Probabilistic Model Checking

Lecture 9
Model Checking for CTMCs

Alessandro Abate

Department of Computer Science
University of Oxford

Overview

- CSL model checking
  - basic algorithm
  - untimed properties
  - time-bounded until
  - the S (steady-state) operator

- Rewards
  - reward structures for CTMCs
  - properties: extension of CSL
  - model checking

CSL: Continuous Stochastic Logic

- CSL syntax:

ψ is true with probability ~p

  - $\phi ::= \text{true} \mid a \mid \phi \wedge \phi \mid \neg\phi \mid P_{\sim p}\,[\psi] \mid S_{\sim p}\,[\phi]$ (state formulae)
  - $\psi ::= X\,\phi \quad \mid \quad \phi\, U^{I}\, \phi$ (path formulae)

"next"

"time bounded until"

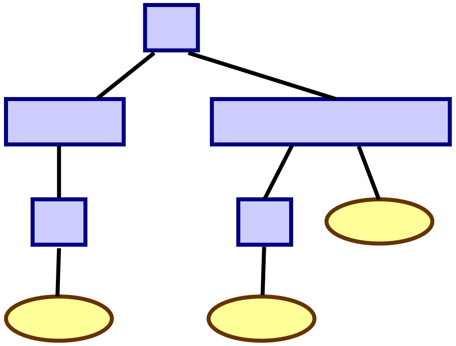
in the "long run" ϕ is true with probability ~p

  - where a is an atomic proposition, I an interval of $\mathbb{R}_{\geq 0}$, $p \in [0,1]$ and $\sim \in \{<,>,\leq,\geq\}$

CSL model checking for CTMCs

- Algorithm for CSL model checking [BHHK03]
  - inputs: CTMC $C=(S,s_{init},\mathbf{R},L)$, CSL formula ϕ
  - output: $Sat(\phi) = \{ s \in S \mid s \vDash \phi \}$, the set of states satisfying ϕ
- Often, also consider quantitative results
  - e.g. compute result of $P_{=?}$ [$F^{[0,t]}$ minimum] for $0 \leq t \leq 100$

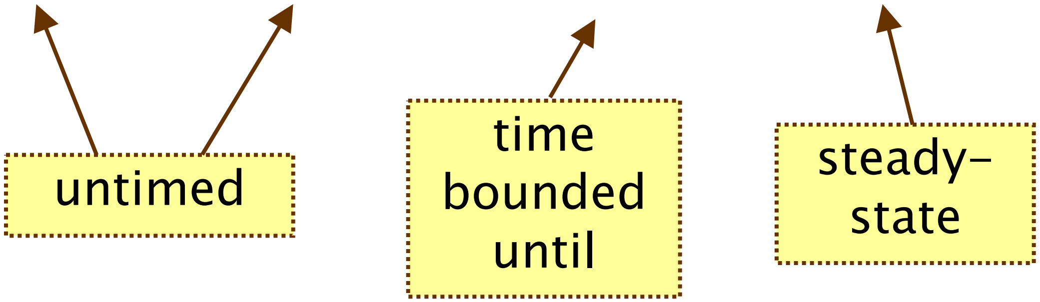
- Basic algorithm similar to PCTL for DTMCs
  - proceeds by induction on parse tree of ϕ
- For the non-probabilistic operators:
  - $Sat(true) = S$
  - $Sat(a) = \{ s \in S \mid a \in L(s) \}$
  - $Sat(\neg\phi) = S \setminus Sat(\phi)$
  - $Sat(\phi_1 \wedge \phi_2) = Sat(\phi_1) \cap Sat(\phi_2)$



CSL model checking for CTMCs

- Main task: computing probabilities for $P_{\sim p}\,[\cdot]$ and $S_{\sim p}\,[\cdot]$
  - $\phi ::= \text{true} \mid a \mid \phi \wedge \phi \mid \neg\phi \mid$

    $$P_{\sim p}\,[\,X\ \phi\,] \mid P_{\sim p}\,[\,\phi\ U\ \phi\,] \mid P_{\sim p}\,[\,\phi\ U^{I}\ \phi\,] \mid S_{\sim p}\,[\,\phi\,]$$



untimed

time bounded until

steady-state

  - where $\phi_1\ U\ \phi_2 \equiv \phi_1\ U^{[0,\infty)}\ \phi_2$

Untimed properties

- Untimed properties can be verified on the embedded DTMC
  - properties of the form: $P_{\sim p}$ [X ϕ] or $P_{\sim p}$ [ϕ_1 U ϕ_2]
  - use algorithms for checking PCTL against DTMCs

