# Probabilistic Model Checking

Marta Kwiatkowska
Gethin Norman
Dave Parker

University of Oxford

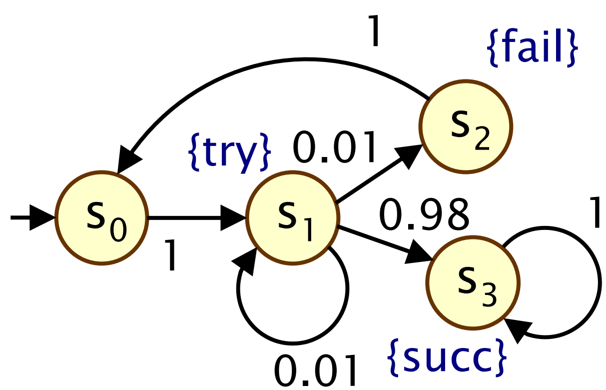
## Part 4 – Markov Decision Processes

# Overview

- Nondeterminism
- Markov decision processes (MDPs)
  - definition, examples, adversaries, probabilities
- Properties of MDPs: The logic PCTL
  - syntax, semantics, equivalences, …
- PCTL model checking
  - algorithms, examples, …
- Costs and rewards

# Recap: DTMCs

- Discrete-time Markov chains (DTMCs)
  - discrete state space,  transitions are discrete time-steps
  - from each state, choice of successor state (i.e. which transition) is determined by a discrete probability distribution




- DTMCs are fully probabilistic
  - well suited to modelling, for example, simple random algorithms or synchronous probabilistic systems where components move in lock-step

# Nondeterminism

- But, some aspects of a system may not be probabilistic and should not be modelled probabilistically; for example:

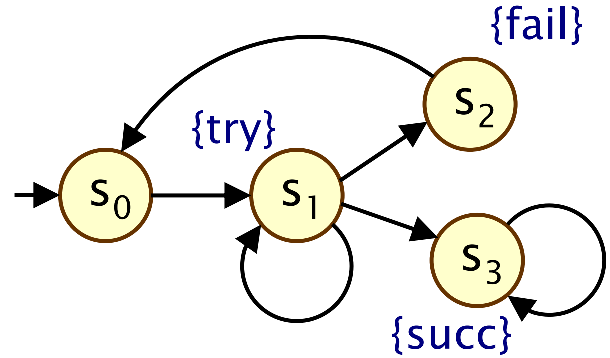
- Concurrency – scheduling of parallel components
  - e.g. randomised distributed algorithms – multiple probabilistic processes operating asynchronously

- Unknown environments
  - e.g. probabilistic security protocols – unknown adversary

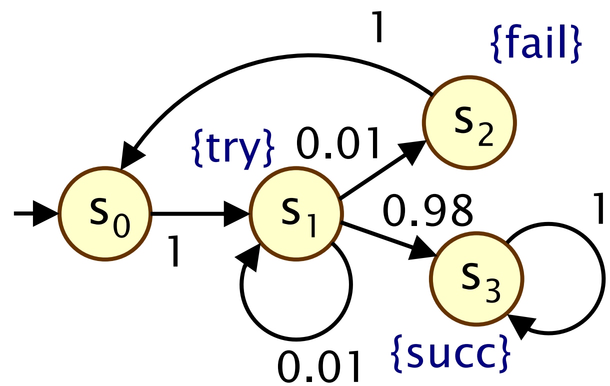
- Underspecification – unknown model parameters
  - e.g. a probabilistic communication protocol designed for message propagation delays of between $d_{min}$ and $d_{max}$

# Probability vs. nondeterminism

- Labelled transition system
  - $(S,s_0,R,L)$ where $R \subseteq S \times S$
  - choice is nondeterministic




- Discrete-time Markov chain
  - $(S,s_0,P,L)$ where $P : S \times S \rightarrow [0,1]$
  - choice is probabilistic



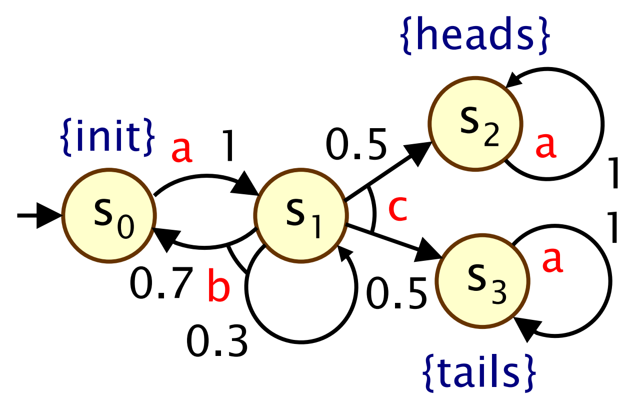

- How to combine?

# Overview

- Nondeterminism
- Markov decision processes (MDPs)
  - definition, examples, adversaries, probabilities
- Properties of MDPs: The logic PCTL
  - syntax, semantics, equivalences, …
- PCTL model checking
  - algorithms, examples, …
- Costs and rewards

# Markov decision processes

- Markov decision processes (MDPs)
  - extension of DTMCs which allow nondeterministic choice
- Like DTMCs:
  - discrete set of states representing possible configurations of the system being modelled
  - transitions between states occur in discrete time-steps
- Probabilities and nondeterminism
  - in each state, a nondeterministic choice between several discrete probability distributions over successor states

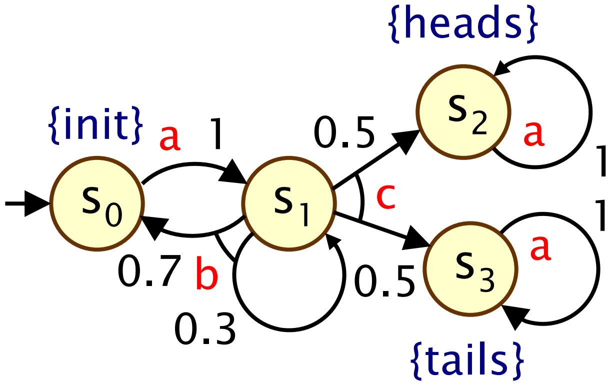


# Markov decision processes

- Formally, an MDP M is a tuple $(S, s_{init}, \mathbf{Steps}, L)$ where:
  - S is a finite set of states ("state space")
  - $s_{init} \in S$ is the initial state
  - **Steps** : $S \rightarrow 2^{Act \times Dist(S)}$ is the transition probability function

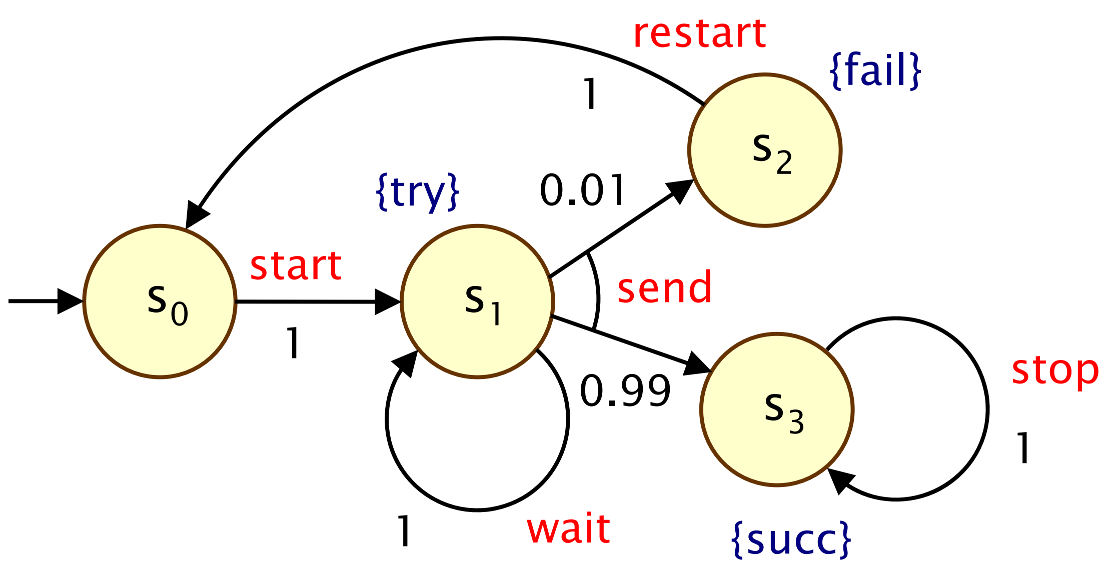
    where Act is a set of actions and Dist(S) is the set of discrete probability distributions over the set S
  - $L : S \rightarrow 2^{AP}$ is a labelling with atomic propositions

