Model Checking Continuous-Time Markov Chains

Joost-Pieter Katoen

Software Modeling and Verification Group

RWTH Aachen University

associated to University of Twente, Formal Methods and Tools

Lecture at Quantitative Model Checking School, March 4, 2010

Content of this lecture

- Continuous Stochastic Logic
  - syntax, semantics, examples
- CSL model checking
  - basic algorithms and complexity
- Bisimulation
  - definition, minimization algorithm, examples
- Priced continuous-time Markov chains
  - motivation, definition, some properties

Content of this lecture

⇒ Continuous Stochastic Logic
  - syntax, semantics, examples
- CSL model checking
  - basic algorithms and complexity
- Bisimulation
  - definition, minimization algorithm, examples
- Priced continuous-time Markov chains
  - motivation, definition, some properties

Continuous-time Markov chain

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

- S is a countable (today: finite) set of *states*
- $\mathbf{P} : S \times S \rightarrow [0, 1]$, a *stochastic matrix*
  - $\mathbf{P}(s, s')$ is one-step probability of going from state s to state s'
  - s is called *absorbing* iff $\mathbf{P}(s, s) = 1$
- $r : S \rightarrow \mathbb{R}_{>0}$, the *exit-rate function*
  - $r(s)$ is the rate of exponential distribution of residence time in state s

CTMC paths

- An infinite path σ in a CTMC $\mathcal{C} = (S, \mathbf{P}, r, L)$ is of the form:

$$\sigma \;=\; s_0 \xrightarrow{t_0} s_1 \xrightarrow{t_1} s_2 \xrightarrow{t_2} s_3 \ldots\ldots$$

  with s_i is a state in S, $t_i \in \mathbb{R}_{>0}$ is a duration, and $\mathbf{P}(s_i, s_{i+1}) > 0$.

- A Borel space on infinite paths exists (cylinder construction)
  - reachability, timed reachability, and ω-regular properties are measurable

- Let *Paths*(s) denote the set of infinite path starting in state s

Reachability probabilities

- Let $\mathcal{C} = (S, \mathbf{P}, r, L)$ be a finite CTMC and $G \subseteq S$ a set of states
- Let $\Diamond G$ be the set of infinite paths in $\mathcal{C}$ reaching a state in G
- Question: what is the probability of $\Diamond G$ when starting from s?
  - what is the probability mass of all infinite paths from s that eventually hit G?
- As state residence times are not relevant for $\Diamond G$, this is simple

Probabilistic reachability

- $\Pr(s, \Diamond G)$ is the least solution of the set of linear equations:

$$\Pr(s, \Diamond G) = \begin{cases} 1 & \text{if } s \in G \\ \sum_{s' \in S} \mathbf{P}(s, s') \cdot \Pr(s', \Diamond G) & \text{otherwise} \end{cases}$$

- Unique solution by pre-computing *Sat*$(\forall \Diamond G)$ and *Sat*$(\exists \Diamond G)$
  - this is a standard graph analysis (as in CTL model checking)
- This is the same as in Christel's first lecture this morning

Continuous stochastic logic (CSL)

- CSL equips the until-operator with a time interval:
  - let interval $I \subseteq \mathbb{R}_{\geqslant 0}$ with rational bounds, e.g., $I = [0, 17]$
  - $\Phi \, \mathsf{U}^I \, \Psi$ asserts that a Ψ-state can be reached via Φ-states
    . . . while reaching the Ψ-state at some time $t \in I$
- CSL contains a probabilistic operator $\mathbb{P}$ with arguments
  - a path formula, e.g., *good* $\mathsf{U}^{[0,12]}$ *bad*, and
  - a probability interval $J \subseteq [0, 1]$ with rational bounds, e.g., $J = [0, \frac{1}{2}]$
- CSL contains a long-run operator $\mathbb{L}$ with arguments
  - a state formula, e.g., $a \wedge b$ or $\mathbb{P}_{=1}(\Diamond \Phi)$, and
  - a probability interval $J \subseteq [0, 1]$ with rational bounds

The branching-time logic CSL

- For $a \in AP$, $J \subseteq [0,1]$ and $I \subseteq \mathbb{R}_{\geqslant 0}$ intervals with rational bounds:

$$\boxed{\begin{array}{c} \Phi \;::=\; a \;\big|\; \neg\Phi \;\big|\; \Phi \wedge \Phi \;\big|\; \mathbb{L}_J(\Phi) \;\big|\; \mathbb{P}_J(\varphi) \\[2ex] \varphi \;::=\; \Phi \,\mathsf{U}\, \Phi \;\big|\; \Phi \,\mathsf{U}^{I}\, \Phi \end{array}}$$

- $s_0 t_0 s_1 t_1 s_2 \ldots \models \Phi \,\mathsf{U}^I\, \Psi$ if Ψ is reached at $t \in I$ and prior to t, Φ holds

- $s \models \mathbb{P}_J(\varphi)$ if the probability of the set of φ-paths starting in s lies in J

- $s \models \mathbb{L}_J(\Phi)$ if starting from s, the probability of being in Φ on the long run lies in J

Derived operators

$$\Diamond \Phi \;=\; \mathit{true}\, \mathsf{U}\, \Phi$$

$$\Diamond^{\leqslant t}\, \Phi \;=\; \mathit{true}\, \mathsf{U}^{\leqslant t}\, \Phi$$

$$\mathbb{P}_{\leqslant p}(\Box \Phi) \;=\; \mathbb{P}_{\geqslant 1-p}(\Diamond \neg \Phi)$$

$$\mathbb{P}_{]p,q]}(\Box^{\leqslant t}\, \Phi) \;=\; \mathbb{P}_{[1-q,1-p[}(\Diamond^{\leqslant t}\, \neg \Phi)$$

abbreviate $\mathbb{P}_{[0,0.5]}(\varphi)$ by $\mathbb{P}_{\leqslant 0.5}(\varphi)$ and $\mathbb{P}_{]0,1]}(\varphi)$ by $\mathbb{P}_{>0}(\varphi)$ and so on

Timed reachability formulas

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

$$\mathbb{P}_{\geqslant 0.92}\left(\mathit{legal}\ \mathsf{U}^{\leqslant 3.1}\ \mathit{goal}\right)$$

- Almost surely stay in a legal state for at least 10 sec:

$$\mathbb{P}_{=1}\left(\Box^{\leqslant 10}\ \mathit{legal}\right)$$

- Combining these two constraints:

$$\mathbb{P}_{\geqslant 0.92}\left(\mathit{legal}\ \mathsf{U}^{\leqslant 3.1}\ \mathbb{P}_{=1}\left(\Box^{\leqslant 10}\ \mathit{legal}\right)\right)$$

Long-run formulas

- The long-run probability of being in a safe state is at most 0.00001:

$$\mathbb{L}_{\leqslant 10^{-5}}\,(\textit{safe})$$

- On the long run, with at least "five nine" likelihood almost surely a goal state can be reached within one sec.:

$$\mathbb{L}_{\geqslant 0.99999}\left(\mathbb{P}_{=1}(\Diamond^{\leqslant 1}\textit{goal})\right)$$

- The probability to reach a state that in the long run guarantees more than five-nine safety exceeds $\frac{1}{2}$:

$$\mathbb{P}_{>0.5}\,(\Diamond\,\mathbb{L}_{>0.99999}(\textit{safe}))$$

CSL semantics

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

$s \models a$	iff	$a \in L(s)$
$s \models \neg\,\Phi$	iff	not $(s \models \Phi)$
$s \models \Phi \wedge \Psi$	iff	$(s \models \Phi)$ and $(s \models \Psi)$
$s \models \mathbb{L}_J(\Phi)$	iff	$\lim_{t\to\infty} \Pr\{\, \sigma \in \textit{Paths}(s) \mid \sigma@t \models \Phi \,\} \in J$
$s \models \mathbb{P}_J(\varphi)$	iff	$\Pr\{\, \sigma \in \textit{Paths}(s) \mid \sigma \models \varphi \,\} \in J$
$\sigma \models \Phi \,\mathsf{U}^I\, \Psi$	iff	$\exists t \in I.\ ((\forall t' \in [0,t).\, \sigma@t' \models \Phi) \wedge \sigma@t \models \Psi)$

where $\sigma@t$ is the state along σ that is occupied at time t

Content of this lecture

- Continuous Stochastic Logic
  - syntax, semantics, examples

$\Rightarrow$ CSL model checking
  - basic algorithms and complexity

- Bisimulation
  - definition, minimization algorithm, examples

- Priced continuous-time Markov chains
  - motivation, definition, some properties

CSL model checking

- Let $\mathcal{C}$ be a finite CTMC and Φ a CSL formula.
- Problem: determine the states in $\mathcal{C}$ satisfying Φ
- Determine *Sat*(Φ) by a recursive descent over parse tree of Φ
- For the propositional fragment ($\neg, \wedge, a$): do as for CTL
- How to check formulas of the form $\mathbb{P}_J(\varphi)$?
  - φ is an until-formula: do as for PCTL, i.e., linear equation system
  - φ is a time-bounded until-formula: integral equation system
- How to check formulas of the form $\mathbb{L}_J(\Psi)$?
  - graph analysis + solving linear equation system(s)

Model-checking the long-run operator

- For a strongly-connected CTMC:

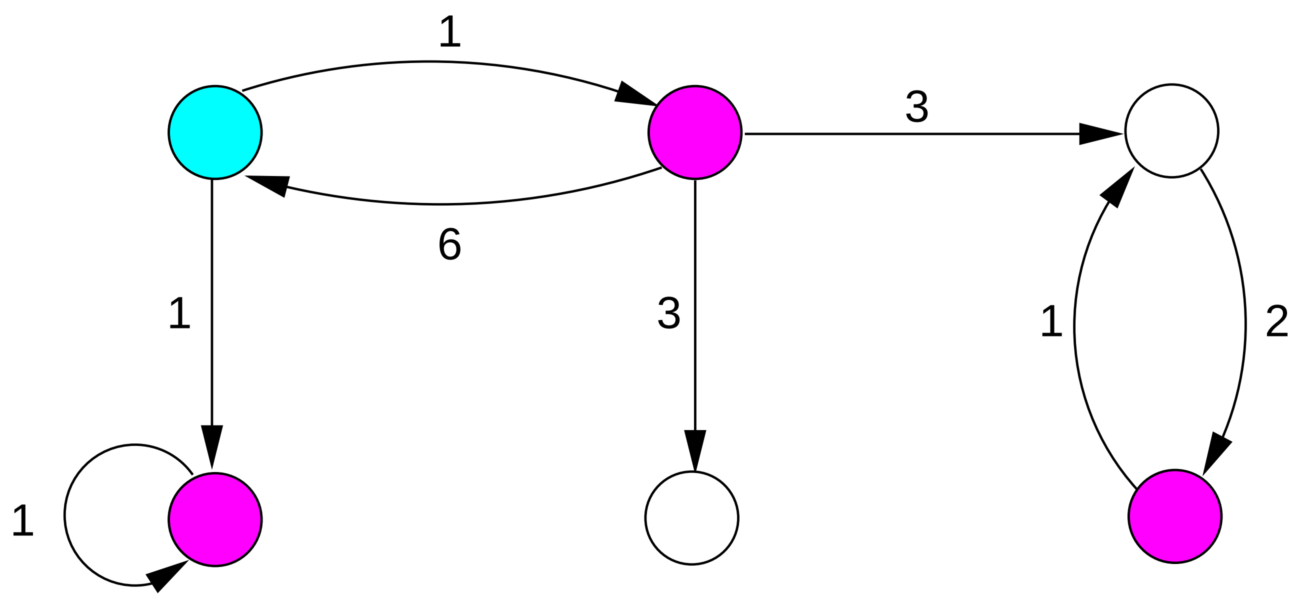
$$s \in \textit{Sat}(\mathbb{L}_J(\Phi)) \quad \text{iff} \quad \sum_{s' \in \textit{Sat}(\Phi)} p(s') \in J$$

