

Probabilistic Computation Tree Logic

Lecture #21 of Advanced Model Checking

Joost-Pieter Katoen

Lehrstuhl 2: Software Modeling & Verification

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

July 17, 2009

thanks to Dave Parker (Oxford) for his slides

Discrete-time Markov chains

A **DTMC** \mathcal{M} is a tuple $(S, \mathbf{P}, \iota_{init}, AP, L)$ with:

- S is a countable nonempty set of **states**
- $\mathbf{P} : S \times S \rightarrow [0, 1]$, **transition probability function** s.t. $\sum_{s'} \mathbf{P}(s, s') = 1$
 - $\mathbf{P}(s, s')$ is the probability to jump from s to s' in one step
 - s is **absorbing** if $\mathbf{P}(s, s) = 1$
- $\iota_{init} : S \rightarrow [0, 1]$, the **initial distribution** with $\sum_{s \in S} \iota_{init}(s) = 1$
 - $\iota_{init}(s)$ is the probability that system starts in state s
 - state s for which $\iota_{init}(s) > 0$ is an **initial state**
- $L : S \rightarrow 2^{AP}$, the **labelling function**

Probabilistic CTL

- Temporal logic for describing properties of MCs
- Extension of the temporal logic CTL
 - key addition is the **probabilistic** operator \mathbb{P}
 - which “replaces” CTL’s universal and existential path quantification
- Example:
 - $\mathbb{P}_{\leq 0.001} [\diamond^{\leq 500} \text{bad}]$
 - “with probability at least 10^{-3} a bad-state is reached within 500 steps”

PCTL syntax

- For $a \in AP$, $J \subseteq [0, 1]$ an interval with rational bounds, and natural n :

$$\begin{array}{l} \Phi ::= \text{true} \mid a \mid \Phi \wedge \Phi \mid \neg \Phi \mid \mathbb{P}_J(\varphi) \\ \varphi ::= \bigcirc \Phi \mid \Phi_1 \cup \Phi_2 \mid \Phi_1 \cup^{\leq n} \Phi_2 \end{array}$$

- $s_0 s_1 s_2 \dots \models \Phi \cup^{\leq n} \Psi$ if Φ holds until Ψ holds within n steps
- $s \models \mathbb{P}_J(\varphi)$ if probability that paths starting in s fulfill φ lies in J

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

Derived operators

$$\Diamond \Phi = \text{true} \cup \Phi$$

$$\Diamond^{\leq n} \Phi = \text{true} \cup^{\leq n} \Phi$$

$$\mathbb{P}_{\leq p}(\Box \Phi) = \mathbb{P}_{\geq 1-p}(\Diamond \neg \Phi)$$

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

operators like weak until W or release R can be derived analogously

Example properties

- With probability ≥ 0.92 , a goal state is reached legally:

$$\mathbb{P}_{\geq 0.92} (\neg \textit{illegal} \text{ U } \textit{goal})$$

- ... in maximally 137 steps: $\mathbb{P}_{\geq 0.92} (\neg \textit{illegal} \text{ U}^{\leq 137} \textit{goal})$

- ... once there, remain there almost surely for the next 31 steps:

$$\mathbb{P}_{\geq 0.92} \left(\neg \textit{illegal} \text{ U}^{\leq 137} \mathbb{P}_{=1}(\Box^{[0,31]} \textit{goal}) \right)$$

PCTL semantics (1)

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

$$s \models a \quad \text{iff} \quad a \in L(s)$$

$$s \models \neg \Phi \quad \text{iff} \quad \text{not } (s \models \Phi)$$

$$s \models \Phi \wedge \Psi \quad \text{iff} \quad (s \models \Phi) \text{ and } (s \models \Psi)$$

$$s \models \mathbb{P}_J(\varphi) \quad \text{iff} \quad \Pr(s \models \varphi) \in J$$

where $\Pr(s \models \varphi) = \Pr_s\{\pi \in \text{Paths}(s) \mid \pi \models \varphi\}$

PCTL semantics (2)

A *path* in \mathcal{M} is an infinite sequence $s_0 s_1 s_2 \dots$ with $\mathbf{P}(s_i, s_{i+1}) > 0$

