Probabilistic Model Checking

Marta Kwiatkowska
Gethin Norman
Dave Parker

University of Oxford

Part 5 – Continuous–Time Markov Chains
Overview

- Exponential distributions

- Continuous-time Markov chains (CTMCs)
  - definition, paths, probabilities, steady-state, transient, ...

- Properties of CTMCs: The logic CSL
  - syntax, semantics, equivalences, ...

- CSL model checking
  - algorithm, examples, ...

- Costs and rewards
Exponential distribution

- Continuous random variable $X$ is exponential with parameter $\lambda>0$ if the density function is given by

$$f_X(t) = \begin{cases} \lambda \cdot e^{-\lambda t} & \text{if } t > 0 \\ 0 & \text{otherwise} \end{cases}$$

- Cumulative distribution function (P[$X \leq t$]) of $X$:

$$F_X(t) = \int_0^t \lambda \cdot e^{-\lambda x} dx = \left[-e^{-\lambda x}\right]_0^t = 1 - e^{-\lambda t}$$

- $P[X > t] = e^{-\lambda t}$

- Expectation $E[X] = \int_0^\infty x \cdot \lambda \cdot e^{-\lambda x} dx = \frac{1}{\lambda}$

- Variance $\text{Var}[X] = \frac{1}{\lambda^2}$
Exponential distribution – Examples

- The more $\lambda$ increases, the faster the c.d.f. approaches 1
Exponential distribution

- Adequate for modelling many real-life phenomena
  - failure rates
  - inter-arrival times
  - continuous process to change state

- Can approximate general distributions arbitrarily closely

- Maximal entropy if just the mean is known
  - i.e. best approximation when only mean is known
Exponential distribution – Memoryless

- **Memoryless property:** \( P[ X>t_1+t_2 | X>t_1 ] = P[ X>t_2 ] \)

- Exponential distribution is the **only** continuous distribution which is memoryless

\[
P[ X>t_1+t_2 | X>t_1 ] = \frac{P[ X>t_1+t_2 \land X>t_1 ]}{P[ X>t_1 ]} = \frac{P[ X>t_1+t_2 ]}{P[ X>t_1 ]} = e^{-\lambda \cdot (t_1+t_2)} / e^{-\lambda \cdot t_1}
\]

\[
= (e^{-\lambda \cdot t_1} \cdot e^{-\lambda \cdot t_2}) / e^{-\lambda \cdot t_1}
\]

\[
= e^{-\lambda \cdot t_2}
\]

\[
= P[ X>t_2 ]
\]

recall \( P[X>t] = e^{-\lambda \cdot t} \)
Exponential distribution – Properties

• **Minimum** of two independent exponential distributions is an exponential distribution (parameter is sum)

\[
P[\min(X_1, X_2) \leq t] = 1 - P[\min(X_1, X_2) > t]
\]

\[
= 1 - P[X_1 > t \land X_2 > t]
\]

\[
= 1 - P[X_1 > t] \cdot P[X_2 > t]
\]

\[
= 1 - e^{-\lambda_1 \cdot t} \cdot e^{-\lambda_2 \cdot t}
\]

\[
= 1 - e^{-(\lambda_1 + \lambda_2) \cdot t}
\]

\[
= 1 - P[Y > t] = P[Y \leq t]
\]

- recall \( P[X > t] = e^{-\lambda \cdot t} \)
Overview

• Exponential distributions

• **Continuous-time Markov chains (CTMCs)**
  – definition, paths, probabilities, steady-state, transient, …

• Properties of CTMCs: The logic CSL
  – syntax, semantics, equivalences, …

• CSL model checking
  – algorithm, examples, …

• Costs and rewards
Continuous–time Markov chains

- **Continuous–time Markov chains (CTMCs)**
  - labelled transition systems augmented with rates
  - discrete states
  - *continuous* time–steps
  - delays *exponentially distributed*

- **Suited to modelling:**
  - reliability models
  - control systems
  - queueing networks
  - biological pathways
  - chemical reactions
  - ...
Continuous–time Markov chains

- Formally, a CTMC $C$ is a tuple $(S, s_{\text{init}}, R, L)$ where:
  - $S$ is a finite set of states ("state space")
  - $s_{\text{init}} \in S$ is the initial state
  - $R : S \times S \to \mathbb{R}_{\geq 0}$ is the transition rate matrix
  - $L : S \to 2^{\text{AP}}$ is a labelling with atomic propositions

- Transition rate matrix assigns rates to each pair of states
  - used as a parameter to the exponential distribution
  - transition between $s$ and $s'$ when $R(s,s') > 0$
  - probability triggered before $t$ time units $1 - e^{-R(s,s') \cdot t}$
Continuous-time Markov chains

• What happens when there exists multiple $s'$ with $R(s,s') > 0$?
  – first transition triggered determines the next state
  – called the race condition

• Time spent in a state before a transition:
  – minimum of exponential distributions
  – exponential with parameter given by summation:
    \[ E(s) = \sum_{s' \in S} R(s,s') \]
  – $E(s)$ is the exit rate of state $s$
  – state absorbing if $E(s) = 0$ (no outgoing transitions)
  – probability of leaving a state $s$ within $[0,t]$ equals $1 - e^{-E(s) \cdot t}$
Embedded DTMC

- Can determine the probability of each transition occurring
  - independent of the time at which it occurs

- Embedded DTMC: \( \text{emb}(C) = (S, s_{\text{init}}, P^{\text{emb}(C)}, L) \)
  - state space, initial state and labelling as the CTMC
  - for any \( s, s' \in S \)

\[
P^{\text{emb}(C)}(s, s') = \begin{cases} 
R(s, s')/E(s) & \text{if } E(s) > 0 \\
1 & \text{if } E(s) = 0 \text{ and } s = s' \\
0 & \text{otherwise}
\end{cases}
\]

- Alternative characterisation of the behaviour:
  - remain in \( s \) for delay exponentially distributed with rate \( E(s) \)
  - probability next state is \( s' \) is given by \( P^{\text{emb}(C)}(s, s') \)
Continuous-time Markov chains

- **Infinitesimal generator matrix**

\[ Q(s, s') = \begin{cases} 
R(s, s') & s \neq s' \\
- \sum_{s \neq s'} R(s, s') & \text{otherwise}
\end{cases} \]

- **Alternative definition: a CTMC is:**
  - a family of random variables \{ X(t) \mid t \in \mathbb{R}_{\geq 0} \}
  - \( X(t) \) are observations made at time instant \( t \)
  - i.e. \( X(t) \) is the state of the system at time instant \( t \)
  - which satisfies...

