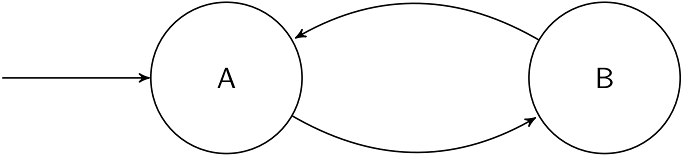
Probabilistic Models

Stefan Ratschan

Katedra číslicového návrhu
Fakulta informačních technologií
České vysoké učení technické v Praze

Evropský sociální fond Praha & EU: Investujeme do vaší budoucnosti

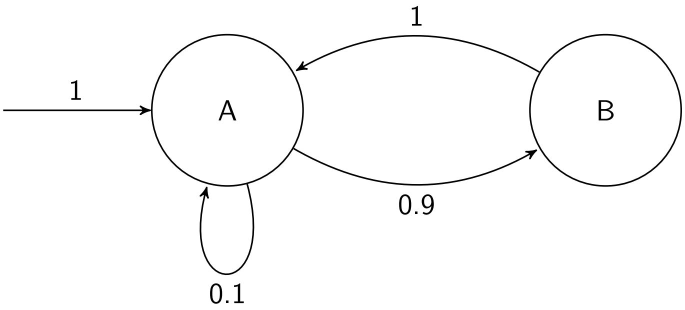
Motivation



$\models \mathbf{GF}b? \quad (\mathcal{I}(b) = \{B\})$
$\mathcal{P}(\mathbf{GF}b) = 1$

What is this? MI-SPI?

Markov Chains



Variant: *Probabilistic Transition System*:

- Countable set of states S
- *Initial distribution* $S_0 : S \rightarrow [0, 1]$ s.t. $\sum_{s \in S} S_0(s) = 1$
- *Transition probability matrix* $R : S \times S \rightarrow [0, 1]$ s.t.
  for all $s \in S$, $\sum_{s' \in S} R(s, s') = 1$

Probabilistic Transition Systems

- Countable set of states S
- *Initial distribution* $S_0 : S \to [0, 1]$ s.t. $\sum_{s \in S} S_0(s) = 1$
- *Transition probability matrix* $R : S \times S \to [0, 1]$ s.t.
  for all $s \in S$, $\sum_{s' \in S} R(s, s') = 1$

Graphical representation: no edge $=$ zero transition probability

Distribution after n steps, stationary distribution, ... see MI-SPI.

For classical transition systems
a given sequence of states either is a path or it is no path

Terminology here: *path*: any sequence of states (i.e., element of Σ_S).

For a given path $s_0, \dots,$ its probability?
$S_0(s_0) \prod_{i=0}^{\infty} R(s_i, s_{i+1})$?

Example: $ABABAB \cdots = 0.9 \times 1 \times 0.9 \times 1 \times 0.9 \times \dots$ So: usually 0?

Probability of Paths

We know the probability of one transition, two transitions, ...

Probability of a finite sequence of states:

$$\begin{aligned}\mathcal{P}_{(S,S_0,R)}(s_0,\dots,s_{n-1}) &:= S_0(s_0)R(s_0,s_1)\dots R(s_{n-2},s_{n-1})\\ &= S_0(s_0)\prod_{i\in\{1,\dots,n-1\}} R(s_{i-1},s_i)\end{aligned}$$

Infinite sequences of states?

σ-algebra, measurability, probability measure (see BI-PST)

Fundamental idea:

Measure probability of system staying in a set of paths

Probability of Set of Paths

Which sets of paths can we assign probability to?

Intuition: Follow finite sequence of states, then anything.