Semantics of path-formulas is defined as in CTL:

$$\pi \models \bigcirc \Phi \quad \text{iff} \quad s_1 \models \Phi$$

$$\pi \models \Phi \cup \Psi \quad \text{iff} \quad \exists n \geq 0. (s_n \models \Psi \wedge \forall 0 \leq i < n. s_i \models \Phi)$$

$$\pi \models \Phi \cup^{\leq n} \Psi \quad \text{iff} \quad \exists k \geq 0. (k \leq n \wedge s_k \models \Psi \wedge \forall 0 \leq i < k. s_i \models \Phi)$$

Measurability

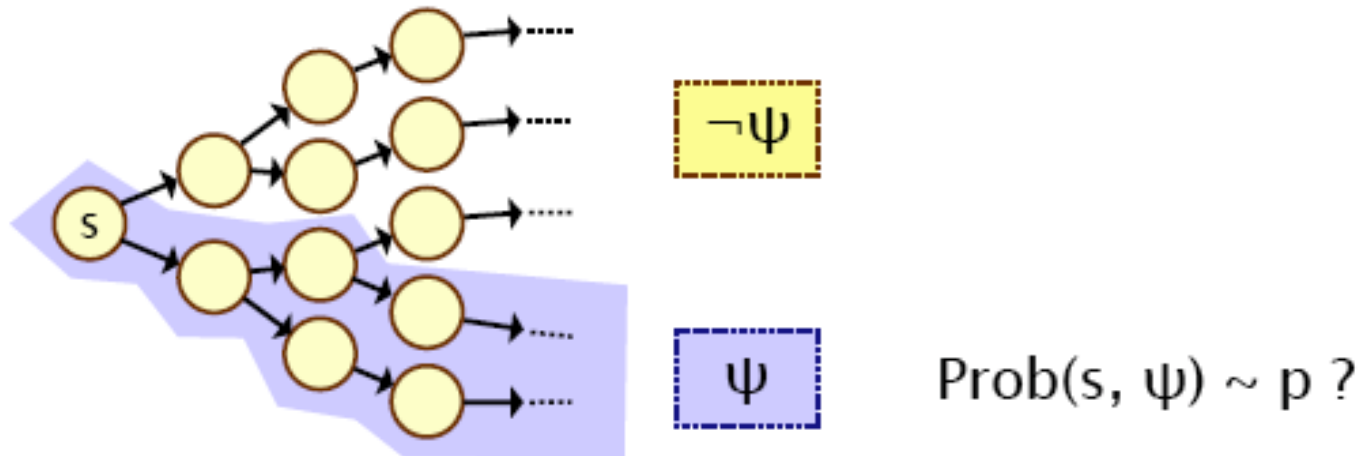
For any PCTL path formula φ and state s of DTMC \mathcal{M}
the set $\{\pi \in Paths(s) \mid \pi \models \varphi\}$ is measurable

Three cases:

- $\bigcirc \Phi$:
 - cylinder sets constructed from paths of length one
- $\Phi \cup^{\leq n} \Psi$:
 - (finite number of) cylinder sets from paths of length at most n
- $\Phi \cup \Psi$:
 - countable union of paths satisfying $\Phi \cup^{\leq n} \Psi$ for all $n \geq 0$

Semantics of \mathbb{P} -operator

- $s \models \mathbb{P}_J(\varphi)$ if:
 - the probability that a path starting in s fulfills φ lies in J
- Formally:
 - $s \models \mathbb{P}_J(\varphi)$ iff $\text{Pr}_s\{\pi \in \text{Paths}(s) \mid \pi \models \varphi\} \in J$