$\Longrightarrow$ 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

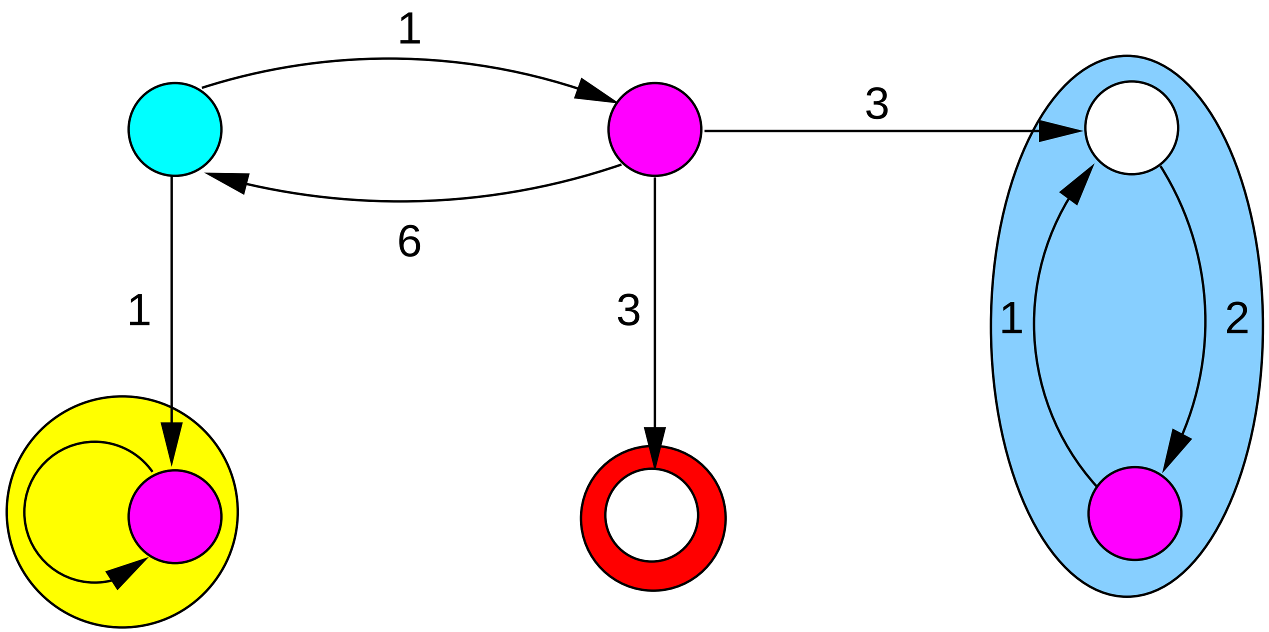
$$s \in \textit{Sat}(\mathbb{L}_J(\Phi)) \quad \text{iff} \quad \sum_{B} \left(\Pr\{\, s \models \Diamond B \,\} \cdot \sum_{s' \in B \cap \textit{Sat}(\Phi)} p^B(s') \right) \in J$$

Verifying long-run properties: an example



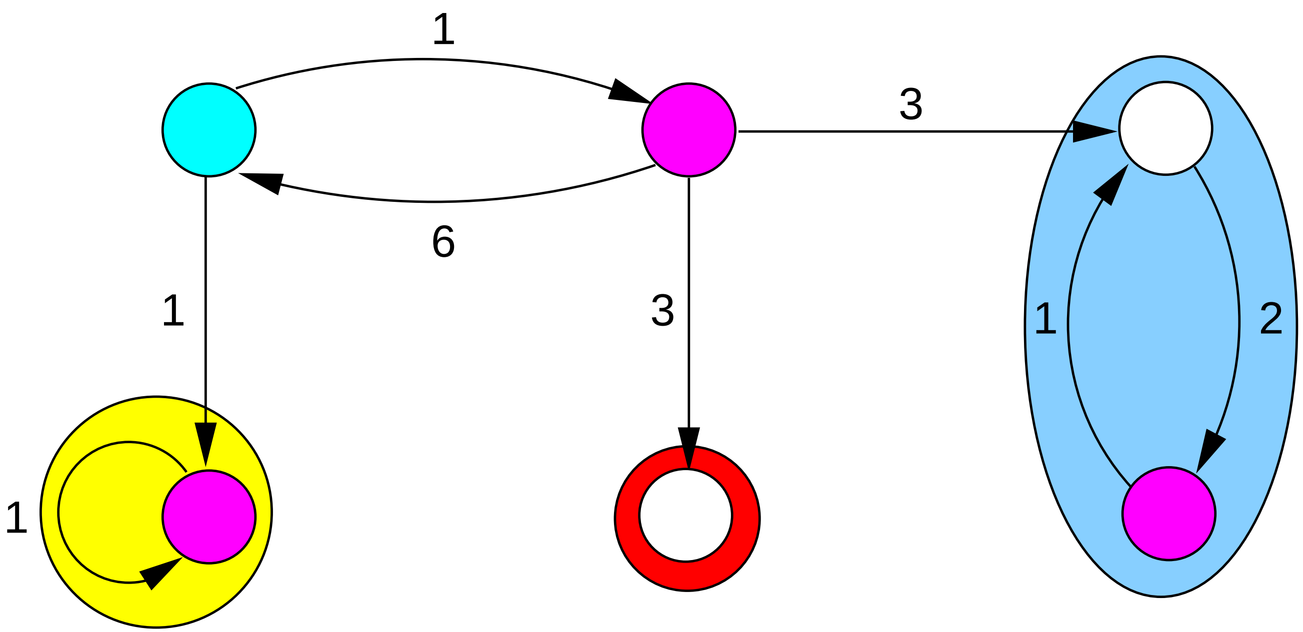
determine the bottom strongly-connected components

Verifying long-run properties: an example



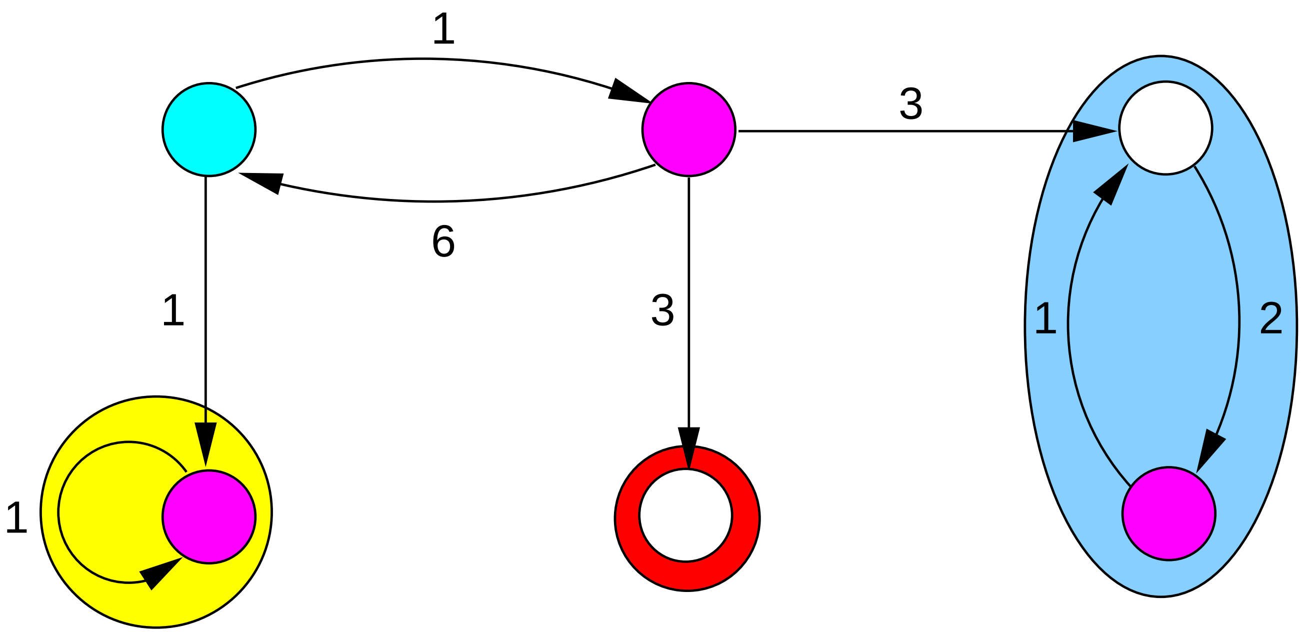
$$s \models \mathbb{L}_{>\frac{3}{4}}(\textit{magenta}) \quad \text{iff} \quad \Pr\{s \models \Diamond at_{yellow}\} \cdot p^{yellow}(\textit{magenta}) + \Pr\{s \models \Diamond at_{blue}\} \cdot p^{blue}(\textit{magenta}) > \frac{3}{4}$$

Verifying long-run properties: an example



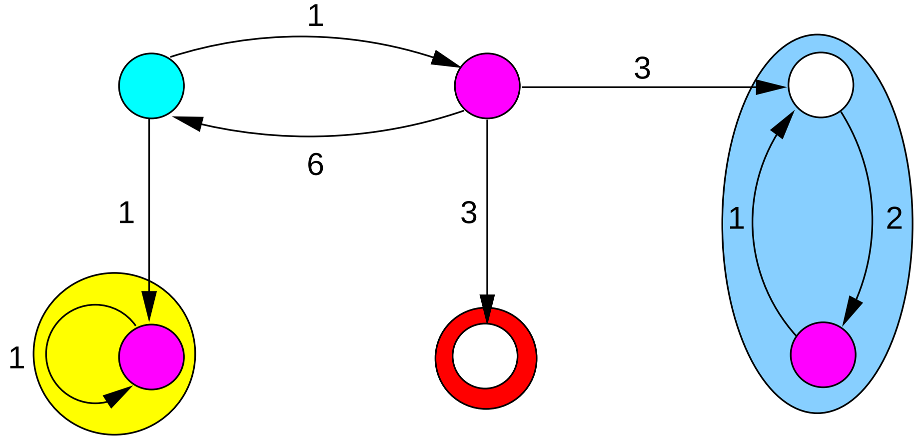
$$s \models \mathbb{L}_{>\frac{3}{4}}(\textit{magenta}) \quad \text{iff} \quad \Pr\{s \models \Diamond at_{yellow}\} \cdot \underbrace{p^{yellow}(\textit{magenta})}_{=1}$$
$$+ \Pr\{s \models \Diamond at_{blue}\} \cdot \underbrace{p^{blue}(\textit{magenta})}_{=\frac{2}{3}} > \frac{3}{4}$$

Verifying long-run properties: an example



$s \models \mathbb{L}_{>\frac{3}{4}}(\textit{magenta})$ iff $\Pr\{s \models \Diamond at_{yellow}\} + \frac{2}{3}\Pr\{s \models \Diamond at_{blue}\} > \frac{3}{4}$