$C_{s_0,\ldots,s_{n-1}}$ (*cylinder*):
set of all paths with prefix $s_0, \ldots, s_{n-1}$

Especially: $C_{.} = \Sigma_S$

Probability of cylinders:

$$\mathcal{P}_{(S,S_0,R)}(C_{s_0,\ldots,s_{n-1}}) := \mathcal{P}_{(S,S_0,R)}(s_0, \ldots, s_{n-1})$$

Especially $\mathcal{P}_{(S,S_0,R)}(C_{.}) = \mathcal{P}_{(S,S_0,R)}(\Sigma_S) = 1.$

Shortcut: $\mathcal{P}()$ for $\mathcal{P}_{(S,S_0,R)}()$

Probability of Set of Paths: Generalization

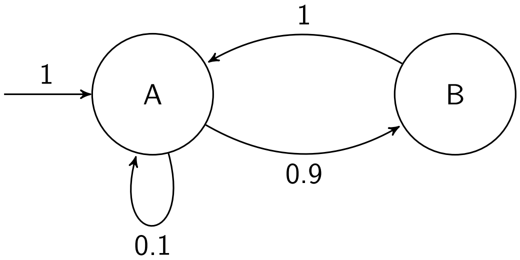
Extension from cylinders to
a probability measure on all sets of paths
that will be relevant for us (viz. Banach-Tarski paradox)

Hence:

- $\mathcal{P}(\emptyset) = 0$
- $\mathcal{P}(\Sigma_S) = 1$
- $\mathcal{P}(A \cup B) = \mathcal{P}(A) + \mathcal{P}(B)$, if $A \cap B = \emptyset$.

Specification of Properties

Various probabilistic variants of temporal logic. Here: LTL:



$\mathcal{I}(b) = \{B\}$

$\not\models \mathbf{GF}b$

$\mathcal{P}(\mathbf{GF}b) = 1$

Non-probabilistic case: $\models \phi$ iff
for all paths π of the transition system, $\pi \models \phi$

Probabilistic case, instead $\mathcal{P}_{(S,S_0,R)}(\phi)$, where

$$\mathcal{P}_{(S,S_0,R)}(\phi) := \mathcal{P}_{(S,S_0,R)}(\{\pi \in \Sigma_S \mid \pi \models \phi\})$$

The set $\{\pi \in \Sigma_S \mid \pi \models \phi\}$ is always measurable [Vardi, 1985].

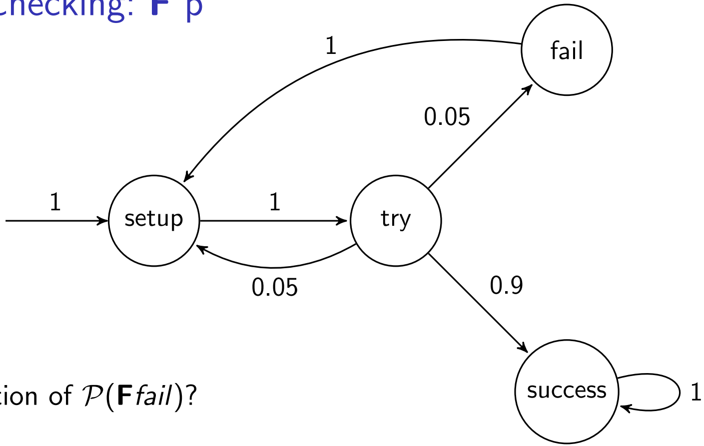
How to compute this?

Further Plan

1. **F** p, with *p* a state property
2. larger part of LTL

Model Checking: **F** p

Example:



Computation of $\mathcal{P}(\mathbf{F}\, \mathit{fail})$?

$$\mathcal{P}(C_{setup,try,fail} \cup C_{setup,try,setup,try,fail} \cup \ldots) =$$
$$\mathcal{P}(C_{setup,try,fail}) + \mathcal{P}(C_{setup,try,setup,try,fail}) + \cdots =$$
$$\mathcal{P}(setup, try, fail) + \mathcal{P}(setup, try, setup, try, fail) + \cdots =$$
$$1 \times 1 \times 0.05 + 1 \times 1 \times 0.05 \times 1 \times 0.05 \cdots = \sum_{n=1}^{\infty} 0.05^n$$

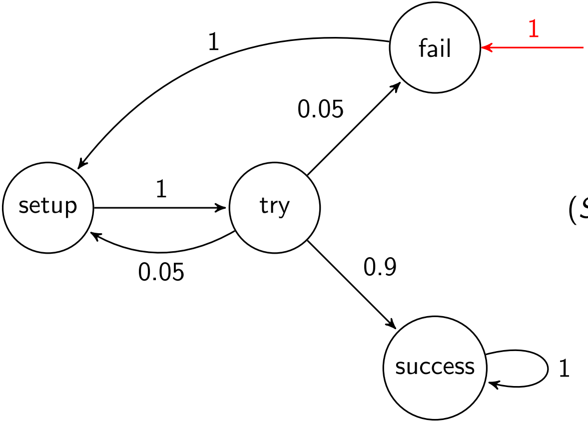
In general: inconvenient!