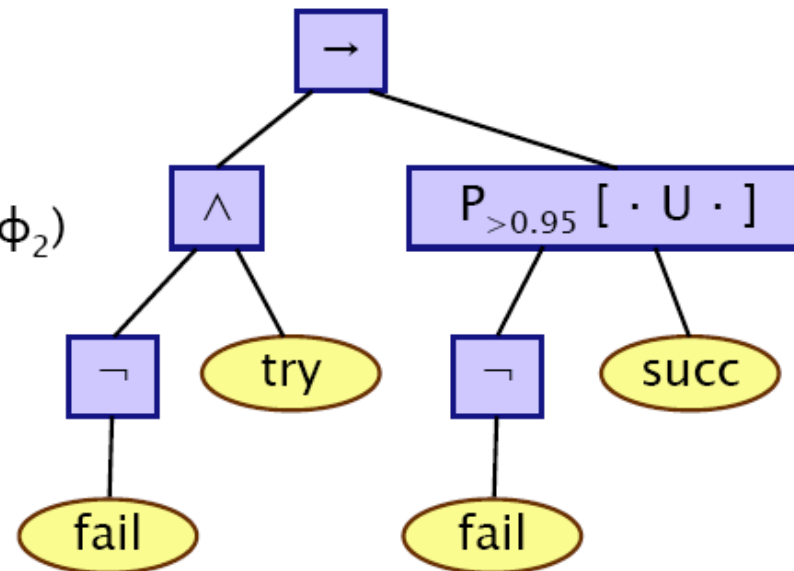


PCTL model checking

- Check whether state s in a DTMC satisfies a PCTL formula:
 - compute **recursively** the set $Sat(\Phi)$ of states that satisfy Φ
 - check whether state s belongs to $Sat(\Phi)$ \Rightarrow **bottom-up traversal** of the parse tree of Φ (like for CTL)
- For the propositional fragment: as for CTL
- **How to compute $Sat(\Phi)$ for the probabilistic operators?**

Bottom-up computation

- Basic algorithm proceeds by induction on parse tree of ϕ
 - example: $\phi = (\neg \text{fail} \wedge \text{try}) \rightarrow P_{>0.95} [\neg \text{fail} \cup \text{succ}]$
- For the non-probabilistic operators:
 - $\text{Sat}(\text{true}) = S$
 - $\text{Sat}(a) = \{ s \in S \mid a \in L(s) \}$
 - $\text{Sat}(\neg \phi) = S \setminus \text{Sat}(\phi)$
 - $\text{Sat}(\phi_1 \wedge \phi_2) = \text{Sat}(\phi_1) \cap \text{Sat}(\phi_2)$
- For the $P_{\sim p} [\psi]$ operator
 - need to compute the probabilities $\text{Prob}(s, \psi)$ for all states $s \in S$



Next formulas

- Alternative formulation: $s \models \mathbb{P}_J(\bigcirc \Phi)$ if and only if $Prob(s, \bigcirc \Phi) \in J$
- Next: $Prob(s, \bigcirc \Phi)$ equals $\sum_{s' \in Sat(\Phi)} \mathbf{P}(s, s')$
- Matrix-vector multiplication:

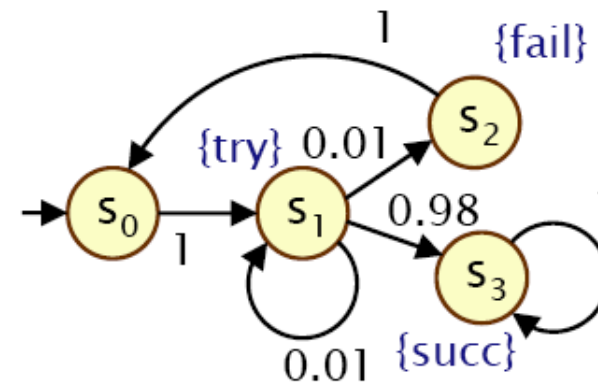
$$\left(Prob(s, \bigcirc \Phi) \right)_{s \in S} = \mathbf{P} \cdot \iota_\Phi$$

where ι_Φ is the characteristic vector of $Sat(\Phi)$, i.e.,
 $\iota_\Phi(s) = 1$ if and only if $s \in Sat(\Phi)$

Example

- Model check: $P_{\geq 0.9} [X (\neg \text{try} \vee \text{succ})]$
 - $\text{Sat} (\neg \text{try} \vee \text{succ}) = (S \setminus \text{Sat}(\text{try})) \cup \text{Sat}(\text{succ})$
 $= (\{s_0, s_1, s_2, s_3\} \setminus \{s_1\}) \cup \{s_3\} = \{s_0, s_2, s_3\}$
 - $\text{Prob}(X (\neg \text{try} \vee \text{succ})) = \mathbf{P} \cdot \underline{(\neg \text{try} \vee \text{succ})} = \dots$

$$= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0.01 & 0.01 & 0.98 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0.99 \\ 1 \\ 1 \end{bmatrix}$$