Verifying long-run properties: an example



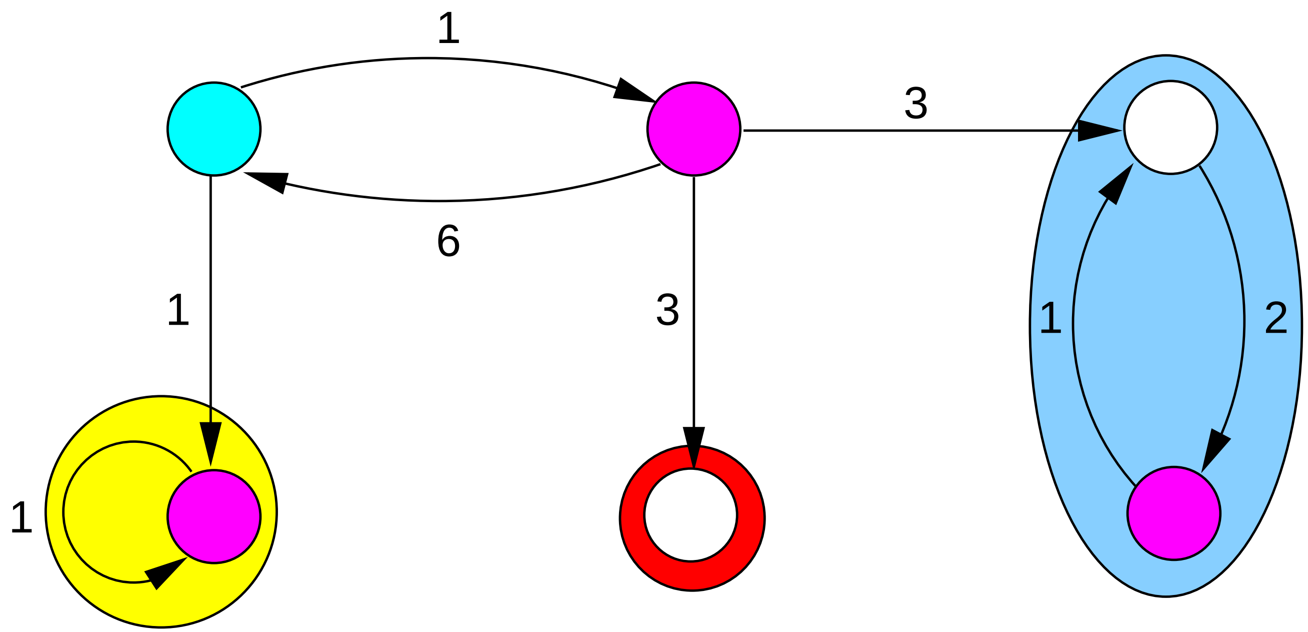
$s \models \mathbb{L}_{>\frac{3}{4}}(\textit{magenta})$ iff $\Pr\{s \models \Diamond at_{yellow}\} + \frac{2}{3}\Pr\{s \models \Diamond at_{blue}\} > \frac{3}{4}$

$$\Pr\{s \models \Diamond at_{yellow}\} = \frac{1}{2} + \frac{1}{2}\Pr\{s' \models \Diamond at_{yellow}\}$$

$$\Pr\{s' \models \Diamond at_{yellow}\} = \frac{1}{2}\Pr\{s \models \Diamond at_{yellow}\}$$

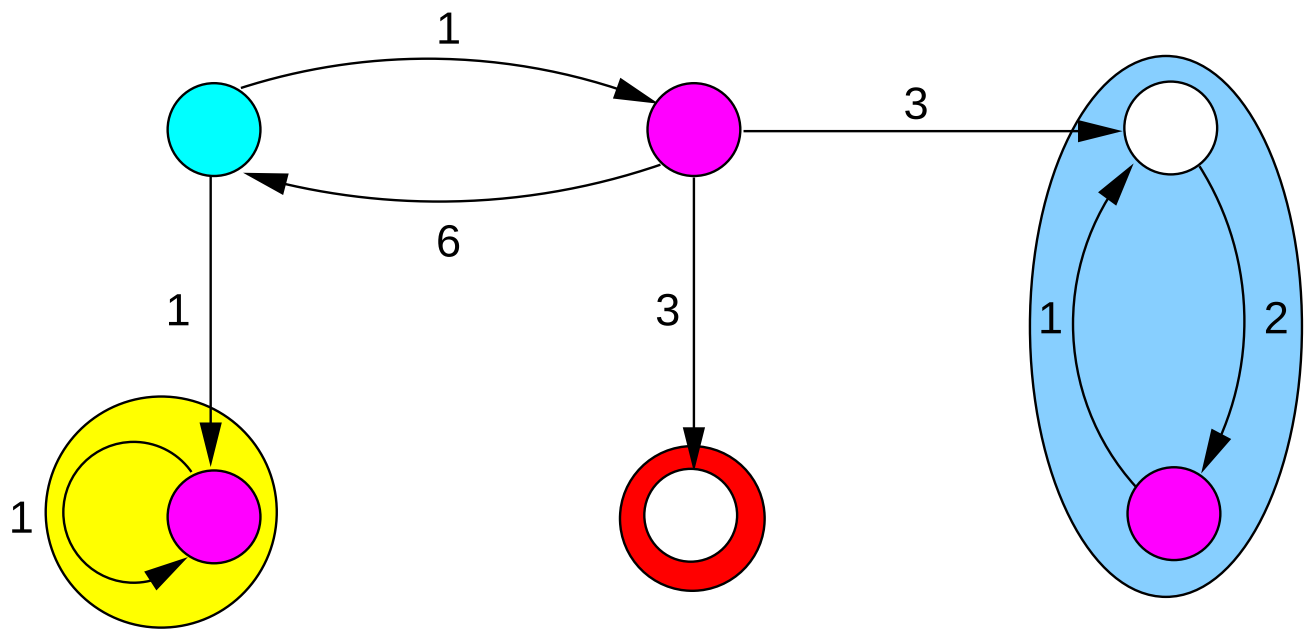
$$\Rightarrow \Pr\{s \models \Diamond at_{yellow}\} = \frac{1}{2}\sum_{k=0}^{\infty}\left(\frac{1}{4}\right)^k = \frac{2}{3}$$

Verifying long-run properties: an example



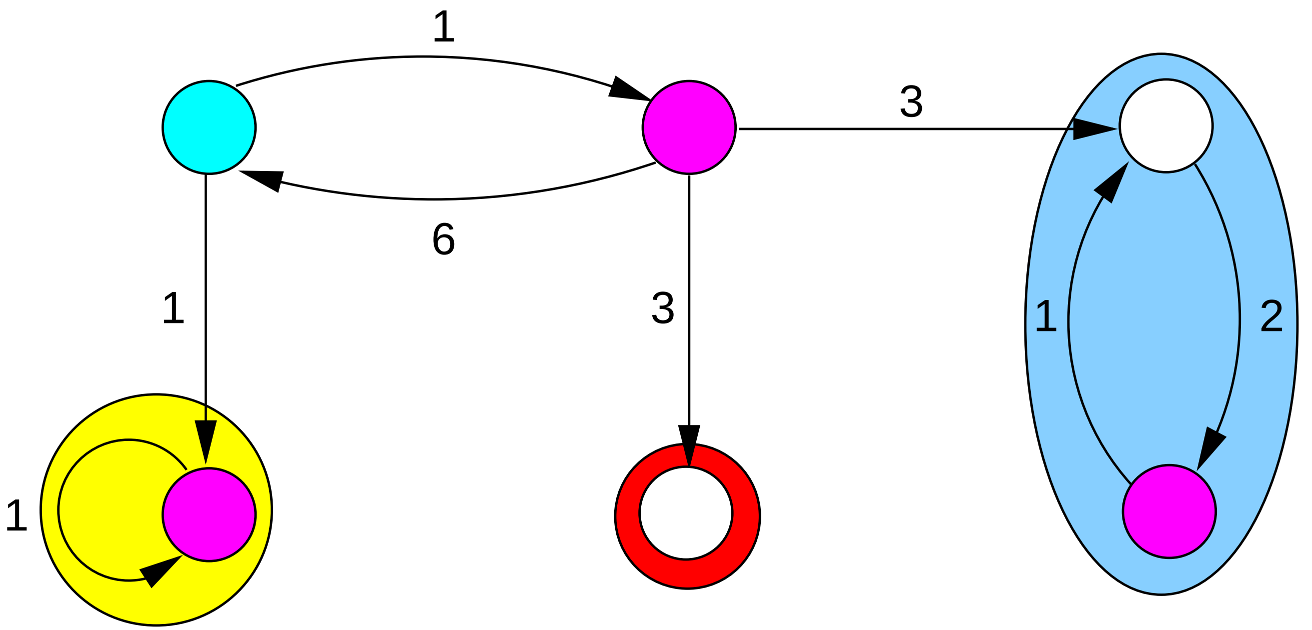
$$\textcolor{blue}{s} \models \mathbb{L}_{>\frac{3}{4}}(\textcolor{magenta}{\textit{magenta}}) \quad \text{iff} \quad \underbrace{\Pr\{\textcolor{blue}{s} \models \Diamond at_{yellow}\}}_{\frac{2}{3}} + \frac{2}{3} \underbrace{\Pr\{\textcolor{blue}{s} \models \Diamond at_{blue}\}}_{\frac{1}{6}} > \frac{3}{4}$$

Verifying long-run properties: an example



$$s \models \mathbb{L}_{>\frac{3}{4}}(\textit{magenta}) \quad \text{iff} \qquad \frac{2}{3} + \frac{2}{3} \cdot \frac{1}{6} > \frac{3}{4}$$

Verifying long-run properties: an example



Thus: $s \models \mathbb{L}_{>\frac{3}{4}}(\textit{magenta})$ as $\underbrace{\frac{2}{3} + \frac{2}{3} \cdot \frac{1}{6}}_{\frac{7}{9}} > \frac{3}{4}$

Time-bounded reachability

- $s \models \mathbb{P}_J\left(\Phi \, \mathsf{U}^I \, \Psi\right)$ if and only if $\Pr\{s \models \Phi \, \mathsf{U}^I \, \Psi\} \in J$
- For $I = [0, t]$, $\Pr\{s \models \Phi \, \mathsf{U}^{\leqslant t} \Psi\}$ is the least solution of:
  - 1 if $s \in$ *Sat*(Ψ)
  - if $s \in$ *Sat*$(\Phi) -$ *Sat*(Ψ):

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

  - 0 otherwise

Reduction to transient analysis

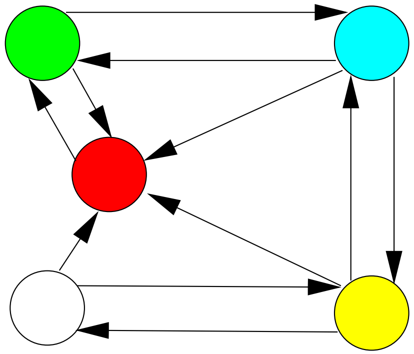
- For an arbitrary CTMC $\mathcal{C}$ and property $\varphi = \Phi \, \mathsf{U}^{\leqslant t} \, \Psi$ we have:
  - φ is fulfilled once a Ψ-state is reached before t along a Φ-path
  - φ is violated once a $\neg\,(\Phi \,\vee\, \Psi)$-state is visited before t

- This suggests to transform the CTMC $\mathcal{C}$ as follows:
  - make all Ψ-states and all $\neg\,(\Phi \,\vee\, \Psi)$-states absorbing

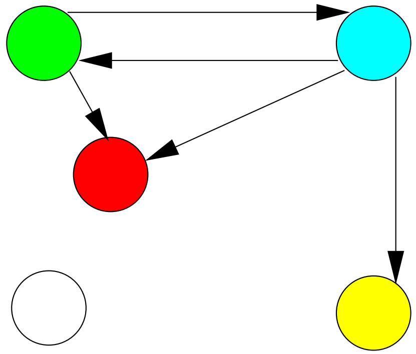
- Theorem: $\underbrace{s \models \mathbb{P}_J(\Phi \, \mathsf{U}^{\leqslant t} \, \Psi)}_{\text{in } \mathcal{C}}$ iff $\underbrace{s \models \mathbb{P}_J(\Diamond^{=t} \, \Psi)}_{\text{in } \mathcal{C}'}$