Behavior from a Certain State

Temporal operators are defined based on suffixes etc.

The system will already be in a non-initial state.

How to analyze how system behaves then?



$(S, 1_{fail}, R)$

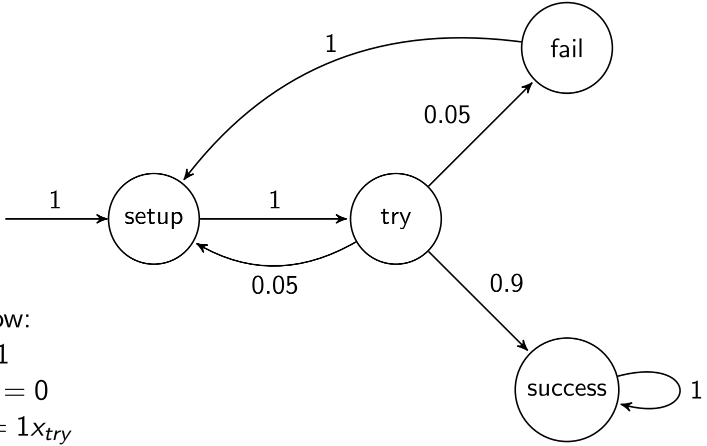
Behavior from a Certain State: Formally

$$1_s(t) := \begin{cases} 1, & \text{if } s = t \\ 0, & \text{otherwise} \end{cases}$$

$(S, 1_s, R)$: changed prob. trans. system that starts in s with probability 1.

Model Checking **F** p

For every state s, $x_s := \mathcal{P}_{(S, 1_s, R)}(\mathbf{F} \mathit{fail})$



We know:
$x_{fail} = 1$
$x_{success} = 0$
$x_{setup} = 1 x_{try}$
$x_{try} = 0.05 x_{fail} + 0.05 x_{setup} + 0.9 x_{success}$

What is this? Linear systems of equations! Solution: $x_{setup} = 0.05263\ldots$

Model Checking **F** p: In General

$$
\begin{aligned}
\mathcal{P}_{(S,1_s,R)}(\mathbf{F}p) &= \mathcal{P}_{(S,1_s,R)}(p \vee [\neg p \wedge \mathbf{XF}p]) \\
&= \mathcal{P}_{(S,1_s,R)}(p) + \mathcal{P}_{(S,1_s,R)}(\neg p \wedge \mathbf{XF}p) \\
&= \mathcal{P}_{(S,1_s,R)}(p) + \mathcal{P}_{(S,1_s,R)}(\neg p)\mathcal{P}_{(S,1_s,R)}(\mathbf{XF}p)
\end{aligned}
$$

if $s \models p$:

$$= 1 + 0 = 1$$

if $s \not\models p$:

$$= 0 + \sum_{s' \in S} R(s,s')\mathcal{P}_{(S,1_{s'},R)}(\mathbf{F}p) = \sum_{s' \in S} R(s,s')\mathcal{P}_{(S,1_{s'},R)}(\mathbf{F}p)$$

Linear system of equations, for every $s \in S$, equation

$$
x_s = \begin{cases}
1, \text{if } s \models p, \\
0, \text{if no state satisfying } p \text{ is reachable from } s, \\
\sum_{s' \in S} R(s,s')x_{s'}, \text{ otherwise.}
\end{cases}
$$

"reachable": in transition graph (ignoring probabilities)

Generalization

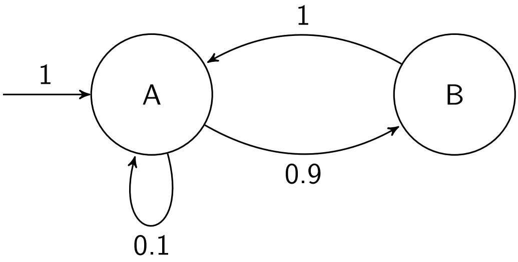
Observation: $\mathbf{G}\,\phi$ is equivalent to $\neg\mathbf{F}\neg\phi$.