- Certain qualitative time-bounded until formulae can also be verified on the embedded DTMC
  - for any (non-empty) interval I

    $s \vDash P_{\sim 0} [\ \phi_1\ U^{I}\ \phi_2\ ]$ if and only if $s \vDash P_{\sim 0} [\phi_1\ U^{[0,\infty)}\ \phi_2\ ]$

  - can use precomputation algorithm Prob0

Model checking – Time-bounded until

- Compute $\mathrm{Prob}(s, \phi_1\ U^I\ \phi_2)$ for all states where I is an arbitrary interval of the non-negative real numbers
- Lemmas:
  - $\mathrm{Prob}(s, \phi_1\ U^I\ \phi_2) = \mathrm{Prob}(s, \phi_1\ U^{cl(I)}\ \phi_2)$

    where cl(I) denotes the closure of the interval I
  - $\mathrm{Prob}(s, \phi_1\ U^{[0,\infty)}\ \phi_2) = \mathrm{Prob}^{emb(C)}(s, \phi_1\ U\ \phi_2)$

    where emb(C) is the embedded DTMC

- Therefore, 3 remaining cases to consider:
  - $I = [0,t]$ for some $t \in \mathbb{R}_{\geq 0}$, $I = [t,t']$ for some $t \leq t' \in \mathbb{R}_{\geq 0}$ and $I = [t,\infty)$ for some $t \in \mathbb{R}_{\geq 0}$

- Two methods: 1. Integral equations; 2. Uniformisation

Time-bounded until: integral equations

- Computing the probabilities reduces to determining the least solution of the following set of integral equations
  - (note similarity to bounded until for DTMCs)
- Prob(s, ϕ_1 $U^{[0,t]}$ ϕ_2) equals
  - 1 if s $\in$ Sat(ϕ_2),
  - 0 if s $\in$ Sat($\neg\phi_1 \wedge \neg\phi_2$),
  - and otherwise equals

$$\int_0^t \sum_{s' \in S} \left(P^{emb(C)}(s,s') \cdot E(s) \cdot e^{-E(s) \cdot x} \right) \mathrm{Prob}(s', \phi_1 \ U^{[0,t-x]} \ \phi_2) \ dx$$

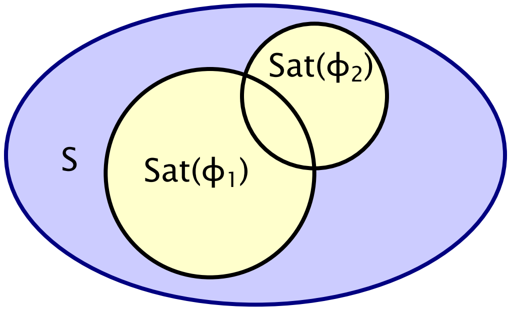
probability of moving from s to s’ at time x

probability, in state s’, of satisfying until before t-x time units elapse

- One possibility: solve these integrals numerically
  - numerical integration, e.g. trapezoidal, Simpson, Romberg
  - expensive, possible issues with numerical stability

Time-bounded until: uniformisation

- Reduction to transient analysis…



- Make all ϕ_2 states absorbing
  - from such a state $\phi_1\ U^{[0,x]}\ \phi_2$ holds with probability 1

- Make all $\neg\phi_1 \land \neg\phi_2$ states absorbing
  - from such a state $\phi_1\ U^{[0,x]}\ \phi_2$ holds with probability 0

- Formally: Construct CTMC $C[\phi_2][\neg\phi_1 \land \neg\phi_2]$
  - where for CTMC $C=(S,s_{init},\mathbf{R},L)$, let $C[\theta]=(S,s_{init},\mathbf{R}[\theta],L)$, where θ state formula, $\mathbf{R}[\theta](s,s')=\mathbf{R}(s,s')$ if $s \notin Sat(\theta)$ and 0 otherwise

Time-bounded until: uniformisation

- Problem then reduces to calculating transient probabilities of the CTMC $C[\phi_2][\neg\phi_1 \wedge \neg\phi_2]$:

$$\mathrm{Prob}(s, \phi_1\, \mathrm{U}^{[0,t]}\, \phi_2) = \sum_{s' \in \mathrm{Sat}(\phi_2)} \underline{\pi}_{s,t}^{C[\phi_2][\neg\phi_1 \wedge \neg\phi_2]}(s')$$

transient probability: starting in state s, the probability of being in state s’ at time t