- Notes:
  - Steps(s) is always non-empty, i.e. no deadlocks
  - the use of actions to label distributions is optional



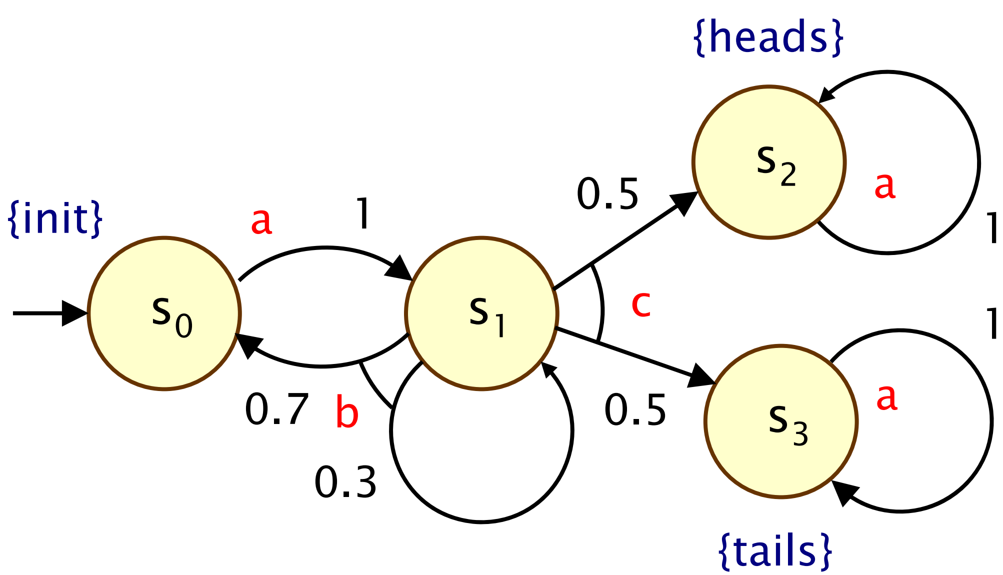
# Simple MDP example

- Modification of the simple DTMC communication protocol
  - after one step, process starts trying to send a message
  - then, a nondeterministic choice between: (a) waiting a step because the channel is unready; (b) sending the message
  - if the latter, with probability 0.99 send successfully and stop
  - and with probability 0.01, message sending fails, restart



# Simple MDP example 2

- Another simple MDP example with four states
  - from state $s_0$, move directly to $s_1$ (action a)
  - in state $s_1$, nondeterminstic choice between actions b and c
  - action b gives a probabilistic choice: self-loop or return to $s_0$
  - action c gives a 0.5/0.5 random choice between heads/tails



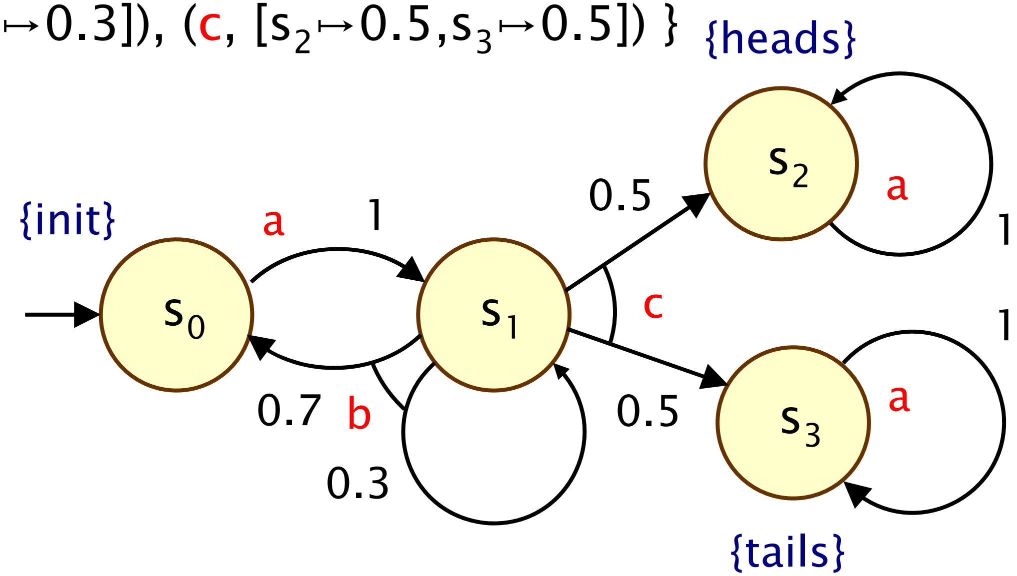
# Simple MDP example 2

$M = (S, s_{init}, \mathbf{Steps}, L)$

$S = \{s_0, s_1, s_2, s_3\}$
$s_{init} = s_0$

$AP = \{init, heads, tails\}$
$L(s_0) = \{init\}$,
$L(s_1) = \emptyset$,
$L(s_2) = \{heads\}$,
$L(s_3) = \{tails\}$

$\mathbf{Steps}(s_0) = \{ (a, s_1 \mapsto 1) \}$
$\mathbf{Steps}(s_1) = \{ (b, [s_0 \mapsto 0.7, s_1 \mapsto 0.3]), (c, [s_2 \mapsto 0.5, s_3 \mapsto 0.5]) \}$
$\mathbf{Steps}(s_2) = \{ (a, s_2 \mapsto 1) \}$
$\mathbf{Steps}(s_3) = \{ (a, s_3 \mapsto 1) \}$



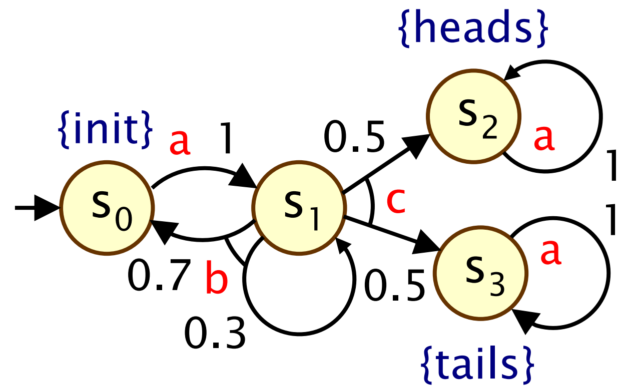
# The transition probability function

- It is often useful to think of the function **Steps** as a matrix
  - non-square matrix with $|S|$ columns and $\Sigma_{s \in S}$ $|$**Steps**$(s)|$ rows

- Example (for clarity, we omit actions from the matrix)