So: if we can handle $\mathbf{F}$ and $\neg$, then also $\mathbf{G}$.

Example: Compute $\mathcal{P}_{(S,1_s,R)}(\mathbf{GF}b)$ by computing $\mathcal{P}_{(S,1_s,R)}(\neg\mathbf{F}\neg\mathbf{F}b)$

Probability of Temporal Logic Formulas: Intuition

Recursive computation, example:



$\mathcal{I}(b) = \{B\}$

$\mathbf{GF}b$

$\neg\mathbf{F}\neg\mathbf{F}b$

	$\mathcal{P}_{(S,1_A,R)}()$	$\mathcal{P}_{(S,1_B,R)}()$
b	0	1
$\mathbf{F}b$	1	1
$\neg\mathbf{F}b$	0	0
$\mathbf{F}\neg\mathbf{F}b$	0	0
$\neg\mathbf{F}\neg\mathbf{F}b$	1	1

$\mathcal{P}_{(S,S_0,R)}(\mathbf{GF}b) = \mathcal{P}_{(S,1_A,R)}(\mathbf{GF}b) = 1$

Probability of Temporal Logic Formulas: Plan

Computation of

1. $\mathcal{P}_{(S,1_s,R)}(\phi), s \in S$ for formulas ϕ without temporal operators, and then recursively

   $\mathcal{P}_{(S,1_s,R)}(\mathbf{X}\phi), s \in S$
2. $\mathcal{P}_{(S,1_s,R)}(\mathbf{F}\phi), s \in S$ from $\mathcal{P}_{(S,1_s,R)}(\phi), s \in S$.
   $\mathcal{P}_{(S,1_s,R)}(\neg\phi), s \in S$
3. $\mathcal{P}_{(S,S_0,R)}(\phi)$ from $\mathcal{P}_{(S,1_s,R)}(\phi), s \in S$

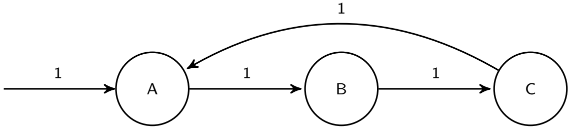
LTL Model Checking: Without Temporal Operators

For a formula ϕ without any temporal operator

$$\mathcal{P}_{(S,1_s,R)}(\phi) = \begin{cases} 1, & \text{if } (s, \dots) \models \phi \\ 0, & \text{otherwise} \end{cases}$$

Intuition: Probability that ϕ holds on the initial state s

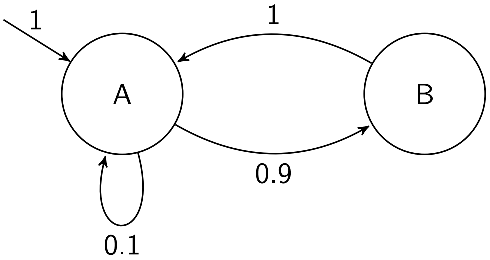
Example:



For a state property p, $(s, \dots) \models p$ iff $s \models p$

if $\mathcal{I}(p) = \{B, C\}$: $\mathcal{P}_{(S,1_A,R)}(p) = 0$
if $\mathcal{I}(p) = \{A, B\}$: $\mathcal{P}_{(S,1_A,R)}(p) = 1$

LTL Model Checking: $\mathbf{X}\,\phi$



$\mathcal{P}_{(S,1_A,R)}(\phi) = 0$
$\mathcal{P}_{(S,1_B,R)}(\phi) = 1$

Example:

$$\mathcal{P}_{(S,1_A,R)}(\mathbf{X}\phi) = \sum_{s'\in S} R(A,s')\mathcal{P}_{(S,1_{s'},R)}(\phi) =$$
$$= 0.1\mathcal{P}_{(S,1_A,R)}(\phi) + 0.9\mathcal{P}_{(S,1_B,R)}(\phi) = 0.9$$