Time-bounded until: uniformisation

- Can now adapt uniformisation to computing the vector of probabilities $\underline{\text{Prob}}(\phi_1\ U^{[0,t]}\ \phi_2)$
  - recall Π_t – matrix of transient probabilities: $\Pi_t(s,s')=\underline{\pi}_{s,t}(s')$
  - can be computed via uniformisation: $\Pi_t = \sum_{i=0}^{\infty} \gamma_{q\cdot t,i} \cdot \left(\mathbf{P}^{unif(C)}\right)^i$

- Combining with: $\text{Prob}(s, \phi_1\ U^{[0,t]}\ \phi_2) = \sum_{s' \in Sat(\phi_2)} \underline{\pi}_{s,t}^{C[\phi_2][\neg\phi_1 \wedge \neg\phi_2]}(s')$

$$\begin{aligned}\underline{\text{Prob}}(\phi_1\ U^{[0,t]}\ \phi_2) &= \Pi_t^{C[\phi_2][\neg\phi_1 \wedge \neg\phi_2]} \cdot \underline{\phi_2} \\ &= \left(\sum_{i=0}^{\infty} \gamma_{q\cdot t,i} \cdot \left(\mathbf{P}^{unif(C[\phi_2][\neg\phi_1 \wedge \neg\phi_2])}\right)^i\right) \cdot \underline{\phi_2} \\ &= \sum_{i=0}^{\infty} \left(\gamma_{q\cdot t,i} \cdot \left(\mathbf{P}^{unif(C[\phi_2][\neg\phi_1 \wedge \neg\phi_2])}\right)^i \cdot \underline{\phi_2}\right)\end{aligned}$$

  - (note analogy: for transient analysis, we post-multiplied from initial vector, now we pre-multiply with Sat-indicator vector)

Time-bounded until: uniformisation

- Have shown that we can calculate the probabilities as:

$$\underline{\mathrm{Prob}}(\phi_1\ \mathrm{U}^{[0,t]}\ \phi_2) = \sum_{i=0}^{\infty}\left(\gamma_{q\cdot t,i}\cdot \left(\mathbf{P}^{\mathrm{unif}(\mathbf{C}[\phi_2][\neg\phi_1\wedge\neg\phi_2])}\right)^{\mathrm{i}} \cdot \underline{\phi_2} \right)$$

- Infinite summation can be truncated using techniques by Fox and Glynn [FG88]

- Can compute iteratively to avoid matrix powers:

$$\left(\mathbf{P}^{\mathrm{unif}(\mathrm{C})}\right)^{0} \cdot \underline{\phi_2} = \underline{\phi_2}$$

$$\left(\mathbf{P}^{\mathrm{unif}(\mathrm{C})}\right)^{\mathrm{i+1}} \cdot \underline{\phi_2} = \mathbf{P}^{\mathrm{unif}(\mathrm{C})} \cdot \left(\left(\mathbf{P}^{\mathrm{unif}(\mathrm{C})}\right)^{\mathrm{i}} \cdot \underline{\phi_2} \right)$$

  - (note slight imprecision in Greek gamma var, 1st equation)

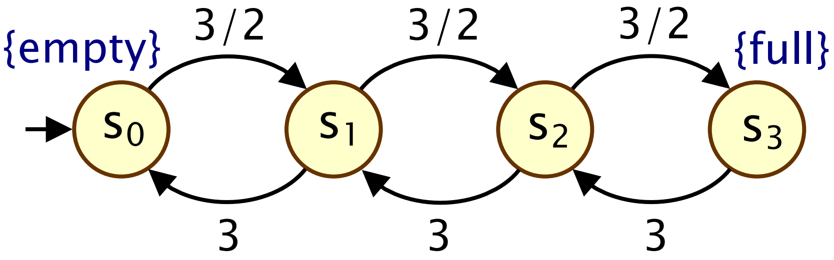
Time−bounded until − Example

- $P_{>0.65}$ [$F^{[0,7.5]}$ full] ≡ $P_{>0.65}$ [true $U^{[0,7.5]}$ full]
  - “probability of the queue becoming full within 7.5 time units”
- State s_3 satisfies full and no states satisfy ¬true
  - in C[full][¬true ∧¬ full] only state s_3 made absorbing

$$\begin{bmatrix} 2/3 & 1/3 & 0 & 0 \\ 2/3 & 0 & 1/3 & 0 \\ 0 & 2/3 & 0 & 1/3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

