_{i+1}) > 0 \quad \text{and} \quad t_i > 0$$

non time-divergent paths have probability zero

Semantics of path-formulas is defined by:

$$\sigma \models \bigcirc^I \Phi \qquad \text{iff } \sigma[1] \models \Phi \text{ and } t_0 \in I$$

$$\sigma \models \Phi \,\mathsf{U}^I\, \Psi \qquad \text{iff } \exists t \in I.\ ((\forall t' \in [0, t).\ \sigma@t' \models \Phi) \wedge \sigma@t \models \Psi)$$

where $\sigma@t$ denotes the state in the path σ at time t

Model-checking CSL

- Check which states in a CTMC satisfy a CSL formula:
  - compute recursively the set $Sat(\Phi)$ of states that satisfy Φ

  $\Rightarrow$ recursive descent computation over the parse tree of Φ
- For the non-stochastic part: as for CTL
- For all probabilistic formulae not involving a time bound: as for PCTL
  - using the *embedded DTMC*
- How to compute $Sat(\Phi)$ for the stochastic timed operators?

Model-checking the steady-state operator

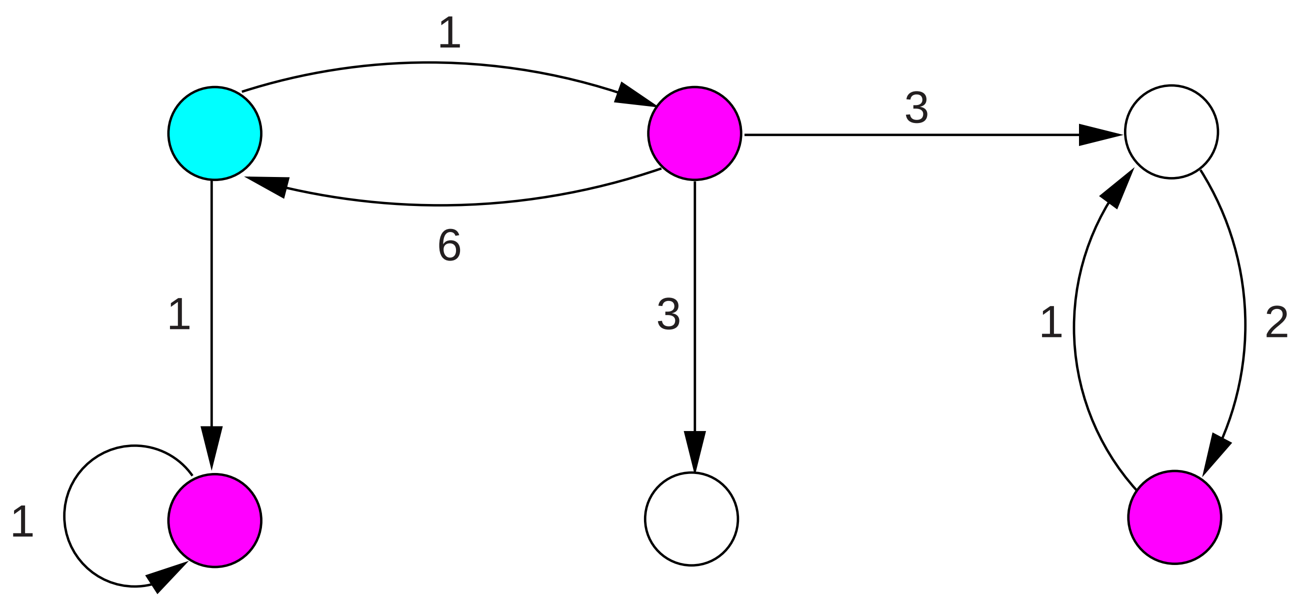
- For an ergodic (i.e., strongly-connected) CTMC:

$$s \in \mathit{Sat}(\mathbb{S}_{\trianglelefteq p}(\Phi)) \text{ iff } \sum_{s' \in \mathit{Sat}(\Phi)} \pi_{s'} \trianglelefteq p$$

$\Longrightarrow$ this boils down to a standard steady-state analysis

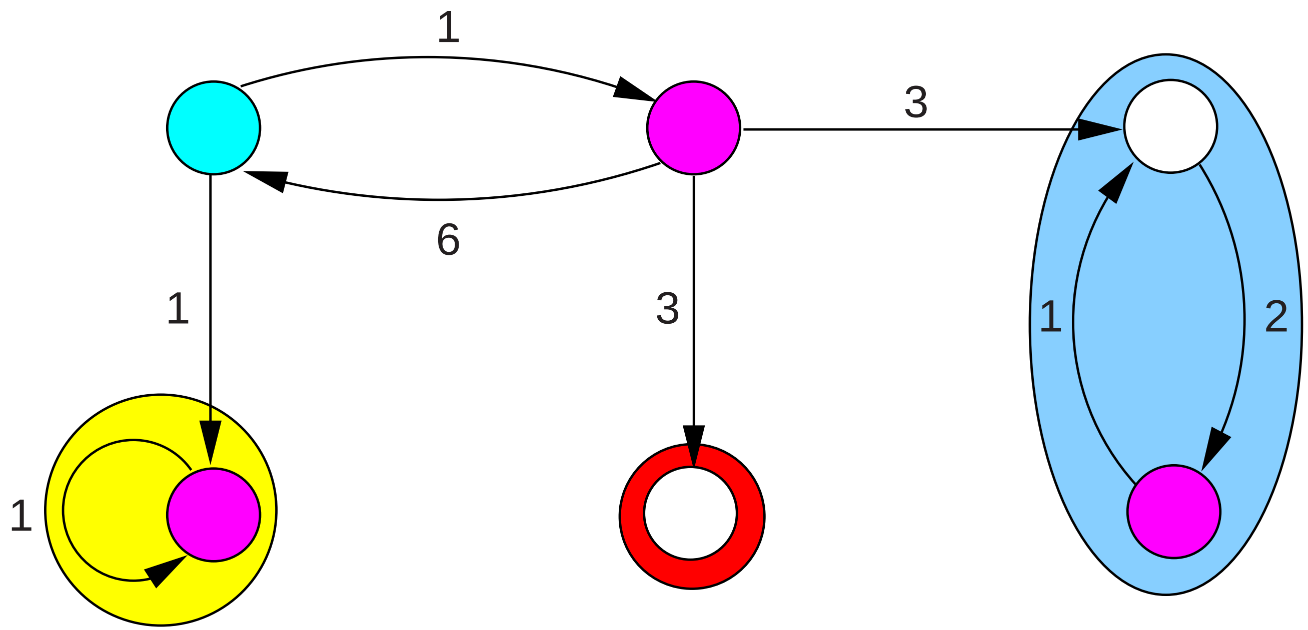
- For an arbitrary CTMC:
  - determine the *bottom* strongly-connected components (BSCCs)
  - for BSCC B determine the steady-state probability of a Φ-state
  - compute the probability to reach BSCC B from state s
  - check whether $\displaystyle\sum_{B} \left(\Pr\{\text{ reach } B \text{ from } s \,\} \cdot \sum_{s' \in B \cap \mathit{Sat}(\Phi)} \pi^{B}_{s'} \right) \trianglelefteq p$

Verifying steady-state properties: an example



determine the bottom strongly-connected components

Verifying steady-state properties: an example



$$s \models \mathbb{S}_{>0.75}(\textit{magenta}) \quad \text{iff} \quad Prob(s, \Diamond at_{yellow})\cdot\pi^{yellow}(\textit{magenta})$$
$$+ Prob(s, \Diamond at_{blue})\cdot\pi^{blue}(\textit{magenta}) > 0.75$$