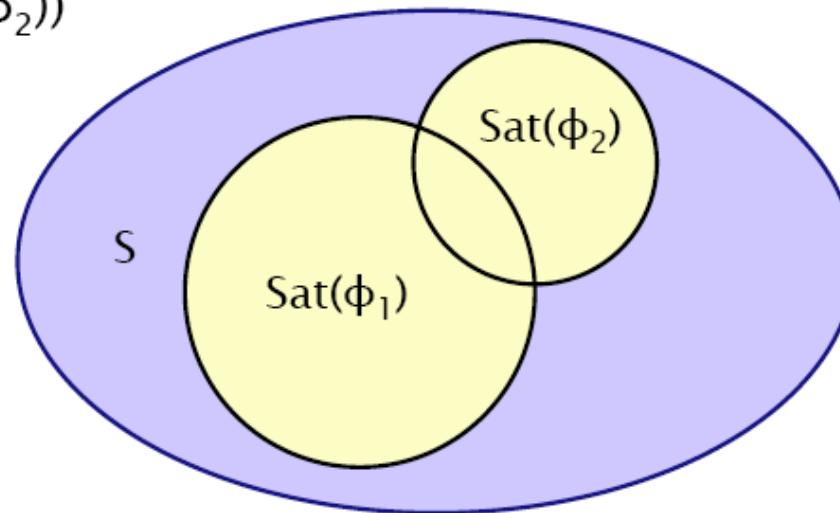
- Results:
 - $\text{Prob}(X (\neg \text{try} \vee \text{succ})) = [0, 0.99, 1, 1]$
 - $\text{Sat}(P_{\geq 0.9} [X (\neg \text{try} \vee \text{succ})]) = \{s_1, s_2, s_3\}$

Bounded until

- $s \models \mathbb{P}_J(\Phi \text{ U}^{\leq h} \Psi)$ if and only if $Prob(s, \Phi \text{ U}^{\leq h} \Psi) \in J$
- $Prob(s, \Phi \text{ U}^{\leq h} \Psi)$ is the least solution of: (Hansson & Jonsson, 1990)
 - 1 if $s \models \Psi$
 - for $h > 0$ and $s \models \Phi \wedge \neg \Psi$:
$$\sum_{s' \in S} P(s, s') \cdot Prob(s', \Phi \text{ U}^{\leq h-1} \Psi)$$
 - 0 otherwise
- Standard reachability for $\mathbb{P}_{>0}(\Phi \text{ U}^{\leq h} \Psi)$ and $\mathbb{P}_{\geq 1}(\Phi \text{ U}^{\leq h} \Psi)$
 - for efficiency reasons (avoiding solving system of linear equations)

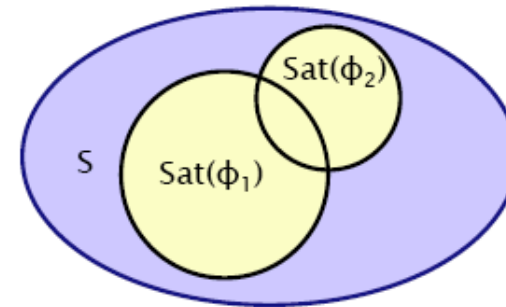
Bounded until

- Computation of probabilities for PCTL $U^{\leq k}$ operator
 - $\text{Sat}(P_{\sim p}[\phi_1 U^{\leq k} \phi_2]) = \{s \in S \mid \text{Prob}(s, \phi_1 U^{\leq k} \phi_2) \sim p\}$
 - need to compute $\text{Prob}(s, \phi_1 U^{\leq k} \phi_2)$ for all $s \in S$
- First identify (some) states where **probability is trivially 1/0**
 - $S^{\text{yes}} = \text{Sat}(\phi_2)$
 - $S^{\text{no}} = S \setminus (\text{Sat}(\phi_1) \cup \text{Sat}(\phi_2))$



Bounded until

- $S^{\text{yes}} = \text{Sat}(\phi_2)$
- $S^{\text{no}} = S \setminus (\text{Sat}(\phi_1) \cup \text{Sat}(\phi_2))$
- And let:
 - $S^? = S \setminus (S^{\text{yes}} \cup S^{\text{no}})$