matrix of unif(C[full][¬true ∧¬full]) with uniformisation rate $\max_{s \in S} E(s) = 4.5$ $(=3+3/2)$

s_3 made absorbing



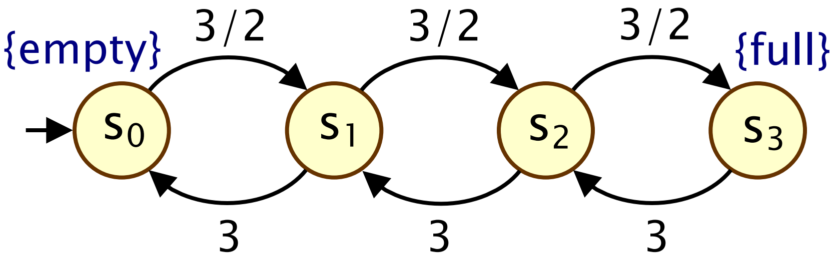
Time-bounded until – Example

- Computing the summation of matrix-vector multiplications

$$\underline{Prob}(\phi_1 \, U^{[0,t]} \, \phi_2) = \sum_{i=0}^{\infty} \left(\gamma_{q \cdot t, i} \cdot \left(\mathbf{P}^{unif(C[\phi_2][\neg\phi_1 \wedge \neg\phi_2])} \right)^i \cdot \underline{\phi_2} \right)$$

  - yields $\underline{Prob}(F^{[0,7.5]} \text{ full }) \approx [0.6482, 0.6823, 0.7811, 1]$

- $P_{>0.65}[F^{[0,7.5]} \text{ full }]$ satisfied in states s_1, s_2 and s_3



Time-bounded until – $P_{\sim p}\,[\phi_1\ U^{[t,t']}\ \phi_2]$

- In this case the computation can be split into two parts:
- 1. Probability of remaining in ϕ_1 states until time t
  - can be computed as transient probabilities on the CTMC where states satisfying $\neg\phi_1$ have been made absorbing
- 2. Probability of reaching a ϕ_2 state, while remaining in states satisfying ϕ_1, within the time interval [0,t'–t]
  - i.e. computing Prob($\phi_1\ U^{[0,t'-t]}\ \phi_2$)

$$\mathrm{Prob}(s, \phi_1\ U^{[t,t']}\ \phi_2) = \sum_{s'\in \mathrm{Sat}(\phi_1)} \underline{\pi}^{C[\neg\phi_1]}_{s,t}(s') \cdot \mathrm{Prob}(s', \phi_1\ U^{[0,t'-t]}\ \phi_2)$$

sum over states satisfying ϕ_1

Probability of reaching state s' at time t and satisfying ϕ_1 up until this point

probability $\phi_1\ U^{[0,t'-t]}\ \phi_2$ holds in s'

Time–bounded until – $P_{\sim p} [\phi_1 \, U^{[t,t']} \, \phi_2]$

- Let $\mathrm{Prob}_{\phi_1}(s, \phi_1 U^{[0,t'-t]} \phi_2) = \mathrm{Prob}(s, \phi_1 U^{[0,t'-t]} \phi_2)$ if $s \in \mathrm{Sat}(\phi_1)$, and 0 otherwise
- From the previous slide we have:

$$\underline{\mathrm{Prob}}(\phi_1 \, U^{[t,t']} \, \phi_2) = \Pi_t^{C[\neg\phi_1]} \cdot \underline{\mathrm{Prob}}_{\phi_1}(\phi_1 \, U^{[0,t'-t]} \, \phi_2)$$

$$= \left(\sum_{i=0}^{\infty} \gamma_{q \cdot t, i} \cdot \left(\mathbf{P}^{\mathrm{unif}(C[\neg\phi_1])} \right)^i \right) \cdot \underline{\mathrm{Prob}}_{\phi_1}(\phi_1 \, U^{[0,t'-t]} \, \phi_2)$$

$$= \sum_{i=0}^{\infty} \left(\gamma_{q \cdot t, i} \cdot \left(\mathbf{P}^{\mathrm{unif}(C[\neg\phi_1])} \right)^i \cdot \underline{\mathrm{Prob}}_{\phi_1}(\phi_1 \, U^{[0,t'-t]} \, \phi_2) \right)$$

  - summation can be truncated using Fox and Glynn [FG88]
  - can compute iteratively (only scalar and matrix-vector operations)