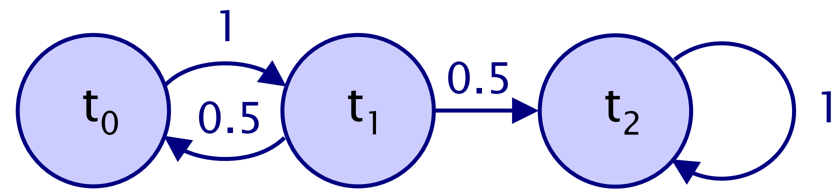
**Steps**$(s_0) = \{ (a, s_1 \mapsto 1) \}$
**Steps**$(s_1) = \{ (b, [s_0 \mapsto 0.7, s_1 \mapsto 0.3]), (c, [s_2 \mapsto 0.5, s_3 \mapsto 0.5]) \}$
**Steps**$(s_2) = \{ (a, s_2 \mapsto 1) \}$
**Steps**$(s_3) = \{ (a, s_3 \mapsto 1) \}$

$$\textbf{Steps} = \left[ \begin{array}{cccc} 0 & 1 & 0 & 0 \\ \hline 0.7 & 0.3 & 0 & 0 \\ 0 & 0 & 0.5 & 0.5 \\ \hline 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{array} \right]$$

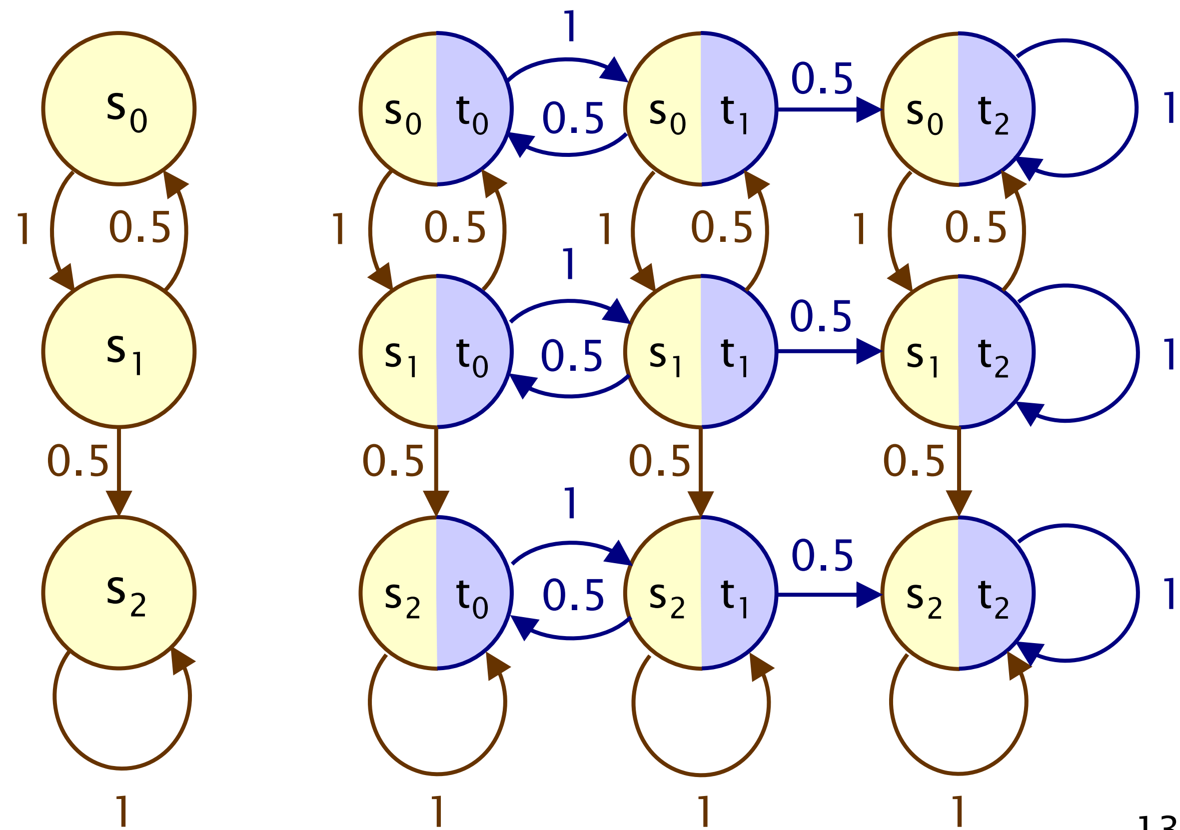


# Example – Parallel composition

Asynchronous parallel composition of two 3-state DTMCs




Action labels omitted here



# Paths and probabilities

- A (finite or infinite) path through an MDP
  - is a sequence of states and action/distribution pairs
  - e.g. $s_0(a_0,\mu_0)s_1(a_1,\mu_1)s_2\ldots$
  - such that $(a_i,\mu_i) \in$ **Steps**$(s_i)$ and $\mu_i(s_{i+1}) > 0$ for all $i \geq 0$
  - represents an execution (i.e. one possible behaviour) of the system which the MDP is modelling
  - note that a path resolves both types of choices: nondeterministic and probabilistic
- To consider the probability of some behaviour of the MDP
  - first need to resolve the nondeterministic choices
  - …which results in a DTMC
  - …for which we can define a probability measure over paths

# Adversaries

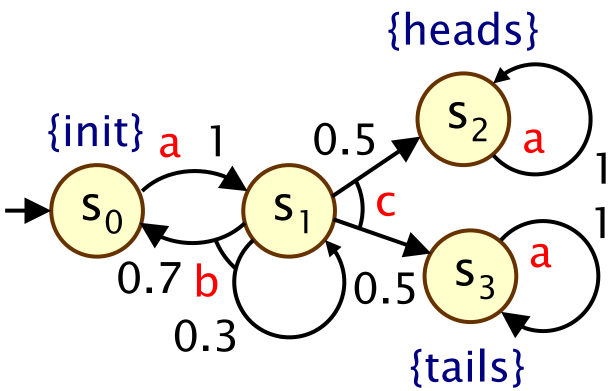
- An adversary resolves nondeterministic choice in an MDP
  - adversaries are also known as “schedulers” or “policies”

- Formally:
  - an adversary A of an MDP M is a function mapping every finite path $\omega = s_0(a_1,\mu_1)s_1 \ldots s_n$ to an element of Steps($s_n$)

- For each A can define a probability measure $Pr^A_s$ over paths
  - constructed through an infinite state DTMC ($Path^A_{fin}(s), s, \mathbf{P}^A_s$)
  - states of the DTMC are the finite paths of A starting in state s
  - initial state is s (the path starting in s of length 0)
  - $\mathbf{P}^A_s(\omega,\omega') = \mu(s)$ if $\omega' = \omega(a, \mu)s$ and $A(\omega) = (a,\mu)$
  - $\mathbf{P}^A_s(\omega,\omega') = 0$ otherwise

# Adversaries – Examples

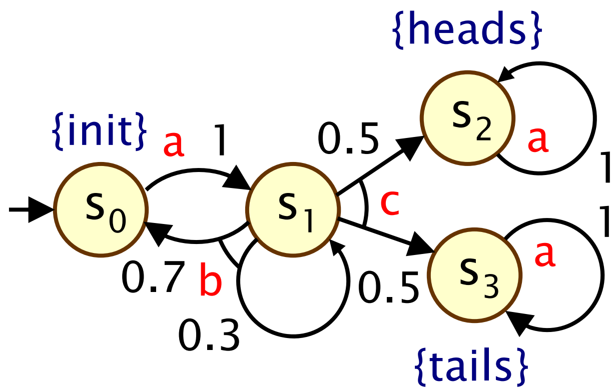
- Consider the previous example MDP
  - note that $s_1$ is the only state for which $|\mathbf{Steps}(s)| > 1$
  - i.e. $s_1$ is the only state for which an adversary makes a choice
  - let $\mu_b$ and $\mu_c$ denote the probability distributions associated with actions b and c in state $s_1$