- **Memoryless (Markov property)**

\[ P[X(t_k) = s_k \mid X(t_{k-1}) = s_{k-1}, \ldots, X(t_0) = s_0] = P[X(t_k) = s_k \mid X(t_{k-1}) = s_{k-1}] \]
Simple CTMC example

- Modelling a queue of jobs
  - initially the queue is empty
  - jobs arrive with rate 3/2
  - jobs are served with rate 3
  - maximum size of the queue is 3
Simple CTMC example

\[ C = (S, s_{\text{init}}, R, L) \]

\[ S = \{s_0, s_1, s_2, s_3\} \]

\[ s_{\text{init}} = s_0 \]

\[ AP = \{\text{empty}, \text{full}\} \]

\[ L(s_0) = \{\text{empty}\} \quad L(s_1) = L(s_2) = \emptyset \quad \text{and} \quad L(s_3) = \{\text{full}\} \]

\[
R = \begin{bmatrix}
0 & 3/2 & 0 & 0 \\
3 & 0 & 3/2 & 0 \\
0 & 3 & 0 & 3/2 \\
0 & 0 & 3 & 0
\end{bmatrix}
\]

\[
P_{\text{emb}(C)} = \begin{bmatrix}
0 & 1 & 0 & 0 \\
2/3 & 0 & 1/3 & 0 \\
0 & 2/3 & 0 & 1/3 \\
0 & 0 & 1 & 0
\end{bmatrix}
\]

\[
Q = \begin{bmatrix}
-3/2 & 3/2 & 0 & 0 \\
3 & -9/2 & 3/2 & 0 \\
0 & 3 & -9/2 & 3/2 \\
0 & 0 & 3 & -3
\end{bmatrix}
\]

transition rate matrix

embedded DTMC

infinitesimal generator matrix
Paths of a CTMC

- **Infinite path** $\omega$ is a sequence $s_0t_0s_1t_1s_2t_2\ldots$ such that
  - $R(s_i,s_{i+1}) > 0$ and $t_i \in \mathbb{R}_{>0}$ for all $i \in \mathbb{N}$
  - amount of time spent in the $j$th state: $\text{time}(\omega,j)=t_j$
  - state occupied at time $t$: $\omega@t=s_j$
    where $j$ smallest index such that $\sum_{i \leq j} t_j \geq t$

- **Finite path** is a sequence $s_0t_0s_1t_1s_2t_2\ldots s_{k-1}s_k$ such that
  - $R(s_i,s_{i+1}) > 0$ and $t_i \in \mathbb{R}_{>0}$ for all $i<k$
  - $s_k$ is absorbing ($R(s,s') = 0$ for all $s' \in S$)
  - amount of time spent in the $i$th state only defined for $j \leq k$:
    - $\text{time}(\omega,j)=t_j$ if $j<k$ and $\text{time}(\omega,j)=\infty$ if $j=k$
  - state occupied at time $t$: if $t \leq \sum_{i \leq k} t_j$ then $\omega@t$ as above otherwise $t>\sum_{i \leq k} t_j$ then $\omega@t=s_k$
Probability space

- **Sample space**: Path(s) (set of all paths from a state s)
- **Events**: sets of infinite paths
- **Basic events**: sets of paths with common finite prefix
  - probability of a single finite path is zero
  - include time intervals in cylinders

- **Cylinder is a sequence** $s_0, l_0, s_1, l_1, ..., l_{n-1}, s_n$
  - $s_0, s_1, s_2, ..., s_n$ sequence of states where $R(s_i, s_{i+1}) > 0$ for $i < n$
  - $l_0, l_1, l_2, ..., l_{n-1}$ sequence of nonempty intervals of $\mathbb{R}_{\geq 0}$

- **$C(s_0, l_0, s_1, l_1, ..., l_{n-1}, s_n)$** set of (infinite and finite paths):
  - $\omega(i) = s_i$ for all $i \leq n$ and time($\omega, i$) $\in l_i$ for all $i < n$
Probability space

- Define measure over cylinders by induction

  - \( \Pr_s(C(s)) = 1 \)

  - \( \Pr_s(C(s, l, s_1, l_1, \ldots, l_{n-1}, s_n, l', s')) \) equals

    \[
    \Pr_s(C(s, l, s_1, l_1, \ldots, l_{n-1}, s_n)) \cdot \mathcal{P}^{\text{emb}(C)}(s_n, s') \cdot \left(e^{-E(s_n) \cdot \inf l'} - e^{-E(s_n) \cdot \sup l'}\right)
    \]
Probability space

- Probability space \((\text{Path}(s), \Sigma_{\text{Path}(s)}, \text{Pr}_s)\)

- Sample space \(\Omega = \text{Path}(s)\) (infinite and finite paths)

- Event set \(\Sigma_{\text{Path}(s)}\)
  - least \(\sigma\)-algebra on \(\text{Path}(s)\) containing all cylinders starting in \(s\)

- Probability measure \(\text{Pr}_s\)
  - \(\text{Pr}_s\) extends uniquely from probability defined over cylinders

- See [BHHK03] for further details
Probability space – Example

- Cylinder \( C(s_0, [0, 2], s_1) \)

\[
\Pr(C(s_0, [0, 2], s_1)) = \Pr(C(s_0)) \cdot P^{emb(C)}(s_0, s_1) \cdot (e^{-E(s_0) \cdot 0} - e^{-E(s_0) \cdot 2})
\]
\[
= 1 \cdot 1 \cdot (e^{-3/2 \cdot 0} - e^{-3/2 \cdot 2})
\]
\[
= 1 - e^{-3}
\]
\[
\approx 0.95021
\]

- Probability of leaving the initial state \( s_0 \) and moving to state \( s_1 \) within the first 2 time units of operation

![Diagram showing states and transitions](image)

\( s_0 \) \( s_1 \) \( s_2 \) \( s_3 \)

- \( \{\text{empty}\} \) transitions with rate 3/2, \( \{\text{full}\} \) transitions with rate 3.