Checking time-bounded reachability

- $s \models \mathbb{P}_{\trianglelefteq p}(\Phi \, \mathsf{U}^{\leqslant t} \, \Psi)$ if and only if $Prob(s, \Phi \, \mathsf{U}^{\leqslant t} \, \Psi) \trianglelefteq p$
- $Prob(s, \Phi \, \mathsf{U}^{\leqslant t} \Psi)$ is the least solution of: (Baier, Katoen & Hermanns, 1999)
  - 1 if $s \models \Psi$
  - if $s \models \Phi \wedge \neg \Psi$:

$$\int_0^t \sum_{s' \in S} \underbrace{\mathbf{P}(s, s') \cdot E(s) \cdot e^{-\mathbf{E}(s) \cdot x}}_{\text{probability to move to state } s' \text{ at time } x} \cdot \underbrace{Prob(s', \Phi \, \mathsf{U}^{\leqslant t-x} \, \Psi)}_{\text{probability to fulfill } \Phi \, \mathsf{U} \, \Psi \text{ before time } t-x \text{ from } s'} \, dx$$

  - 0 otherwise

Reduction to transient analysis

(Baier, Haverkort, Hermanns & Katoen, 2000)

- Make all Ψ- and all $\neg(\Phi \vee \Psi)$-states absorbing in $\mathcal{C}$
- Check $\Diamond^{=t}\,\Psi$ in the obtained CTMC $\mathcal{C}'$
- This is a standard transient analysis in $\mathcal{C}'$:

$$\sum_{s' \models \Psi} \Pr\{\sigma \in \mathit{Paths}(s) \mid \sigma@t = s'\}$$

  - compute by solving linear differential equations, or discretization

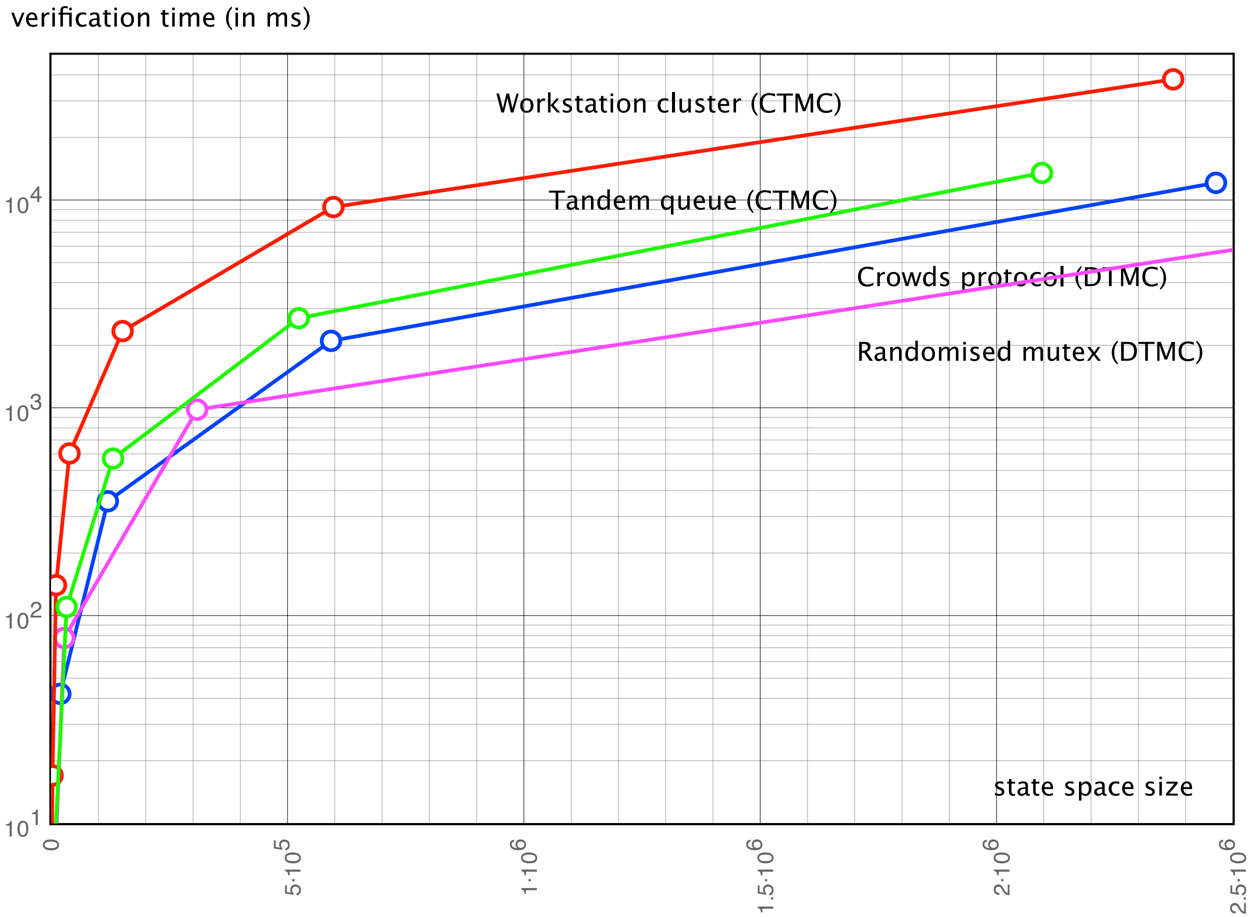
$\Rightarrow$ Discretization + matrix-vector multiplication + Poisson probabilities

Markov reward model checker (MRMC)

(Zapreev & Meyer-Kayser, 2000/2005)

- Supports DTMCs, CTMCs and cost-based extensions thereof
  - temporal logics: P(R)CTL and CS(R)L
  - bounded until, long run properties, and interval bounded until
- Sparse-matrix representation
- Command-line tool (in C)
  - experimental platform for new (e.g., reward) techniques
  - back-end of GreatSPN, PEPA WB, PRISM and stochastic GG tool
  - freely downloadable under Gnu GPL license
- Experiments: Pentium 4, 2.66 GHz, 1 GB RAM

Verification times



Probabilistic bisimulation

- Let $\mathcal{D} = (S, \mathbf{P}, L)$ be a DTMC and R an equivalence relation on S
- R is a *probabilistic bisimulation* on S if for any $(s, s') \in R$:

$$L(s) = L(s') \text{ and } \mathbf{P}(s, C) = \mathbf{P}(s', C) \quad \text{for all} \quad C \text{ in } S/R$$

where $\mathbf{P}(s, C) = \sum_{s' \in C} \mathbf{P}(s, s')$ (Larsen & Shou, 1989)

- $s \sim s'$ if $\exists$ a probabilistic bisimulation R on S with $(s, s') \in R$

$$s \sim s' \Leftrightarrow (\forall \Phi \in \textit{PCTL} : s \models \Phi \text{ if and only if } s' \models \Phi)$$