- Then it follows: $s \models_{\mathcal{C}'} \mathbb{P}_J(\Diamond^{=t} \, \Psi)$ iff $\underbrace{\sum_{s' \models \Psi} p_{s'}(t)}_{\text{transient probs in } \mathcal{C}'} \in J$

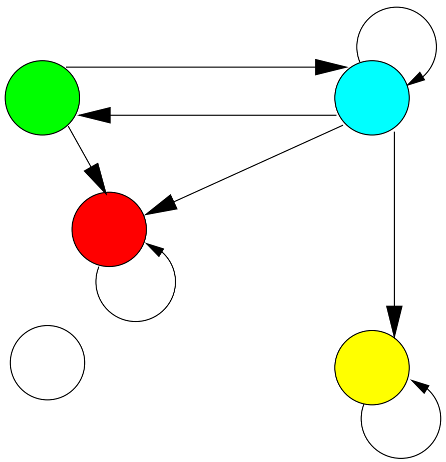
Example: TMR with $\mathbb{P}_J((\textit{green} \vee \textit{blue})\, \mathsf{U}^{[0,3]}\, \textit{red})$



transformation



uniformisation



recursive computation

like PCTL

bounded until

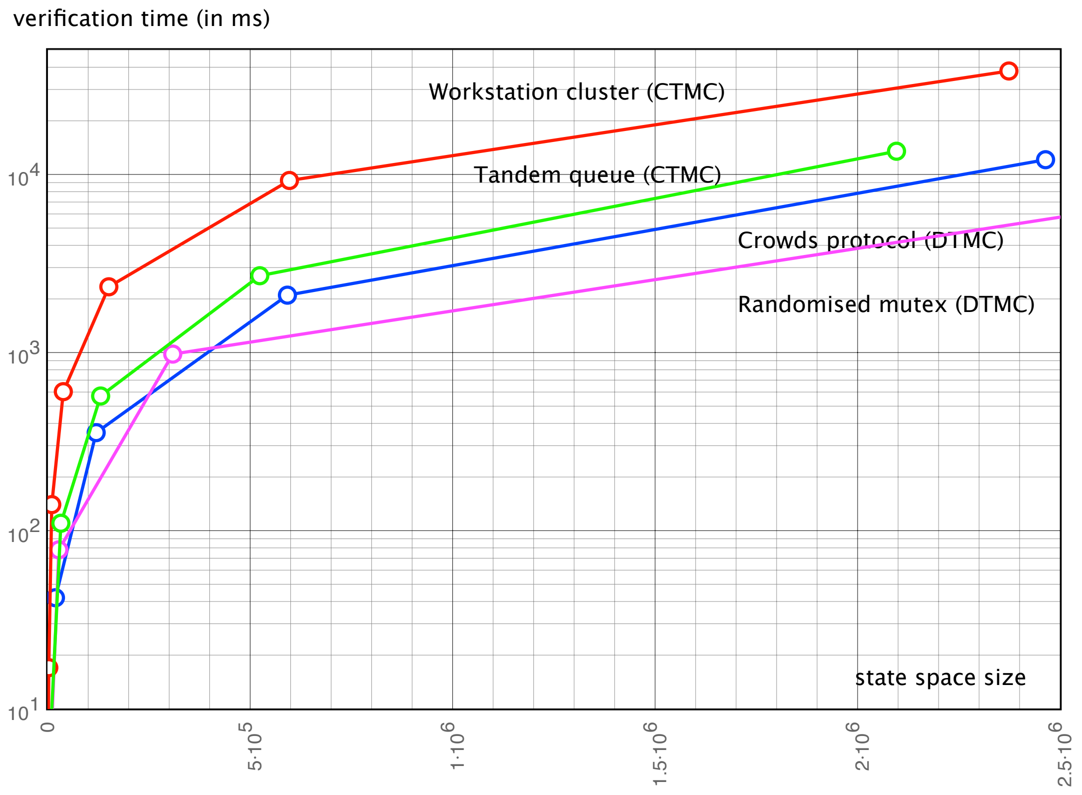
Interval-bounded reachability

- For any path σ that fulfills $\Phi \, \mathsf{U}^{[t,t']} \, \Psi$ with $0 < t \leqslant t'$:
  - Φ holds continuously up to time t, and
  - the suffix of σ starting at time t fulfills $\Phi \, \mathsf{U}^{[0,t'-t]} \, \Psi$
- Approach: divide the problem into two:

$$\underbrace{\sum_{s' \models \Phi} p^{\mathcal{C}'}(s, s', t)}_{\text{check } \Box^{[0,t]} \, \Phi} \cdot \underbrace{\sum_{s'' \models \Psi} p^{\mathcal{C}''}(s', s'', t'-t)}_{\substack{\text{check } \Phi \, \mathsf{U}^{[0,t'-t]} \, \Psi \\ \text{with starting distribution } \underline{p}^{\mathcal{C}'}(t)}}$$

  - where CTMC $\mathcal{C}'$ equals $\mathcal{C}$ with all Φ-states absorbing
  - and CTMC $\mathcal{C}''$ equals $\mathcal{C}$ with all Ψ and $\neg \, (\Phi \vee \Psi)$-states absorbing

Verification times



command-line tool MRMC ran on a Pentium 4, 2.66 GHz, 1 GB RAM laptop

Reachability probabilities

	Nondeterminism no	Nondeterminism yes
Reachability	linear equation system DTMC	linear programming MDP
Timed reachability	transient analysis CTMC	discretisation + linear programming CTMDP

Summary of CSL model checking

- Recursive descent over the parse tree of Φ
- Long-run operator: graph analysis + linear system(s) of equations
- Time-bounded until: CTMC transformation and uniformization
- Worst case time-complexity: $\mathcal{O}(|\Phi| \cdot (|\,\mathbf{R}\,| \cdot r \cdot t_{max} + |\,S\,|^{2.81}))$

  with $|\Phi|$ the length of Φ, uniformization rate r, t_{max} the largest time bound in Φ
- Tools:

  PRISM (symbolic), MRMC (explicit state), YMER (simulation), VESTA (simulation), . . .

Content of this lecture

- Continuous Stochastic Logic
  - syntax, semantics, examples
- CSL model checking
  - basic algorithms and complexity

$\Rightarrow$ Bisimulation
  - definition, minimization algorithm, examples

- Priced continuous-time Markov chains
  - motivation, definition, some properties

Probabilistic bisimulation

- Traditional LTL/CTL model checking: (Fisler & Vardi, 1998)
  - significant reductions in state space (upto logarithmic)
  - cost of bisimulation minimisation significantly exceeds model checking time
- Pros:
  - fully automated and efficient abstraction technique
  - enables compositional minimization
- Our interest:

does bisimulation minimization as pre-computation step
of probabilistic model checking pay off?

Probabilistic bisimulation

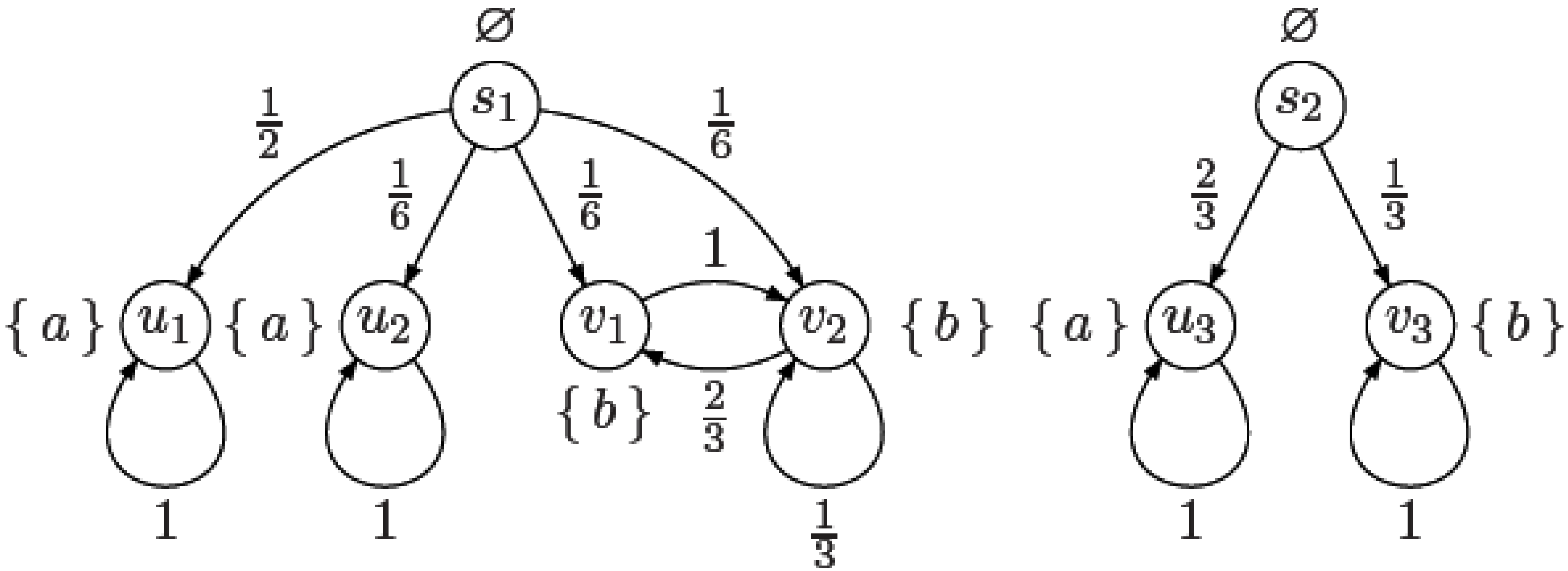
- Let $\mathcal{C} = (S, \mathbf{P}, r, L)$ be a CTMC and R an equivalence relation on S
- R is a probabilistic bisimulation on S if for any $(s, s') \in R$ it holds:
  1. $L(s) = L(s')$
  2. $r(s) = r(s')$
  3. $\mathbf{P}(s, C) = \mathbf{P}(s', C)$ for all $C \in S/R$, where $\mathbf{P}(s, C) = \sum_{u \in C} \mathbf{P}(s, u)$

  Note that the last two conditions together equal $\mathbf{R}(s, C) = \mathbf{R}(s', C)$.