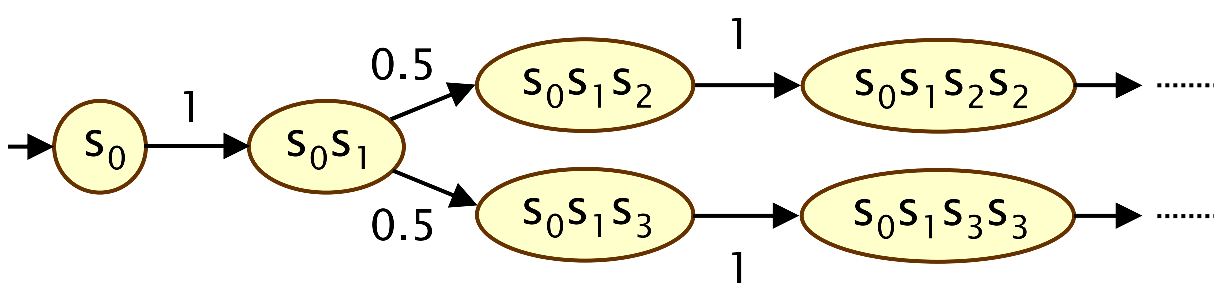



- Adversary $A_1$
  - picks action c the first time
  - $A_1(s_0 s_1)=(c,\mu_c)$
- Adversary $A_2$
  - picks action b the first time, then c
  - $A_2(s_0 s_1)=(b,\mu_b)$, $A_2(s_0 s_1 s_1)=(c,\mu_c)$, $A_2(s_0 s_1 s_0 s_1)=(c,\mu_c)$

# Adversaries – Examples

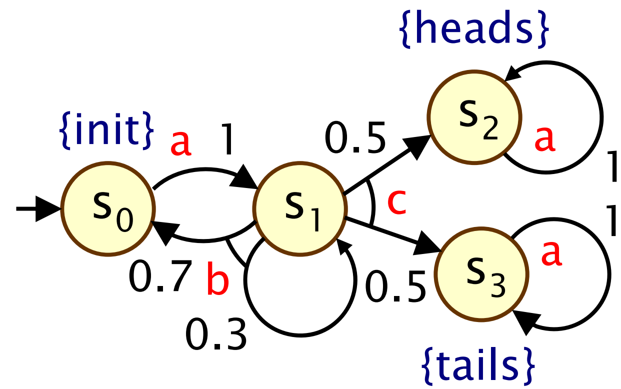
- Fragment of DTMC for adversary $A_1$
  - $A_1$ picks action c the first time

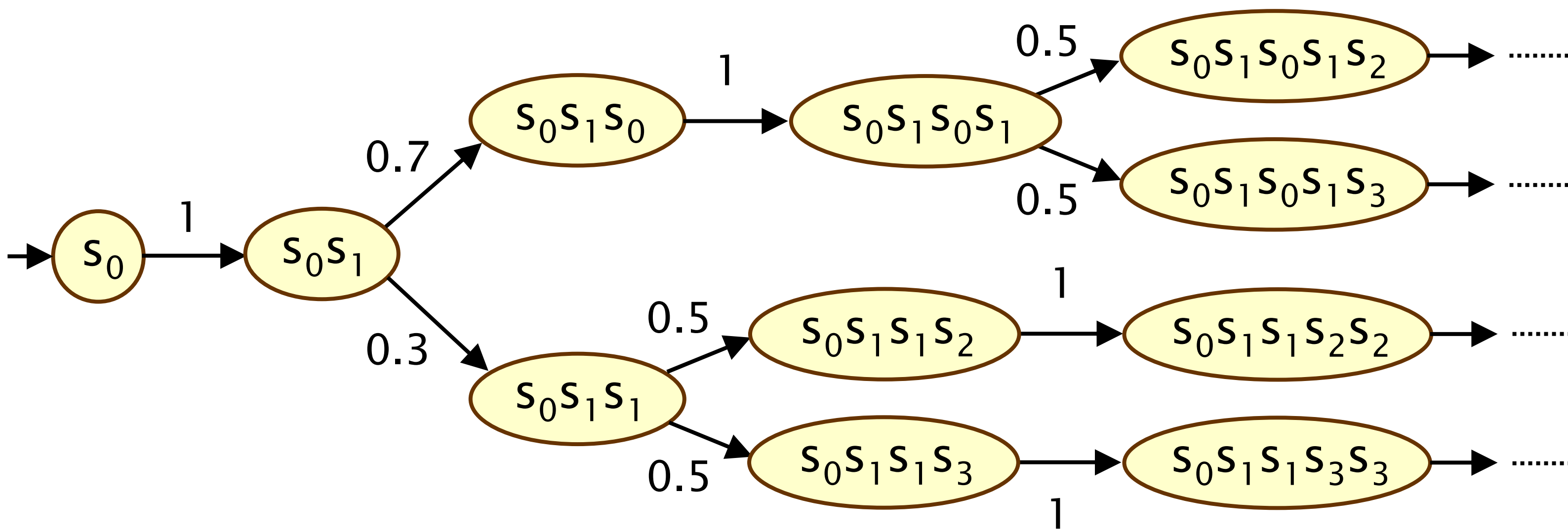


# Adversaries – Examples

- Fragment of DTMC for adversary $A_2$
  - $A_2$ picks action b, then c



# Overview

- Nondeterminism
- Markov decision processes (MDPs)
  - definition, examples, adversaries, probabilities
- Properties of MDPs: The logic PCTL
  - syntax, semantics, equivalences, …
- PCTL model checking
  - algorithms, examples, …
- Costs and rewards

# PCTL

- Temporal logic for describing properties of MDPs
  - identical syntax to the logic PCTL for DTMCs

ψ is true with probability ~p

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

"next" "bounded until" "until"

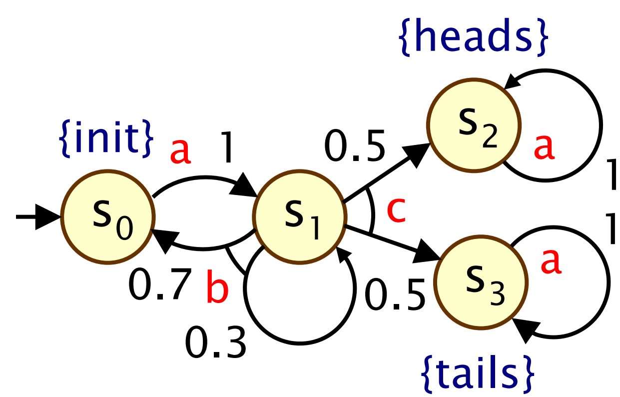
  - where a is an atomic proposition, used to identify states of interest, $p \in [0,1]$ is a probability, $\sim \in \{<,>,\leq,\geq\}$, $k \in \mathbb{N}$

# PCTL semantics for MDPs

- PCTL formulas interpreted over states of an MDP
  - $s \models \phi$ denotes $\phi$ is "true in state s" or "satisfied in state s"

- Semantics of (non-probabilistic) state formulas:
  - identical to those for DTMCs
  - for a state s of the MDP $(S, s_{init}, \mathbf{Steps}, L)$:
  - $s \models a \quad \Leftrightarrow \quad a \in L(s)$
  - $s \models \phi_1 \wedge \phi_2 \quad \Leftrightarrow \quad s \models \phi_1$ and $s \models \phi_2$
  - $s \models \neg\phi \quad \Leftrightarrow \quad s \models \phi$ is false

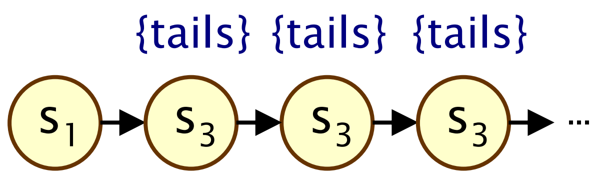
- Examples
  - $s_3 \models \text{tails}$
  - $s_1 \models \neg\, \text{heads} \wedge \neg\text{tails}$