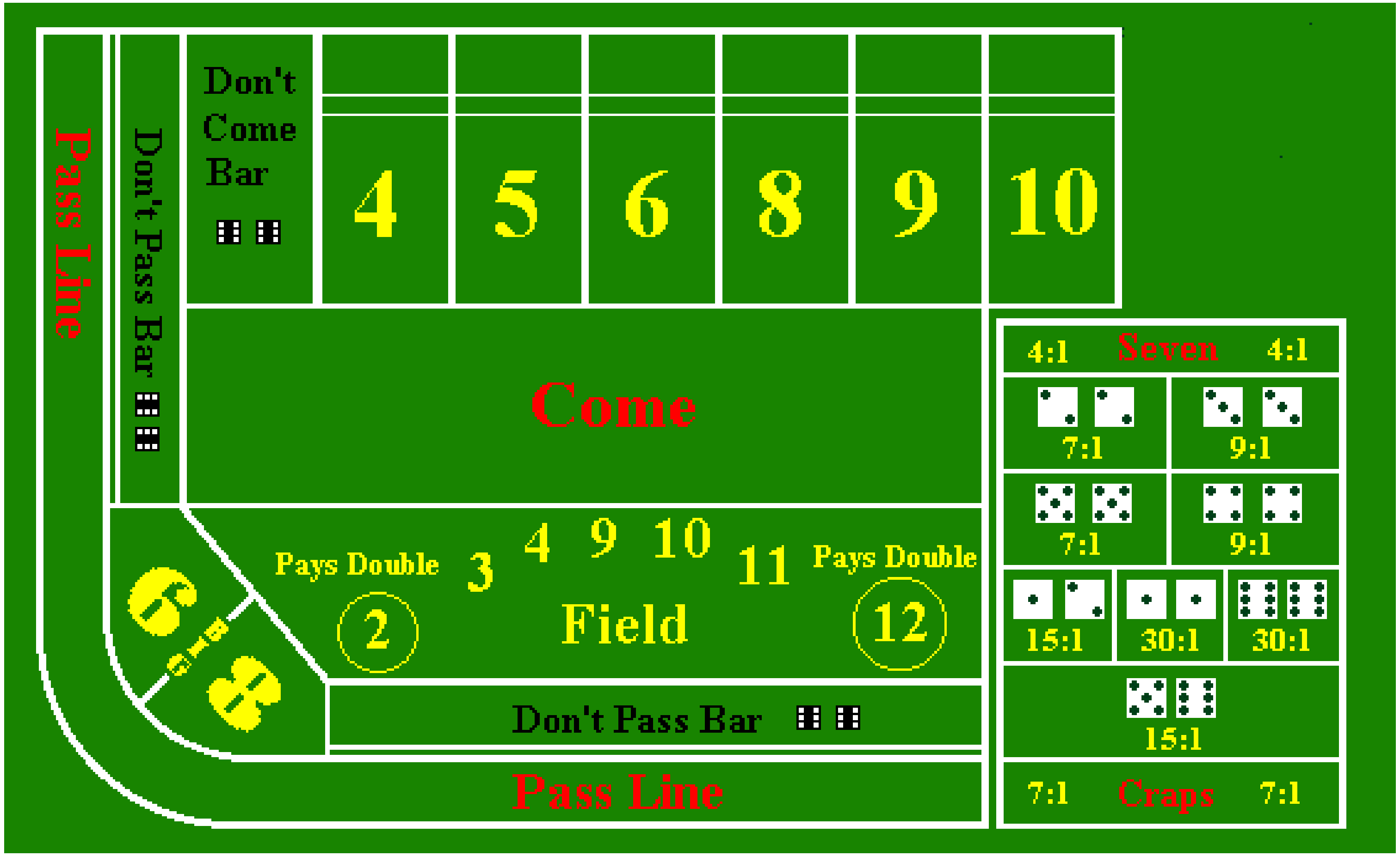
Quotient DTMC under $\sim$

$$\mathcal{D}/\sim \; = \; (S', \mathbf{P}', L'),$$ the quotient of $\mathcal{D} = (S, \mathbf{P}, L)$ under $\sim$:

- $S' = S/\sim = \; \{ \, [s]_\sim \mid s \in S \, \}$
- $\mathbf{P}'([s]_\sim, C) = \mathbf{P}(s, C)$
- $L'([s]_\sim) = L(s)$

get $\mathcal{D}/\sim$ by partition-refinement in time $\mathcal{O}(M \cdot \log N + |AP| \cdot N)$ (Derisavi et al., 2001)

Craps



Pass Line
Don't Pass Bar
Don't
Come
Bar
4
5
6
8
9
10
Come
4:1
Seven
4:1
7:1
9:1
7:1
9:1
15:1
30:1
30:1
15:1
7:1
Craps
7:1
Pays Double
3
4
9
10
11
Pays Double
2
Field
12
6
8
Don't Pass Bar
Pass Line