- States s and s' are bisimilar, denoted $s \sim s'$, if:

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

Example



for simplicity, all states have the same exit rate (= uniform CTMC)

Quotient Markov chain

For $\mathcal{C} = (S, \mathbf{R}, L)$ and probabilistic bisimulation $\sim \subseteq S \times S$ let

$$\mathcal{C}/\sim \; = \; (S', \mathbf{R}', L'), \qquad \text{the quotient of } \mathcal{C} \text{ under } \sim$$

where

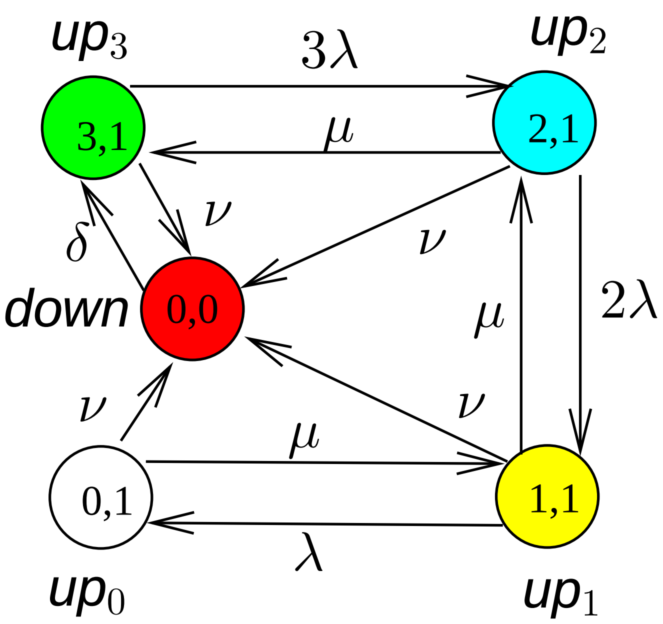
- $S' = S/\sim \; = \; \{\, [s]_\sim \mid s \in S \,\}$ with $[s]_\sim \; = \; \{\, s' \in S \mid s \sim s' \,\}$
- $\mathbf{R}' : S' \times S' \to [0, 1]$ is defined such that for each $s \in S$ and $C \in S$:

$$\mathbf{R}'\,([s]_\sim, C) \;=\; \mathbf{R}(s, C)$$

- $L'([s]_\sim) = L(s)$

it follows that $\mathcal{C} \;\sim\; \mathcal{C}/\sim$

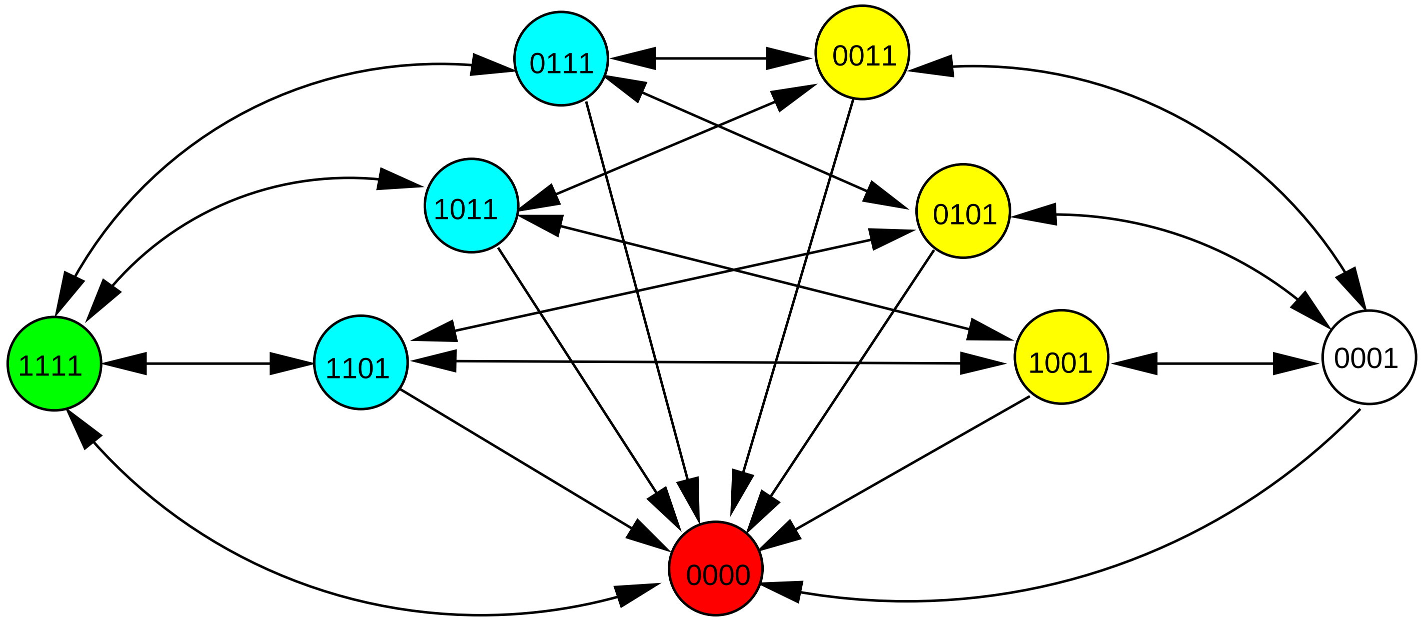
Modelling a TMR system as a CTMC



- processor failure rate is λ fph;
  its repair rate is μ rph
- voter failure rate is ν fph;
  its repair rate is δ rph
- rate matrix: e.g., $\mathbf{R}((3,1),(2,1)) = 3\lambda$
- exit rates: e.g., $r((3,1)) = 3\lambda{+}\nu$
- probability matrix: e.g.,

$$\mathbf{P}((3,1),(2,1)) = \frac{3\lambda}{3\lambda{+}\nu}$$

A bisimilar TMR model



$$\mathbf{R}'([s]_{\sim_m}, C) = \mathbf{R}(s, C) = \sum_{s' \in C} \mathbf{R}(s, s')$$

Preservation of state probabilities

- Let $\mathcal{C} = (S, \mathbf{R}, L)$ be a CTMC with initial distribution $\underline{p}(0)$
- For any $C \in S_0/\sim$ we have:

$$\underline{p}'_C(t) = \sum_{s \in C} \underline{p}_s(t) \qquad \text{for any } t \geqslant 0$$

- If the steady-state distribution exists, then it follows:

$$\underline{p}'_C = \lim_{t \to \infty} \underline{p}'_C(t) = \lim_{t \to \infty} \sum_{s \in C} \underline{p}_s(t) = \sum_{s \in C} \underline{p}_s$$

Logical characterization

For any finite CTMC with states s and s':

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

The quotient under the coarsest bisimulation can be obtained by partition-refinement in time-complexity $\mathcal{O}(|\mathbf{R}| \cdot \log |S|)$

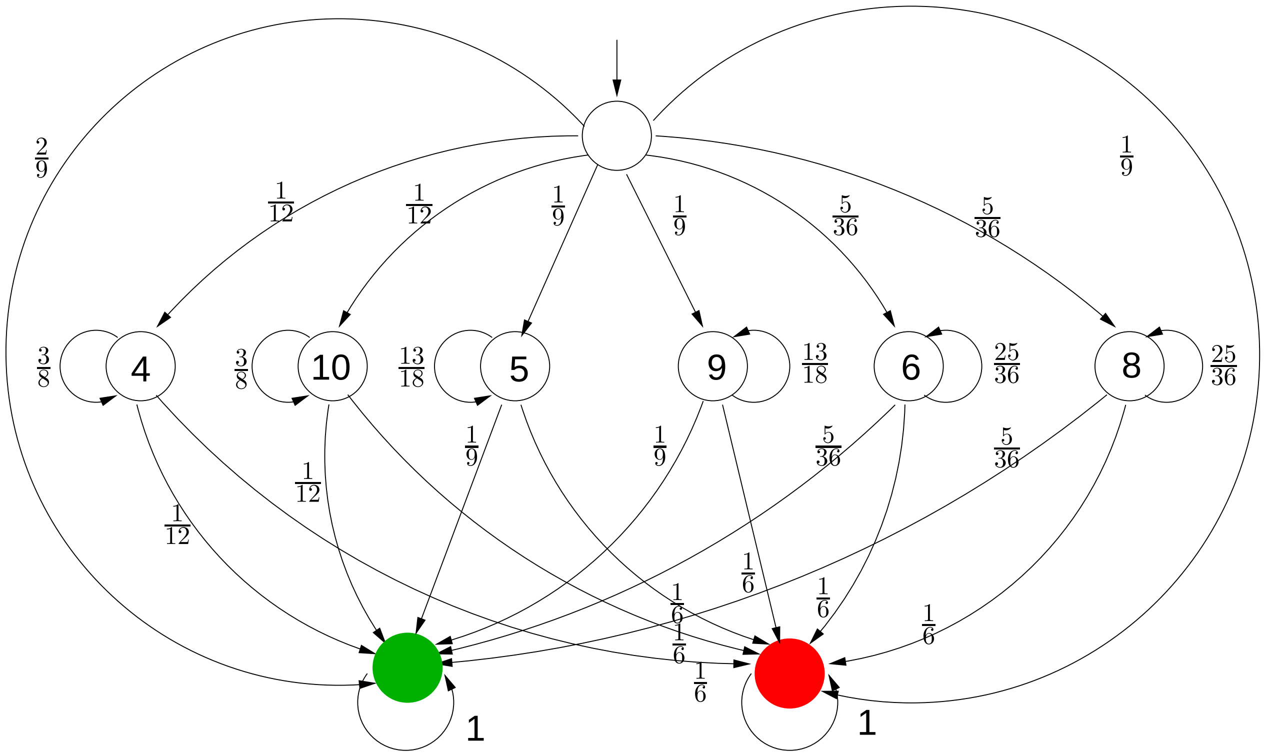
Craps

- 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”)

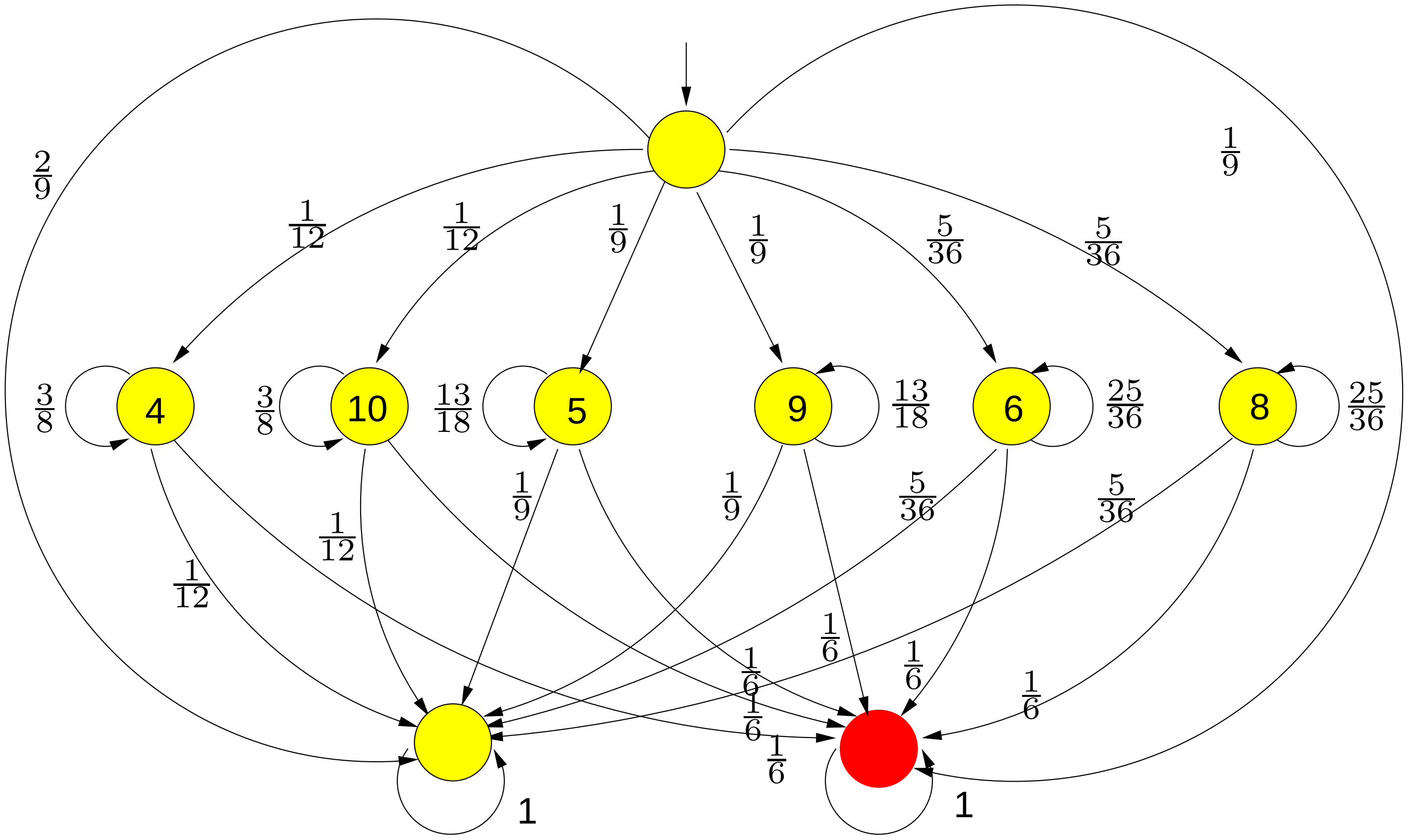
- Repeat until 7 or the “point” is thrown:
  - outcome 7: loss (“seven-out”)
  - outcome the point: win
  - any other outcome: roll again