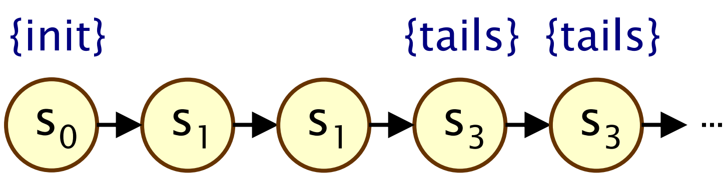
# PCTL semantics for MDPs

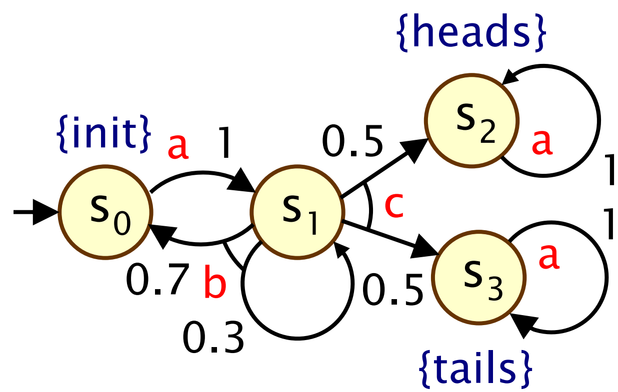
- Semantics of path formulas identical to DTMCs:
  - for a path $\omega = s_0(a_1,\mu_1)s_1(a_2,\mu_2)s_2\ldots$ in the MDP:
  - $\omega \vDash X\ \phi \quad \Leftrightarrow \quad s_1 \vDash \phi$
  - $\omega \vDash \phi_1\ U^{\leq k}\ \phi_2 \quad \Leftrightarrow \quad \exists i \leq k$ such that $s_i \vDash \phi_2$ and $\forall j<i,\ s_j \vDash \phi_1$
  - $\omega \vDash \phi_1\ U\ \phi_2 \quad \Leftrightarrow \quad \exists k \geq 0$ such that $\omega \vDash \phi_1\ U^{\leq k}\ \phi_2$

- Some examples of satisfying paths:
  - X tails



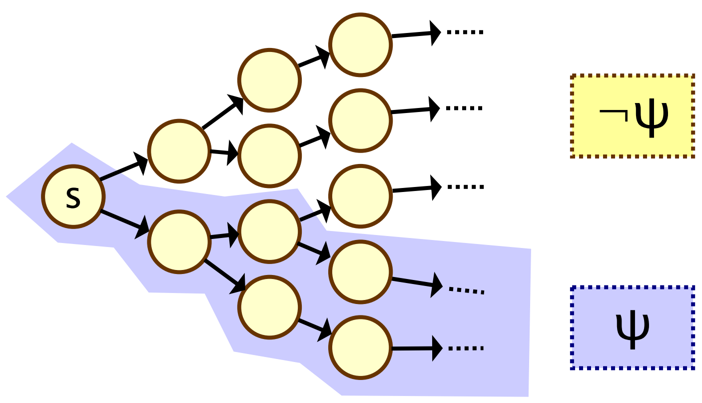

  - ¬heads U tails



# PCTL semantics for MDPs

- Semantics of the probabilistic operator P
  - can only define probabilities for a specific adversary A
  - $s \vDash P_{\sim p}$ [ $\psi$ ] means "the probability, from state s, that $\psi$ is true for an outgoing path satisfies ~p for all adversaries A"
  - formally $s \vDash P_{\sim p} [\ \psi\ ] \Leftrightarrow Prob^A(s, \psi) \sim p$ for all adversaries A
  - where $Prob^A(s, \psi) = Pr^A_s \{\ \omega \in Path^A(s) \mid \omega \vDash \psi\ \}$




$Prob^A(s, \psi) \sim p$

# Minimum and maximum probabilities

- Letting:
  - $p_{max}(s, \psi) = \sup_A \mathrm{Prob}^A(s, \psi)$
  - $p_{min}(s, \psi) = \inf_A \mathrm{Prob}^A(s, \psi)$
- We have:
  - if $\sim \in \{\geq,>\}$, then $s \models P_{\sim p} [\ \psi\ ] \quad \Leftrightarrow \quad p_{min}(s, \psi) \sim p$
  - if $\sim \in \{<,\leq\}$, then $s \models P_{\sim p} [\ \psi\ ] \quad \Leftrightarrow \quad p_{max}(s, \psi) \sim p$
- Model checking $P_{\sim p}[\ \psi\ ]$ reduces to the computation over all adversaries of either:
  - the minimum probability of $\psi$ holding
  - the maximum probability of $\psi$ holding

# Classes of adversary

- A more general semantics for PCTL over MDPs
  - parameterise by a class of adversaries Adv

- Only change is:
  - $s \vDash_{Adv} P_{\sim p} [\psi] \iff Prob^{A}(s, \psi) \sim p$ for all adversaries $A \in Adv$

- Original semantics obtained by taking Adv to be the set of all adversaries for the MDP

- Alternatively, take Adv to be the set of all fair adversaries
  - path fairness: if a state is occurs on a path infinitely often, then each non-deterministic choice occurs infinite often
  - see e.g. [BK98]

# PCTL derived operators

- Same equivalences as for DTMCs:
  - $\text{false} \equiv \neg\text{true}$ (false)
  - $\phi_1 \vee \phi_2 \equiv \neg(\neg\phi_1 \wedge \neg\phi_2)$ (disjunction)
  - $\phi_1 \rightarrow \phi_2 \equiv \neg\phi_1 \vee \phi_2$ (implication)
  - $\mathrm{F}\ \phi \equiv \text{true}\ \mathrm{U}\ \phi$ (eventually)
  - $\mathrm{F}^{\leq k}\ \phi \equiv \text{true}\ \mathrm{U}^{\leq k}\ \phi$
  - $\mathrm{G}\ \phi \equiv \neg(\mathrm{F}\ \neg\phi) \equiv \neg(\text{true}\ \mathrm{U}\ \neg\phi)$ (always)
  - $\mathrm{G}^{\leq k}\ \phi \equiv \neg(\mathrm{F}^{\leq k}\ \neg\phi)$
  - $\mathrm{P}_{\geq p}\ [\ \mathrm{G}\ \phi\ ] \quad \equiv \quad \mathrm{P}_{\leq 1-p}\ [\ \mathrm{F}\ \neg\phi\ ]$
  - etc.

# Qualitative properties

- PCTL can express qualitative properties of MDPs
  - like for DTMCs, can relate these to CTL's AF and EF operators
  - need to be careful with "there exists" and adversaries

- $P_{\geq 1}$ [ F $\phi$ ] is (similar to but) weaker than AF $\phi$
  - $P_{\geq 1}$ [ F $\phi$ ] $\Leftrightarrow$ $Prob^A(s, F\ \phi) \geq 1$ for all adversaries A
  - recall that "probability$\geq$1" is weaker than "for all"

- We can construct the following equivalence for EF $\phi$
  - $s \vDash EF\ \phi \Leftrightarrow$ there exists a finite path from s to a $\phi$–state