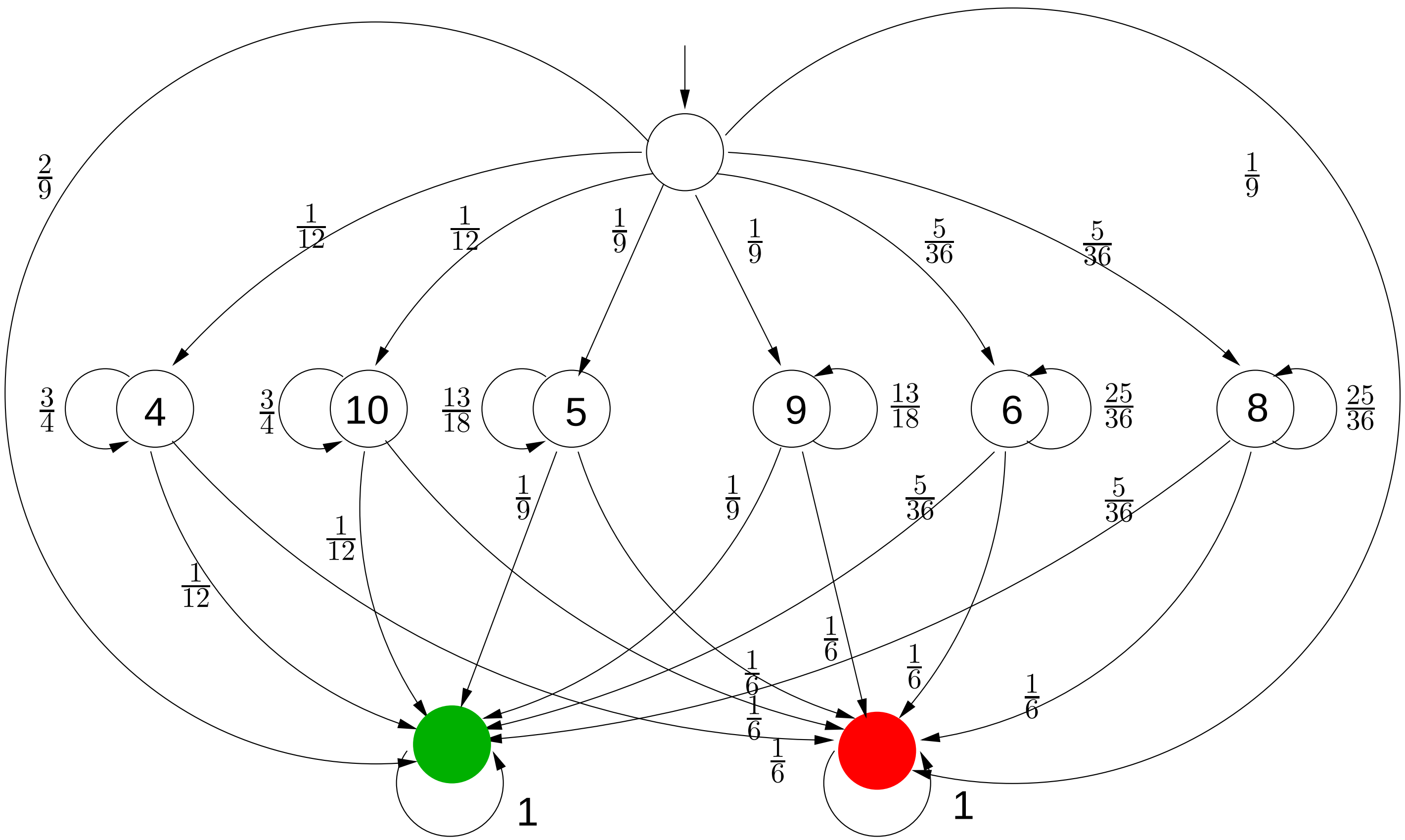
Craps

- Roll two dice and bet on outcome

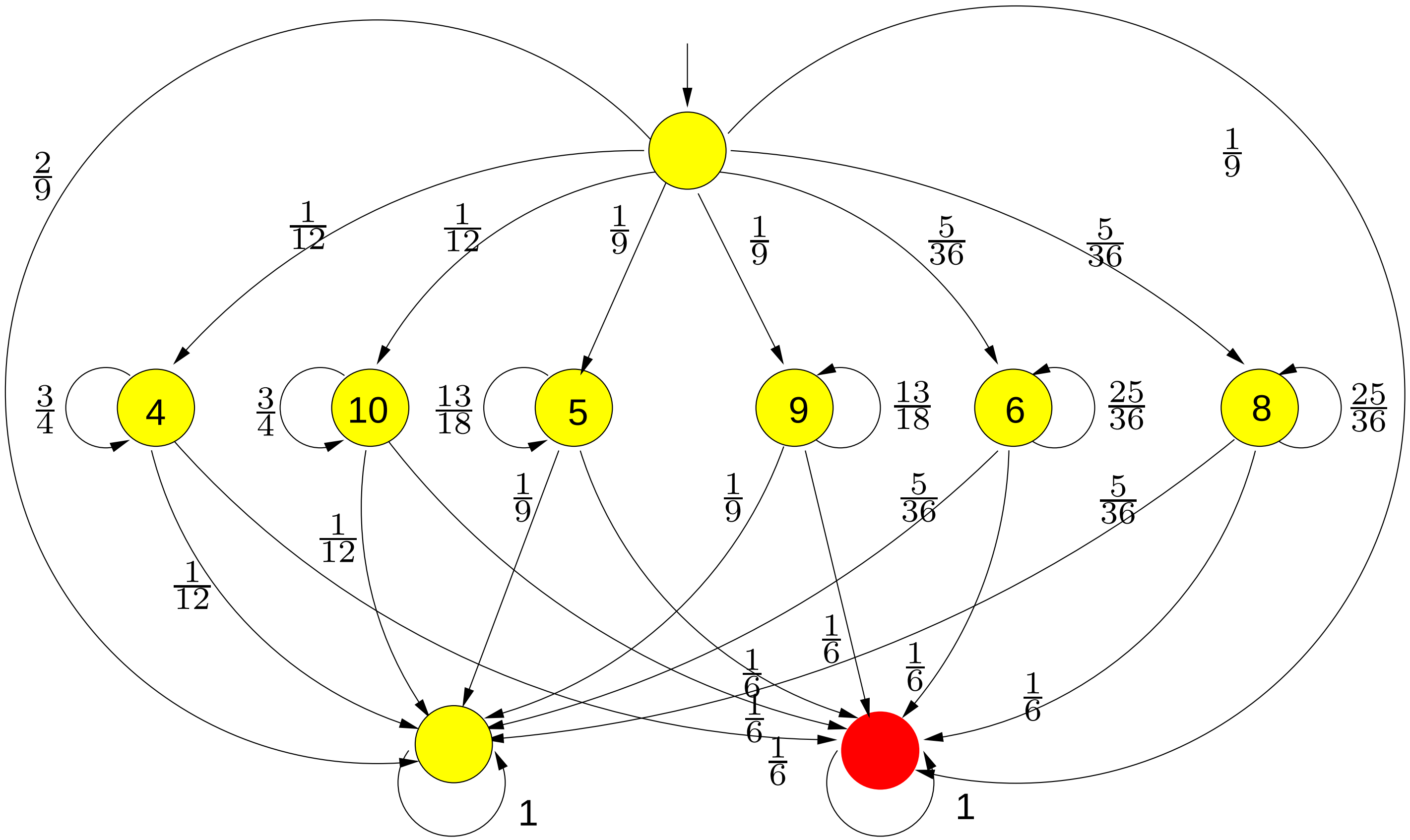
- Come-out roll ("pass line" wager):
  - outcome 7 or 11: win
  - outcome 2, 3, and 12: loss ("craps")
  - any other outcome: roll again (outcome is "point")

- Repeat until 7 or the "point" is thrown:
  - outcome 7: loss ("seven-out")
  - outcome the point: win
  - any other outcome: roll again