\( s_0 \) \( s_1 \) \( s_2 \) \( s_3 \)
Transient and steady-state behaviour

- **Transient behaviour**
  - state of the model at a particular *time instant*
  - $\pi_{s,t}^C(s')$ is probability of, having started in state $s$, being in state $s'$ at time $t$
  - $\pi_{s,t}^C(s') = \Pr_s\{ \omega \in \text{Path}^C(s) \mid \omega_{@t} = s' \}$

- **Steady-state behaviour**
  - state of the model in the *long-run*
  - $\pi_{s}^C(s')$ is probability of, having started in state $s$, being in state $s'$ in the long run
  - $\pi_{s}^C(s') = \lim_{t \to \infty} \pi_{s,t}^C(s')$
  - the percentage of time, in long run, spent in each state
Computing transient probabilities

- $\Pi_t$ – matrix of transient probabilities
  - $\Pi_t(s,s') = \pi_{s,t}(s')$

- $\Pi_t$ solution of the differential equation: $\Pi_t' = \Pi_t \cdot Q$
  - $Q$ infinitesimal generator matrix

- Can be expressed as a matrix exponential and therefore evaluated as a power series
  \[
  \Pi_t = e^{Q \cdot t} = \sum_{i=0}^{\infty} \frac{(Q \cdot t)^i}{i!}
  \]
  - computation potentially unstable
  - probabilities instead computed using the uniformised DTMC
Uniformisation

- Uniformised DTMC \( \text{unif}(C) = (S, s_{\text{init}}, P_{\text{unif}(C)}, L) \) of \( C = (S, s_{\text{init}}, R, L) \)
  - set of states, initial state and labelling the same as \( C \)
  - \( P_{\text{unif}(C)} = I + Q/q \)
  - \( q \geq \max\{E(s) \mid s \in S\} \) is the uniformisation rate

- Each time step (epoch) of uniformised DTMC corresponds to one exponentially distributed delay with rate \( q \)
  - if \( E(s) = q \) transitions the same as embedded DTMC (residence time has the same distribution as one epoch)
  - if \( E(s) < q \) add self loop with probability \( 1 - E(s)/q \) (residence time longer than \( 1/q \) so one epoch may not be ‘long enough’)
Uniformisation

- Using the uniformised DTMC the transient probabilities can be expressed by:

\[
\Pi_t = e^{Q \cdot t} = e^{q \cdot (p_{\text{unif}(C)} - I) \cdot t} = e^{(q \cdot t) \cdot p_{\text{unif}(C)}} \cdot e^{-q \cdot t}
\]

\[
= e^{-q \cdot t} \cdot \left( \sum_{i=0}^{\infty} \frac{(q \cdot t)^i}{i!} \cdot \left( p_{\text{unif}(C)} \right)^i \right)
\]

\[
= \sum_{i=0}^{\infty} \left( e^{-q \cdot t} \cdot \frac{(q \cdot t)^i}{i!} \right) \cdot \left( p_{\text{unif}(C)} \right)^i
\]

\[
= \sum_{i=0}^{\infty} \gamma_{q \cdot t, i} \cdot \left( p_{\text{unif}(C)} \right)^i
\]

\(p_{\text{unif}(C)}\) stochastic (all entries in \([0,1]\) & rows sum to 1), therefore computations with \(P\) more numerically stable than \(Q\).
Uniformisation

\[ \Pi_t = \sum_{i=0}^{\infty} \gamma_{q \cdot t, i} \cdot (P_{\text{unif}(C)})^i \]

- \((P_{\text{unif}(C)})^i\) is probability of jumping between each pair of states in \(i\) steps

- \(\gamma_{q \cdot t, i}\) is the \(i\)th Poisson probability with parameter \(q \cdot t\)
  - the probability of \(i\) steps occurring in time \(t\), given each has delay exponentially distributed with rate \(q\)

- Can truncate the summation using the techniques of Fox and Glynn [FG88], which allow efficient computation of the Poisson probabilities
Uniformisation

- **Computing** $\pi_{s,t}$ for a fixed state $s$ and time $t$
  - can be computed **efficiently** using **matrix–vector operations**
  - pre-multiply the matrix $\Pi_t$ by the initial distribution
  - in this $\pi_{s,0}$ where $\pi_{s,0}(s')$ equals 1 if $s=s'$ and 0 otherwise

\[
\pi_{s,t} = \pi_{s,0} \cdot \Pi_t = \pi_{s,0} \cdot \sum_{i=0}^{\infty} \gamma_{q,t,i} \cdot \left( P_{\text{unif}(C)} \right)^i
\]

\[
= \sum_{i=0}^{\infty} \gamma_{q,t,i} \cdot \pi_{s,0} \cdot \left( P_{\text{unif}(C)} \right)^i
\]

- compute iteratively to avoid the computation of matrix powers

\[
\left( \pi_{s,t} \cdot P_{\text{unif}(C)} \right)^{i+1} = \left( \pi_{s,t} \cdot P_{\text{unif}(C)} \right)^i \cdot P_{\text{unif}(C)}
\]
Overview

• Exponential distributions

• Continuous-time Markov chains (CTMCs)
  – definition, paths, probabilities, steady-state, transient, …

