

Reachability in Markov Decision Processes

Lecture #22 of Advanced Model Checking

Joost-Pieter Katoen

Lehrstuhl 2: Software Modeling & Verification

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

July 22, 2009

thanks to Dave Parker (Oxford) for his slides

The importance of nondeterminism

- **Concurrency** – scheduling of parallel components
 - in randomised distributed algorithms several components interact asynchronously
- **Abstraction**
 - partition state space of a Markov chain in similar (but not identical) states
- **Unknown environments**
 - do not stipulate how the environment will behave, security: unknown adversary

“There is nothing mysterious about nondeterminism, it arises from the deliberated decision to ignore the factors which influence the selection”

Markov decision process

- Markov decision processes

- extension of Markov chains which allow nondeterministic choice
- in fact, a combination of Markov chains and labeled transition systems

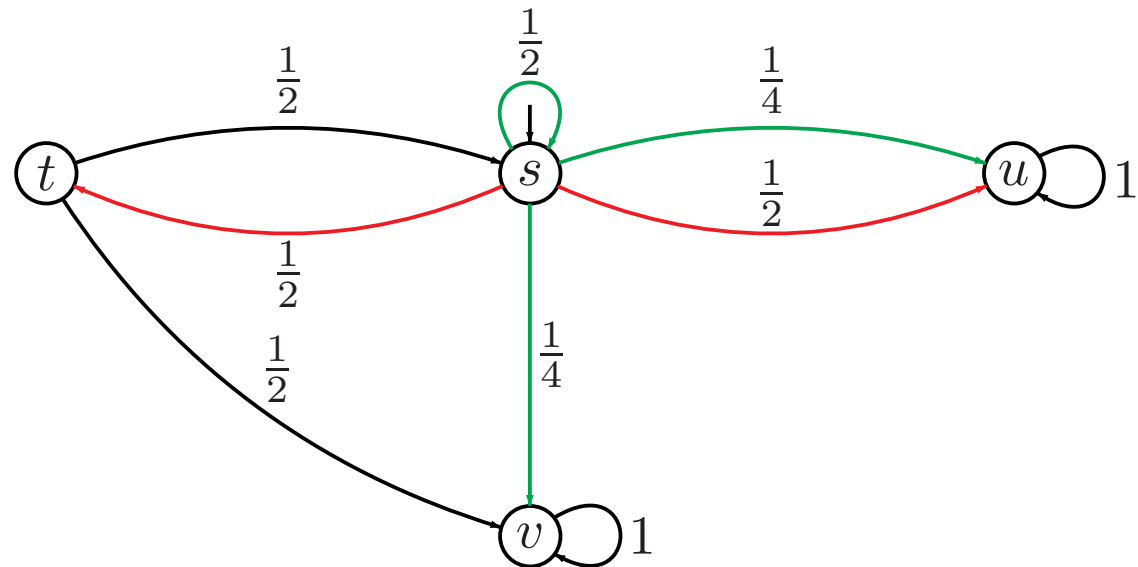
- Like Markov chains:

- discrete set of states representing the possible system configurations
- transitions between states occur in discrete time-steps

- Probabilities and nondeterminism

in each state, a nondeterministic choice between several probability distributions over successor states

Markov decision process



Markov decision process

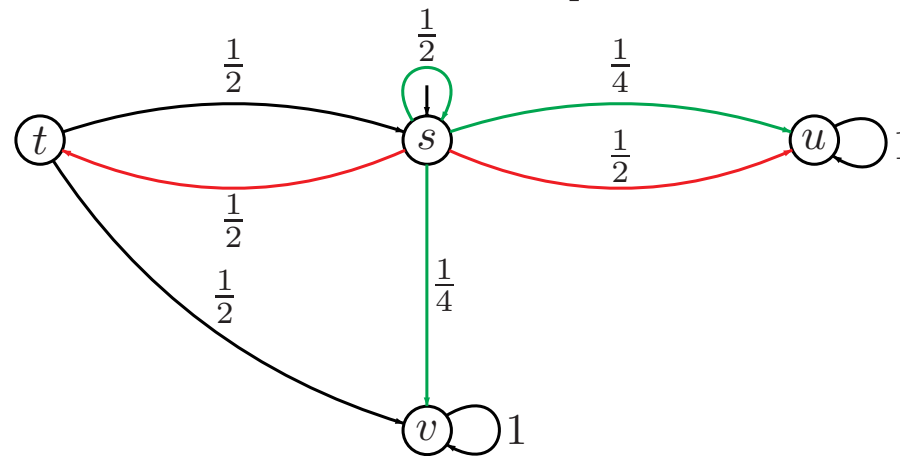
An MDP $\mathcal{M} = (S, Act, \mathbf{P}, \iota_{\text{init}}, AP, L)$ where

- S is a finite set of **states**
- Act is a finite set of **actions** and AP a set of **atomic propositions**
- $\mathbf{P} : S \times Act \times S \rightarrow [0, 1]$, **transition probability function** such that:

$$\text{for all } s \in S \text{ and } \alpha \in Act : \sum_{s' \in S} \mathbf{P}(s, \alpha, s') \in \{0, 1\}$$

- $\iota_{\text{init}} \in \text{Distr}(S)$, **initial state distribution**
- $L : S \rightarrow 2^{AP}$, **labeling** function of the states

Markov decision process

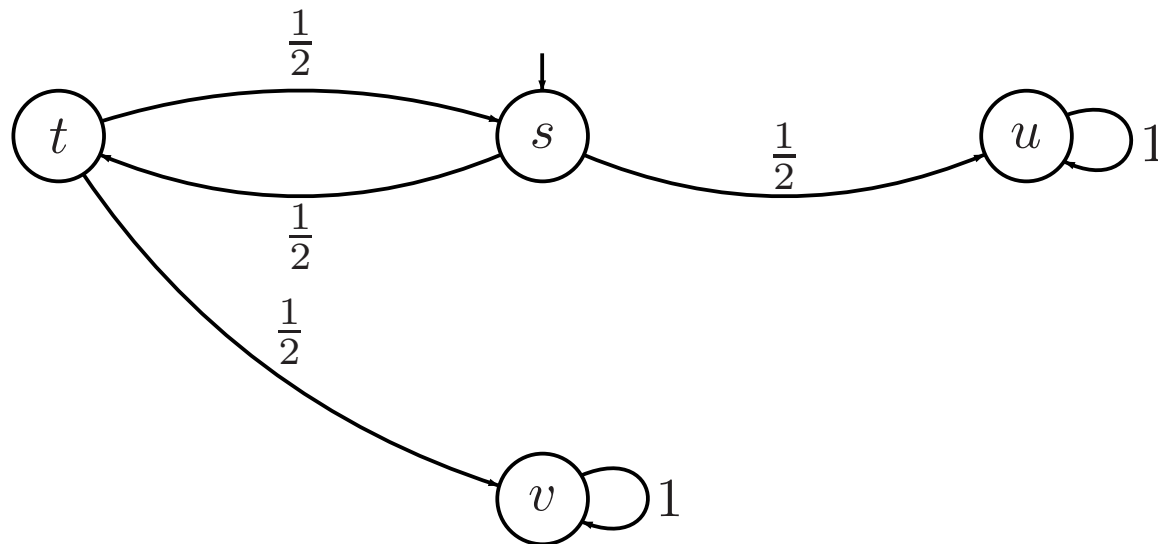


- Initial distribution: $\iota_{\text{init}}(s) = 1$ and $\iota_{\text{init}}(t) = \iota_{\text{init}}(u) = \iota_{\text{init}}(v) = 0$
- Set of enabled actions in state s is $\text{Act}(s) = \{ \alpha, \beta \}$ where
 - $\mathbf{P}(s, \alpha, s) = \frac{1}{2}$, $\mathbf{P}(s, \alpha, t) = 0$ and $\mathbf{P}(s, \alpha, u) = \mathbf{P}(s, \alpha, v) = \frac{1}{4}$
 - $\mathbf{P}(s, \beta, s) = \mathbf{P}(s, \beta, v) = 0$, and $\mathbf{P}(s, \beta, t) = \mathbf{P}(s, \beta, u) = \frac{1}{2}$
- $\text{Act}(t) = \{ \alpha \}$ with $\mathbf{P}(t, \alpha, s) = \mathbf{P}(t, \alpha, u) = \frac{1}{2}$ and 0 otherwise