A DTMC model of Craps

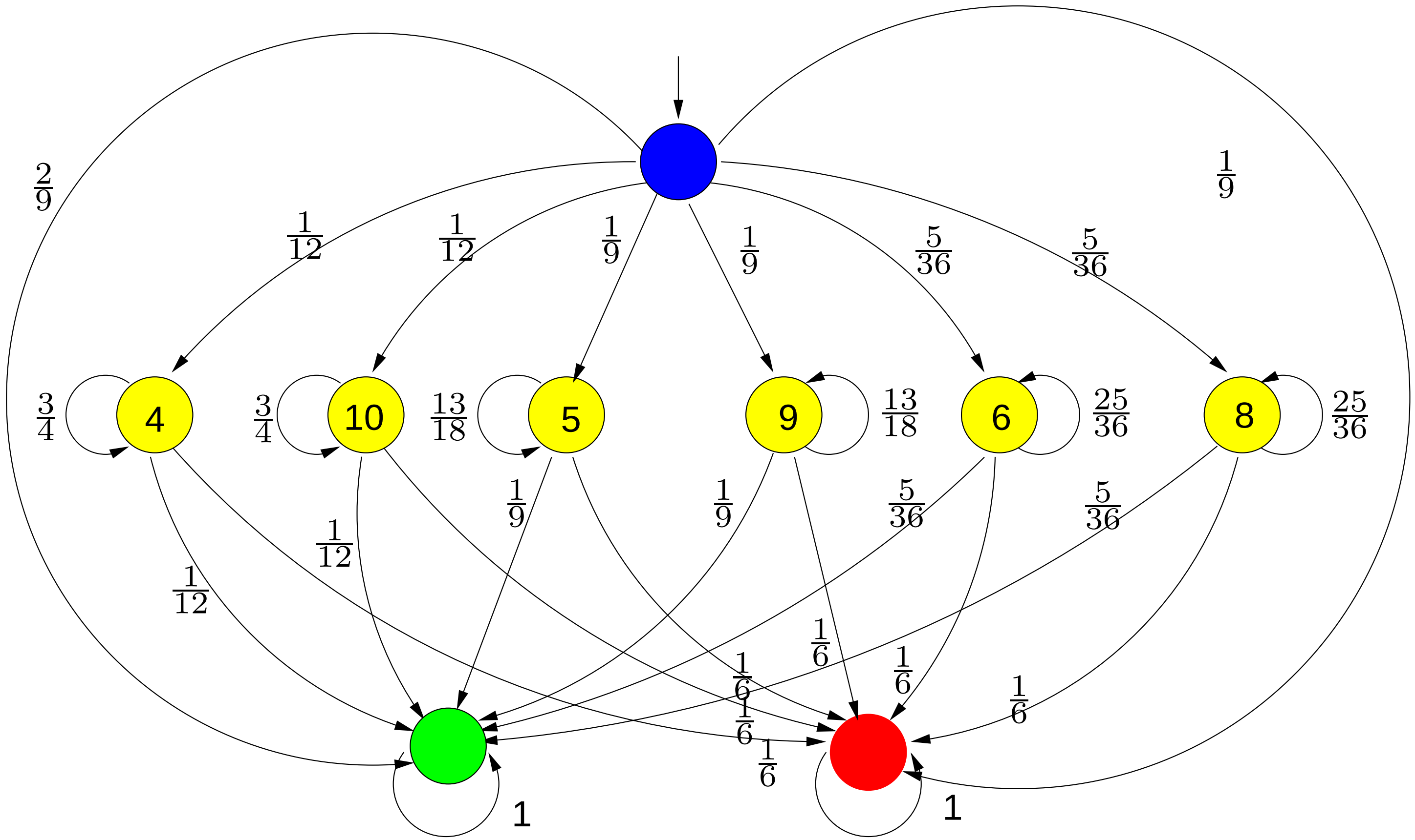


Minimizing Craps



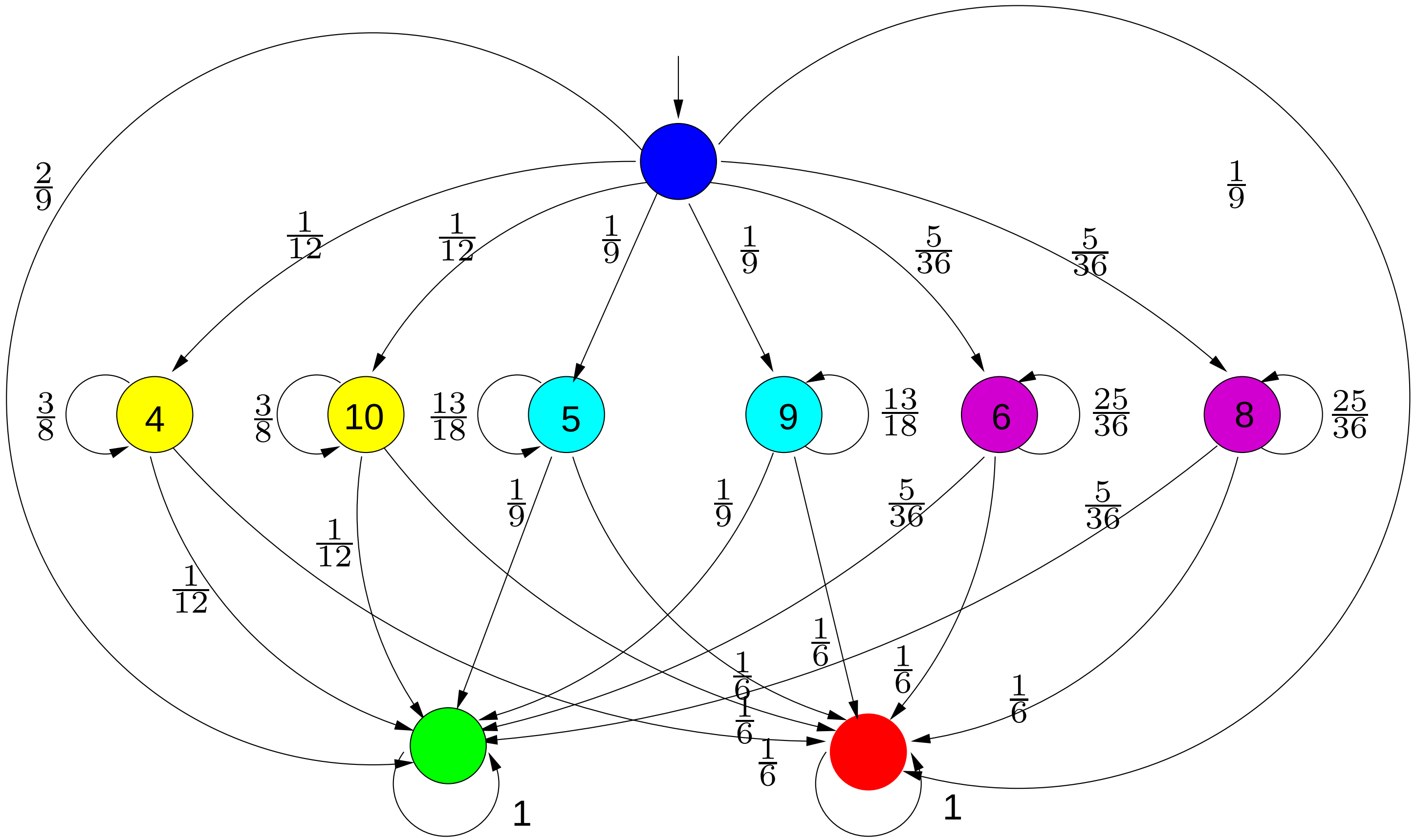
initial partitioning for the atomic propositions $AP = \{ \mathit{loss} \}$

A first refinement



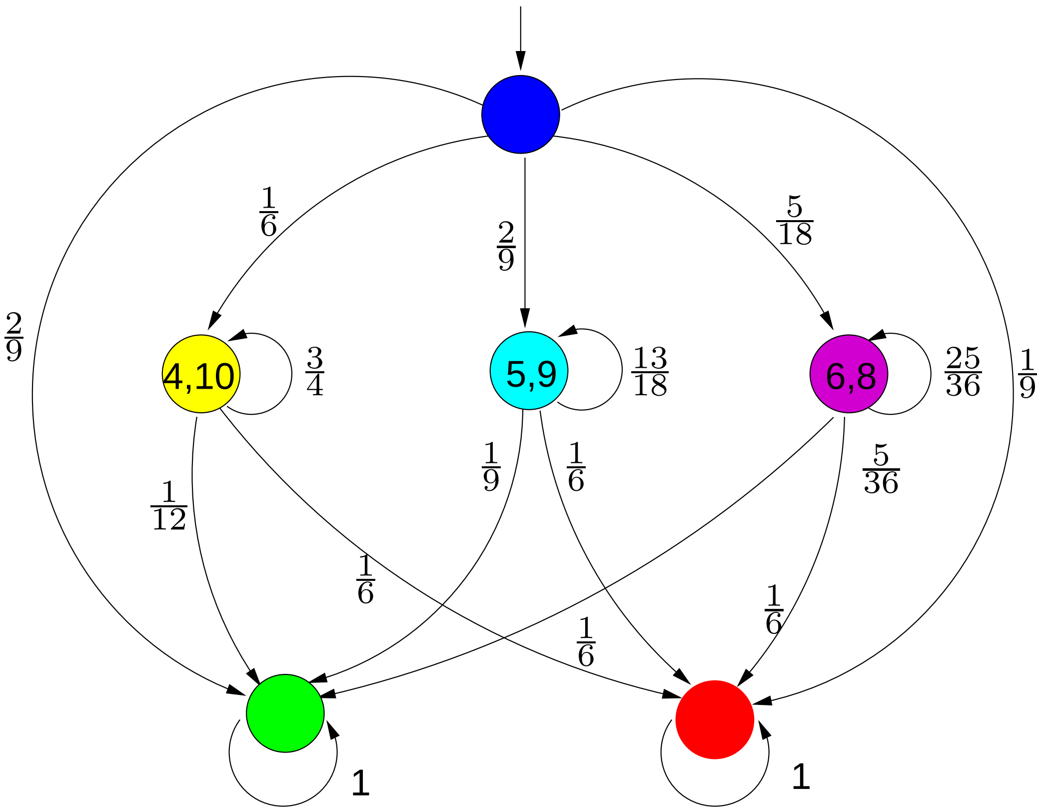
refine ("split") with respect to the set of red states

A second refinement



refine (“split”) with respect to the set of green states

Quotient DTMC



Property-driven bisimulation

- For DTMC $\mathcal{D}$, set F of PCTL-formulas, and equivalence R on S
- R is a probabilistic F-bisimulation on S if for any $(s, s') \in R$:

$$L_F(s) = L_F(s') \text{ and } \mathbf{P}(s, C) = \mathbf{P}(s', C) \quad \text{for all} \quad C \text{ in } S/R$$

where $L_F(s) = \{ \Phi \in F \mid s \models \Phi \}$ (Baier et al., 2000)