    $\Leftrightarrow Prob^A(s, F\ \phi) > 0$ for some adversary A

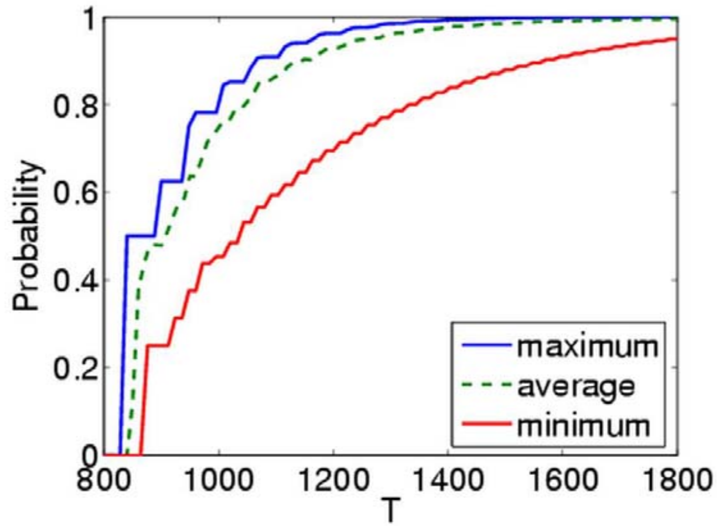
    $\Leftrightarrow$ not $Prob^A(s, F\ \phi) \leq 0$ for all adversaries A

    $\Leftrightarrow \neg P{\leq}0$ [ F $\phi$ ]

# Quantitative properties

- For PCTL properties with P as the outermost operator
  - we allow a quantitative form
  - for MDPs, there are two types: $\mathrm{Pmin}_{=?}$ [ ψ ] and $\mathrm{Pmax}_{=?}$ [ ψ ]
  - i.e. "what is the minimum/maximum probability (over all adversaries) that path formula ψ is true?"
  - model checking is no harder since compute the values of $p_{min}(s, \psi)$ or $p_{max}(s, \psi)$ anyway
  - useful to spot patterns/trends

- Example CSMA/CD protocol
  - "min/max probability that a message is sent within the deadline"



# Some real PCTL examples

- Byzantine agreement protocol
  - $\mathrm{Pmin}_{=?}$ [ F (agreement $\wedge$ rounds$\leq$2) ]
  - “what is the minimum probability that agreement is reached within two rounds?”

- CSMA/CD communication protocol
  - $\mathrm{Pmax}_{=?}$ [ F collisions=k ]
  - “what is the maximum probability of k collisions?”

- Self-stabilisation protocols
  - $\mathrm{Pmin}_{=?}$ [ $F^{\leq t}$ stable ]
  - “what is the minimum probability of reaching a stable state within k steps?”

# Overview

- Nondeterminism
- Markov decision processes (MDPs)
  - definition, examples, adversaries, probabilities
- Properties of MDPs: The logic PCTL
  - syntax, semantics, equivalences, …
- PCTL model checking
  - algorithms, examples, …
- Costs and rewards

# PCTL model checking for MDPs

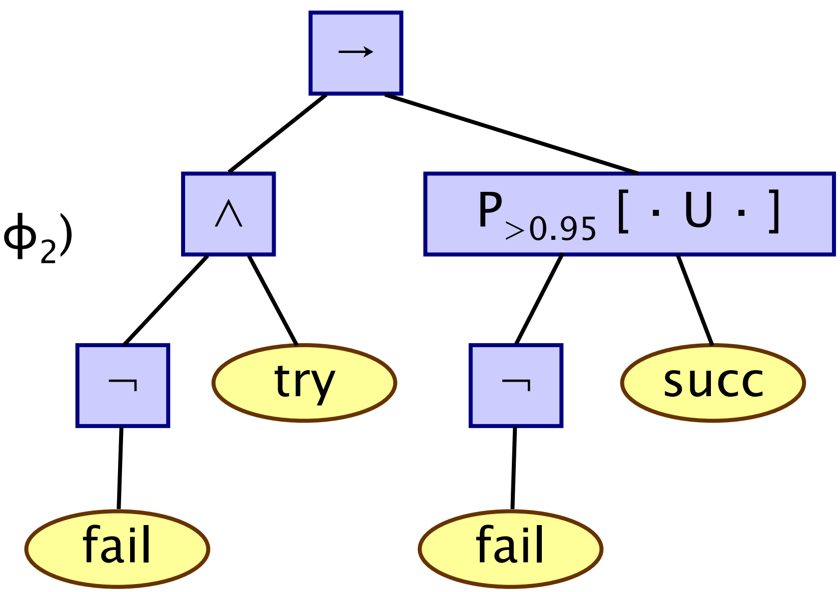
- Algorithm for PCTL model checking [BdA95]
  - inputs: MDP $M=(S,s_{init},\mathbf{Steps},L)$, PCTL formula $\phi$
  - output: $Sat(\phi) = \{ s \in S \mid s \vDash \phi \}$ = set of states satisfying $\phi$

- What does it mean for a MDP M to satisfy a formula $\phi$?
  - sometimes require $s \vDash \phi$ for all $s \in S$, i.e. $Sat(\phi) = S$
  - sometimes sufficient to check $s_{init} \vDash \phi$, i.e. if $s_{init} \in Sat(\phi)$

- Focus on quantitative results
  - e.g. compute result of Pmin=? [ F error ]
  - e.g. compute result of Pmax=? [ $F^{\leq k}$ error ] for $0 \leq k \leq 100$

# PCTL model checking for MDPs

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

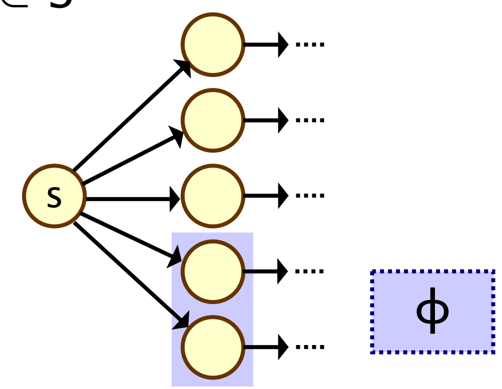


# PCTL model checking for MDPs

- Remains to consider $P_{\sim p}$ [ $\psi$ ] formulae
  - reduces compute either $p_{min}(s, \psi)$ or $p_{max}(s, \psi)$ for all $s \in S$
  - dependent on whether $\sim \in \{\geq,>\}$ or $\sim \in \{<,\leq\}$
- Present algorithms for computing $p_{min}(s, \psi)$
  - the case when $\sim \in \{\geq,>\}$
- Computation of $p_{min}(s, \psi)$ is dual
  - replace “min” with “max” and “for all” with “there exists”

# PCTL next for MDPs

- Computation of probabilities for PCTL next operator
  - $Sat(P_{\sim p}[\ X\ \phi\ ]) = \{\ s \in S \mid p_{min}(s, X\ \phi) \sim p\ \}$
  - need to compute $p_{min}(s, X\ \phi)$ for all $s \in S$




- Recall in the DTMC case
  - sum outgoing probabilities for transitions to ϕ-states
  - $Prob(s, X\ \phi) = \Sigma_{s' \in Sat(\phi)}\ P(s,s')$

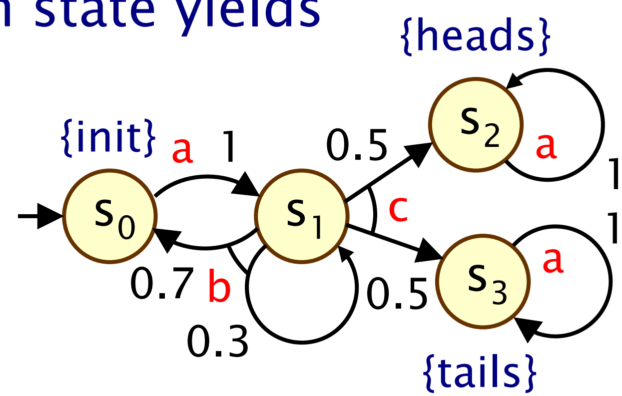
- For MDPs perform computation for each distribution available in s and then take minimum:
  - $p_{min}(s, X\ \phi) = \min\{\ \Sigma_{s' \in Sat(\phi)}\ \mu(s') \mid (a,\mu) \in Steps(s)\ \}$