Intuitive operational behaviour of MDPs

- Initial state is determined randomly according to ℓ_{init}
- On entering state s , choose an enabled action **nondeterministically**
 - let $Act(s) = \{ \alpha \mid \exists s' \in S. \mathbf{P}(s, \alpha, s') > 0 \}$ the enabled actions in s
 - the probability of selecting $\alpha \in Act(s)$ is **unknown**
- After selecting $\alpha \in Act(s)$, next state is t with probability $\mathbf{P}(s, \alpha, t)$
- Note: an MDP for which $|Act(s)| = 1$ for each s is a Markov chain

(Discrete-time) Markov chain



a Markov chain is an MDP in which each state has a single enabled action

Paths in an MDP

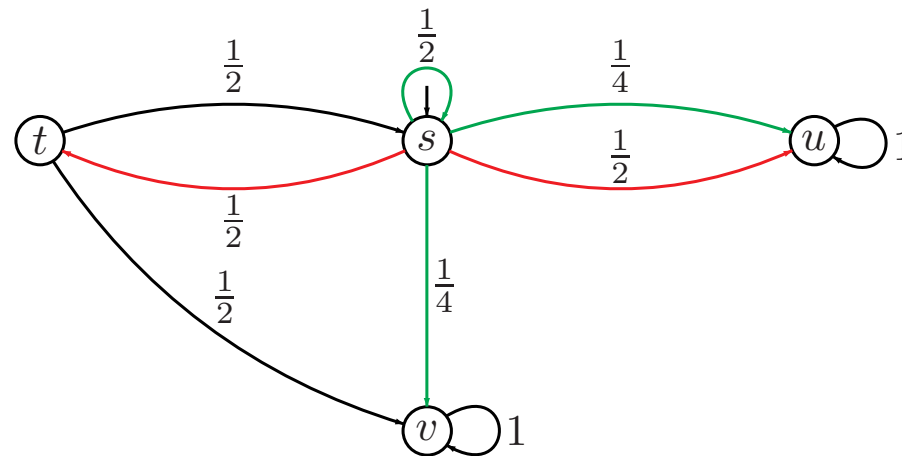
- An MDP is **finite** if the state space S , Act , and AP are finite
- The **state graph** of MDP \mathcal{M} is the digraph $G = (V, E)$
 - $V = S$ are the states of \mathcal{M} , and $(s, t) \in E$ iff $\mathbf{P}(s, \alpha, t) > 0$ for some α
- **Paths** in an MDP are alternating sequences of states and actions

$$\pi = s_0 \xrightarrow{\alpha_1} s_1 \xrightarrow{\alpha_2} s_2 \xrightarrow{\alpha_3} \dots$$

such that $\mathbf{P}(s_i, \alpha_{i+1}, s_{i+1}) > 0$ for all $i \geq 0$

- Paths represent possible behaviours of an MDP

Example paths



$$s \xrightarrow{\alpha} s \xrightarrow{\alpha} s \xrightarrow{\beta} t \xrightarrow{\alpha} s \xrightarrow{\beta} u \dots$$

$$s \xrightarrow{\beta} t \xrightarrow{\alpha} s \xrightarrow{\beta} t \xrightarrow{\alpha} s \dots\dots\dots$$

Probability measure for MDPs?

nondeterministically choosing between a fair and unfair coin

Policies

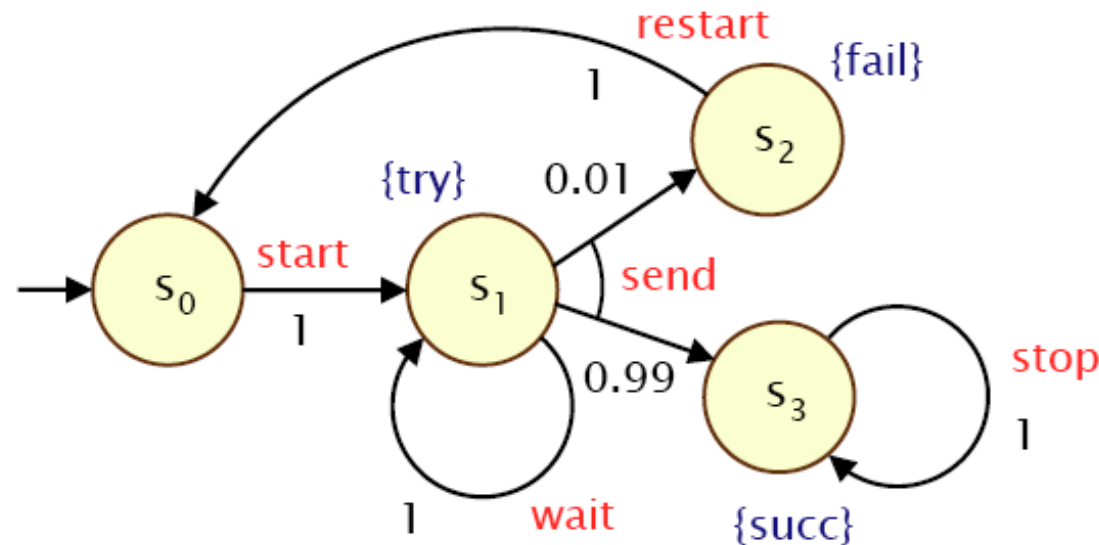
- To consider the probability of a path of an MDP
 - first need to **resolve the nondeterministic choices**
 - . . . which results in a Markov chain
 - . . . for which a probability measure over infinite paths was defined
- A **policy** resolves nondeterministic choices in an MDP
 - alternative terminology: adversary, scheduler, tactic, strategy, . . .
- Formally: $\mathfrak{S} : S^+ \rightarrow Act$ is a **policy** with

$$\mathfrak{S}(\underbrace{s_0 s_1 \dots s_n}_{\text{history}}) \in \underbrace{\{ \alpha \mid \exists s \in S. \mathbf{P}(s_n, \alpha, s) > 0 \}}_{Act(s_n)}$$

note: actions are not part of the history since $\alpha_{i+1} = \mathfrak{S}(s_0 \dots s_i)$

Another simple MDP

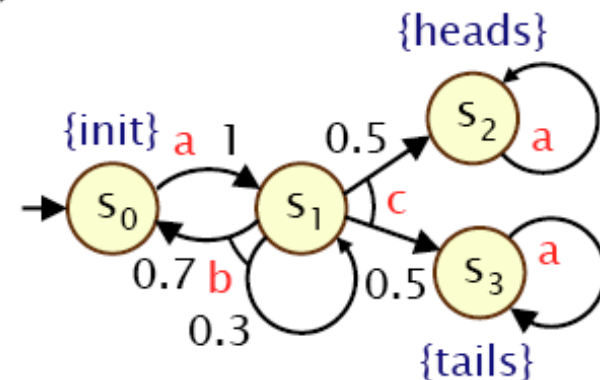
- after one step, process **starts** trying to send a message
- then, a nondeterministic choice between: (a) **waiting** a step because the channel is unread; (b) **sending** the message
- if the latter, with probability 0.99 send **successfully** and **stop**
- and with probability 0.01, message sending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



Example policies

- $\text{Path}^A(s) \subseteq \text{Path}(s)$
 - (infinite) paths from s where nondeterminism resolved by A
 - i.e. paths $s_0(a_0, \mu_0)s_1(a_1, \mu_1)s_2 \dots$
 - for which $A(s_0(a_0, \mu_0)s_1 \dots s_n)) = (a_n, \mu_n)$