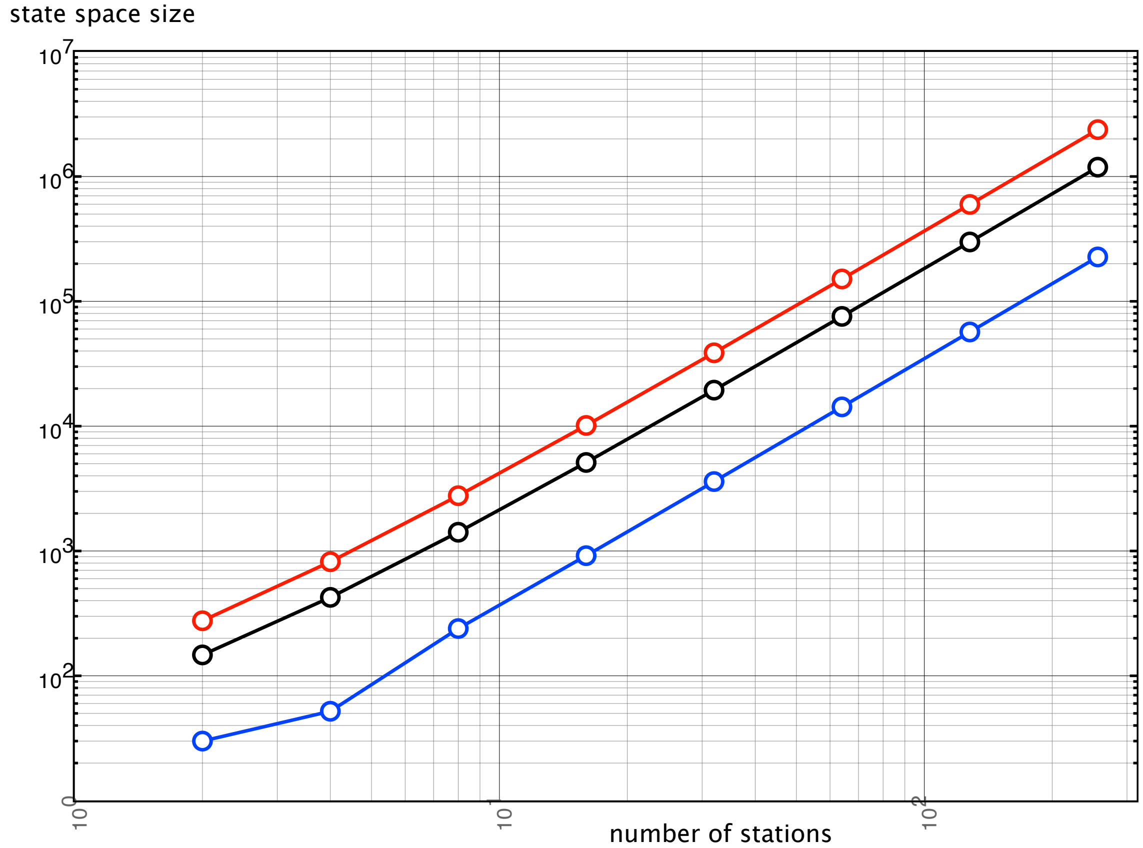
- $s \sim_F s'$ if $\exists$ a probabilistic F-bisimulation R on S with $(s, s') \in R$

$$s \sim_F s' \;\Leftrightarrow\; (\forall \Phi \in \mathit{PCTL}_F : s \models \Phi \text{ if and only if } s' \models \Phi)$$

Minimization for Φ until Ψ

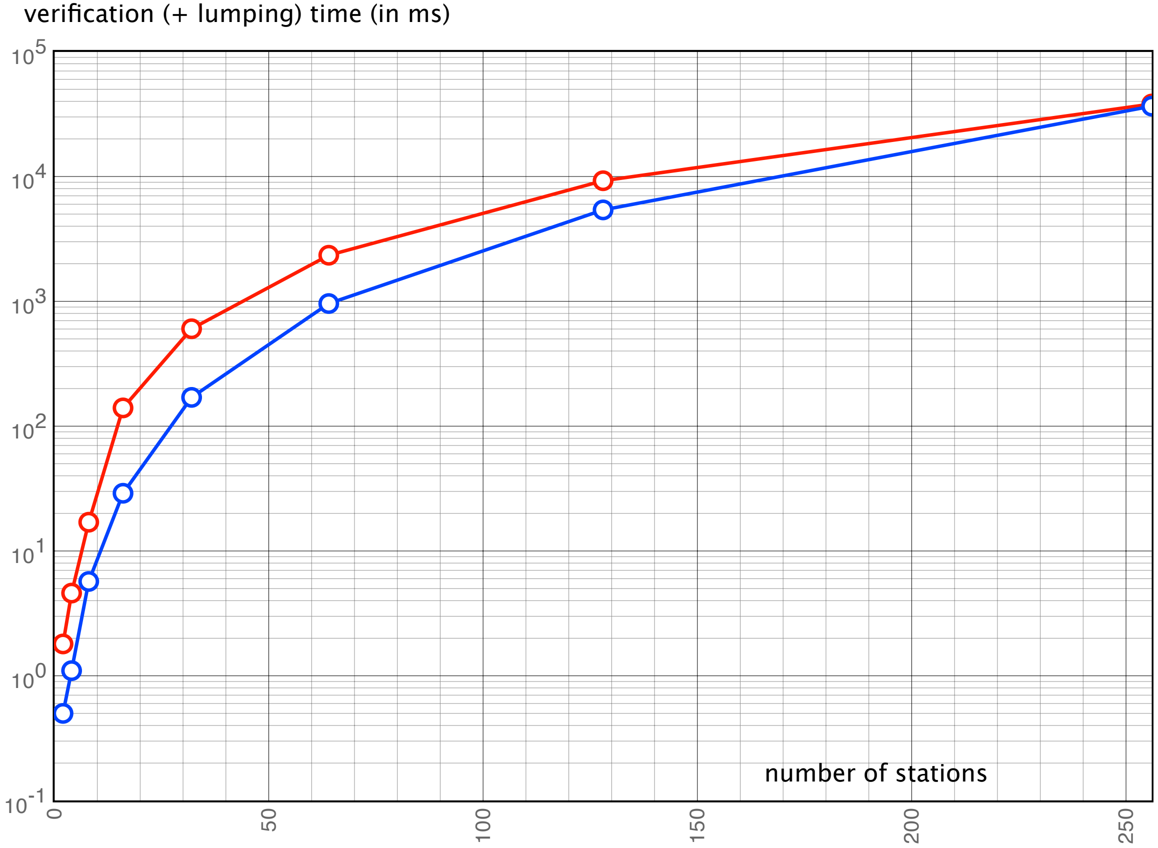
- Initial partition for $\sim$: $s_\Pi = \{ s' \mid L(s') = L(s) \}$
  - independent of the formula to be checked
- Now: exploit the structure of the formula to be checked
- Bounded until:
  - take $F = \{ \Psi, \neg\Phi \wedge \neg\Psi, \Phi \wedge \neg\Psi \}$
  - initial partition $\Pi = \{ s_\Psi, s_{\neg\Phi\wedge\neg\Psi}, \mathit{Sat}(\Phi \wedge \neg\Psi) \}$
  - or, for non-recurrent DTMCs: $\mathcal{P}_{\leqslant 0}(\Phi \,\mathsf{U}\, \Psi)$ instead of $\neg\Phi \wedge \neg\Psi$
- Standard until:
  - take $F = \{ \underbrace{\mathcal{P}_{\geqslant 1}(\Phi \,\mathsf{U}\, \Psi)}_{\text{single state in } \Pi}, \underbrace{\mathcal{P}_{\leqslant 0}(\Phi \,\mathsf{U}\, \Psi)}_{\text{single state in } \Pi}, \mathcal{P}_{>0}(\Phi \,\mathsf{U}\, \Psi) \wedge \mathcal{P}_{<1}(\Phi \,\mathsf{U}\, \Psi) \}$