• Properties of CTMCs: The logic CSL
  – syntax, semantics, equivalences, …

• CSL model checking
  – algorithm, examples, …

• Costs and rewards
CSL

- **Temporal logic for describing properties of CTMCs**
  - CSL = Continuous Stochastic Logic \cite{ASSB00,BHHK03}
  - extension of (non-probabilistic) temporal logic CTL
- **Key additions:**
  - probabilistic operator \( P \) (like PCTL)
  - steady state operator \( S \)
- **Example:** \( \text{down} \rightarrow P_{>0.75} [ \neg \text{fail} U^{[1,2]} \text{up} ] \)
  - when a shutdown occurs, the probability of a system recovery being completed between 1 and 2 hours without further failure is greater than 0.75
- **Example:** \( S_{<0.1}[\text{insufficient_routers}] \)
  - in the long run, the chance that an inadequate number of routers are operational is less than 0.1
• **CSL syntax:**

- $\phi ::= true \mid a \mid \phi \land \phi \mid \neg \phi \mid P_{\sim p} [\psi] \mid S_{\sim p} [\phi]$ (state formulas)

- $\psi ::= X \phi \mid \phi U I \phi$ (path formulas)

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

- **A CSL formula is always a state formula**
  - path formulas only occur inside the $P$ operator
CSL semantics for CTMCs

- **CSL formulas interpreted over states of a CTMC**
  - $s \models \phi$ denotes $\phi$ is “true in state $s$” or “satisfied in state $s$”

- **Semantics of state formulas:**
  - for a state $s$ of the CTMC $(S, s_{init}, R, L)$:
    - $s \models a \iff \ a \in L(s)$
    - $s \models \phi_1 \land \phi_2 \iff s \models \phi_1 \text{ and } s \models \phi_2$
    - $s \models \neg \phi \iff s \models \phi \text{ is false}$
    - $s \models P_{\sim p} [\psi] \iff \text{Prob}(s, \psi) \sim p$
    - $s \models S_{\sim p} [\phi] \iff \sum_{s'} \pi_{s}(s') \sim p$

**Probability of, starting in state $s$, satisfying the path formula $\psi$**

**Probability of, starting in state $s$, being in state $s'$ in the long run**
CSL semantics for CTMCs

- **Prob(s, ψ)** is the probability, starting in state s, of satisfying the path formula ψ
  
  \[ \text{Prob}(s, \psi) = \Pr_s \{ \omega \in \text{Path}_s \mid \omega \models \psi \} \]

- **Semantics of path formulas:**
  
  - for a path ω of the CTMC:
    
    \[ \omega \models X \phi \iff \omega(1) \text{ is defined and } \omega(1) \models \phi \]
    
    \[ \omega \models \phi_1 \cup^I \phi_2 \iff \exists t \in I. (\omega@t \models \phi_2 \land \forall t' < t. \omega@t' \models \phi_1) \]

  - if ω(0) is absorbing
    
    ω(1) not defined

  - there exists a time instant in the interval I where ϕ₂ is true and ϕ₁ is true at all preceding time instants
CSL derived operators

• (As for PCTL) can derive basic logical equivalences:
  - $\text{false} \equiv \neg \text{true}$ (false)
  - $\phi_1 \lor \phi_2 \equiv \neg(\neg\phi_1 \land \neg\phi_2)$ (disjunction)
  - $\phi_1 \rightarrow \phi_2 \equiv \neg\phi_1 \lor \phi_2$ (implication)

• The “eventually” operator (path formula)
  - $F \phi \equiv \text{true} U \phi$ (F = “future”) (F = “future”)
  - sometimes written as $\Diamond \phi$ (“diamond”) (“diamond”)
  - “$\phi$ is eventually true”
  - timed version: $F^I \phi \equiv \text{true} U^I \phi$
  - “$\phi$ becomes true in the interval $I$”
More on CSL

• Negation and probabilities
  – $\neg P_{>p} [ \phi_1 U^I \phi_2 ] \equiv P_{\leq p} [ \phi_1 U^I \phi_2 ]$
  – $\neg S_{>p} [ \phi ] \equiv S_{\leq p} [ \phi ]$

• The “always” operator (path formula)
  – $G \phi \equiv \neg (F \neg \phi) \equiv \neg (\text{true } U \neg \phi)$  \hspace{1cm} (G = “globally”)
  – sometimes written as $\Box \phi$ \hspace{1cm} (“box”)
  – “$\phi$ is always true”
  – bounded version: $G^I \phi \equiv \neg (F^I \neg \phi)$
  – “$\phi$ holds throughout the interval $I$”
  – strictly speaking, $G \phi$ cannot be derived from the CSL syntax in this way since there is no negation of path formulas
  – but, as for PCTL, we can derive $P_{\sim p} [ G \phi ]$ directly...
Derivation of $P_{\sim p} [ G \phi ]$

- $s \models P_{>p} [ G \phi ] \iff \text{Prob}(s, G \phi) > p$
  $\iff \text{Prob}(s, \neg (F \neg \phi)) > p$
  $\iff 1 - \text{Prob}(s, F \neg \phi) > p$
  $\iff \text{Prob}(s, F \neg \phi) < 1 - p$
  $\iff s \models P_{<1-p} [ F \neg \phi ]$

- Other equivalences:
  - $P_{\geq p} [ G \phi ] \equiv P_{\leq 1-p} [ F \neg \phi ]$
  - $P_{<p} [ G \phi ] \equiv P_{>1-p} [ F \neg \phi ]$
  - $P_{>p} [ G' \phi ] \equiv P_{<1-p} [ F' \neg \phi ]$
Quantitative properties

- Consider CSL formulae $P_{\sim p} [\psi]$ and $S_{\sim p} [\phi]$
  - if the probability is unknown, how to choose the bound $p$?

- When the outermost operator of a CSL formula is $P$ or $S$
  - allow bounds of the form $P_{=?} [\psi]$ and $S_{=?} [\phi]$
  - what is the probability that path formula $\psi$ is true?
  - what is the long-run probability that $\phi$ holds?