- Adversary A_1
 - (picks action c the first time)
 - $\text{Path}^{A_1}(s_0) = \{ s_0s_1s_2^\omega, s_0s_1s_3^\omega \}$

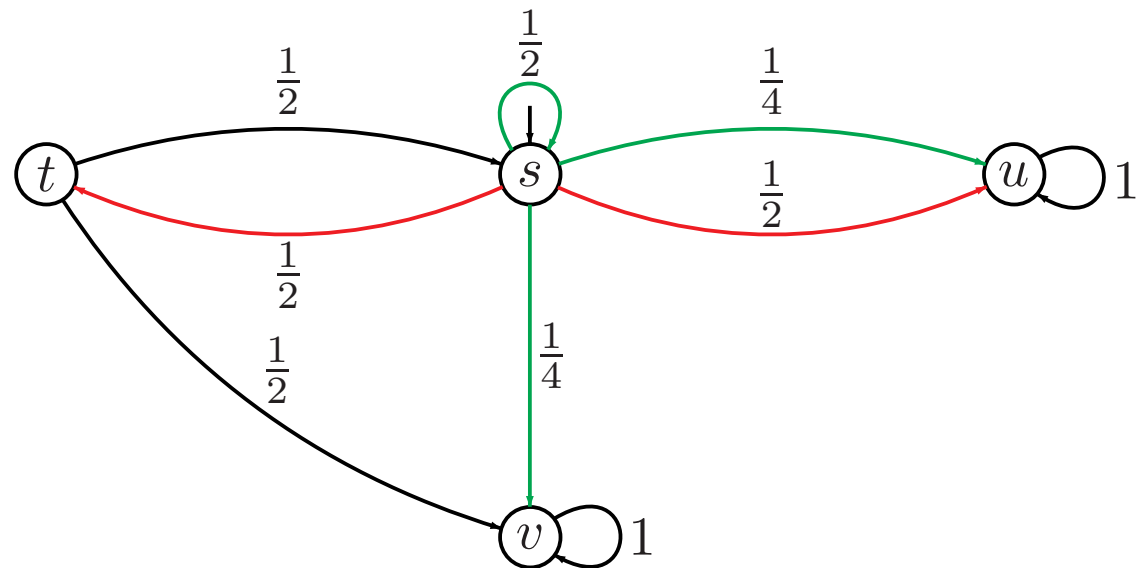


- Adversary A_2
 - (picks action b the first time, then c)
 - $\text{Path}^{A_2}(s_0) = \{ s_0s_1s_0s_1s_2^\omega, s_0s_1s_0s_1s_3^\omega, s_0s_1s_1s_2^\omega, s_0s_1s_1s_3^\omega \}$

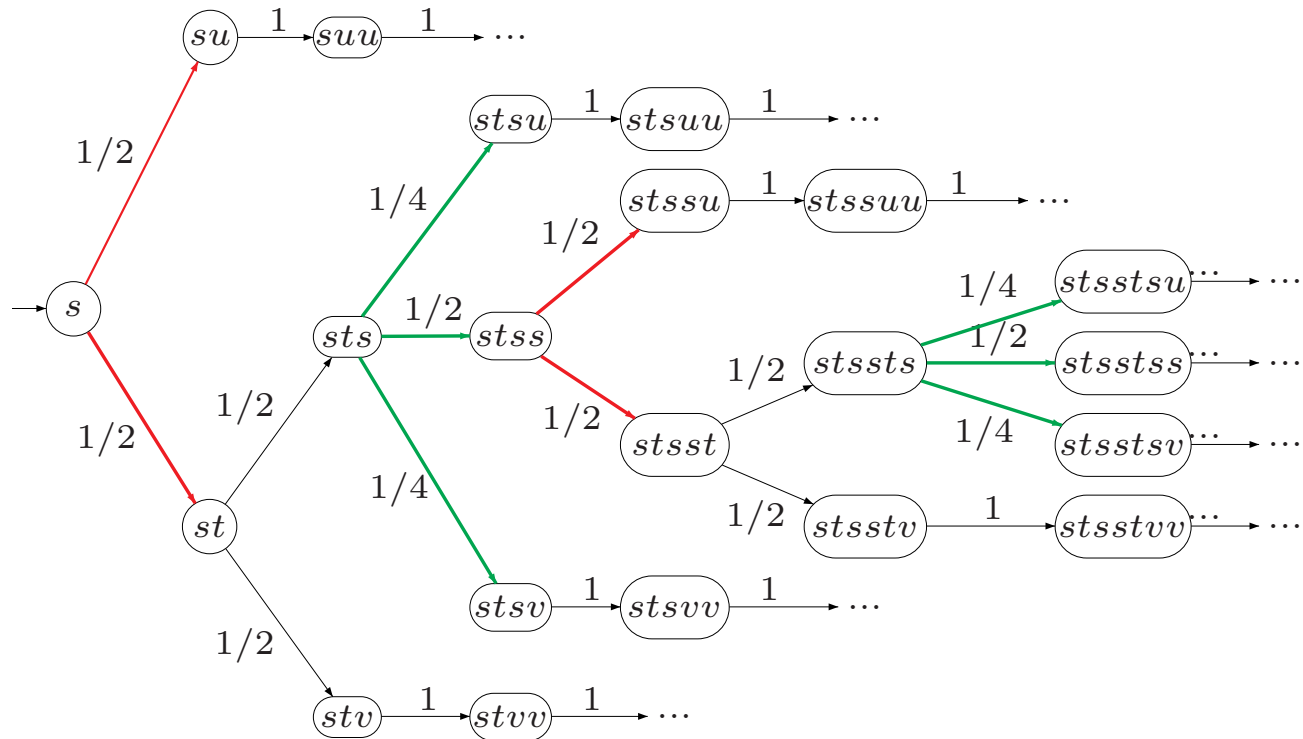
Applying a policy: induced Markov chain

- Policy \mathfrak{S} for MDP \mathcal{M} induces an infinite-state **MC** $\mathcal{M}_{\mathfrak{S}}$
- Unfold MDP \mathcal{M} , resolving all nondeterminism according to \mathfrak{S}
- This yields the infinite **Markov chain** $\mathcal{M}_{\mathfrak{S}} = (S_{\mathfrak{S}}, \mathbf{P}_{\mathfrak{S}}, \iota_{\text{init}}, L)$ with:
 - $S_{\mathfrak{S}} = S^+$, nonempty state sequences in MDP \mathcal{M}
 - $\mathbf{P}_{\mathfrak{S}}(\pi, \pi \rightarrow s) = \mathbf{P}(\text{last}(\pi), \mathfrak{S}(\pi), s)$ and 0 otherwise
 - $L(\text{last}(\pi)) = L(\pi)$
- 1-to-1 correspondence between $\text{Paths}^{\mathfrak{S}}(s)$ and paths in $\mathcal{M}_{\mathfrak{S}}$
- This gives us a **probability measure** $Pr^{\mathcal{M}_{\mathfrak{S}}}$ over $\text{Paths}^{\mathfrak{S}}(s)$
 - from probability measure over infinite paths of MC $\mathcal{M}_{\mathfrak{S}}$

Markov decision process



Applying a policy



Example policy = alternate between red and green

Reachability probabilities in MDPs

- Reachability probability of set $B \subseteq S$ from state s :

$$Pr^{\mathcal{G}}(s \models \Diamond B) = Pr_s^{\mathcal{M}^{\mathcal{G}}} \{ \pi \in Paths(s) \mid \pi \models \Diamond B \}$$