Workstation cluster (Haverkort et al., 2001)



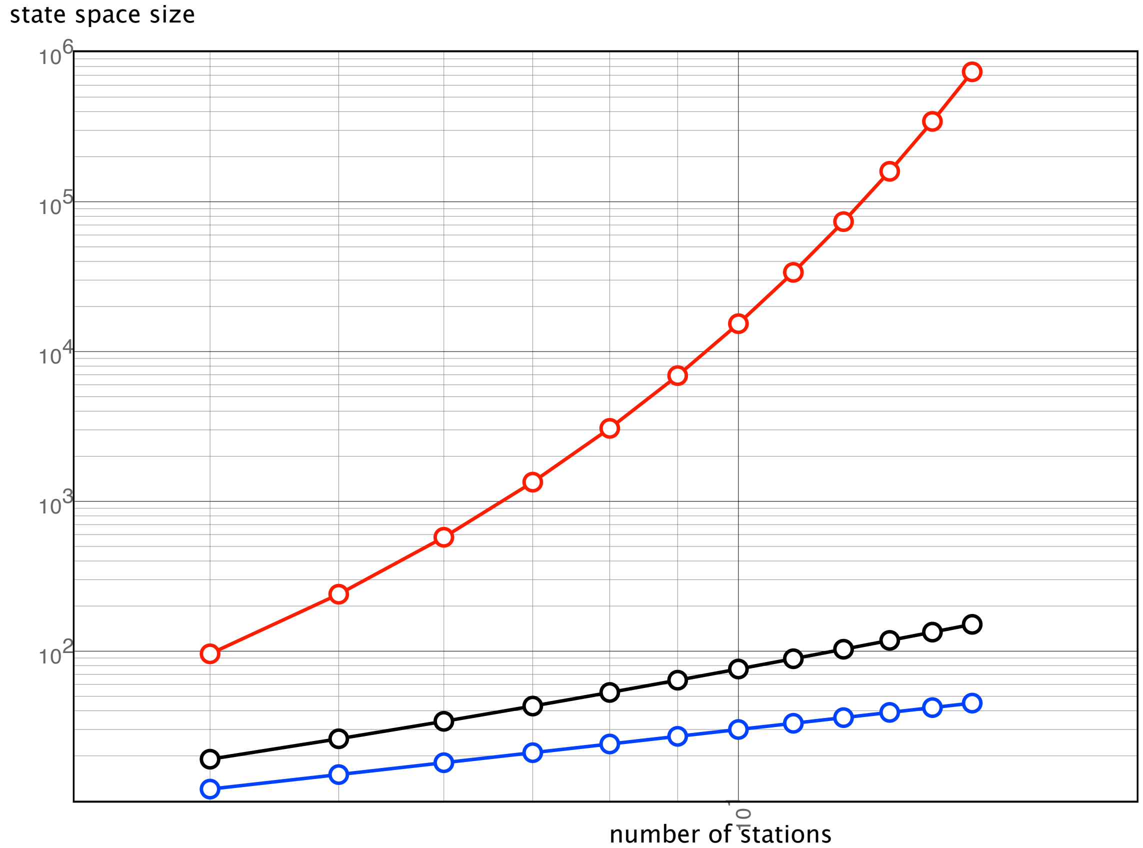
state space reductions for $\mathbb{P}_{\leqslant q}(\textit{minimum} \, \mathsf{U}^{\leqslant 510} \, \textit{premium})$

Workstation cluster



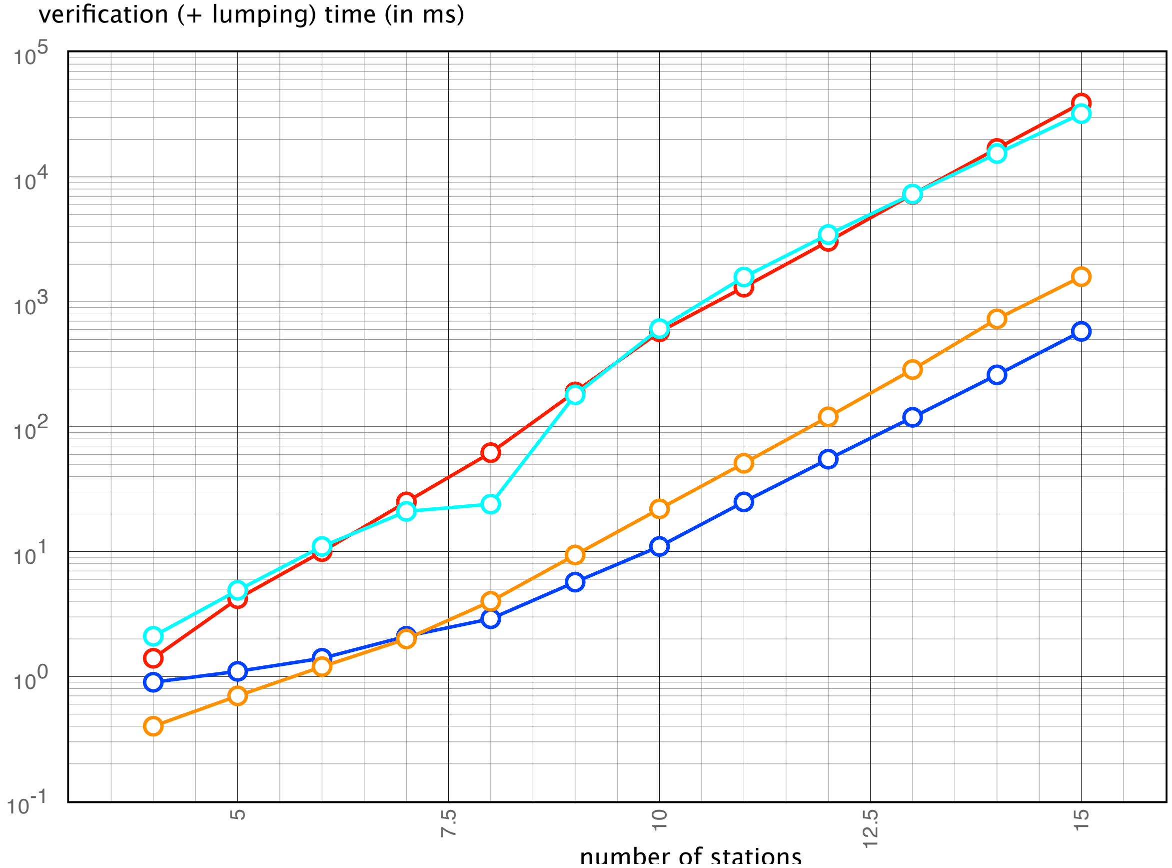
verification (+ lumping) times (in ms) for $\mathbb{P}_{\leqslant q}(\textit{minimum} \, \mathsf{U}^{\leqslant 510} \, \textit{premium})$

Cyclic polling system (Ibe & Trivedi, 1989)



state space reductions for $\mathbb{P}_{\leqslant q}(\neg \mathit{serve}_1 \ \mathsf{U}^{\leqslant 1010} \ \mathit{serve}_1)$ and $\mathbb{P}_{\leqslant q}(\neg \mathit{serve}_1 \ \mathsf{U} \ \mathit{serve}_1)$