- Compute solution of recursive equations:

$$\text{Prob}(s, \phi_1 \text{ U}^{\leq k} \phi_2) = \begin{cases} 1 & \text{if } s \in S^{\text{yes}} \\ 0 & \text{if } s \in S^{\text{no}} \\ 0 & \text{if } s \in S^? \text{ and } k = 0 \\ \sum_{s' \in S} P(s, s') \cdot \text{Prob}(s', \phi_1 \text{ U}^{\leq k-1} \phi_2) & \text{if } s \in S^? \text{ and } k > 0 \end{cases}$$

Bounded until

- Simultaneous computation of vector $\underline{\text{Prob}}(\phi_1 \text{ U}^{\leq k} \phi_2)$
 - i.e. probabilities $\text{Prob}(s, \phi_1 \text{ U}^{\leq k} \phi_2)$ for all $s \in S$
- Iteratively define in terms of matrices and vectors
 - define matrix \mathbf{P}' as follows: $\mathbf{P}'(s, s') = \mathbf{P}(s, s')$ if $s \in S^?$,
 $\mathbf{P}'(s, s') = 1$ if $s \in S^{\text{yes}}$ and $s = s'$, $\mathbf{P}'(s, s') = 0$ otherwise
 - $\underline{\text{Prob}}(\phi_1 \text{ U}^{\leq 0} \phi_2) = \underline{\phi}_2$
 - $\underline{\text{Prob}}(\phi_1 \text{ U}^{\leq k} \phi_2) = \mathbf{P}' \cdot \underline{\text{Prob}}(\phi_1 \text{ U}^{\leq k-1} \phi_2)$
 - requires **k matrix-vector multiplications**
- Note that we could express this in terms of matrix powers
 - $\underline{\text{Prob}}(\phi_1 \text{ U}^{\leq k} \phi_2) = (\mathbf{P}')^k \cdot \underline{\phi}_2$ and compute $(\mathbf{P}')^k$ in $\log_2 k$ steps
 - but this is actually inefficient: $(\mathbf{P}')^k$ is much less sparse than \mathbf{P}'

Bounded until

- Model check: $P_{>0.98} [F^{\leq 2} \text{ succ}] \equiv P_{>0.98} [\text{true} U^{\leq 2} \text{ succ}]$
 - $\text{Sat}(\text{true}) = S = \{s_0, s_1, s_2, s_3\}$, $\text{Sat}(\text{succ}) = \{s_3\}$
 - $S^{\text{yes}} = \{s_3\}$, $S^{\text{no}} = \emptyset$, $S^? = \{s_0, s_1, s_2\}$, $\mathbf{P}' = \mathbf{P}$
 - $\text{Prob}(\text{true} U^{\leq 0} \text{ succ}) = \underline{\text{succ}} = [0, 0, 0, 1]$

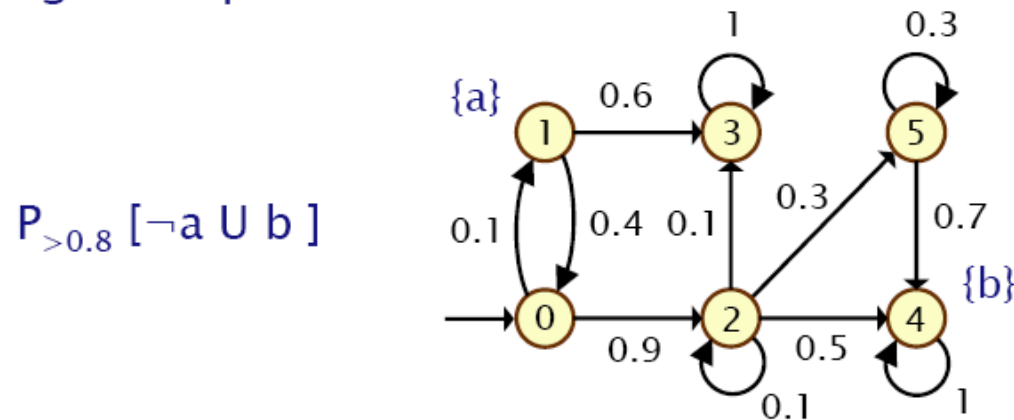
$$\text{Prob}(\text{true} U^{\leq 1} \text{ succ}) = \mathbf{P}' \cdot \underline{\text{succ}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0.01 & 0.01 & 0.98 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0.98 \\ 0 \\ 1 \end{bmatrix}$$