Time-bounded until – $P_{\sim p}\,[\phi_1\ U^{[t,\infty)}\ \phi_2]$

- Letting $\text{Prob}_{\phi 1}(s,\ \phi_1 U^{[0,\infty)} \phi_2) = \text{Prob}(s,\ \phi_1 U^{[0,\infty)} \phi_2)$ if $s \in Sat(\phi_1)$ and 0 otherwise, we have:

$$\begin{aligned}
\underline{\text{Prob}}(\phi_1\ U^{[t,\infty]}\ \phi_2) &= \Pi_t^{C[\neg\phi_1]} \cdot \underline{\text{Prob}}_{\phi_1}^{emb(C)}(\phi_1\ U\ \phi_2) \\
&= \left(\sum_{i=0}^{\infty} \gamma_{q \cdot t,i} \cdot \left(\mathbf{P}^{unif(C[\neg\phi_1])} \right)^i \right) \cdot \underline{\text{Prob}}_{\phi_1}^{emb(C)}(\phi_1\ U\ \phi_2) \\
&= \sum_{i=0}^{\infty} \left(\gamma_{q \cdot t,i} \cdot \left(\mathbf{P}^{unif(C[\neg\phi_1])} \right)^i \cdot \underline{\text{Prob}}_{\phi_1}^{emb(C)}(\phi_1\ U\ \phi_2) \right)
\end{aligned}$$

  - summation can be truncated using Fox and Glynn [FG88]
  - can compute iteratively (only scalar and matrix-vector operations)

Model Checking – $S_{\sim p}[\,\phi\,]$

- A state s satisfies the formula $S_{\sim p}[\phi]$ if $\Sigma_{s' \models \phi}\ \underline{\pi}^{C}_{s}(s') \sim p$
  - $\underline{\pi}^{C}_{s}(s')$ is probability, having started in state s, of being in state s' in the long run
- Thus reduces to computing and then summing steady-state probabilities for the CTMC (recall results earlier)

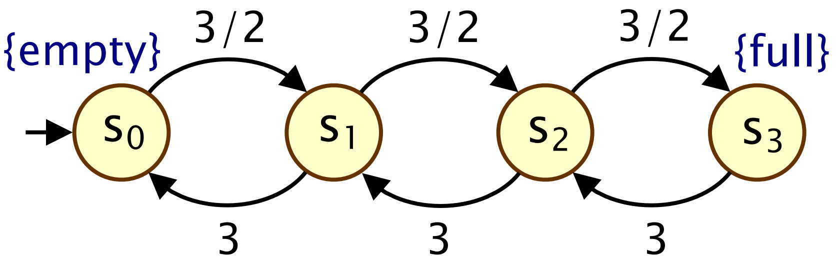
- If CTMC is irreducible:
  - solution of one system of linear equations
- If CTMC is reducible:
  - determine set of BSCCs for the CTMC
  - solve two systems of linear equations, for each BSCC T:

1. one to obtain the vector $\underline{\text{ProbReach}}^{emb(C)}(T)$
2. the other to compute the steady state probabilities $\underline{\pi}^{T}$ for T

$S_{\sim p}$ [ϕ] – Example

- $S_{<0.1}$[full]
- CTMC is irreducible (comprises a single BSCC)
  - steady state probabilities independent of starting state
  - can be computed by solving $\underline{\pi} \cdot \mathbf{Q} = 0$ and $\Sigma\, \underline{\pi}(s) = 1$

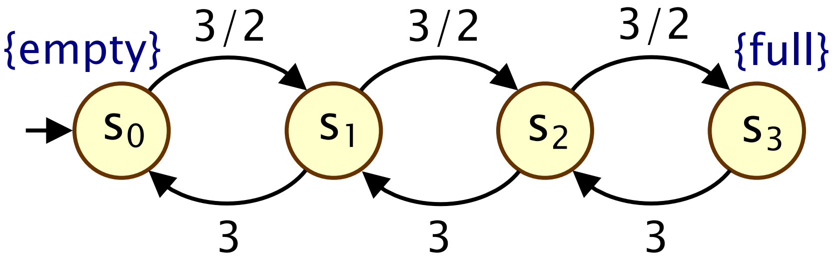
$$\mathbf{Q} = \begin{bmatrix} -3/2 & 3/2 & 0 & 0 \\ 3 & -9/2 & 3/2 & 0 \\ 0 & 3 & -9/2 & 3/2 \\ 0 & 0 & 3 & -3 \end{bmatrix}$$