In general

$$\mathcal{P}_{(S,1_s,R)}(\mathbf{X}\phi) = \mathcal{P}_{(S,1_s,R)}(\{\pi \mid \pi \models \mathbf{X}\phi\}) =$$
$$= \mathcal{P}_{(S,1_s,R)}(\{\pi \mid \pi^1 \models \phi\}) = \sum_{s'\in S} R(s,s')\mathcal{P}_{(S,1_{s'},R)}(\phi)$$

LTL Model Checking: Negation

$$\mathcal{P}_{(S,1_s,R)}(\neg\phi) + \mathcal{P}_{(S,1_s,R)}(\phi) =$$

$$\begin{aligned}
&= \mathcal{P}_{(S,1_s,R)}(\{\pi \in \Sigma_S \mid \pi \models \neg\phi\}) + \mathcal{P}_{(S,1_s,R)}(\{\pi \in \Sigma_S \mid \pi \models \phi\}) \\
&= \mathcal{P}_{(S,1_s,R)}(\{\pi \in \Sigma_S \mid \pi \models \neg\phi\} \cup \{\pi \in \Sigma_S \mid \pi \models \phi\}) \\
&= \mathcal{P}_{(S,1_s,R)}(\Sigma_S) \\
&= 1
\end{aligned}$$

So:

$$\mathcal{P}_{(S,1_s,R)}(\neg\phi) = 1 - \mathcal{P}_{(S,1_s,R)}(\phi)$$

LTL Model Checking: $\mathbf{F}\,\phi$

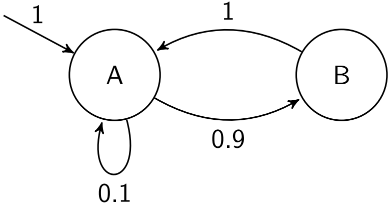
$$
\begin{aligned}
\mathcal{P}_{(S,1_s,R)}(\mathbf{F}\phi) &= \mathcal{P}_{(S,1_s,R)}(\phi \vee [\neg\phi \wedge \mathbf{XF}\phi]) = \\
&= \mathcal{P}_{(S,1_s,R)}(\phi) + \mathcal{P}_{(S,1_s,R)}([\neg\phi \wedge \mathbf{XF}\phi]) = \\
&= \mathcal{P}_{(S,1_s,R)}(\phi) + \mathcal{P}_{(S,1_s,R)}(\neg\phi)\mathcal{P}_{(S,1_s,R)}(\mathbf{XF}\phi) = \\
&= \mathcal{P}_{(S,1_s,R)}(\phi) + (1 - \mathcal{P}_{(S,1_s,R)}(\phi))\mathcal{P}_{(S,1_s,R)}(\mathbf{XF}\phi) = \\
&= \mathcal{P}_{(S,1_s,R)}(\phi) + (1 - \mathcal{P}_{(S,1_s,R)}(\phi)) \sum_{s' \in S} R(s,s')\mathcal{P}_{(S,1_{s'},R)}(\mathbf{F}\phi)
\end{aligned}
$$

Linear system of equations in variables $\mathcal{P}_{(S,1_s,R)}(\mathbf{F}\phi)$,
for every $s \in S$, equation

$$
\mathcal{P}_{(S,1_s,R)}(\mathbf{F}\phi) =
\begin{cases}
0, \text{if for all states } t \text{ reachable from } s, \mathcal{P}_{(S,1_t,R)}(\phi) = 0, \\
\mathcal{P}_{(S,1_s,R)}(\phi) + (1 - \mathcal{P}_{(S,1_s,R)}(\phi)) \sum_{s' \in S} R(s,s')\mathcal{P}_{(S,1_{s'},R)}(\mathbf{F}\phi), \text{ otherwise.}
\end{cases}
$$

unique solution [Courcoubetis and Yannakakis, 1995, Lemma 3.1.1.1]

Combination



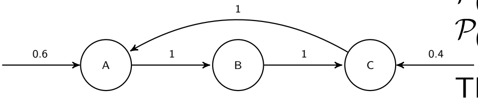
	$\mathcal{P}_{(S,1_A,R)}()$	$\mathcal{P}_{(S,1_B,R)}()$
b	0	1
$\mathbf{F}b$	1	1
$\neg\mathbf{F}b$	0	0
$\mathbf{F}\neg\mathbf{F}b$	0	0
$\neg\mathbf{F}\neg\mathbf{F}b$	1	1