- ω -regular properties (and many more) are also measurable
- $\forall \mathcal{G}. Pr^{\mathcal{G}}(s \models \Diamond B) \leq \varepsilon$ implies $\forall \mathcal{G}. Pr^{\mathcal{G}}(s \models \Box \neg B) \geq 1 - \varepsilon$

Reachability probabilities in MDPs

- Reachability probability of set $B \subseteq S$ from state s :

$$Pr^{\mathcal{G}}(s \models \Diamond B) = Pr_s^{\mathcal{M}^{\mathcal{G}}} \{ \pi \in Paths(s) \mid \pi \models \Diamond B \}$$

$$- \forall \mathcal{G}. Pr^{\mathcal{G}}(s \models \Diamond B) \leq \varepsilon \text{ implies } \forall \mathcal{G}. Pr^{\mathcal{G}}(s \models \Box \neg B) \geq 1 - \varepsilon$$

- Analysis focuses on obtaining lower- and upperbounds, e.g.,

$$Pr^{\min}(s \models \Diamond B) = \inf_{\mathcal{G}} Pr^{\mathcal{G}}(s \models \Diamond B) \quad \text{and}$$

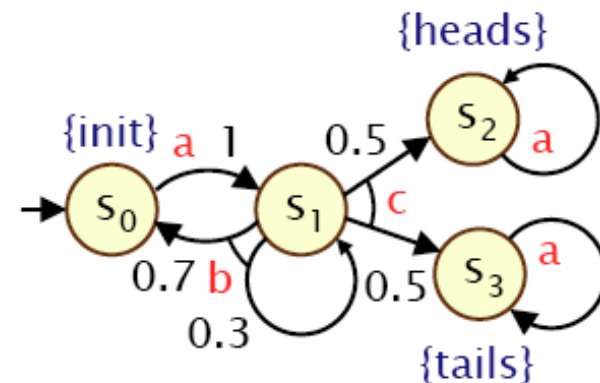
$$Pr^{\max}(s \models \Diamond B) = \sup_{\mathcal{G}} Pr^{\mathcal{G}}(s \models \Diamond B)$$

note: \mathcal{G} ranges over all, potentially infinitely many, policies

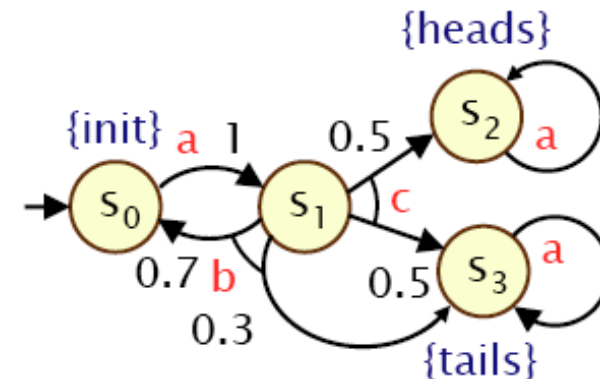
- And on determining policies for these bounds

Example reachability probabilities

- $\text{Prob}^{A^1}(s_0, F \text{ tails}) = 0.5$
- $\text{Prob}^{A^2}(s_0, F \text{ tails}) = 0.5$
 - (where A_i picks b $i-1$ times then c)
- ...
- $p_{\max}(s_0, F \text{ tails}) = 0.5$
- $p_{\min}(s_0, F \text{ tails}) = 0$



- $\text{Prob}^{A^1}(s_0, F \text{ tails}) = 0.5$
- $\text{Prob}^{A^2}(s_0, F \text{ tails}) = 0.3 + 0.7 \cdot 0.5 = 0.65$
- $\text{Prob}^{A^3}(s_0, F \text{ tails}) = 0.3 + 0.7 \cdot 0.3 + 0.7 \cdot 0.7 \cdot 0.5 = 0.755$
- ...
- $p_{\max}(s_0, F \text{ tails}) = 1$
- $p_{\min}(s_0, F \text{ tails}) = 0.5$



Classes of policies

- **Memoryless** policy: always pick the same action in a given state

- for state sequences $s_1 \dots s_m$ and $t_1 \dots t_n$ with $s_m = t_n$:

$$\mathfrak{S}(s_1 \dots s_m) = \mathfrak{S}(t_1 \dots t_n)$$

- selection is based on given state only; induces a finite-state MC

- **Finite-memory** policy: selection is modeled by a DFA

- selection is based on current state of MDP and current state of DFA

- **Counting** policy:

- selection is based on number of visits to states so far

Optimality for maximum reachability

Let \mathcal{M} be a finite MDP with state space S , $s \in S$ and $B \subseteq S$

The values $x_s = Pr^{\max}(s \models \Diamond B)$ are the unique solution of:

- If $s \in B$, then $x_s = 1$.
- If $s \not\models \exists \Diamond B$, then $x_s = 0$.
- If $s \notin B$ and $s \models \exists \Diamond B$, then

$$x_s = \max \left\{ \sum_{t \in S} \mathbf{P}(s, \alpha, t) \cdot x_t \mid \alpha \in \mathbf{Act}(s) \right\}$$

this is an instance of the Bellman equation for dynamic programming

Optimality for minimum reachability

Let \mathcal{M} be a finite MDP with state space S , $s \in S$ and $B \subseteq S$

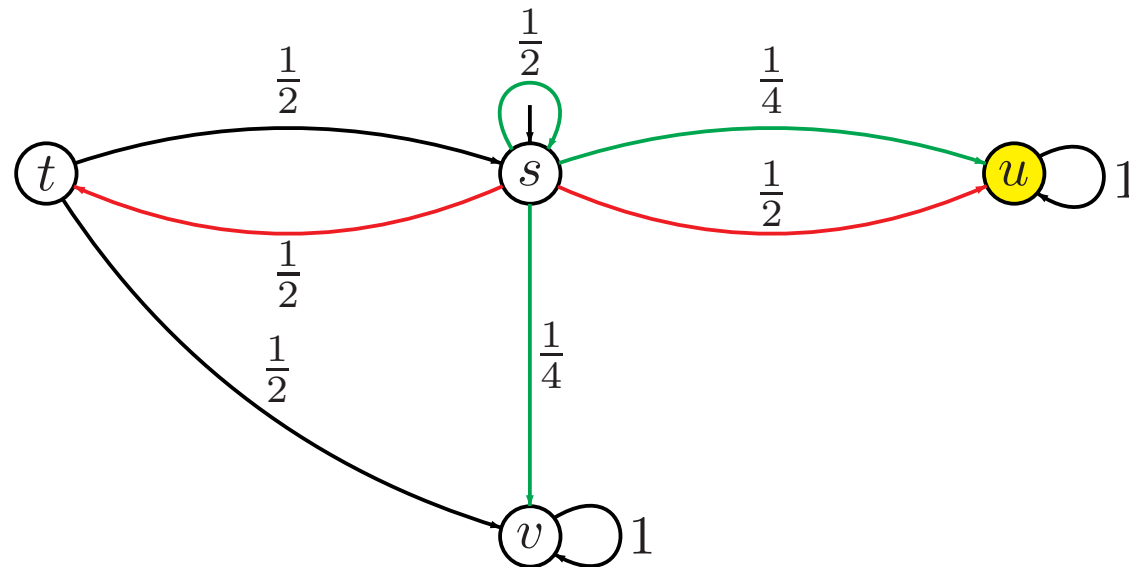
The values $x_s = Pr^{\min}(s \models \Diamond B)$ are the unique solution of:

- If $s \in B$, then $x_s = 1$.
- If $s \not\models \forall \Diamond B$, then $x_s = 0$.
- If $s \notin B$ and $s \models \forall \Diamond B$, then

$$x_s = \min \left\{ \sum_{t \in S} \mathbf{P}(s, \alpha, t) \cdot x_t \mid \alpha \in \mathbf{Act}(s) \right\}$$

this is an instance of the Bellman equation for dynamic programming

Example (max)



equation system for reachability objective $\Diamond \{ u \}$ is:

$$x_u = 1 \text{ and } x_v = 0$$

$$x_s = \max \left\{ \frac{1}{2}x_s + \frac{1}{4}x_u + \frac{1}{4}x_v, \frac{1}{2}x_u + \frac{1}{2}x_t \right\} \quad \text{and} \quad x_t = \frac{1}{2}x_s + \frac{1}{2}x_v$$