A DTMC model of Craps

- Come-out roll:
  - 7 or 11: win
  - 2, 3, or 12: loss
  - else: roll again
- Next roll(s):
  - 7: loss
  - point: win
  - else: roll again

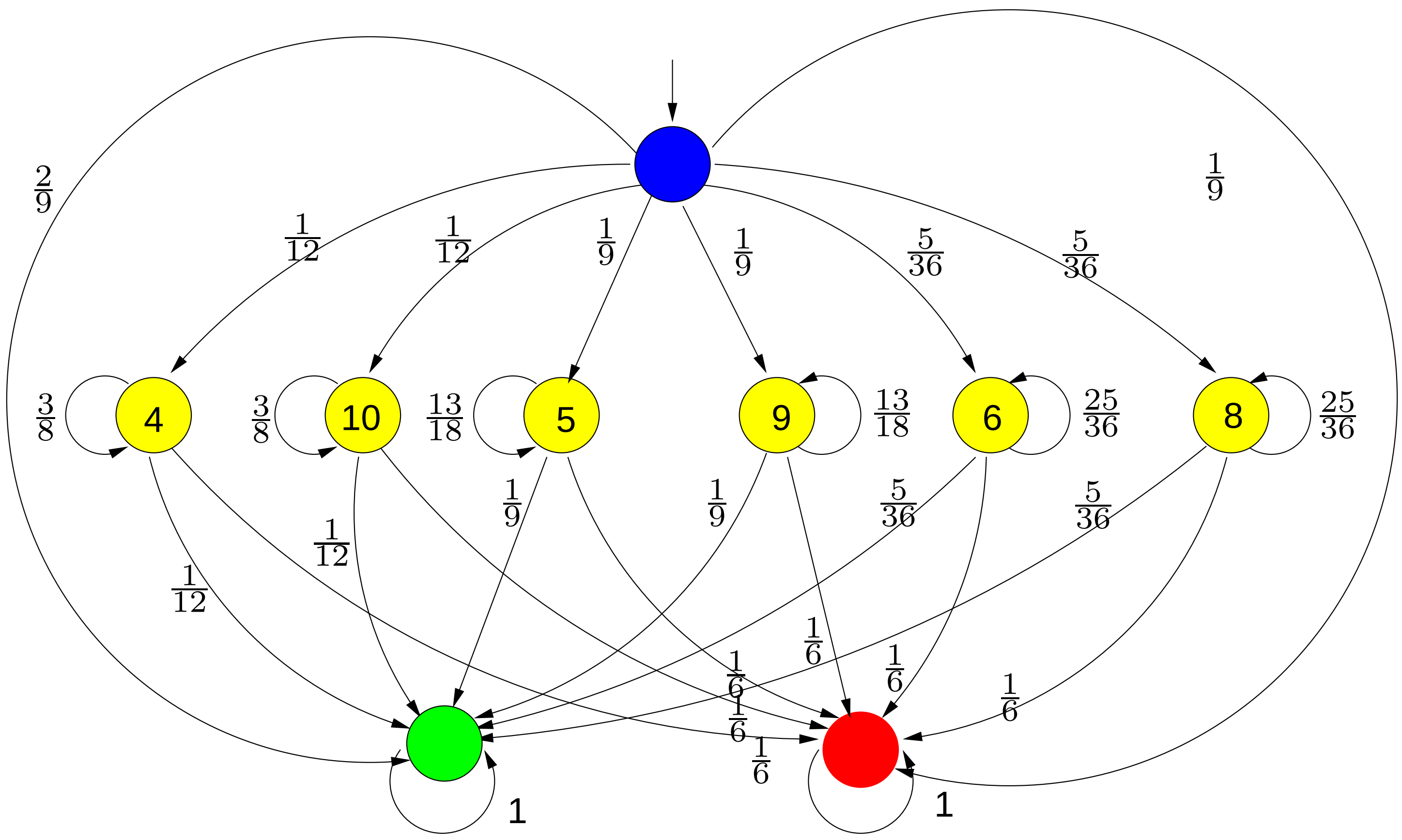


Minimizing Craps



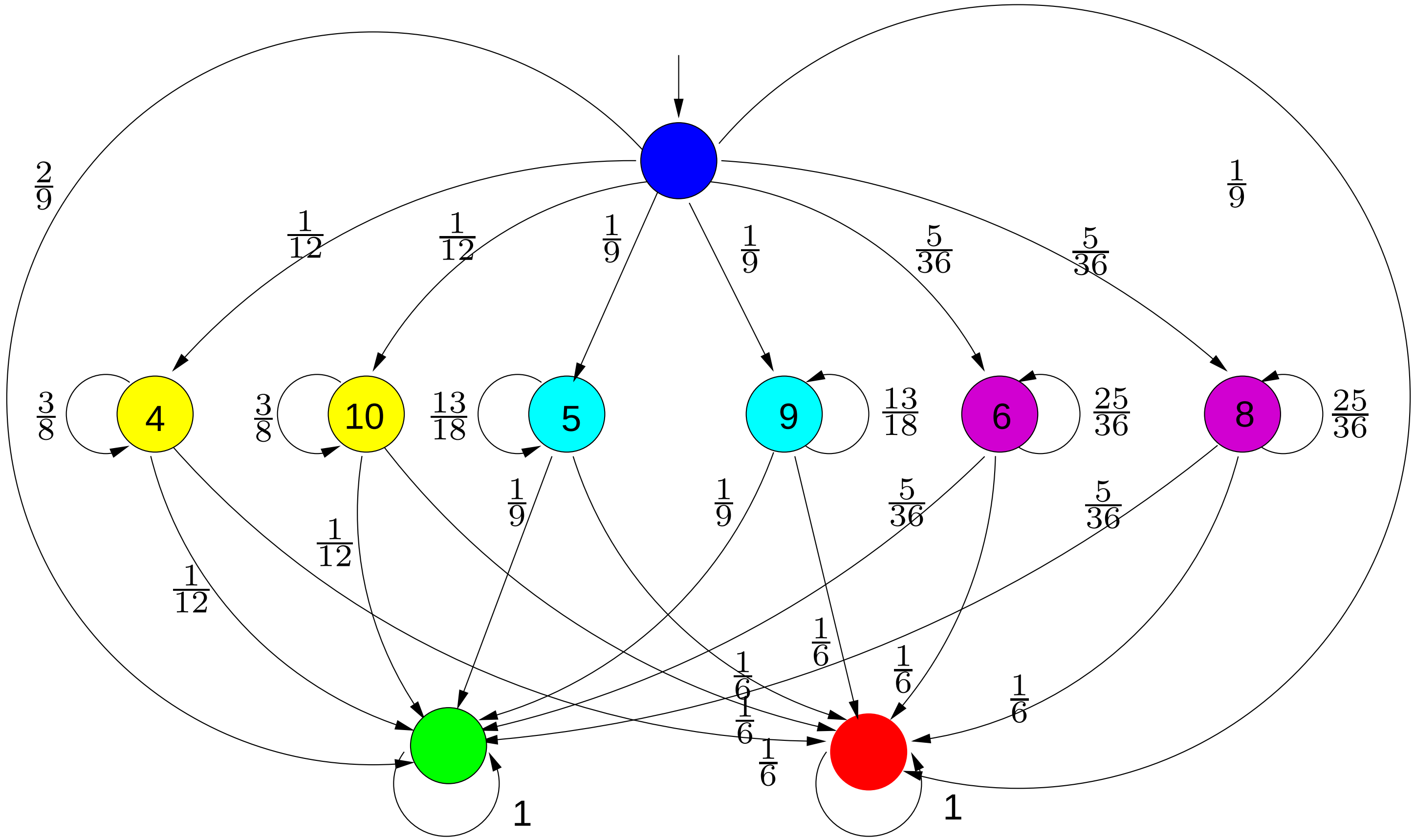
initial partitioning for the atomic propositions $AP = \{ \mathit{loss} \}$

A first refinement



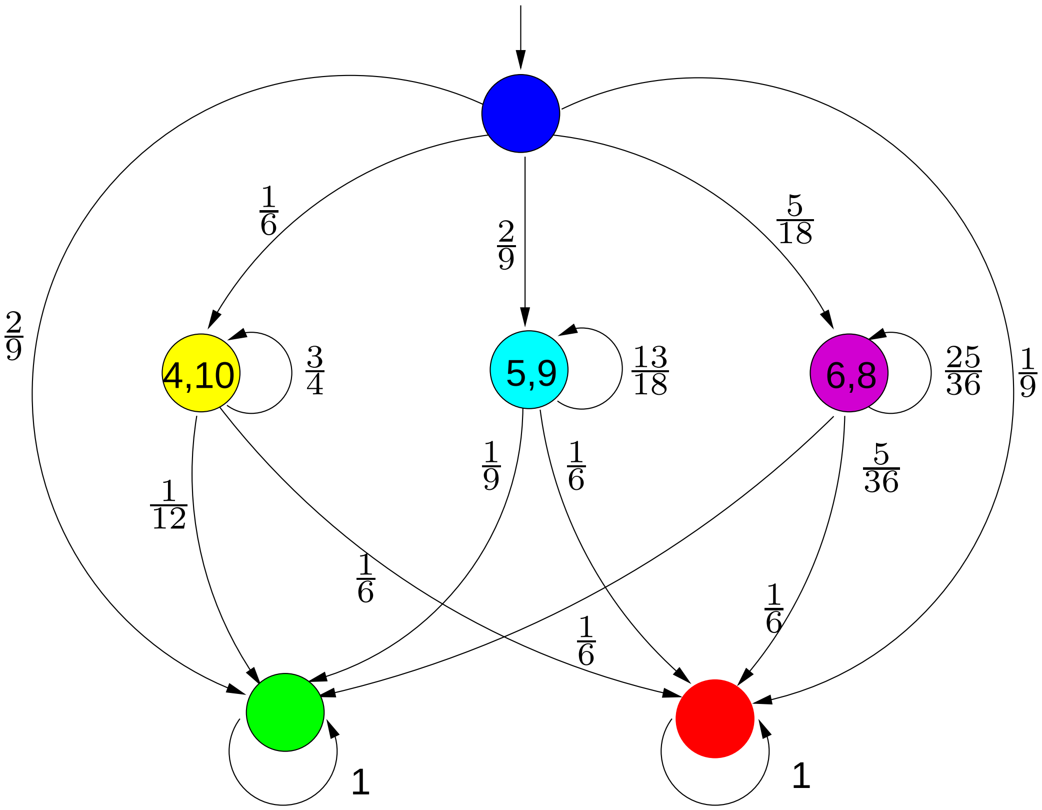
refine (“split”) with respect to the set of red states