$\mathcal{P}_{(S,1_A,R)}(\mathbf{F}b) = 0.1\mathcal{P}_{(S,1_A,R)}(\mathbf{F}b) + 0.9\mathcal{P}_{(S,1_B,R)}(\mathbf{F}b), \mathcal{P}_{(S,1_B,R)}(\mathbf{F}b) = 1.$
$\mathcal{P}_{(S,1_A,R)}(\mathbf{F}b) = 0.1\mathcal{P}_{(S,1_A,R)}(\mathbf{F}b) + 0.9$
$0.9\mathcal{P}_{(S,1_A,R)}(\mathbf{F}b) = 0.9$
$\mathcal{P}_{(S,1_A,R)}(\mathbf{F}b) = 1$

$\mathcal{P}_{(S,1_A,R)}(\neg\mathbf{F}b) = 0; \mathcal{P}_{(S,1_B,R)}(\neg\mathbf{F}b) = 0.$
$\mathcal{P}_{(S,1_A,R)}(\mathbf{F}\neg\mathbf{F}b) = 0; \mathcal{P}_{(S,1_B,R)}(\mathbf{F}\neg\mathbf{F}b) = 0.$
$\mathcal{P}_{(S,1_A,R)}(\neg\mathbf{F}\neg\mathbf{F}b) = 1; \mathcal{P}_{(S,1_B,R)}(\neg\mathbf{F}\neg\mathbf{F}b) = 1.$

$\mathcal{P}_{(S,S_0,R)}(\neg\mathbf{F}\neg\mathbf{F}b) = 1\mathcal{P}_{(S,1_A,R)}(\neg\mathbf{F}\neg\mathbf{F}b) + 0\mathcal{P}_{(S,1_B,R)}(\neg\mathbf{F}\neg\mathbf{F}b) = 1$

Overall Probability



$\mathcal{P}_{(S,1_A,R)}(\phi) = 0.2$
$\mathcal{P}_{(S,1_B,R)}(\phi) = 0.7$
$\mathcal{P}_{(S,1_C,R)}(\phi) = 0.1$

Then: $\mathcal{P}_{(S,S_0,R)}(\phi) = 0.6\mathcal{P}_{(S,1_A,R)}(\phi) + 0.4\mathcal{P}_{(S,1_C,R)}(\phi)$

$$\begin{aligned}
\mathcal{P}_{(S,S_0,R)}(\phi) &= \mathcal{P}_{(S,S_0,R)}(\{\pi \in \Sigma_S \mid \pi \models \phi\}) \\
&= \mathcal{P}_{(S,S_0,R)}(\bigcup_{s\in S}\{\pi \in \Sigma_S \mid \pi(0) = s, \pi \models \phi\}) \\
&= \sum_{s\in S} \mathcal{P}_{(S,S_0,R)}(\{\pi \in \Sigma_S \mid \pi(0) = s, \pi \models \phi\}) \\
&= \sum_{s\in S} S_0(s)\mathcal{P}_{(S,1_s,R)}(\{\pi \in \Sigma_S \mid \pi(0) = s, \pi \models \phi\}) \\
&= \sum_{s\in S} S_0(s)\mathcal{P}_{(S,1_s,R)}(\{\pi \in \Sigma_S \mid \pi \models \phi\}) \\
&= \sum_{s\in S} S_0(s)\mathcal{P}_{(S,1_s,R)}(\phi)
\end{aligned}$$

Probabilistic LTL Model Checking: Summary

We can handle state properties, **X**, **F**, $\neg$,
and hence also with **G**.