$S_{\sim p}[\ \phi\ ]$ – Example

$$\begin{array}{rcrcrcrcl}
-3/2 \cdot \underline{\pi}(s_0) & + & 3 \cdot \underline{\pi}(s_1) & & & & & = & 0 \\
3/2 \cdot \underline{\pi}(s_0) & - & 9/2 \cdot \underline{\pi}(s_1) & + & 3 \cdot \underline{\pi}(s_2) & & & = & 0 \\
 & & 3/2 \cdot \underline{\pi}(s_1) & - & 9/2 \cdot \underline{\pi}(s_2) & + & 3 \cdot \underline{\pi}(s_3) & = & 0 \\
 & & & & 3/2 \cdot \underline{\pi}(s_2) & - & 3 \cdot \underline{\pi}(s_3) & = & 0
\end{array}$$

$$\underline{\pi}(s_0) + \underline{\pi}(s_1) + \underline{\pi}(s_2) + \underline{\pi}(s_3) = 1$$



- solution: $\underline{\pi} = [\ 8/15,\ 4/15,\ 2/15,\ 1/15\ ]$
- $\Sigma_{s' \vDash Sat(full)}\ \underline{\pi}\,(s') = 1/15 < 0.1$
- so all states satisfy $S_{<0.1}[\text{ full }]$

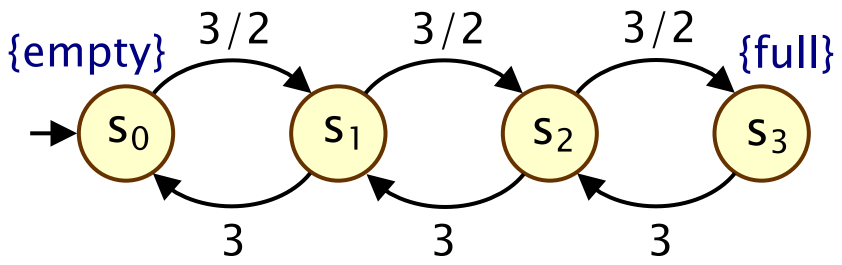
Rewards (or costs)

- Like DTMCs, we can augment CTMCs with rewards
  - real-valued quantities assigned to states and/or transitions
  - can be interpreted in two ways: instantaneous/cumulative
  - properties considered: expected value of rewards
  - formal property specifications as an extension of CSL

- For a CTMC $(S,s_{init},\mathbf{R},L)$, a reward structure is a pair $(\underline{\rho},\iota)$
  - $\underline{\rho} : S \rightarrow \mathbb{R}_{\geq 0}$ is a vector of state rewards
  - $\iota : S \times S \rightarrow \mathbb{R}_{\geq 0}$ is a matrix of transition rewards

- For cumulative reward-based properties of CTMCs
  - state rewards interpreted as rate at which reward is gained
  - if the CTMC remains in state s for $t \in \mathbb{R}_{>0}$ time units, a reward of $t \cdot \underline{\rho}(s)$ is acquired

Reward structures – Examples



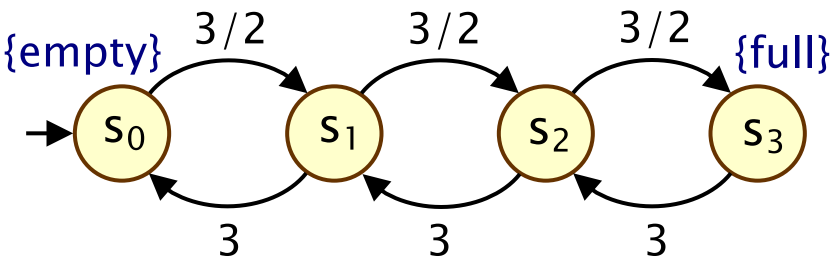
instantaneous

- Example: "size of message queue"
  - $\underline{\rho}(s_i)=i$ and $\iota(s_i,s_j)=0\ \forall i,j$

cumulative

- Example: "time for which queue is not full"
  - $\underline{\rho}(s_i)=1$ for $i<3$, $\underline{\rho}(s_3)=0$ and $\iota(s_i,s_j)=0\ \forall i,j$