- Model checking is no harder: compute the values anyway

- As we have seen, useful for spotting patterns and trends
CSL example – Workstation cluster

- **Case study: Cluster of workstations [HHK00]**
  - two sub-clusters (N workstations in each cluster)
  - star topology with a central switch
  - components can break down, single repair unit
  - minimum QoS: at least $\frac{3}{4}$ of the workstations operational and connected via switches
  - premium QoS: all workstations operational and connected via switches


CSL example – Workstation cluster

- $P=?[\text{true} \ U^{[0,t]} \ \neg\text{minimum}]$
  - the chance that the QoS drops below minimum within $t$ hours

- $\neg\text{minimum} \rightarrow P_{<0.1}[F^{[0,t]} \ \neg\text{minimum}]$
  - when facing insufficient QoS, the probability of facing the same problem after $t$ hours is less than 0.1

- $S=?[\text{minimum}]$
  - the probability in the long run of having minimum QoS

- $\text{minimum} \rightarrow P_{>0.8}[\text{minimum} \ U^{[0,t]} \ \text{premium}]$
  - the probability of going from minimum to premium QoS within $t$ hours without violating minimum QoS is at least 0.8

- $P=?[\neg\text{minimum} \ U^{[t,\infty)} \ \text{minimum}]$
  - the chance it takes more than $t$ time units to recover from insufficient QoS
Overview

• Exponential distributions

• Continuous-time Markov chains (CTMCs)
  – definition, paths, probabilities, steady-state, transient, ...

• Properties of CTMCs: The logic CSL
  – syntax, semantics, equivalences, ...

• CSL model checking
  – algorithm, examples, ...

• Costs and rewards
CSL model checking for CTMCs

• Algorithm for CSL model checking [BHHK03]
  – inputs: CTMC $C = (S, s_{init}, R, L)$, CSL formula $\phi$
  – output: $\text{Sat}(\phi) = \{ s \in S \mid s \models \phi \}$, the set of states satisfying $\phi$

• What does it mean for a CTMC $C$ to satisfy a formula $\phi$?
  – check that $s \models \phi$ for all states $s \in S$, i.e. $\text{Sat}(\phi) = S$
  – know if $s_{init} \models \phi$, i.e. if $s_{init} \in \text{Sat}(\phi)$

• Sometimes, focus on quantitative results
  – e.g. compute result of $P=?[\text{true} \ U^{[0,13.5]}$ minimum $]$
  – e.g. compute result of $P=?[\text{true} \ U^{[0,t]}$ minimum $]$ for $0 \leq t \leq 100$
CSL model checking for CTMCs

- Basic algorithm proceeds by induction on parse tree of $\phi$
  - example: $\phi = S_{<0.1}[\neg \text{fail }] \rightarrow P_{>0.95} [ \neg \text{fail } U^1 \text{ succ } ]$

- For the non-probabilistic operators:
  - $\text{Sat}(\text{true}) = S$
  - $\text{Sat}(a) = \{ s \in S \mid a \in L(s) \}$
  - $\text{Sat}(\neg \phi) = S \setminus \text{Sat}(\phi)$
  - $\text{Sat}(\phi_1 \land \phi_2) = \text{Sat}(\phi_1) \cap \text{Sat}(\phi_2)$
Untimed properties

- Untimed properties can be verified on the embedded DTMC
  - properties of the form: $P_{~p} [ X \phi ]$ or $P_{~p} [ \phi_1 U[0,\infty) \phi_2 ]$
  - 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 $I$

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

  - can use pre–computation algorithm Prob0
Untimed properties

- $s \models P_{\sim 1} [\phi_1 \cup [0, \infty) \phi_2]$ does not imply $s \models P_{\sim 1} [\phi_1 \cup [0, \infty) \phi_2]$

- Consider the following example
  - with probability 1 eventually reach state $s_1$
    
    $s_0 \models P_{\geq 1} [\phi_1 \cup [0, \infty) \phi_2]$  
  - probability of remaining in state $s_0$ until time-bound $t$ is greater than zero for any $t$
    
    $s_0 \models \neg P_{\geq 1} [\phi_1 \cup [0, t] \phi_2]$  
  - $s_0 \models \neg P_{\geq 1} [\phi_1 \cup [0, t] \phi_2]$
Model checking – Time-bounded until

- Compute \( \text{Prob}(s, \phi_1 \ U^I \phi_2) \) for all states where \( I \) is an arbitrary interval of the non-negative real numbers
  
  - \( \text{Prob}(s, \phi_1 \ U^I \phi_2) = \text{Prob}(s, \phi_1 \ U^{\text{cl}(I)} \phi_2) \)
    where \( \text{cl}(I) \) closure of the interval \( I \)
  
  - \( \text{Prob}(s, \phi_1 \ U^{[0,\infty)} \phi_2) = \text{Prob}^{\text{emb}(C)}(s, \phi_1 \ U \phi_2) \)
    where \( \text{emb}(C) \) is the embedded DTMC

- Therefore, remains to consider the cases when
  
  - \( I = [0,t] \) for some \( t \in \mathbb{R}_{\geq 0} \)
  
  - \( I = [t,t'] \) for some \( t,t' \in \mathbb{R}_{\geq 0} \) such that \( t \leq t' \)
  
  - \( I = [t,\infty) \) for some \( t \in \mathbb{R}_{\geq 0} \)
Model checking – \( P_{\sim p}[\phi_1 \ U^{[0,t]} \ \phi_2] \)

- Computing the probabilities reduces to determining the least solution of the following set of integral equations:

- \( \text{Prob}(s, \phi_1 \ U^{[0,t]} \ \phi_2) \) equals
  - 1 if \( s \in \text{Sat}(\phi_2) \),
  - 0 if \( s \in \text{Sat}(\neg \phi_1 \ \land \ \neg \phi_2) \)
  - and otherwise equals

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

- Probability in state \( s' \) of satisfying until before \( t-x \) time units elapse
- Probability of moving from \( s \) to \( s' \) at time \( x \)
- Integrate over \( x \) between 0 and \( t \)
Model checking – $P_{\sim p} [\phi_1 \ U^{0,t} \ \phi_2]$