A second refinement



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

Quotient DTMC



IEEE 802.11 group communication protocol

	original CTMC			lumped CTMC		red. factor	
OD	states	transitions	ver. time	blocks	lump + ver. time	states	time
4	1125	5369	121.9	71	13.5	**15.9**	**9.00**
12	37349	236313	7180	1821	642	**20.5**	**11.2**
20	231525	1590329	50133	10627	5431	**21.8**	**9.2**
28	804837	5750873	195086	35961	24716	**22.4**	**7.9**
36	2076773	15187833	5103900	91391	77694	**22.7**	**6.6**
40	3101445	22871849	7725041	135752	127489	**22.9**	**6.1**

all verification times concern timed reachability properties

BitTorrent-like P2P protocol

			symmetry reduction				
original CTMC			reduced CTMC			red. factor	
N	states	ver. time	states	red. time	ver. time	states	time
2	1024	5.6	528	12	2.9	**1.93**	0.38
3	32768	410	5984	100	59	**5.48**	2.58
4	1048576	22000	52360	360	820	**20.0**	18.3

			bisimulation minimisation				
original CTMC			lumped CTMC			red. factor	
N	states	ver. time	blocks	lump time	ver. time	states	time
2	1024	5.6	56	1.4	0.3	**18.3**	3.3
3	32768	410	252	170	1.3	**130**	2.4
4	1048576	22000	792	10200	4.8	**1324**	2.2

bisimulation may reduce a factor 66 after (manual) symmetry reduction

Content of this lecture

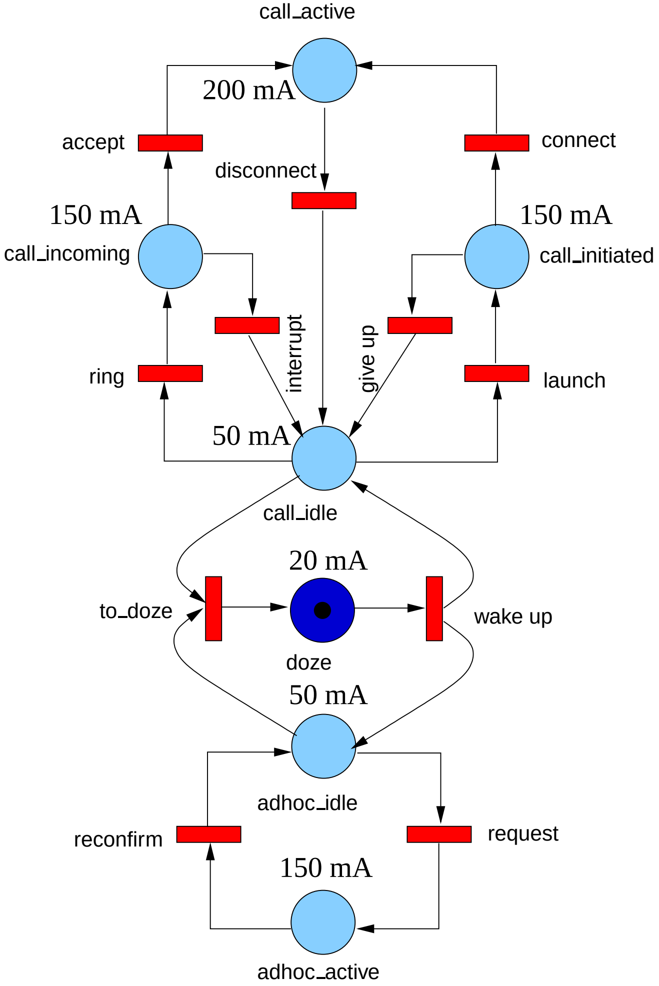
- Continuous Stochastic Logic
  - syntax, semantics, examples
- CSL model checking
  - basic algorithms and complexity
- Bisimulation
  - definition, minimization algorithm, examples

$\Rightarrow$ Priced continuous-time Markov chains
  - motivation, definition, some properties

Power consumption in mobile ad-hoc networks

- Single battery-powered mobile phone with ad-hoc traffic
- Two types of traffic: ad-hoc traffic and ordinary calls
  - offer transmission capabilities for data transfer between third parties (altruism)
  - normal call traffic
- Prices are used to model power consumption
  - in *doze* mode (20 mA), calls can neither be made nor received
  - active calls are assumed to consume 200 mA
  - ad-hoc traffic and call handling takes 120 mA; idle mode costs 50 mA
  - total battery capacity is 750 mAh; price equals one mA

A priced stochastic Petri net model



transition	mean time (in min)	rate (per h)
accept	20	180
connect	10	360
disconnect	4	15
doze	5	12
give up	1	60
interrupt	1	60
launch	80	0.75
reconfirm	4	15
request	10	6
ring	80	0.75
wake up	16	3.75

Required properties

- The probability to receive a call within 24 hours exceeds 0.23
- The probability to receive a call while having consumed at most 80% power exceeds 0.99
- The probability to launch a call before consuming at most 80% power within 24 hours – while using the phone only for ad-hoc transfer beforehand – exceeds 0.78

Priced continuous-time Markov chains

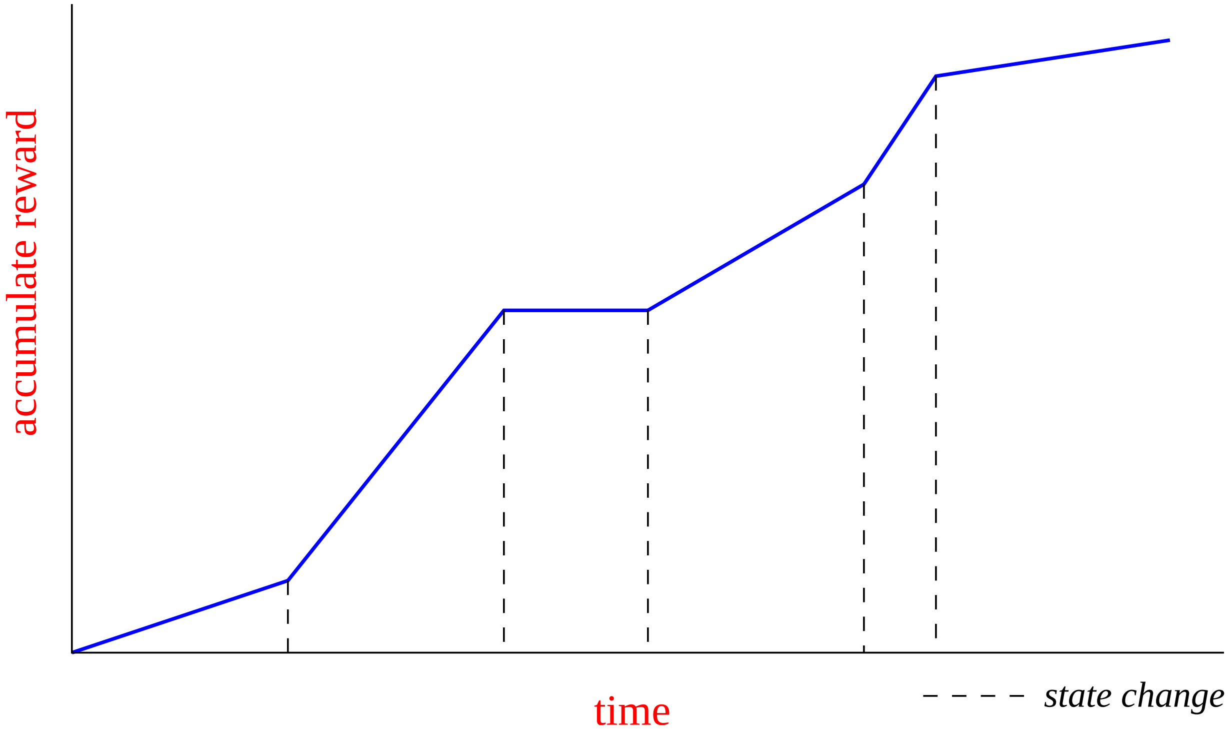
A CMRM is a triple $(S, \mathbf{R}, L, \rho)$ where:

- S is a set of states, $\mathbf{R}$ a rate matrix and L a labelling (as before)
- $\rho : S \rightarrow \mathbb{R}_{\geqslant 0}$ is a price function

Interpretation:

- Staying t time units in state s costs $\rho(s) \cdot t$

Cumulating price



Time- and cost-bounded reachability

- In $\geqslant$ 92% of the cases, a goal state is reached with *cost at most 62*:

$$\mathcal{P}_{\geqslant 0.92} \left(\neg \, \textit{illegal} \;\; \mathsf{U}_{\leqslant 62} \;\; \textit{goal} \right)$$

- . . . . . . within 133.4 time units: $\mathcal{P}_{\geqslant 0.92} \left(\neg \, \textit{illegal} \;\; \mathsf{U}^{\leqslant 133.4}_{\leqslant 62} \;\; \textit{goal} \right)$

- Possible to put constraints on:
  - the *likelihood* with which certain behaviours occur,
  - the *time frame* in which certain events should happen, and
  - the *prices* (or: rewards) that are allowed to be made.

Checking time- and cost-bounded reachability

- $s \models \mathbb{P}_L(\Phi \, \mathsf{U}^I_J \, \Psi)$ if and only if $\Pr\{s \models \Phi \, \mathsf{U}^I_J \, \Psi\} \in L$
- For $I = [0, t]$ and $J = [0, r]$, $\Pr\{s \models \Phi \, \mathsf{U}^{\leqslant t}_{\leqslant r} \Psi\}$ is the least solution of:
  - 1 if $s \models \Psi$
  - if $s \models \Phi$ and $s \not\models \Psi$:

$$\int_{K(s)} \sum_{s' \in S} \mathbf{R}(s, s') \cdot e^{-r(s) \cdot x} \cdot \Pr\{s' \models \Phi \, \mathsf{U}^{\leqslant t-x}_{\leqslant r-\rho(s)\cdot x} \, \Psi\} \, dx$$

    where $K(s) = \{ \, x \in I \mid \rho(s) \cdot x \in J \, \}$ is subset of I whose price lies in J

  - 0 otherwise

Duality: model transformation

- Key concept: exploit duality of time advancing and price increase

- The dual of an MRM $\mathcal{C}$ with $\rho(s) > 0$ into MRM $\mathcal{C}^*$:

$$\mathbf{R}^*(s, s') \;=\; \frac{\mathbf{R}(s, s')}{\rho(s)} \quad \text{and} \quad \rho^*(s) \;=\; \frac{1}{\rho(s)}$$

  state space S and the state-labelling L in $\mathcal{C}$ are unaffected

- So, accelerate state s if $\rho(s) < 1$ and slow it down if $\rho(s) > 1$

Duality theorem

- Transform any state-formula by swapping price and time bounds:

$$\left(\Phi \, \mathsf{U}_{J}^{I} \, \Psi\right) * \;=\; \Phi^{*} \, \mathsf{U}_{I}^{J} \, \Psi^{*}$$

- Duality theorem: $\underbrace{s \models \mathbb{P}_L \left(\Phi \, \mathsf{U}_{J}^{I} \, \Psi\right)}_{\text{in } \mathcal{C}}$ iff $\underbrace{s \models \mathbb{P}_L \left(\Phi^{*} \, \mathsf{U}_{I}^{J} \, \Psi^{*}\right)}_{\text{in } \mathcal{C}^{*}}$

$\Rightarrow$ $\boxed{\text{Verifying } \mathsf{U}_{J} \text{ (in } \mathcal{C}\text{) is identical to model-checking } \mathsf{U}^{J} \text{ (in } \mathcal{C}^{*}\text{)}}$

Proof sketch

$$\Pr_{\mathcal{C}^*}(s \models \Diamond^{\leqslant c}_{\leqslant t} G)$$

$=$ (* for $s \notin G$ *)

$$\int_{K^*} \sum_{s' \in S} \mathbf{R}^*(s, s') \cdot e^{-r^*(s) \cdot x} \cdot \Pr_{\mathcal{C}^*}\left(s' \models \Diamond^{\leqslant c \ominus x}_{\leqslant t \ominus \rho^*(s) \cdot x} G\right) \, dx$$