Combination of those operators: recursive computation of $\mathcal{P}_{(S,1_s,R)}()$

And finally

$$\mathcal{P}_{(S,S_0,R)}(\phi) = \sum_{s \in S} S_0(s)\mathcal{P}_{(S,1_s,R)}(\phi)$$

Can be generalized to full LTL

Negation

$$\mathcal{P}_{(S,1_s,R)}(\neg\phi) = 1 - \mathcal{P}_{(S,1_s,R)}(\phi)$$

and especially

$$\mathcal{P}_{(S,1_s,R)}(\mathbf{G}\phi) = \mathcal{P}_{(S,1_s,R)}(\neg\mathbf{F}\neg\phi) = 1 - \mathcal{P}_{(S,1_s,R)}(\mathbf{F}\neg\phi)$$

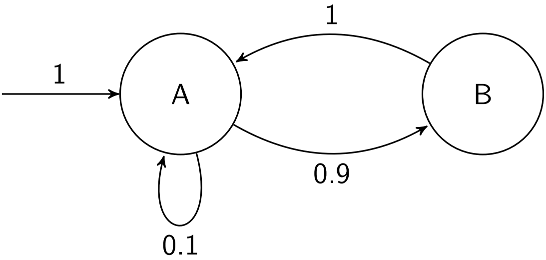
But, attention: $\models \neg\phi$ is not equivalent to $\not\models \phi$:

$$\begin{aligned}
&\models \neg\phi \iff \\
&\forall \pi \,.\, \pi \models \neg\phi \iff \\
&\forall \pi \,.\, \pi \not\models \phi \;\not\Longleftarrow\!\!\!\Longrightarrow \\
\\
&\exists \pi \,.\, \pi \not\models \phi \iff \\
&\neg\forall \pi \,.\, \pi \models \phi \iff \\
&\not\models \phi
\end{aligned}$$

and especially

$$\models \mathbf{G}\phi \text{ iff } \models \neg\mathbf{F}\neg\phi \;\cancel{\text{iff}}\; \not\models \mathbf{F}\neg\phi$$

Nondeterminism versus Probability



$\mathcal{I}(b) = \{B\}$

$\not\models$ $\mathbf{GF}b$ (transition system without probabilities), but
$\mathcal{P}(\mathbf{GF}b) = 1$

Why do we at all need transition systems without probabilities?

We may not have probabilities at all, or
probabilities that do not fulfill the Markov property

Markov Decision Processes

Allow both non-deterministic and probabilistic transitions

Basis for reinforcement learning:

Verification tool: `http://www.prismmodelchecker.org`

Demo: Synchronous Leader Election Protocol

`http://www.prismmodelchecker.org/casestudies/synchronous_leader.php`

Costs and Rewards

Classical view: program either correct or incorrect

Often not sufficient:

- More modest goal: smallest possible impact of bugs
- Battery-powered devices
- Steering aeroplane: energy consumption critical
- Stock trading software: highest possible profit

In many situations it is critical to

- minimize cost, or
- maximize reward.

Probabilistic Transition Systems and Costs

Cost function $c : S \to \mathbb{N}_0$

Cost of finite path: $c(s_0, \dots, s_n) := \sum_{i \in \{0,\dots,n-1\}} c(s_i)$

Cost of property $\mathbf{F}p$ on path π:

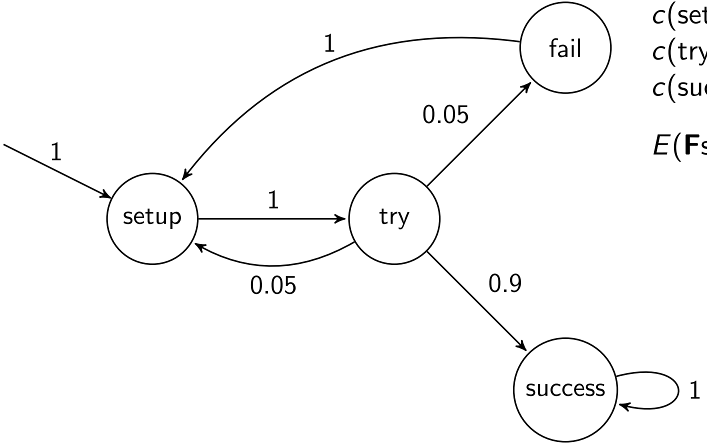
$$c_\pi(\mathbf{F}p) := \begin{cases} c(s_0, \dots, s_n), \text{if } \pi \models \mathbf{F}p, \text{ where} \\ \qquad s_0, \dots, s_n \text{ is the prefix of } \pi \text{ with} \\ \qquad\qquad s_0 \not\models p, \dots, s_{n-1} \not\models p, s_n \models p \\ \infty, \text{ otherwise.} \end{cases}$$

Expected cost:

$$E(\mathbf{F}p) := \sum_{\substack{(s_0, \dots, s_n) \in \Sigma_S, \\ s_0 \not\models p, \dots, s_{n-1} \not\models p, s_n \models p}} c(s_0, \dots, s_n)\mathcal{P}(s_0, \dots, s_n),$$

if $\mathcal{P}(\mathbf{F}p) = 1$ and ∞, otherwise.