$$\text{Prob}(\text{true} U^{\leq 2} \text{ succ}) = \mathbf{P}' \cdot \text{Prob}(\text{true} U^{\leq 1} \text{ succ}) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0.01 & 0.01 & 0.98 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0.98 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0.98 \\ 0.9898 \\ 0 \\ 1 \end{bmatrix}$$

- $\text{Sat}(P_{>0.98} [F^{\leq 2} \text{ succ}]) = \{s_1, s_3\}$

Until

- Computation of probabilities $\text{Prob}(s, \phi_1 \cup \phi_2)$ for all $s \in S$
- First, identify **all** states where the **probability is 1 or 0**
 - $S^{\text{yes}} = \text{Sat}(P_{\geq 1} [\phi_1 \cup \phi_2])$
 - $S^{\text{no}} = \text{Sat}(P_{\leq 0} [\phi_1 \cup \phi_2])$
- Then solve linear equation system for remaining states
- Running example:



Until

- We refer to the first phase (identifying sets S^{yes} and S^{no}) as “precomputation”
 - two algorithms: Prob0 (for S^{no}) and Prob1 (for S^{yes})
 - algorithms work on underlying graph (probabilities irrelevant)
- Important for several reasons
 - ensures unique solution to linear equation system
 - reduces the set of states for which probabilities must be computed numerically
 - gives **exact results** for the states in S^{yes} and S^{no} (no round-off)
 - for model checking of **qualitative** properties ($P_{\sim p}[\cdot]$ where p is 0 or 1), no further computation required

Until

- Probabilities $\text{Prob}(s, \phi_1 \cup \phi_2)$ can now be obtained as the unique solution of the following set of **linear equations**:

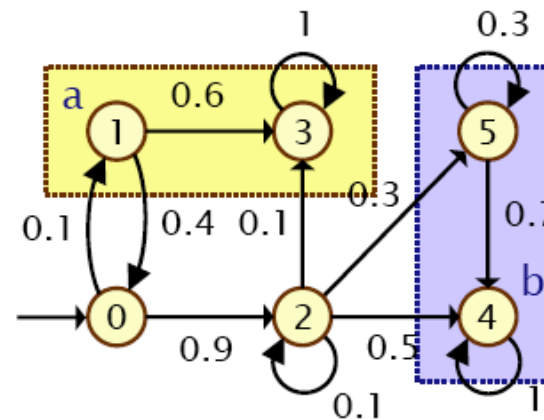
$$\text{Prob}(s, \phi_1 \cup \phi_2) = \begin{cases} 1 & \text{if } s \in S^{\text{yes}} \\ 0 & \text{if } s \in S^{\text{no}} \\ \sum_{s' \in S} P(s, s') \cdot \text{Prob}(s', \phi_1 \cup \phi_2) & \text{otherwise} \end{cases}$$

- Can be reduced to a system in $|S^?|$ unknowns instead of $|S|$ where $S^? = S \setminus (S^{\text{yes}} \cup S^{\text{no}})$

Until

- Example: $P_{>0.8} [\neg a \text{ U } b]$
- Let $x_s = \text{Prob}(s, \neg a \text{ U } b)$

$$S^{\text{no}} = \text{Sat}(P_{\leq 0} [\neg a \text{ U } b])$$



$$S^{\text{yes}} = \text{Sat}(P_{\geq 1} [\neg a \text{ U } b])$$

$$x_1 = x_3 = 0$$

$$x_4 = x_5 = 1$$

$$x_2 = 0.1x_2 + 0.1x_3 + 0.3x_5 + 0.5x_4 = 8/9$$

$$x_0 = 0.1x_1 + 0.9x_2 = 0.8$$

$$\text{Prob}(\neg a \text{ U } b) = \underline{x} = [0.8, 0, 8/9, 0, 1, 1]$$

$$\text{Sat}(P_{>0.8} [\neg a \text{ U } b]) = \{s_2, s_4, s_5\}$$

Reduction to transient analysis

- Make all Ψ - and all $\neg(\Phi \vee \Psi)$ -states absorbing in \mathcal{M}
- Check $\diamond^{=h} \Psi$ in the obtained DTMC \mathcal{M}'
- This is a standard transient analysis in \mathcal{M}' :

$$\sum_{s' \models \Psi} \Pr_s \{ \pi \in \text{Paths}(s) \mid \sigma[h] = s' \}$$