Existence of optimal memoryless policy

There exists a memoryless scheduler \mathfrak{S} such that for any $s \in S$

$$Pr^{\mathfrak{S}}(s \models \Diamond B) = Pr^{\max}(s \models \Diamond B)$$

Proof

Value iteration (max)

Calculate values $x_s = Pr^{\text{max}}(s \models \Diamond B)$ by successive approximation

For the states $s \in Pre^*(B) \setminus B$ we have

$$x_s = \lim_{n \rightarrow \infty} x_s^{(n)}$$

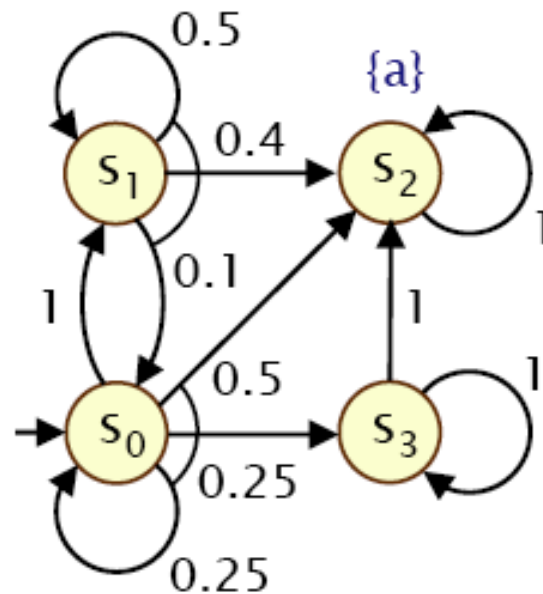
where

$$x_s^{(0)} = 0 \quad \text{and} \quad x_s^{(n+1)} = \text{max} \left\{ \sum_{t \in S} \mathbf{P}(s, \alpha, t) \cdot x_t^{(n)} \mid \alpha \in \mathbf{Act}(s) \right\}$$

for minimal probabilities, a similar strategy works

Value iteration: example

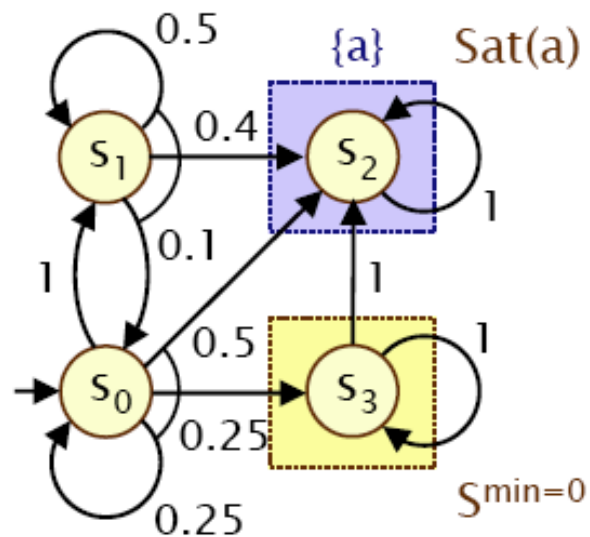
- Minimum/maximum probability of reaching an **a**-state



Value iteration: example

Compute: $p_{\min}(s_i, F a)$

$Sat(a) = \{s_2\}$, $S^{\min=0} = \{s_3\}$, $S^? = \{s_0, s_1\}$



$[x_0^{(n)}, x_1^{(n)}, x_2^{(n)}, x_3^{(n)}]$

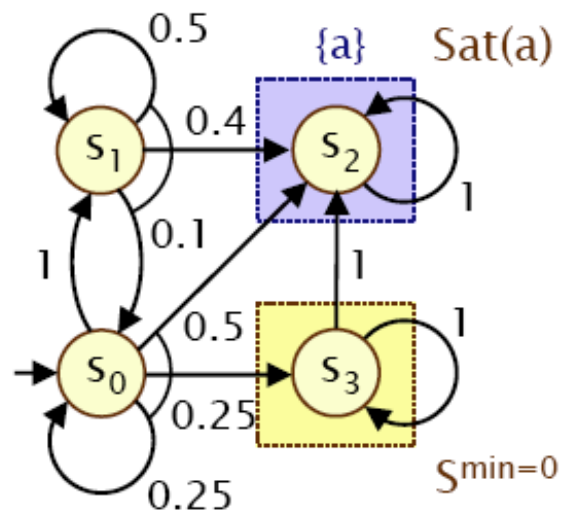
$n=0: [0, 0, 1, 0]$

$n=1: [\min(1 \cdot 0, 0.25 \cdot 0 + 0.25 \cdot 0 + 0.5 \cdot 1),$
 $0.1 \cdot 0 + 0.5 \cdot 0 + 0.4 \cdot 1, 1, 0]$
 $= [0, 0.4, 1, 0]$

$n=2: [\min(1 \cdot 0.4, 0.25 \cdot 0 + 0.25 \cdot 0 + 0.5 \cdot 1),$
 $0.1 \cdot 0 + 0.5 \cdot 0.4 + 0.4 \cdot 1, 1, 0]$
 $= [0.4, 0.6, 1, 0]$

$n=3: \dots$

Value iteration: example



$$\begin{aligned} & \underline{p}_{\min}(F a) \\ & = \\ & [2/3, 14/15, 1, 0] \end{aligned}$$

$$[x_0^{(n)}, x_1^{(n)}, x_2^{(n)}, x_3^{(n)}]$$

$$n=0: [0.000000, 0.000000, 1, 0]$$

$$n=1: [0.000000, 0.400000, 1, 0]$$

$$n=2: [0.400000, 0.600000, 1, 0]$$

$$n=3: [0.600000, 0.740000, 1, 0]$$

$$n=4: [0.650000, 0.830000, 1, 0]$$

$$n=5: [0.662500, 0.880000, 1, 0]$$

$$n=6: [0.665625, 0.906250, 1, 0]$$

$$n=7: [0.666406, 0.919688, 1, 0]$$

$$n=8: [0.666602, 0.926484, 1, 0]$$

...

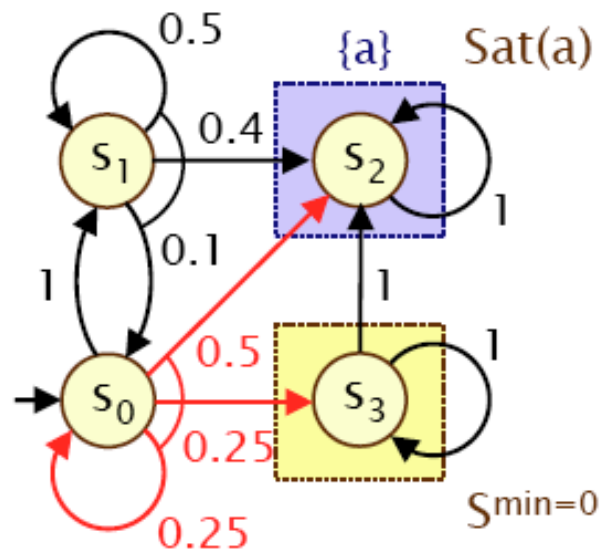
$$n=20: [0.666667, 0.933332, 1, 0]$$

$$n=21: [0.666667, 0.933332, 1, 0]$$

$$\approx [2/3, 14/15, 1, 0]$$

Generating an optimal policy

- Min adversary A_{\min}



$$[x_0^{(n)}, x_1^{(n)}, x_2^{(n)}, x_3^{(n)}]$$

...