- Construct CTMC $C[\phi_2][\neg \phi_1 \ \land \ \neg \phi_2]$
  - where for CTMC $C=(S,s_{init},R,L)$, let $C[\theta]=(S,s_{init},R[\theta],L)$ where $R[\theta](s,s')=R(s,s')$ if $s \not\in \text{Sat}(\theta)$ and 0 otherwise
- Make all $\phi_2$ states absorbing
  - in 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
  - in such a state $\phi_1 \ U^{0,x} \ \phi_2$ holds with probability 0
- Problem then reduces to calculating transient probabilities of the CTMC $C[\phi_2][\neg \phi_1 \ \land \ \neg \phi_2]$: 
  $$\text{Prob}(s, \phi_1 \ U^{0,t} \ \phi_2) = \sum_{s' \in \text{Sat}(\phi_2)} \pi_{s,t}^{C[\phi_2][\neg \phi_1 \ \land \ \neg \phi_2]}(s')$$

transient probability: starting in state the probability of being in state $s'$ at time $t$
Model checking – $P_{\sim p}[\phi_1 \ U^{[0,t]} \phi_2]$

- Can now adapt uniformisation to computing the vector of probabilities $\text{Prob}(\phi_1 \ U^{[0,t]} \phi_2)$
  - recall $\Pi_t$ is matrix of transient probabilities $\Pi_t(s,s')=\pi(s,t(s'))$
  - computed via uniformisation: $\Pi_t = \sum_{i=0}^{\infty} Y_{q,t,i} \cdot (P_{\text{unif}}(C))^i$

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

\[
\text{Prob}(\phi_1 \ U^{[0,t]} \phi_2) = \prod_{t}^{C[\phi_2][\neg \phi_1 \land \neg \phi_2]} \cdot \phi_2 \\
= \left( \sum_{i=0}^{\infty} Y_{q,t,i} \cdot \left( P_{\text{unif}}(C[\phi_2][\neg \phi_1 \land \neg \phi_2])^i \right) \right) \cdot \phi_2 \\
= \sum_{i=0}^{\infty} \left( Y_{q,t,i} \cdot \left( P_{\text{unif}}(C[\phi_2][\neg \phi_1 \land \neg \phi_2])^i \right) \cdot \phi_2 \right)
\]
Model checking – $P_{\sim p}[\Phi_1 \cup^{[0,t]} \Phi_2]$ 

- Have shown that we can calculate the probabilities as:

$$
\text{Prob}(\Phi_1 \cup^{[0,t]} \Phi_2) = \sum_{i=0}^{\infty} \left( \gamma_{q,t,i} \cdot \left( P_{\text{unif}(C[\Phi_2][\neg \Phi_1 \lor \neg \Phi_2])}^i \cdot \Phi_2 \right) \right)
$$

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

- Can compute iteratively to avoid matrix powers:

$$
\left( P_{\text{unif}(C)} \right)^0 \cdot \Phi_2 = \Phi_2
$$

$$
\left( P_{\text{unif}(C)} \right)^{i+1} \cdot \Phi_2 = P_{\text{unif}(C)} \cdot \left( \left( P_{\text{unif}(C)} \right)^i \cdot \Phi_2 \right)
$$
\( P_{\sim p}[\phi_1 \cup^{[0,t]} \phi_2] \) – Example

- \( P_{>0.65}[\text{true} \cup^{[0,7.5]} \text{full}] \)
  - “probability of the queue becoming full within 7.5 time units”
- \textbf{State } s_3 \textbf{ satisfies full and no states satisfy } \neg \text{true}
  - in C[full][\neg \text{true} \land \neg \text{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 \( \text{unif}(C[\text{full}][\neg \text{true} \land \neg \text{full}]) \)
with uniformisation rate \( \max_{s \in S} E(s) = 4.5 \)

\( s_3 \) made absorbing

\( s_0 \) \( s_1 \) \( s_2 \) \( s_3 \) (empty) \( 3/2 \) \( 3/2 \) \( 3/2 \) \{full\}
$P_{\sim p}[\phi_1 \cup [0,t] \phi_2]$ – Example

- Computing the summation of matrix–vector multiplications

$$\Pr ob(\phi_1 \cup [0,t] \phi_2) = \sum_{i=0}^{\infty} \left( y_{q,t,i} \cdot \left( P^{\text{unif}}(C[\phi_2][\neg \phi_1 \wedge \neg \phi_2]) \right)^i \cdot \phi_2 \right)$$

  - yields $\Pr ob(\text{true } \cup [0,7.5] \text{ full}) \approx (0.6482, 0.6823, 0.7811, 1)$

- $P_{>0.65}[\text{ true } \cup [0,7.5] \text{ full }]$ satisfied in states $s_1$, $s_2$ and $s_3$
Model checking – $P_{\sim p}[\phi_1 U^{[t,t']} \phi_2]$

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

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

- Sum over states satisfying $\phi_1$
- Probability of reaching state $s'$ at time $t$ and satisfying $\phi_1$ up until this point
- Probability $\phi_1 U^{[0,t'-t]} \phi_2$ holds in $s'$
Model checking – $P_p[\phi_1 \cup_{t,t'} \phi_2]$

- Letting $\text{Prob}_{\phi_1}(s, \phi_1 \cup_{0,t} \phi_2) = \text{Prob}(s, \phi_1 \cup_{0,t} \phi_2)$ if $s \in \text{Sat}(\phi_1)$ and 0 otherwise, from the previous slide we have:

\[
\begin{align*}
\text{Prob}(\phi_1 \cup_{t,t'} \phi_2) &= \prod_{t}^{C[\neg \phi_1]} \cdot \text{Prob}_{\phi_1}(\phi_1 \cup_{0,t'-t} \phi_2) \\
&= \left( \sum_{i=0}^{\infty} Y_{q,t,i} \cdot \left( P_{\text{unif}(C[-\phi_1])} \right)^i \right) \cdot \text{Prob}_{\phi_1}(\phi_1 \cup_{0,t'-t} \phi_2)
\end{align*}
\]