Computation of Expected Cost

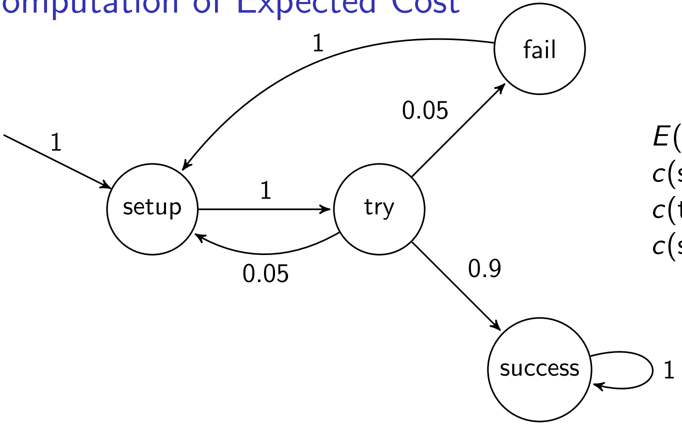


$c(\text{setup}) =$
$c(\text{try}) = c(\text{fail}) =$
$c(\text{success}) = 1$

$E(\mathbf{F}\text{success})$

$$0.9 \times 2 + 0.05 \times 0.9 \times 4 + 0.05 \times 0.9 \times 5 + \dots$$

Computation of Expected Cost



$E(\mathbf{F}\text{success})$,
$c(\text{setup}) =$
$c(\text{try}) = c(\text{fail}) =$
$c(\text{success}) = 1$

$x_s := E(\mathbf{F}\text{success})$, for probabilistic transition system with $S_0(s) = 1$

$$\begin{aligned}
x_{\text{success}} &= 0\\
x_{\text{try}} &= 1 + 0.05x_{\text{fail}} + 0.9x_{\text{success}} + 0.05x_{\text{setup}}\\
x_{\text{fail}} &= 1 + x_{\text{setup}}\\
x_{\text{setup}} &= 1 + x_{\text{try}}
\end{aligned}$$

$x_{\text{setup}} = 2.05/0.9 \sim 2.278$

Computation of Expected Cost

Assumption: $\mathcal{P}(\mathbf{F}p) = 1$

For the solution of the linear system

$$x_s = \begin{cases} 0, \text{if } s \in \mathcal{I}(p), \\ c(s) + \sum_{s' \in S} R(s, s') x_{s'}, & \text{otherwise}, \end{cases}$$

we get:

$$E(\mathbf{F}p) = \sum_{s \in S} S_0(s) x_s$$

Further Possibilities

Various combinations and variations:

- Probabilistic timed automata
- Probabilistic hybrid systems
- Probabilistic Petri nets
- Continuous time Markov chains (viz. MI-SPI)
- atd.

Conclusion

non-determinism $\neq$ probability

analogue notions (transition system, LTL, etc.)

different algorithms (linear algebra)

performance vs. correctness

Literature

Christel Baier and Joost-Pieter Katoen. *Principles of Model Checking*. The MIT Press, 2008.

Costas Courcoubetis and Mihalis Yannakakis. The complexity of probabilistic verification. *Journal of the ACM (JACM)*, 42(4):857–907, 1995.

M. Kwiatkowska, G. Norman, and D. Parker. Stochastic model checking. In M. Bernardo and J. Hillston, editors, *Formal Methods for the Design of Computer, Communication and Software Systems: Performance Evaluation (SFM'07)*, volume 4486 of *LNCS (Tutorial Volume)*, pages 220–270. Springer, 2007.

Moshe Y. Vardi. Automatic verification of probabilistic concurrent finite state programs. In *26th Annual Symposium on Foundations of Computer Science (sfcs 1985)*, pages 327–338. IEEE, 1985.