$$n=20: [0.666667, 0.933332, 1, 0]$$

$$n=21: [0.666667, 0.933332, 1, 0]$$

$$\approx [2/3, 14/15, 1, 0]$$

$$s_0 : \min(1 \cdot 14/15, 0.5 \cdot 1 + 0.5 \cdot 0 + 0.25 \cdot 2/3) \\ = \min(14/15, 2/3)$$

Maximal reach probabilities as a linear program

Consider a finite MDP with state space S , and $B \subseteq S$

The values $x_s = Pr^{\max}(s \models \Diamond B)$ are the unique solution of the *linear program*:

- If $s \in B$, then $x_s = 1$.
- If $s \not\models \exists \Diamond B$, then $x_s = 0$.
- If $s \notin B$ and $s \models \exists \Diamond B$, then $0 \leq x_s \leq 1$ and for all actions $\alpha \in Act(s)$:

$$x_s \geq \sum_{t \in S} \mathbf{P}(s, \alpha, t) \cdot x_t$$

where $\sum_{s \in S} x_s$ is *minimal*

solution techniques e.g., Simplex, ellipsoid techniques, . . .

Minimal reach probabilities as a linear program

Consider a finite MDP with state space S , and $B \subseteq S$

The values $x_s = Pr^{\min}(s \models \Diamond B)$ are the unique solution of the *linear program*:

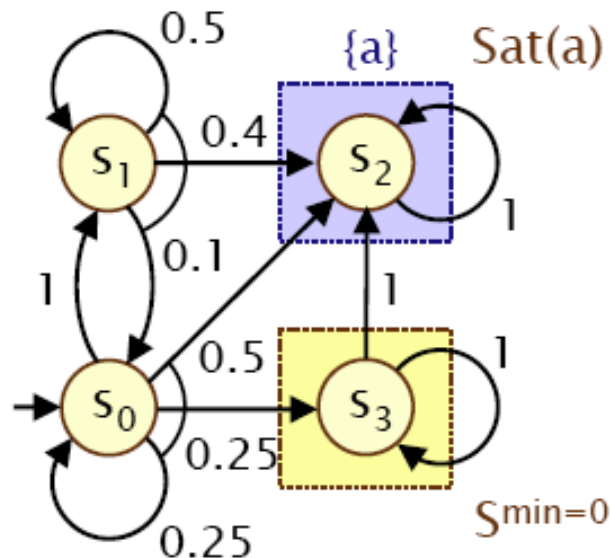
- If $s \in B$, then $x_s = 1$.
- If $s \not\models \forall \Diamond B$, then $x_s = 0$.
- If $s \notin B$ and $s \models \forall \Diamond B$, then $0 \leq x_s \leq 1$ and for all actions $\alpha \in Act(s)$:

$$x_s \leq \sum_{t \in S} \mathbf{P}(s, \alpha, t) \cdot x_t$$

where $\sum_{s \in S} x_s$ is *maximal*

solution techniques e.g., Simplex, ellipsoid techniques, . . .

Example linear optimisation (min)



Let $x_i = p_{\min}(s_i, F a)$

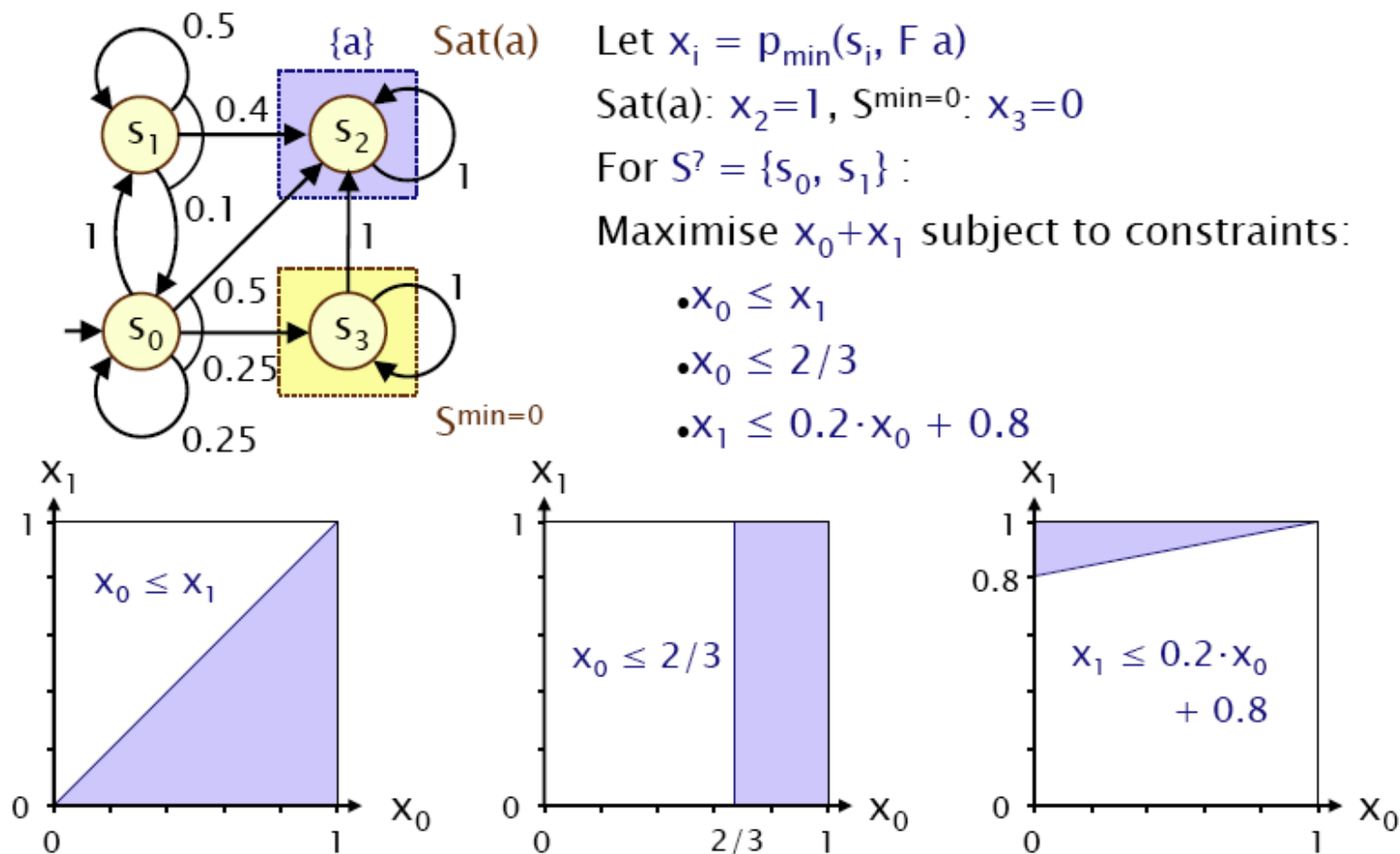
$\text{Sat}(a)$: $x_2 = 1$, $S^{\min=0}$: $x_3 = 0$

For $S^? = \{s_0, s_1\}$:

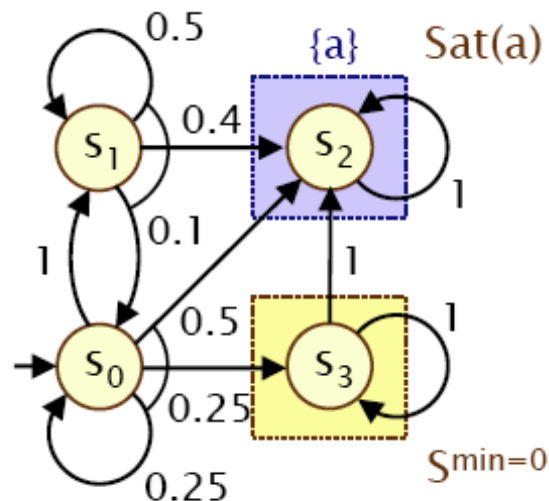
Maximise $x_0 + x_1$ subject to constraints:

- $x_0 \leq x_1$
- $x_0 \leq 0.25 \cdot x_0 + 0.5$
- $x_1 \leq 0.1 \cdot x_0 + 0.5 \cdot x_1 + 0.4$

Example linear optimisation (min)



Example linear optimisation (min)



Let $x_i = p_{\min}(s_i, F a)$

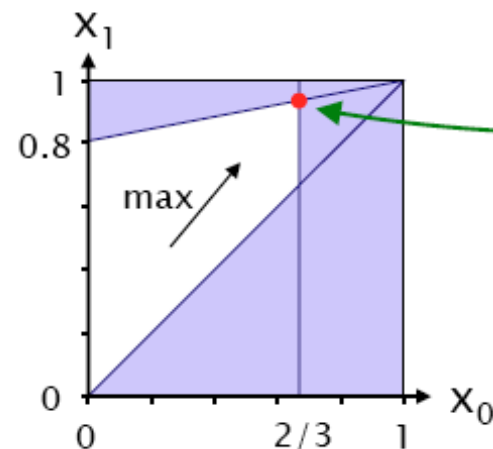
Sat(a): $x_2=1$, $S^{\min=0}$: $x_3=0$

For $S^? = \{s_0, s_1\}$:

Maximise $x_0 + x_1$ subject to constraints:

- $x_0 \leq x_1$
- $x_0 \leq 2/3$
- $x_1 \leq 0.2 \cdot x_0 + 0.8$

$$p_{\min}(F a) = [2/3, 14/15, 1, 0]$$



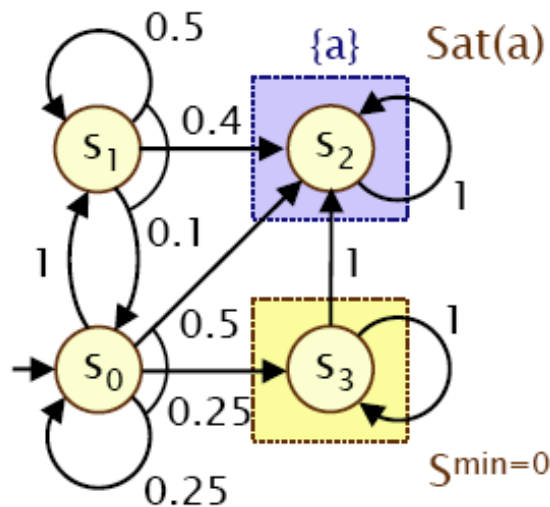
Solution:

$$(x_0, x_1)$$

=

$$(2/3, 14/15)$$

Example linear optimisation (min)



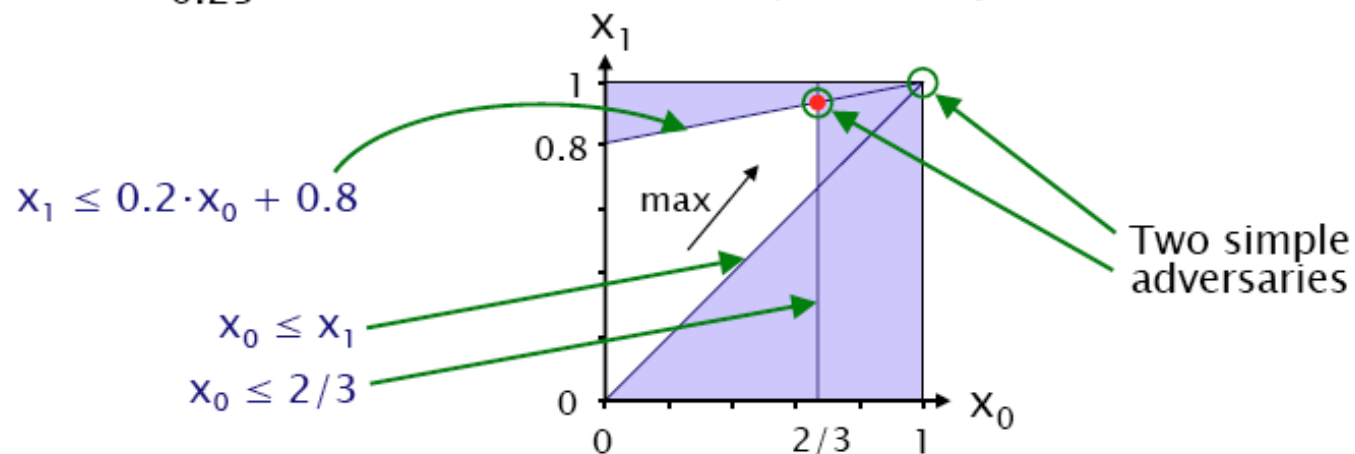
Let $x_i = p_{\min}(s_i, F a)$

Sat(a): $x_2=1$, $S^{\min}=0$: $x_3=0$

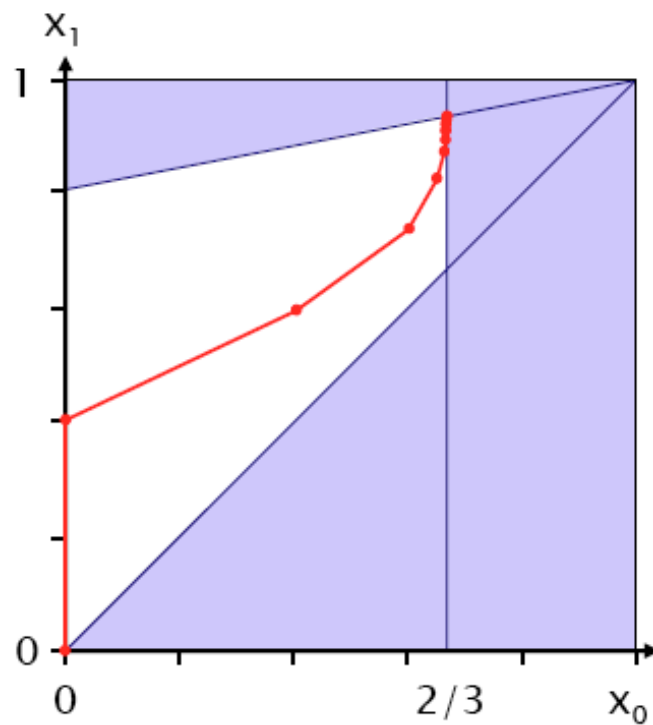
For $S^? = \{s_0, s_1\}$:

Maximise x_0+x_1 subject to constraints:

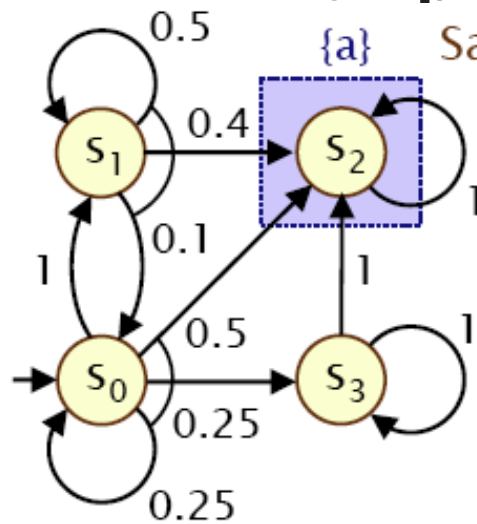
- $x_0 \leq x_1$
- $x_0 \leq 2/3$
- $x_1 \leq 0.2 \cdot x_0 + 0.8$