# PCTL next – Example

- Model check: $P_{\geq 0.5}$ [ X heads ]
  - Sat (heads)= $\{s_2\}$

$$\textbf{Steps} \cdot \underline{heads} = \left[\begin{array}{cccc} 0 & 1 & 0 & 0 \\ \hline 0.7 & 0.3 & 0 & 0 \\ 0 & 0 & 0.5 & 0.5 \\ \hline 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \left[\begin{array}{c} 0 \\ \hline 0 \\ 0.5 \\ \hline 1 \\ \hline 0 \end{array}\right]$$

- Extracting the minimum for each state yields
  - $\underline{p}_{min}$(X heads) = [0, 0, 1, 0]
  - Sat($P_{\geq 0.5}$ [ X heads ]) = $\{s_2\}$



# PCTL bounded until for MDPs

- Computation of probabilities for PCTL $U^{\leq k}$ operator
  - $Sat(P_{\sim p}[\ \phi_1\ U^{\leq k}\ \phi_2\ ]) = \{\ s \in S \mid p_{min}(s, \phi_1\ U^{\leq k}\ \phi_2) \sim p\ \}$
  - need to compute $p_{min}(s, \phi_1\ U^{\leq k}\ \phi_2)$ for all $s \in S$

- First identify states where probability is trivially 1 or 0
  - $S^{yes} = Sat(\phi_2)$
  - $S^{no} = S \setminus (Sat(\phi_1) \cup Sat(\phi_2))$

- For the remaining states $S^{?} = S \setminus (S^{yes} \cup S^{no})$
  - compute $p_{min}(s, \phi_1\ U^{\leq k}\ \phi_2)$ through the recursive equations:

    If $k=0$, then $p_{min}(s, \phi_1\ U^{\leq k}\ \phi_2)$ equals 0

    If $k>0$, then $p_{min}(s, \phi_1\ U^{\leq k}\ \phi_2)$ equals

    $$\min\{\ \Sigma_{s' \in S}\ \mu(s') \cdot p_{min}(s, \phi_1\ U^{\leq k-1}\ \phi_2) \mid (a,\mu) \in Steps(s)\ \}$$

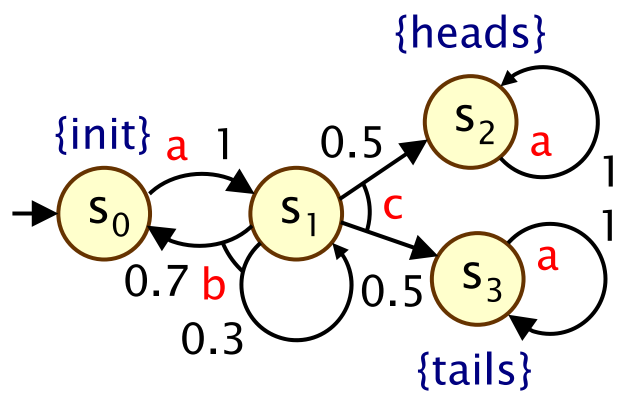
# PCTL bounded until for MDPs

- Simultaneous computation of vector $\underline{p}_{min}(\phi_1\ U^{\leq k}\ \phi_2)$
  - i.e. probabilities $p_{min}(s, \phi_1\ U^{\leq k}\ \phi_2)$ for all $s \in S$

- Recursive definition in terms of matrices and vectors
  - similar to DTMC case
  - requires k matrix-vector multiplications
  - in addition requires k minimum operations

# PCTL bounded until – Example

- Model check: $P_{<0.95}$ [ $F^{\leq 3}$ init ] $\equiv$ $P_{<0.95}$ [ true $U^{\leq 3}$ init ]
  - Sat (true) = S and Sat (init) = $\{s_0\}$
  - $S^{yes} = \{s_0\}$
  - $S^{no} = \varnothing$,
  - $S^{?} = \{s_1, s_2, s_3\}$
- The vector of probabilities is computed successively as:
  - $\underline{p}_{max}$(true $U^{\leq 0}$ init ) = [1,0,0,0]
  - $\underline{p}_{max}$(true $U^{\leq 1}$ init ) = [1,0.7,0,0]
  - $\underline{p}_{max}$(true $U^{\leq 2}$ init ) = [1,0.91,0,0]
  - $\underline{p}_{max}$(true $U^{\leq 3}$ init ) = [1,0.973,0,0]
- Hence, the result is:
  - Sat($P_{<0.95}$ [ $F^{\leq 3}$ init ]) = $\{s_2, s_3\}$



# PCTL until for MDPs

- Computation of probabilities $p_{min}(s, \phi_1 \text{ U } \phi_2)$ for all $s \in S$
- First identify all states where the probability is 1 or 0
- Set of states for which $p_{min}(s, \phi_1 \text{ U } \phi_2)=1$
  - for all adversaries the probability of satisfying $\phi_1 \text{ U } \phi_2$ is 1
  - $S^{yes} = Sat(P_{\geq 1} [\ \phi_1 \text{ U } \phi_2\ ])$
- Set of states for which $p_{min}(s, \phi_1 \text{ U } \phi_2)=0$
  - there exists an adversary for which the probability of satisfying $\phi_1 \text{ U } \phi_2$ is 0
  - not all adversaries satisfy $\phi_1 \text{ U } \phi_2$ with probability $>0$
  - $S^{no} = Sat(\neg\ P_{>0} [\ \phi_1 \text{ U } \phi_2\ ])$

# PCTL until for MDPs

- When computing $p_{max}(s, \phi_1 \text{ U } \phi_2)$...

- Set of states for which $p_{max}(s, \phi_1 \text{ U } \phi_2)=1$
  - there exists an adversary for which the probability of satisfying $\phi_1 \text{ U } \phi_2$ is 1
  - not all adversaries satisfy $\phi_1 \text{ U } \phi_2$ with probability $<1$
  - $S^{yes} = Sat(\neg P_{<1} [\ \phi_1 \text{ U } \phi_2\ ])$

- Set of states for which $p_{max}(s, \phi_1 \text{ U } \phi_2)=0$
  - for all adversaries the probability of satisfying $\phi_1 \text{ U } \phi_2$ is 0
  - $S^{no} = Sat(P_{\leq 0} [\ \phi_1 \text{ U } \phi_2\ ])$

# PCTL until for MDPs

- As for the DTMC refered to as "precomputation" phase
  - four precomputation algorithms:
  - for minimum probabilities Prob1A and Prob0E
  - for maximum probabilities Prob1E and Prob0A
- Important for several reasons
  - reduces the set of states for which probabilities must be computed numerically
  - for $P_{\sim p}[\cdot]$ where p is 0 or 1, no further computation required
  - gives exact results for the states in $S^{yes}$ and $S^{no}$ (no round-off)

# PCTL until for MDPs

- Probabilities $p_{min}(s, \phi_1 \text{ U } \phi_2)$ are obtained as the unique solution of the following linear optimisation problem:

$$\text{maximize } \sum_{s \in S^?} x_s \text{ subject to the constraints:}$$
$$x_s \leq \sum_{s' \in S^?} \mu(s') \cdot x_{s'} + \sum_{s' \in S^{yes}} \mu(s')$$
$$\text{for all } s \in S^? \text{ and for all } (a, \mu) \in \mathbf{Steps}(s)$$

- Simple case of a more general problem known as the stochastic shortest path problem [BT91]

- This can be solved with (a variety of) standard techniques
  - direct methods, e.g. Simplex, ellipsoid method
  - iterative methods, e.g. policy, value iteration