- Summation can be truncated using Fox and Glynn [FG88]
- Can compute iteratively (only scalar and matrix-vector operations)
Model checking – \( P_{\sim p}[\phi_1 U^{[t,\infty)} \phi_2] \)

- Similar to the case for \( \phi_1 U^{[t,t']} \phi_2 \) except second part is now unbounded, and hence the embedded DTMC can be used
- Probability of remaining in \( \phi_1 \) states until time \( t \)
- Probability of reaching a \( \phi_2 \) state, while remaining in states satisfying \( \phi_1 \)
  - i.e. computing \( \text{Prob}(\phi_1 U^{[0,\infty)} \phi_2) \)

\[
\text{Prob}(s, \phi_1 U^{[0,t]} \phi_2) = \sum_{s' \in \text{Sat}(\phi_1)} \pi_{s,t}^{C[-\phi_1]}(s') \cdot \text{Prob}^{\text{emb}(C)}(s', \phi_1 U \phi_2)
\]

- Probability of reaching state \( s' \) at time \( t \) and satisfying \( \phi_1 \) up until this point
- Probability \( \phi_1 U^{[0,\infty)} \phi_2 \) holds in \( s' \)
Model checking – $P_{\sim p}[\phi_1 \cup_{[t,\infty)} \phi_2]$

- Letting $\text{Prob}_{\phi_1}(s, \phi_1 \cup_{[0,\infty)} \phi_2) = \text{Prob}(s, \phi_1 \cup_{[0,\infty)} \phi_2)$ if $s \in \text{Sat}(\phi_1)$ and 0 otherwise, from the previous slide we have:

\[
\text{Prob}(\phi_1 \cup_{[t,\infty)} \phi_2) = \prod_t \text{Prob}_{\phi_1}^{\text{emb}(C)}(\phi_1 \cup_{[0,\infty)} \phi_2)
\]

\[
= \left( \sum_{i=0}^{\infty} Y_{q,t,i} \cdot \left( p_{\text{unif}(C[-\phi_1])}^i \right) \right) \cdot \text{Prob}_{\phi_1}^{\text{emb}(C)}(\phi_1 \cup_{[0,\infty)} \phi_2)
\]

- 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 $\sum_{s', \models \phi} \pi^C_{s}(s') \sim p$
  - $\pi^C_{s}(s')$ is probability, having started in state $s$, of being in state $s'$ in the long run

- First, consider the simple case when $C$ is irreducible
  - $C$ is irreducible (strongly connected) if there exists a finite path from each state to every other state
  - the steady-state probabilities are independent of the starting state: denote the steady state probabilities by $\pi^C(s')$
  - these probabilities can be computed as the unique solution of the linear equation system:

$$\pi^C \cdot Q = 0 \quad \text{and} \quad \sum_{s \in S} \pi^C(s) = 1$$

$Q$ is the infinitesimal generator matrix of $C$
Model Checking – $S_{\sim_p} [ \phi ]$

- **Equation system can be solved by any standard approach**
  - Direct methods, such as Gaussian elimination
  - Iterative methods, such as Jacobi and Gauss–Seidel

- **The satisfaction of the CSL formula**
  - same for all states (steady state independent of starting state)
  - computed by summing steady state probabilities for all states satisfying $\phi$
Model Checking – $S_p[\phi]$

- We now suppose that C is reducible

- First perform graph analysis to find set bssc(C) of bottom strongly connected components (BSCCs)
  - strongly connected components that cannot be left

- Treating each individual $B \in \text{bscc}(C)$ as an irreducible CTMC compute the steady state probabilities $\pi^B$
  - employ the methods described above

- Calculate the probability of reaching each individual BSCC
  - can be computed in the embedded DTMC
  - if $a_B$ is an atomic proposition true only in the states of B, this probability is given by $\text{Prob}^\text{emb}(C)(s, F a_B)$
Model Checking – $S_{\sim p}[\phi]$

- For any states $s$ and $s'$ the steady state probability $\pi^C_s(s')$ can then be computed as:

$$\pi^C_s(s') = \begin{cases} \text{Prob}^{\text{emb}(C)}(s, F a_B) \cdot \pi^B(s') & \text{if } s' \in B \text{ for some } B \in \text{bscc}(C) \\ 0 & \text{otherwise} \end{cases}$$

- The total work required to compute $\pi^C_s(s')$ for all $s$ and $s'$
  - solve two linear equation systems for each BSCC $B$
    - one to obtain the vector $\text{Prob}^{\text{emb}(C)}(F a_B)$
    - the other to compute the steady state probabilities $\pi^B$
  - computation of the BSCCs requires only analysis of the underlying graph structure and can be performed using classical algorithms based on depth-first search
$S_{\sim_p}[\phi]$ – Example

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

$$Q = \begin{bmatrix}
-\frac{3}{2} & \frac{3}{2} & 0 & 0 \\
3 & -\frac{9}{2} & \frac{3}{2} & 0 \\
0 & 3 & -\frac{9}{2} & \frac{3}{2} \\
0 & 0 & 3 & -3
\end{bmatrix}$$
$S_{\sim p}[ \phi ]$ – Example

\[ -3/2 \cdot \pi(s_0) + 3 \cdot \pi(s_1) = 0 \]
\[ 3/2 \cdot \pi(s_0) - 9/2 \cdot \pi(s_1) + 3 \cdot \pi(s_2) = 0 \]
\[ 3/2 \cdot \pi(s_1) - 9/2 \cdot \pi(s_2) + 3 \cdot \pi(s_3) = 0 \]
\[ 3/2 \cdot \pi(s_2) - 3 \cdot \pi(s_3) = 0 \]

\[ \pi(s_0) + \pi(s_1) + \pi(s_2) + \pi(s_3) = 1 \]

- solution: $\pi = (8/15, 4/15, 2/15, 1/15)$

- $\sum_{s' \not= \text{full}} \pi(s') = 1/15 < 0.1$

- so all states satisfy $S_{<0.1}[\text{full}]$
Overview

• Exponential distributions

• Continuous-time Markov chains (CTMCs)
  – definition, paths, probabilities, steady-state, transient, ...