- compute by $(\mathbf{P}')^h \cdot \iota_{\Psi}$ where ι_{Ψ} is the characteristic vector of $\text{Sat}(\Psi)$

\Rightarrow Matrix-vector multiplication

Time complexity

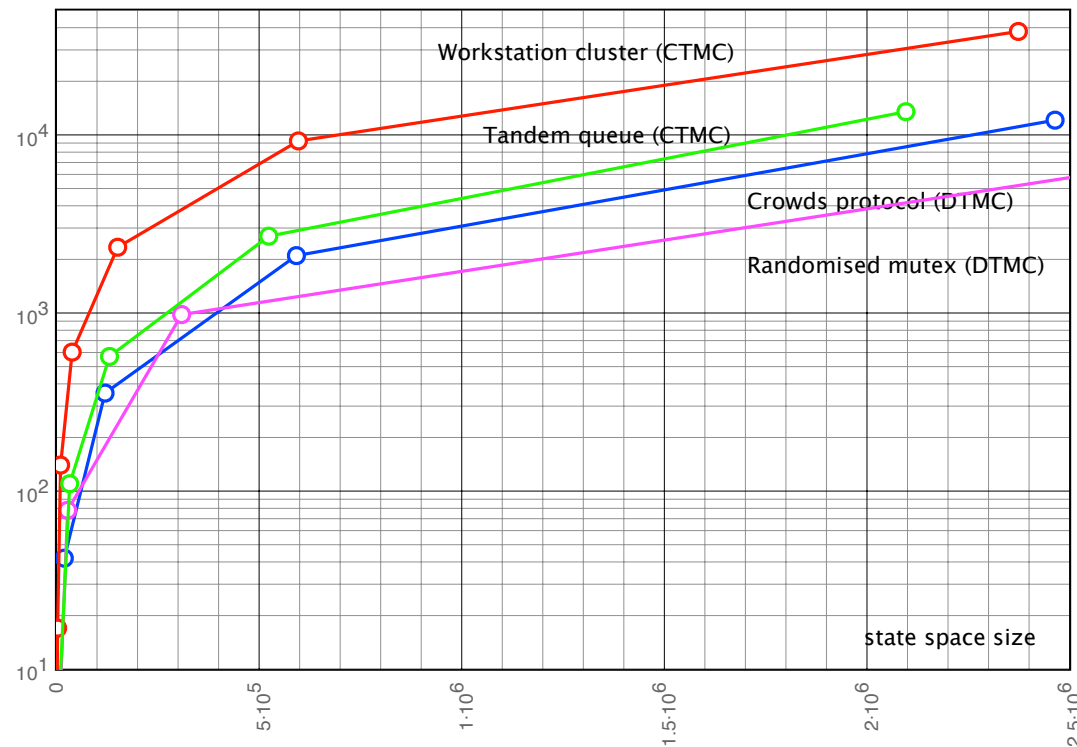
For finite DTMC \mathcal{M} and PCTL formula Φ , $\mathcal{M} \models \Phi$ can be solved in time

$$\mathcal{O}\left(\text{poly}(\text{size}(\mathcal{M})) \cdot n_{\max} \cdot |\Phi|\right)$$

- $n_{\max} = \max\{n \mid \Psi_1 \cup^{\leq n} \Psi_2 \text{ occurs in } \Phi\}$
- and $n_{\max} = 1$ if Φ does not contain the bounded until-operator

Verification times

verification time (in ms)



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

The qualitative fragment of PCTL

- For $a \in AP$ and natural n :

$$\begin{aligned}\Phi &::= \text{true} \mid a \mid \Phi \wedge \Phi \mid \neg \Phi \mid \mathbb{P}_{>0}(\varphi) \mid \mathbb{P}_{=1}(\varphi) \\ \varphi &::= \bigcirc \Phi \mid \Phi_1 \cup \Phi_2\end{aligned}$$