# PCTL until for MDPs

- In the case of maximum probabilities
- Probabilities $p_{max}(s, \phi_1 \; U \; \phi_2)$ are obtained as the unique solution of the following linear optimisation problem:

$$\text{minimize } \sum\nolimits_{s \in S^?} x_s \text{ subject to the constraints :}$$

$$x_s \geq \sum_{s' \in S^?} \mu(s') \cdot x_{s'} + \sum_{s' \in S^{yes}} \mu(s')$$

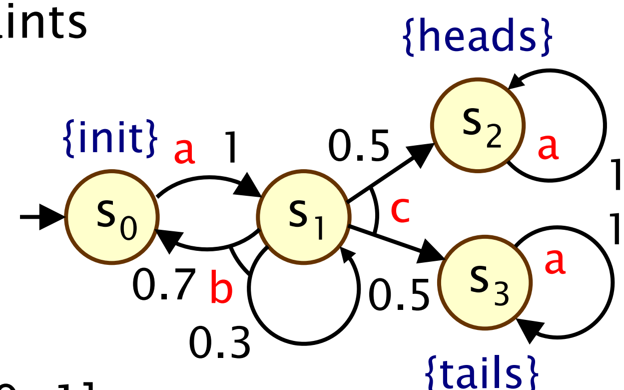
$$\text{for all } s \in S^? \text{ and for all } (a, \mu) \in \mathbf{Steps}(s)$$

# PCTL until – Example

- Model check: $P_{\geq 0.5}$ [ true U (tails $\vee$ init) ]
  - Sat(tails $\vee$ init) = $\{s_0, s_3\}$
  - $S^{no}$ = Sat($\neg P_{>0}$ [true U (tails $\vee$ init)]) = $\{s_2\}$
  - $S^{yes}$ = Sat($P_{\geq 1}$ [true U (tails $\vee$ init)]) = $\{s_0, s_3\}$

- Linear optimisation problem:
  - maximize $x_1$ subject to the constraints

$$x_1 \leq 0.3 \cdot x_1 + 0.7$$
$$x_1 \leq 0.5$$




- Which yields:
  - $\underline{p}_{min}$(true U (tails $\vee$ init)) = [1, 0.5, 0, 1]
  - Sat($P_{\geq 0.5}$ [ try U succ ]) = $\{s_0, s_1, s_3\}$

# Overview

- Nondeterminism
- Markov decision processes (MDPs)
  - definition, examples, adversaries, probabilities
- Properties of MDPs: The logic PCTL
  - syntax, semantics, equivalences, …
- PCTL model checking
  - algorithms, examples, …
- Costs and rewards

# Costs and rewards

- We can augment MDPs with rewards (or costs)
  - real-valued quantities assigned to states and/or actions
  - different from the DTMC case where transition rewards assigned to individual transitions

- For a MDP ($S$,$s_{init}$,**Steps**,$L$), a reward structure is a pair ($\underline{\rho}$,$\boldsymbol{\iota}$)
  - $\underline{\rho} : S \rightarrow \mathbb{R}_{\geq 0}$ is the state reward function
  - $\boldsymbol{\iota} : S \times Act \rightarrow \mathbb{R}_{\geq 0}$ is transition reward function

- As for DTMCs these can be used to compute:
  - elapsed time, power consumption, size of message queue, number of messages successfully delivered, net profit, …

# PCTL and rewards

- Augment PCTL with rewards based properties
  - allow a wide range of quantitative measures of the system
  - basic notion: expected value of rewards

expected reward is ~r

$$\phi \;::=\; \ldots \;\mid\; R_{\sim r}\,[\; I^{=k} \;] \;\mid\; R_{\sim r}\,[\; C^{\leq k} \;] \;\mid\; R_{\sim r}\,[\; F\;\phi \;]$$

“instantaneous” “cumulative” “reachability”

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

- $R_{\sim r}$ [ · ] means “the expected value of · satisfies ~r for all adversaries”

# Types of reward formulas

- Instantaneous: $R_{\sim r}$ [ $I^{=k}$ ]
  - the expected value of the reward at time-step k is ~r for all adversaries
  - "the minimum expected queue size after exactly 90 seconds"

- Cumulative: $R_{\sim r}$ [ $C^{\leq k}$ ]
  - the expected reward cumulated up to time-step k is ~r for all adversaries
  - "the maximum expected power consumption over one hour"

- Reachability: $R_{\sim r}$ [ F $\phi$ ]
  - the expected reward cumulated before reaching a state satisfying $\phi$ is ~r for all adversaries
  - the maximum expected time for the algorithm to terminate

# Reward formula semantics

- Formal semantics of the three reward operators:
  - for a state s in the MDP:
  - $s \vDash R_{\sim r} [\ I^{=k}\ ] \;\Leftrightarrow\; Exp^A(s, X_{I=k}) \sim r$ for all adversaries A
  - $s \vDash R_{\sim r} [\ C^{\leq k}\ ] \;\Leftrightarrow\; Exp^A(s, X_{C\leq k}) \sim r$ for all adversaries A
  - $s \vDash R_{\sim r} [\ F\ \Phi\ ] \;\Leftrightarrow\; Exp^A(s, X_{F\Phi}) \sim r$ for all adversaries A

$Exp^A(s, X)$ denotes the expectation of the random variable $X : Path^A(s) \rightarrow \mathbb{R}_{\geq 0}$ with respect to the probability measure $Pr^A_s$

# Reward formula semantics

- For an infinite path $\omega = s_0(a_0,\mu_0)s_1(a_1,\mu_1)s_2\ldots$

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

$$X_{C\le k}(\omega) = \begin{cases} 0 & \text{if } k = 0 \\ \sum_{i=0}^{k-1} \underline{\rho}(s_i) + \iota(a_i) & \text{otherwise} \end{cases}$$

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

where $k_\phi = \min\{\, i \mid s_i \vDash \phi \,\}$

# Model checking reward formulas

- Instantaneous: $R_{\sim r}$ [ $I^{=k}$ ]
  - similar the to computation of bounded until probabilities
  - solution of recursive equations
- Cumulative: $R_{\sim r}$ [ $C^{\leq k}$ ]
  - extension of bounded until computation
  - solution of recursive equations
- Reachability: $R_{\sim r}$ [ F $\phi$ ]
  - similar to the case for until
  - solve a linear optimization problem

# Model checking summary

- Atomic propositions and logical connectives: trivial
- Probabilistic operator P:
  - $X\ \Phi$ : one matrix-vector multiplications
  - $\Phi_1\ U^{\leq k}\ \Phi_2$ : k matrix-vector multiplications
  - $\Phi_1\ U\ \Phi_2$ : linear optimisation problem in at most $|S|$ variables
- Expected reward operator R
  - $I^{=k}$ : k matrix-vector multiplications
  - $C^{\leq k}$ : k iterations of matrix-vector multiplication + summation
  - $F\ \Phi$ : linear optimisation problem in at most $|S|$ variables

# Model checking complexity

- For model checking of an MDP $(S, s_{init}, \mathbf{Steps}, L)$ and PCTL formula $\phi$ (including reward operators)
  - complexity is linear in $|\Phi|$ and polynomial in $|S|$
- Size $|\phi|$ of $\phi$ is defined as number of logical connectives and temporal operators plus sizes of temporal operators
  - model checking is performed for each operator
- Worst-case operators are $P_{\sim p}$ [ $\phi_1$ U $\phi_2$ ] and $R_{\sim r}$ [ F $\phi$ ]
  - main task: solution of linear optimization problem of size $|S|$
  - can be solved with ellipsoid method (polynomial in $|S|$)
  - and also precomputation algorithms (max $|S|$ steps)

# Summing up…

- Nondeterminism
- Markov decision processes (MDPs)
  - definition, examples, adversaries, probabilities
- Properties of MDPs: The logic PCTL
  - syntax, semantics, equivalences, …
- PCTL model checking
  - algorithms, examples, …
- Costs and rewards