Cyclic polling system

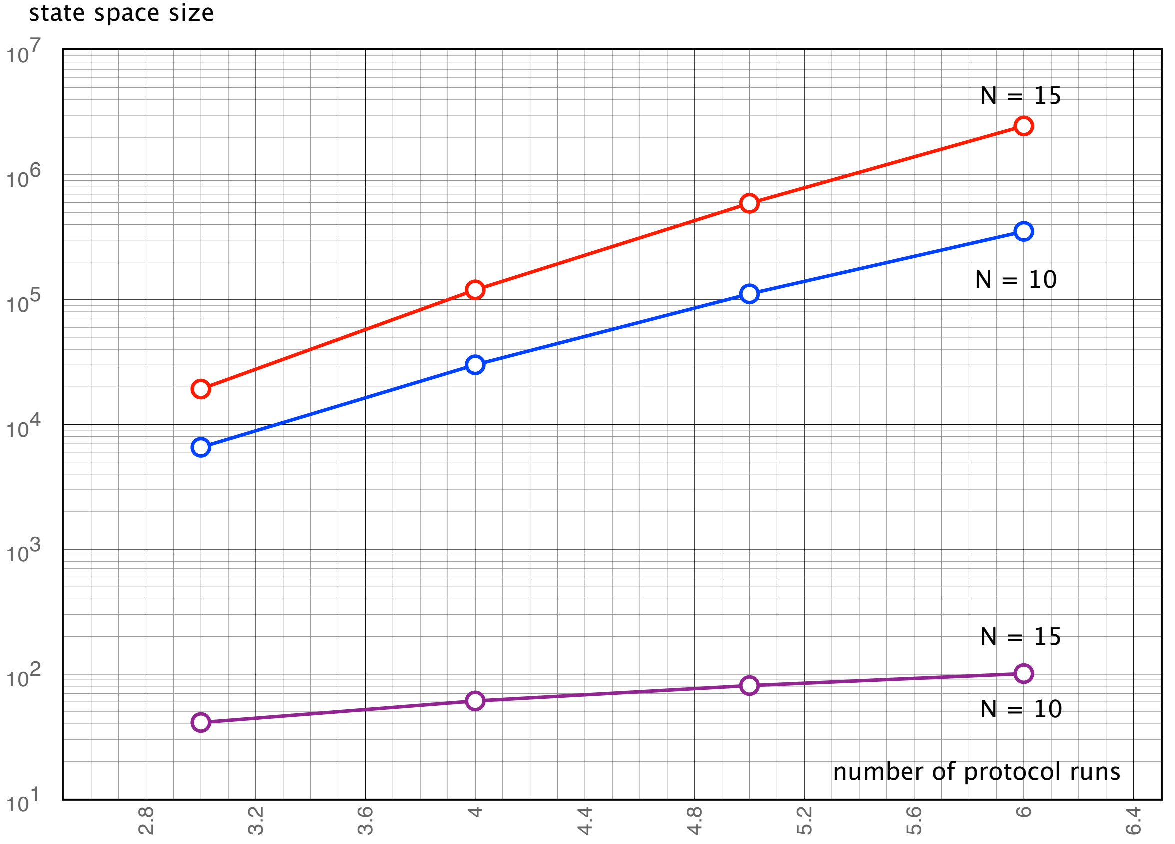


run times for $\mathbb{P}_{\leqslant q}(\neg serve_1 \ \mathsf{U}^{\leqslant 1010} \ serve_1)$ and $\mathbb{P}_{\leqslant q}(\neg serve_1 \ \mathsf{U} \ serve_1)$

Crowds protocol (Reiter & Rubin, 1998)

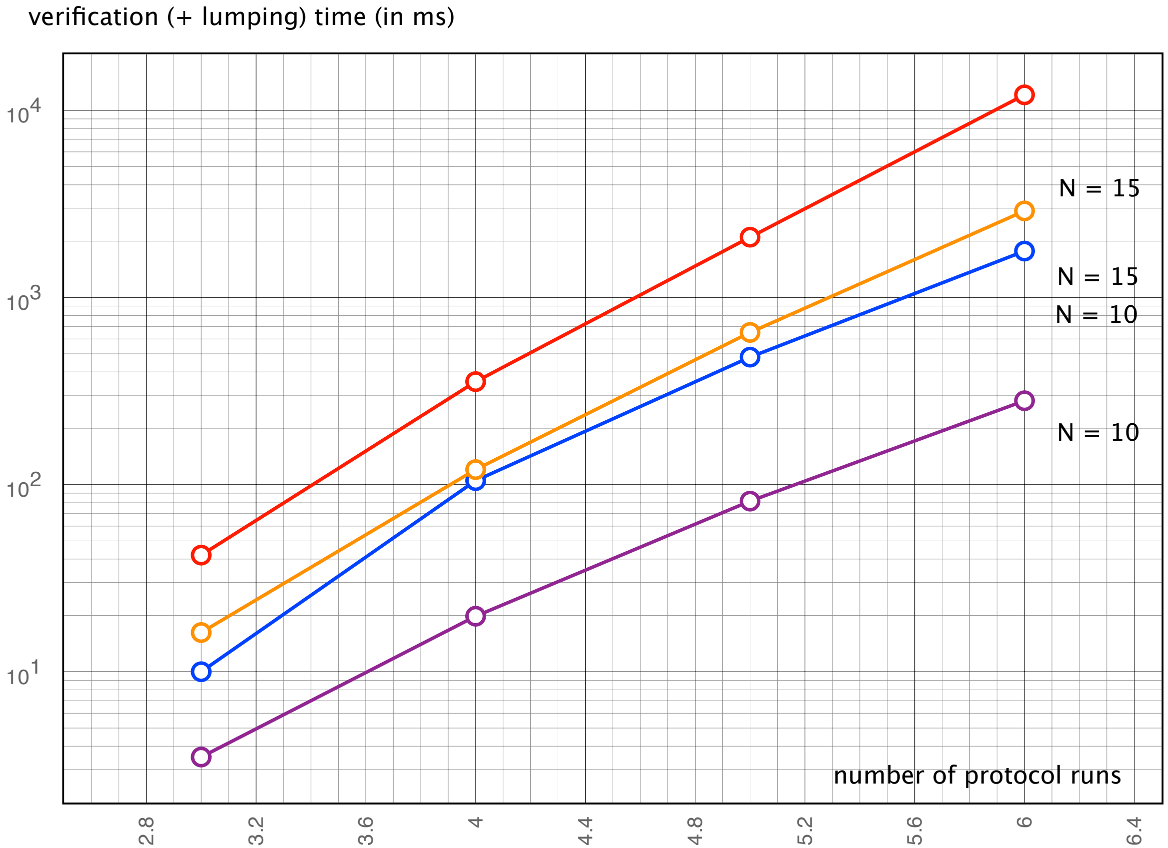
- A protocol for anonymous web browsing (variants: mCrowds, BT-Crowds)
- Hide user's communication by random routing within a crowd
  - sender selects a crowd member randomly using a uniform distribution
  - selected router flips a biased coin:
    * with probability $1 - p$: direct delivery to final destination
    * otherwise: select a next router randomly (uniformly)
  - once a routing path has been established, use it until crowd changes
- Rebuild routing paths on crowd changes (R times)
- Probable innocence:
  - probability real sender is discovered $< \frac{1}{2}$ if $N \geqslant \frac{p}{p-\frac{1}{2}} \cdot (c+1)$
  - where N is crowd's size and c is number of corrupt crowd members

Crowds protocol



state space reductions for eventually observer the real sender more than once

Crowds protocol



run times for eventually observer the real sender more than once

It mostly pays off!

- Significant state space reductions
  - reduction factors varying from 0 to 3 orders of magnitude
  - property-driven bisimulation yields better results
  - . . . even after symmetry reduction
- Mostly a reduction of the total verification time
  - depends on "denseness" and structure of the Markov chain
  - long run: convergence rate of numerical computations
  - reward models: huge reductions of verification time (up to 4 orders)
- Possibility to exploit component-wise minimisation