Reward structures – Examples



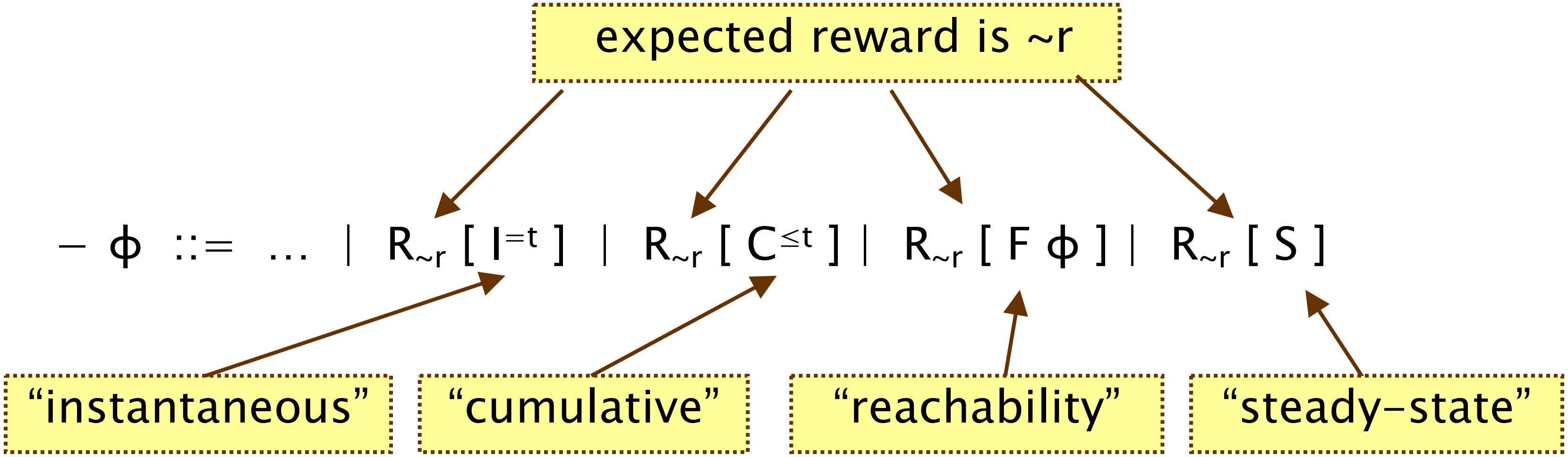
cumulative

- Example: "number of requests served"

$$\rho = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \text{and} \quad \iota = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

CSL and rewards

- PRISM extends CSL to incorporate reward-based properties
  - adds R operator, as in PCTL



expected reward is ~r

- $\phi ::= \ \dots \ \mid R_{\sim r} [\ I^{=t}\ ] \mid R_{\sim r} [\ C^{\leq t}\ ] \mid R_{\sim r} [\ F\ \phi\ ] \mid R_{\sim r} [\ S\ ]$

"instantaneous" "cumulative" "reachability" "steady-state"

- where $r,t \in \mathbb{R}_{\geq 0}$, $\sim \in \{<,>,\leq,\geq\}$

- $R_{\sim r}$ [·] means "the expected value of · satisfies ~r"

Types of reward formulae

- Instantaneous: $R_{\sim r}$ [$I^{=t}$]
  - the expected value of the reward at time instant t is ~r
  - "the expected queue size after 6.7 seconds is at most 2"
- Cumulative: $R_{\sim r}$ [$C^{\leq t}$]
  - the expected reward cumulated up to time instant t is ~r
  - "the expected requests served within the first 4.5 seconds of operation is less than 10"
- Reachability: $R_{\sim r}$ [F ϕ]
  - the expected reward cumulated before reaching ϕ is ~r
  - "the expected requests served before the queue becomes full"
- Steady-state: $R_{\sim r}$ [S]
  - the long-run average expected reward is ~r
  - "expected long-run queue size is at least 1.2"

Reward properties in PRISM

- Quantitative form:
  - e.g. $R_{=?}$ [$C^{\leq t}$]
  - what is the expected reward cumulated up to time instant t?

- Add labels to R operator to distinguish between multiple reward structures defined on the same CTMC
  - e.g. $R_{\{num\_req\}=?}$ [$C^{\leq 4.5}$]
  - "the expected number of requests served within the first 4.5 seconds of operation"
  - e.g. $R_{\{pow\}=?}$ [$C^{\leq 4.5}$]
  - "the expected power consumption within the first 4.5 seconds of operation"

Reward formula semantics

- Formal semantics of the four reward operators:

  - $s \models R_{\sim r}\ [\ I^{=t}\ ]$ $\Leftrightarrow$ $\mathrm{Exp}(s, X_{I=t}) \sim r$
  - $s \models R_{\sim r}\ [\ C^{\leq t}\ ]$ $\Leftrightarrow$ $\mathrm{Exp}(s, X_{C\leq t}) \sim r$
  - $s \models R_{\sim r}\ [\ F\ \Phi\ ]$ $\Leftrightarrow$ $\mathrm{Exp}(s, X_{F\Phi}) \sim r$
  - $s \models R_{\sim r}\ [\ S\ ]$ $\Leftrightarrow$ $\lim_{t\to\infty}(\ 1/t \cdot \mathrm{Exp}(s, X_{C\leq t})\ ) \sim r$

- Where recall:
  - Exp(s, X) denotes the expectation of the random variable $X : \mathrm{Path}(s) \to \mathbb{R}_{\geq 0}$ with respect to the probability measure $\Pr_s$

Reward formula semantics

- Definition of random variables:
  - path $\omega = s_0 t_0 s_1 t_1 s_2 \ldots$

state of ω at time t

$$X_{I=k}(\omega) = \underline{\rho}(\omega@t)$$

time spent in state s_i

time spent in state s_{j_t} before t time units have elapsed

$$X_{C\le t}(\omega) = \sum_{i=0}^{j_t-1}\left(t_i \cdot \underline{\rho}(s_i) + \iota(s_i,s_{i+1})\right) + \left(t - \sum_{i=0}^{j_t-1} t_i\right) \cdot \underline{\rho}(s_{j_t})$$

$$X_{F\phi}(\omega) = \begin{cases} 0 & \text{if } s_0 \in Sat(\phi) \\ \infty & \text{if } s_i \notin Sat(\phi) \text{ for all } i \ge 0 \\ \sum_{i=0}^{k_\phi - 1} t_i \cdot \underline{\rho}(s_i) + \iota(s_i,s_{i+1}) & \text{otherwise} \end{cases}$$

  - where $j_t = \min\{ j \mid \Sigma_{i\le j}\, t_i \ge t \}$ and $k_\phi = \min\{ i \mid s_i \vDash \phi \}$
  - (note: typo in index of first formula: I=k should be I=t)

Model checking reward formulae

- Instantaneous: $R_{\sim r}$ [$I^{=t}$]
  - reduces to transient analysis (state of the CTMC at time t)
  - use uniformisation
- Cumulative: $R_{\sim r}$ [$C^{\leq t}$]
  - extends approach for time-bounded until
  - based on uniformisation
- Reachability: $R_{\sim r}$ [F ϕ]
  - can be computed on the embedded DTMC
  - reduces to solving a system of linear equations
- Steady-state: $R_{\sim r}$ [S]
  - similar to steady-state formulae $S_{\sim r}$ [ϕ]
  - graph based analysis (compute BSCCs)
  - solve systems of linear equations (compute steady state probabilities of each BSCC)

CSL model checking complexity

- For model checking of a CTMC complexity:
  - linear in |Φ| and polynomial in |S|
  - linear in $q \cdot t_{max}$ (t_{max} is maximum finite bound in intervals, q is uniformisation rate)
- $P_{\sim p}[\Phi_1\ U^{[0,\infty)}\ \Phi_2]$, $S_{\sim p}[\Phi]$, $R_{\sim r}\ [F\ \Phi]$ and $R_{\sim r}\ [S]$
  - require solution of linear equation system of size |S|
  - can be solved with Gaussian elimination: cubic in |S|
  - precomputation algorithms (max |S| steps)
- $P_{\sim p}[\Phi_1\ U^I\ \Phi_2]$, $R_{\sim r}\ [C^{\leq t}]$ and $R_{\sim r}\ [I^{=t}]$
  - at most two iterative sequences of matrix-vector products
  - operation is quadratic in the size of the matrix, i.e. |S|
  - total number of iterations bounded by Fox and Glynn
  - the bound is linear in the size of $q \cdot t$

Summing up…

- Model checking a CSL formula ϕ on a CTMC
  - recursive: bottom-up traversal of parse tree of ϕ
- Main work: computing probabilities for P and S operators
  - untimed ($X\ \Phi$, $\Phi_1\ U\ \Phi_2$): perform on embedded DTMC
  - time-bounded until: use uniformisation-based methods, rather than more expensive solution of integral equations
  - other forms of time-bounded until, i.e. $[t_1,t_2]$ and $[t,\infty)$, reduce to two sequential computations like for [0,t]
  - S operator: summation of steady-state probabilities
- Rewards – similar to DTMCs
  - except for continuous-time accumulation of state rewards
  - extension of CSL with R operator
  - model checking of R comparable with that of P