$=$ (* substituting $y = \frac{x}{\rho(s)}$ *)

$$\int_{K} \sum_{s' \in S} \mathbf{R}(s, s') \cdot e^{-r(s) \cdot y} \cdot \Pr_{\mathcal{C}^*}\left(s' \models \Diamond^{\leqslant c \ominus \rho(s) \cdot y}_{\leqslant t \ominus y} G\right) \, dy$$

$=$ (* $\mathcal{C}$ and $\mathcal{C}^*$ have same digraph, equation system has unique solution *)

$$\int_{K} \sum_{s' \in S} \mathbf{R}(s, s') \cdot e^{-r(s) \cdot y} \cdot \Pr_{\mathcal{C}}\left(s' \models \Diamond^{\leqslant c \ominus \rho(s) \cdot y}_{\leqslant t \ominus y} G\right) \, dy$$

$=$ (* $s \notin G$ *)

$$\Pr_{\mathcal{C}^*}\left(s \models \Diamond^{\leqslant t}_{\leqslant c} G\right)$$

Reduction to transient rate probabilities

Consider the formula $\Phi \, \mathsf{U}^{\leqslant t}_{\leqslant c} \, \Psi$ on MRM $\mathcal{C}$

- Approach: *transform* the MRM $\mathcal{C}$ as follows
  - make all Ψ-states and all $\neg \, (\Phi \, \vee \, \Psi)$-states absorbing
  - equip all these absorbing states with price 0

- Theorem: $\underbrace{s \models \mathbb{P}_J(\Phi \, \mathsf{U}^{\leqslant t}_{\leqslant c} \, \Psi)}_{\text{in MRM } \mathcal{C}}$ iff $\underbrace{s \models \mathbb{P}_J(\Diamond^{=t}_{\leqslant c} \, \Psi)}_{\text{in MRM } \mathcal{C}'}$

- This amounts to compute the transient rate distribution in $\mathcal{C}'$

$\Rightarrow$ Algorithms to compute this measure are not widespread!

A discretization approach

- *Discretise* both time and accumulated price as (small) d
  - probability of > 1 transition in d time-units is negligible (Tijms & Veldman 2000)

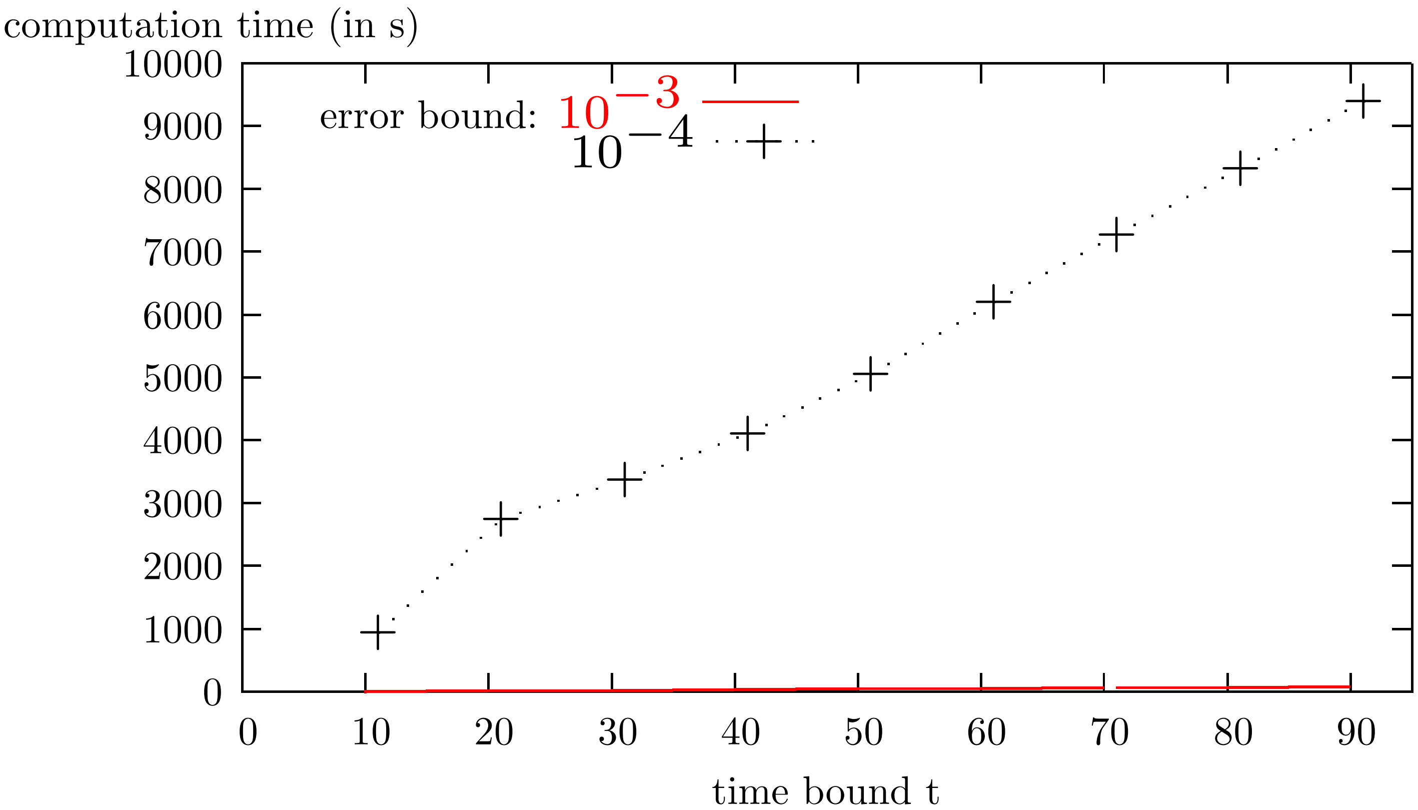
- $$\Pr(s \models \Diamond^{[t,t]}_{\leqslant c} \Psi) \approx \sum_{s' \models \Psi} \sum_{k=1}^{c/d} F^{t/d}(s', k) \cdot d$$

- Initialization: $F^1(s,k) = 1/d$ if $(s,k) = (s_0, \underline{\rho}(s_0))$, and 0 otherwise

- $$F^{j+1}(s,k) = \underbrace{F^j(s, k{-}\rho(s)) \cdot (1{-}r(s) \cdot d)}_{\text{be in state } s \text{ at epoch } j} + \sum_{s' \in S} \underbrace{F^j(s', k{-}\rho(s')) \cdot \mathbf{R}(s', s) \cdot d}_{\text{be in } s' \text{ at epoch } j}$$

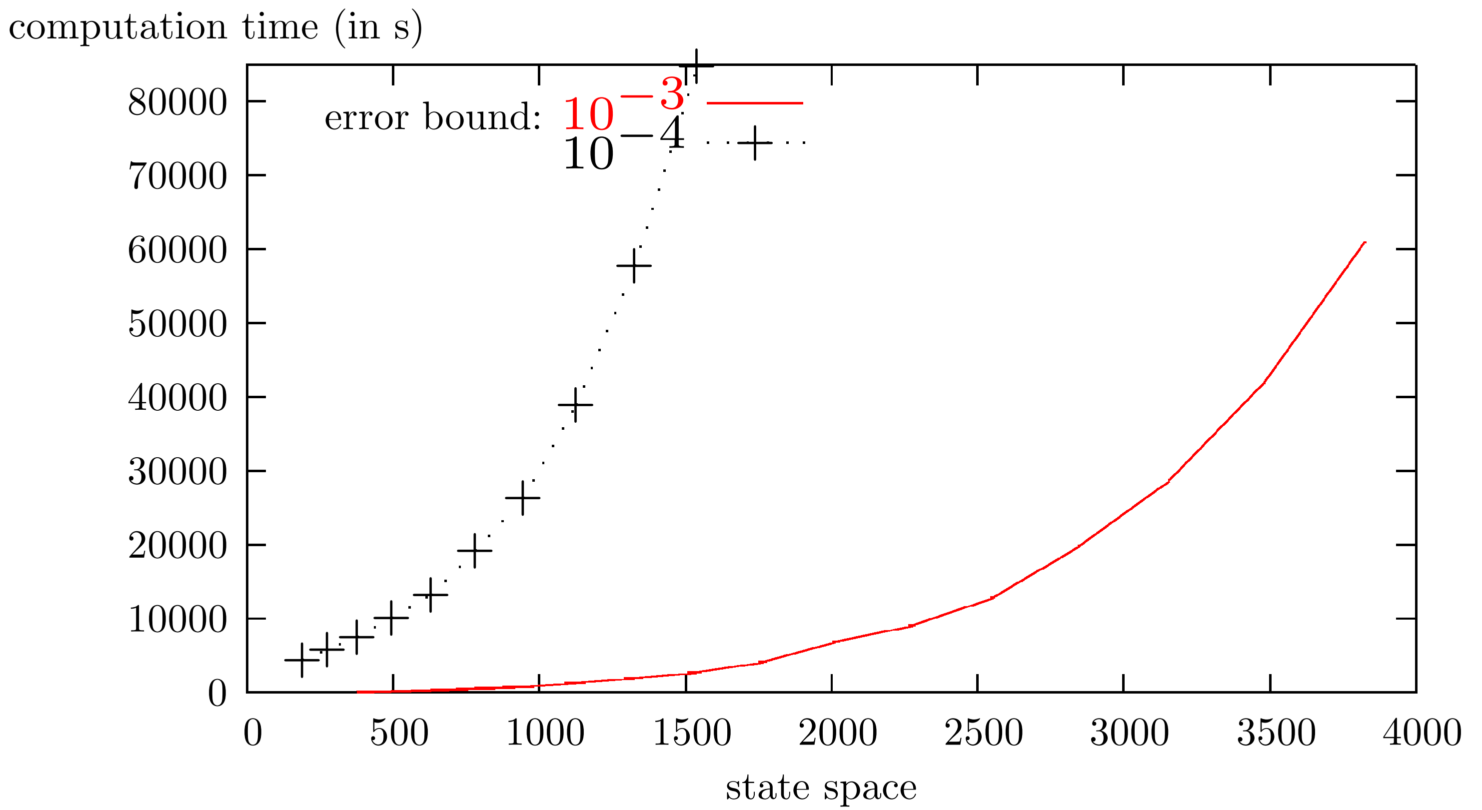
- Time complexity: $\mathcal{O}(|S|^3 \cdot t^2 \cdot d^{-2})$ (for all states)

Discretization



about 300 states; error bound not known

Discretization



Perspectives

- Linear real-time specifications (MTL, timed automata)
- Aggressive abstraction techniques
- Counterexample generation
- Continuous-time Markov decision processes
- Parametric model checking
- Infinite-state model checking
- . . . . . .

CTMC model checking

- . . . . . . is a mature automated technique
- . . . . . . has a broad range of applications
- . . . . . . is supported by powerful software tools
- . . . . . . extendible to prices
- . . . . . . supported by aggressive abstraction

more information: `www.mrmc-tool.org`

- CTMC model checking
  - CSL: [Baier, Haverkort, Hermanns & Katoen, IEEE Trans. Softw. Eng., 2003]
  - linear timed specifications: [Chen, Han, Katoen & Mereacre, LICS 2009]
- Bisimulation minimization
  - [Derisavi, Hermanns & Sanders, IPL 2005], [Valmari & Franceschinis, TACAS 2010]
  - [Katoen, Kemna, Zapreev & Jansen, TACAS 2007]
- Priced continuous-time Markov chain model checking
  - [Baier, Haverkort, Hermanns & Katoen, ICALP 2000]
  - [Baier, Cloth, Haverkort, Hermanns & Katoen, DSN 2005/FMSD 2010]
- CTMC abstraction
  - 3-valued abstraction: [Katoen, Klink, Leucker & Wolf, CONCUR 2008]
  - compositional abstraction: [Katoen, Klink and Neuhäusser, FORMATS 2009]