- The probability bounds $= 0$ and < 1 can be derived:

$$\mathbb{P}_{=0}(\varphi) \equiv \neg \mathbb{P}_{>0}(\varphi) \quad \text{and} \quad \mathbb{P}_{<1}(\varphi) \equiv \neg \mathbb{P}_{=1}(\varphi)$$

- No bounded until, and only > 0 , $= 0$, > 1 and $= 1$ intervals

so: $\mathbb{P}_{=1}(\diamond \mathbb{P}_{>0}(\bigcirc a))$ and $\mathbb{P}_{<1}(\mathbb{P}_{>0}(\diamond a) \cup b)$ are qualitative PCTL formulas

$\mathbb{P}_{=1}$ versus \forall and $\mathbb{P}_{>0}$ versus \exists

- PCTL-formula Φ is *equivalent* to CTL-formula Ψ :
 - $\Phi \equiv \Psi$ if and only if $\text{Sat}_{\mathcal{M}}(\Phi) = \text{Sat}_{TS(\mathcal{M})}(\Psi)$ for each DTMC \mathcal{M}
- $\exists\varphi$ requires φ on **some** paths, $\mathbb{P}_{>0}(\varphi)$ with **positive** probability
 - $\mathbb{P}_{>0}(\bigcirc a) \equiv \exists \bigcirc a$ and $\mathbb{P}_{>0}(\diamond a) \equiv \exists \diamond a$
 - and $\mathbb{P}_{>0}(a \cup b) \equiv \exists a \cup b$
 - but: $\mathbb{P}_{>0}(\Box a) \not\equiv \exists \Box a$
- $\forall\varphi$ requires φ to hold for **all** paths, $\mathbb{P}_{=1}(\varphi)$ for **almost** all
 - $\mathbb{P}_{=1}(\bigcirc a) \equiv \forall \bigcirc a$ and $\mathbb{P}_{=1}(\Box a) \equiv \forall \Box a$
 - but: $\mathbb{P}_{=1}(\diamond a) \not\equiv \forall \diamond a$ whereas $s \models \forall \diamond a$ implies $s \models \mathbb{P}_{=1}(\diamond a)$
 - and $\mathbb{P}_{=1}(a \cup b) \not\equiv \forall a \cup b$

Qualitative PCTL versus CTL

- There is no CTL-formula that is equivalent to $\mathbb{P}_{=1}(\Diamond a)$
- There is no CTL-formula that is equivalent to $\mathbb{P}_{>0}(\Box a)$
- There is no qualitative PCTL-formula that is equivalent to $\forall \Diamond a$
- There is no qualitative PCTL-formula that is equivalent to $\exists \Box a$

PCTL with $\forall \varphi$ and $\exists \varphi$ is more expressive than PCTL

Proofs

Almost sure repeated reachability

Let \mathcal{M} be a finite Markov chain and s a state of \mathcal{M} . Then:

$$s \models \mathbb{P}_{=1}(\Box \mathbb{P}_{=1}(\Diamond a)) \quad \text{iff} \quad \Pr_s\{\pi \in \text{Paths}(s) \mid \pi \models \Box \Diamond a\} = 1$$

this resembles $s \models \forall \Box \forall \Diamond a$ iff for all paths π : $\pi \models \Box \Diamond a$

Repeated reachability probabilities

For finite Markov chain, s a state of \mathcal{M} and interval $J \subseteq [0, 1]$:

$$s \models \underbrace{\mathbb{P}_J(\Diamond \mathbb{P}_{=1}(\Box \mathbb{P}_{=1}(\Diamond a)))}_{=\mathbb{P}_J(\Box \Diamond a)} \quad \text{iff} \quad \Pr(s \models \Box \Diamond a) \in J$$

the probabilities for $\Box \Diamond a$ agree with the probability to reach
a BSCC that contains at least one a -state

Persistence probabilities

For finite Markov chain, s a state of \mathcal{M} and interval $J \subseteq [0, 1]$:

$$s \models \mathbb{P}_J(\Diamond \mathbb{P}_{=1}(\Box a)) \quad \text{iff} \quad \Pr(s \models \Diamond \Box a) \in J$$