• Properties of CTMCs: The logic CSL
  – syntax, semantics, equivalences, ...

• CSL model checking
  – algorithm, examples, ...

• Costs and rewards
Costs and rewards

- **We augment CTMCs with rewards**
  - real-valued quantities assigned to states and/or transitions
  - these can have a wide range of possible interpretations
  - allows a wide range of quantitative measures of the system
  - basic notion: expected value of rewards
  - formal property specifications in an extension of CSL

- **For a CTMC \((S,s_{\text{init}}, R,L)\), a reward structure is a pair \((\rho,\iota)\)**
  - \(\rho : S \rightarrow \mathbb{R}_{\geq 0}\) is a vector of state rewards
  - \(\iota : S \times S \rightarrow \mathbb{R}_{\geq 0}\) is a matrix of transition rewards
  - **continuous time**: reward \(t \cdot \rho(s)\) acquired if the CTMC remains in state \(s\) for \(t \in \mathbb{R}_{\geq 0}\) time units
Reward structures – Example

• Example: “number of requests served”

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

• Example: “size of message queue”
  - \( \rho(s_i) = i \) and \( \iota(s_i, s_j) = 0 \) for all states \( s_i \) and \( s_j \)

\[
\begin{array}{c}
\text{\{empty\}} \\
\text{s_0} \quad \text{s_1} \quad \text{s_2} \quad \text{s_3} \\
\text{3/2} \quad \text{3/2} \quad \text{3/2} \quad \text{\{full\}}
\end{array}
\]
CSL and rewards

- Extend CSL to incorporate reward-based properties
  - add $R$ operator similar to the one in PCTL

  $\phi ::= \ldots \mid R_{\sim r}[\cdot\mid I=t \mid C\leq t \mid F\phi \mid S]$

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

- $R_{\sim r}[\cdot]$ means “the expected value of $\cdot$ satisfies $\sim r$”
Types of reward formulas

- **Instantaneous**: $R_{\sim r} [ I=^t ]$
  - the expected value of the reward at time-instant $t$ is $\sim r$
  - “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 $\sim r$
  - “the expected requests served within the first 4.5 seconds of operation is less than 10”

- **Reachability**: $R_{\sim r} [ F \phi ]$
  - the expected reward cumulated before reaching $\phi$ is $\sim r$
  - “the expected requests served before the queue becomes full”

- **Steady-state** $R_{\sim r} [ S ]$
  - the long-run average expected reward is $\sim r$
  - “expected long-run queue size is at least 1.2”
Reward formula semantics

- **Formal semantics of the four reward operators:**

  - \( s \models R_{\sim r}[I = t] \) \iff \( \text{Exp}(s, X_{I=t}) \sim r \)
  - \( s \models R_{\sim r}[C \leq t] \) \iff \( \text{Exp}(s, X_{C\leq t}) \sim r \)
  - \( s \models R_{\sim r}[F \Phi] \) \iff \( \text{Exp}(s, X_{F\Phi}) \sim r \)
  - \( s \models R_{\sim r}[S] \) \iff \( \lim_{t \to \infty} (\frac{1}{t} \cdot \text{Exp}(s, X_{C\leq t})) \sim r \)

- **where:**
  - \( \text{Exp}(s, X) \) denotes the expectation of the random variable \( X : \text{Path}(s) \to \mathbb{R}_{\geq 0} \) with respect to the probability measure \( \text{Pr}_s \)
Reward formula semantics

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

\[
X_{l=k}(\omega) = \rho(\omega @ t)
\]

\[
X_{c=t}(\omega) = \sum_{i=0}^{j_t-1} (t_i \cdot \rho(s_i) + \iota(s_i, s_{i+1})) + \left( t - \sum_{i=0}^{j_t-1} t_i \right) \cdot \rho(s_{j_t})
\]

\[
X_{F\phi}(\omega) = \begin{cases} 0 & \text{if } s_0 \in \text{Sat}(\phi) \\ \infty & \text{if } s_i \not\in \text{Sat}(\phi) \text{ for all } i \geq 0 \\ \sum_{i=0}^{k_\phi-1} t_i \cdot \rho(s_i) + \iota(s_i, s_{i+1}) & \text{otherwise} \end{cases}
\]

- where $j_t = \min \{ j \mid \sum_{i \leq j} t_i \geq t \}$ and $k_\phi = \min \{ i \mid s_i \models \phi \}$
Model checking reward formulas

- **Instantaneous**: $R_{\sim r} [ l=t ]$
  - 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 [KNP06]
  - based on uniformisation
- **Reachability**: $R_{\sim r} [ F \phi ]$
  - can be computed on the embedded DTMC
  - reduces to solving a system of linear equation
- **Steady–state**: $R_{\sim r} [ S ]$
  - similar to steady state formulae $S_{\sim r} [ \phi ]$
  - graph based analysis (compute BSCCs)
  - solve systems of linear equations (compute steady state probabilities of each BSCC)
Model checking complexity

- For model checking of a CTMC complexity:
  - linear in $|\Phi|$ and polynomial in $|S|$  
  - linear in $q \cdot t_{\text{max}}$ ($t_{\text{max}}$ is maximum finite bound in intervals)

- $P_{\sim p}[\Phi_1 \cup [0,\infty) \Phi_2]$, $S_{\sim p}[\Phi]$, $R_{\sim r} [F \Phi]$ 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)

- $P_{\sim p}[^{\Phi_1} \cup^{\Phi_2}]$, $R_{\sim r} [C \leq t]$ and $R_{\sim r} [I=t]$
  - at most two iterative sequences of matrix–vector product  
  - 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 $q \cdot t$ (q uniformisation rate)
Summing up...

- **Exponential distributions**
- **Continuous-time Markov chains (CTMCs)**
  - definition, paths, probability measure, ...
- **Properties of CTMCs: the logic CSL**
  - syntax, semantics, equivalences, ...
- **CSL model checking**
  - algorithm, examples, ...
- **Costs and rewards**