Value iteration vs. LP


 $[x_0^{(n)}, x_1^{(n)}, x_2^{(n)}, x_3^{(n)}]$
 $n=0: [0.000000, 0.000000, 1, 0]$
 $n=1: [0.000000, 0.400000, 1, 0]$
 $n=2: [0.400000, 0.600000, 1, 0]$
 $n=3: [0.600000, 0.740000, 1, 0]$
 $n=4: [0.650000, 0.830000, 1, 0]$
 $n=5: [0.662500, 0.880000, 1, 0]$
 $n=6: [0.665625, 0.906250, 1, 0]$
 $n=7: [0.666406, 0.919688, 1, 0]$
 $n=8: [0.666602, 0.926484, 1, 0]$
 \dots
 $n=20: [0.666667, 0.933332, 1, 0]$
 $n=21: [0.666667, 0.933332, 1, 0]$
 $\approx [2/3, 14/15, 1, 0]$

Example linear optimisation (max)



Sat(a)

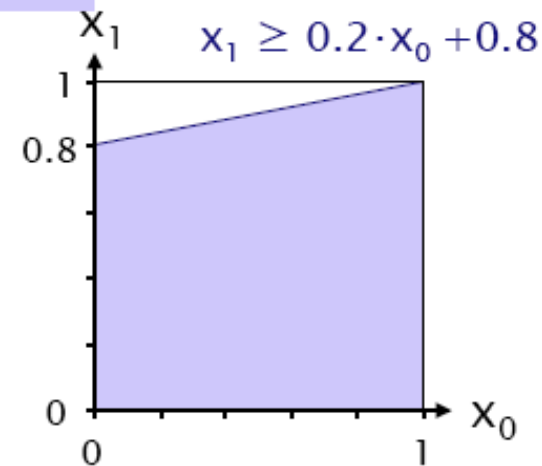
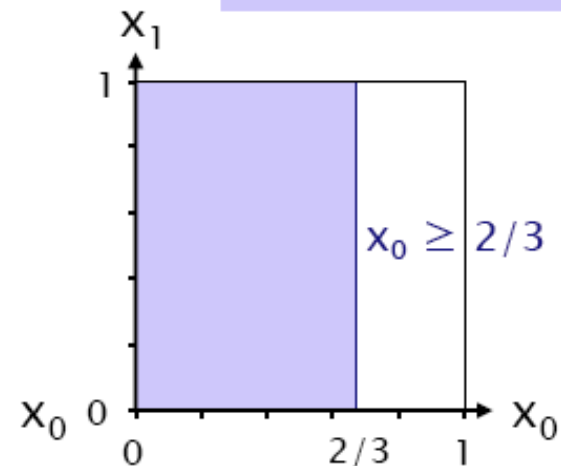
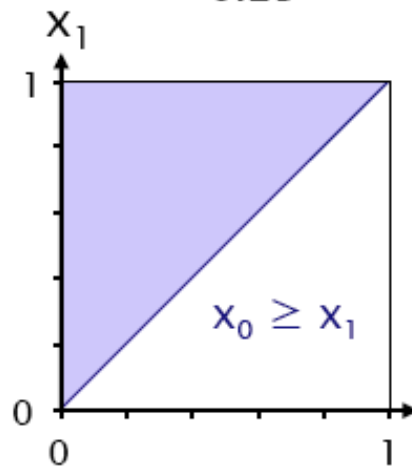
Let $x_i = p_{\max}(s_i, F a)$

Sat(a): $x_2 = 1$, $S^{\max=0} = \emptyset$

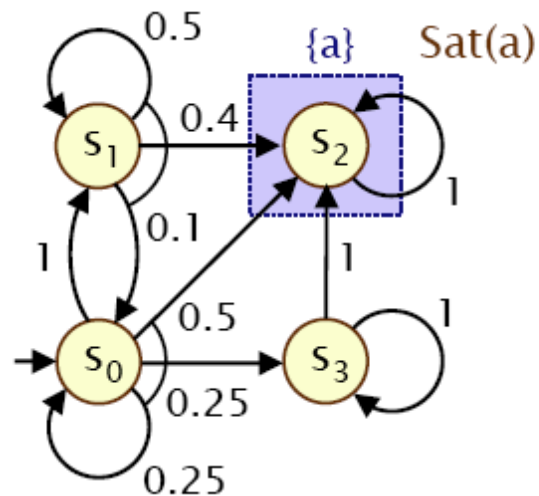
For $S^? = \{s_0, s_1, s_3\}$:

Minimise $x_0 + x_1 + x_3$ subject to constraints:

- $x_0 \geq x_1$
- $x_0 \geq 2/3$
- $x_1 \geq 0.2 \cdot x_0 + 0.8$
- $x_3 \geq x_2$
- $x_3 \geq x_3$



Example linear optimisation (max)



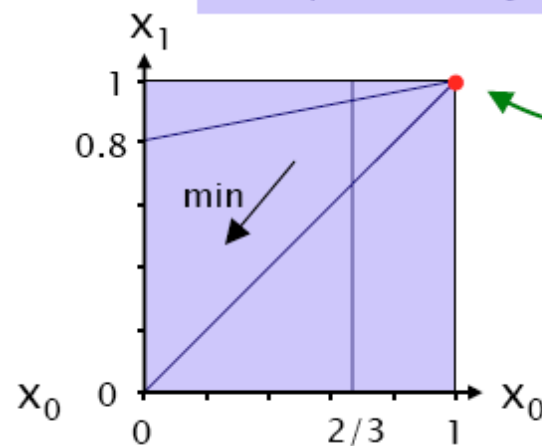
Let $x_i = p_{\max}(s_i, F a)$

Sat(a): $x_2 = 1$, $S^{\max=0} = \emptyset$

For $S^? = \{s_0, s_1, s_3\}$:

Minimise $x_0 + x_1 + x_3$ subject to constraints:

- $x_0 \geq x_1$
- $x_0 \geq 2/3$
- $x_1 \geq 0.2 \cdot x_0 + 0.8$
- $x_3 \geq x_2$
- $x_3 \geq x_3$



(only feasible)

solution:

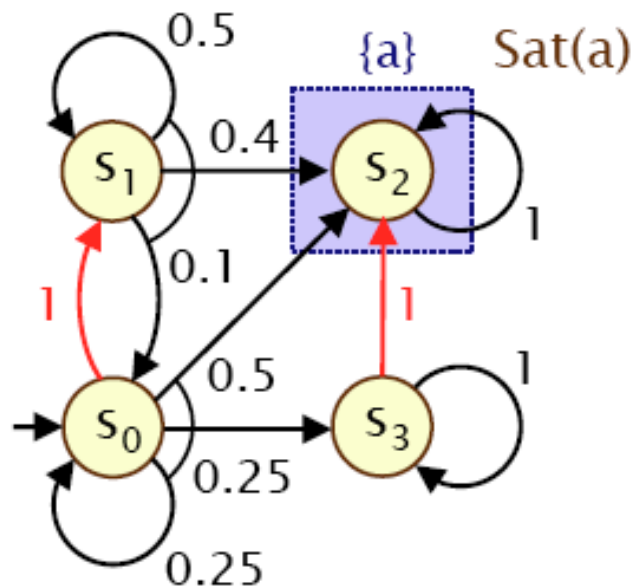
(x_0, x_1, x_2)

=

$(1, 1, 1)$

Example linear optimisation (max)

- Max adversary A_{\max}



Let $x_i = p_{\max}(s_i, F a)$

Sat(a): $x_2 = 1$, $S^{\max=0} = \emptyset$

For $S^? = \{s_0, s_1, s_3\}$:

Minimise $x_0 + x_1 + x_3$ subject to constraints:

- $x_0 \geq x_1$
- $x_3 \geq x_2$
- $x_0 \geq 2/3$
- $x_3 \geq x_3$
- $x_1 \geq 0.2 \cdot x_0 + 0.8$

Solution:

- $(x_0, x_1, x_2) = (1, 1, 1)$