p-Automata: Foundations for Reasoning about Markov Chains

Nir Piterman

Imperial College London

joint work with: Michael Huth and Daniel Wagner

In memory of Amir Pnueli

Markov Chains

- An important modeling formalism in science:
  - Economics.
  - Physics.
  - Biology.
  - Chemistry.
- In CS and Engineering:
  - Performance and queuing models.
  - Randomized algorithms.

Formal Methods for Markov Chains

Formal methods community devoted significant resources:

- Qualitative analysis – 0,1 answers.
- Quantitative analysis – what is the probability.
- Logics for reasoning about Markov chains.
- Probabilistic bisimulation.
- Model checking tools: PRISM (Oxford/Birmingham), LiQuor (Bonn/Dresden), MRMC (Aachen).

Automata in Model Checking

Automata theoretic approach to model checking:

- A unifying approach for: model checking, temporal logics, synthesis, and abstraction
- Linear time through word automata:
  - Translate LTL to word automata.
  - Regular expressions as part of PSL.
- Branching time through tree automata:
  - MSO is satisfiable (Rabin).
  - μ-calculus, CTL, CTL* reasoning.
  - Synthesis of linear specifications.
  - Two player games.
  - Complete abstraction for branching time.

Completeness of Abstraction

Reason about infinite-state systems by abstraction:

- The basis for CEGAR.
- What is the right abstraction domain $\mathcal{D}$?
- Completeness: given an infinite state system M and a branching time property ϕ s.t. $M \models \phi$,there exists a finite $A \in \mathcal{D}$ such that $M \preceq A$ and $A \models \phi$.
- Alternating tree automata are a complete abstraction framework for branching-time logic.

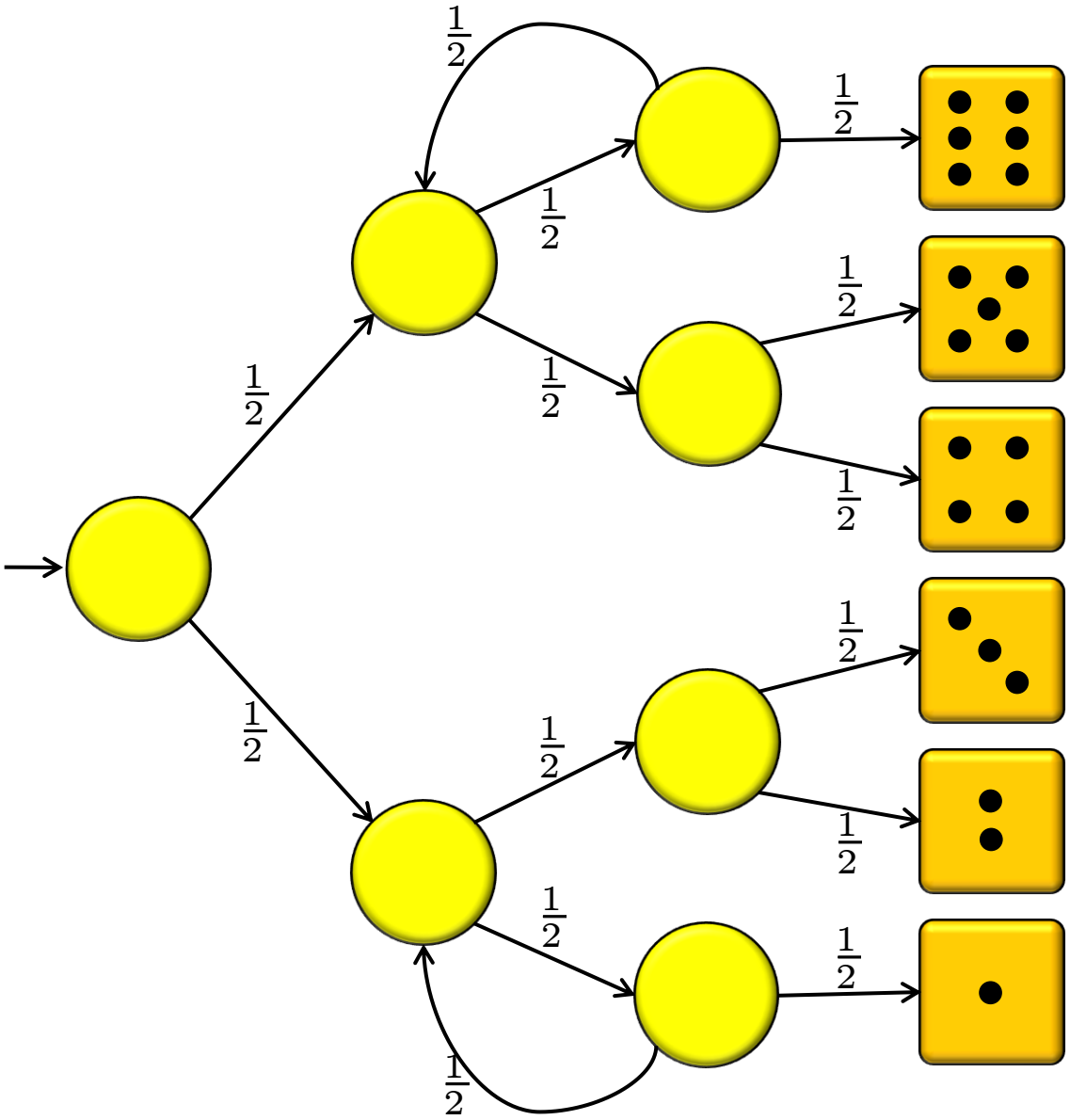
Back to Markov Chains

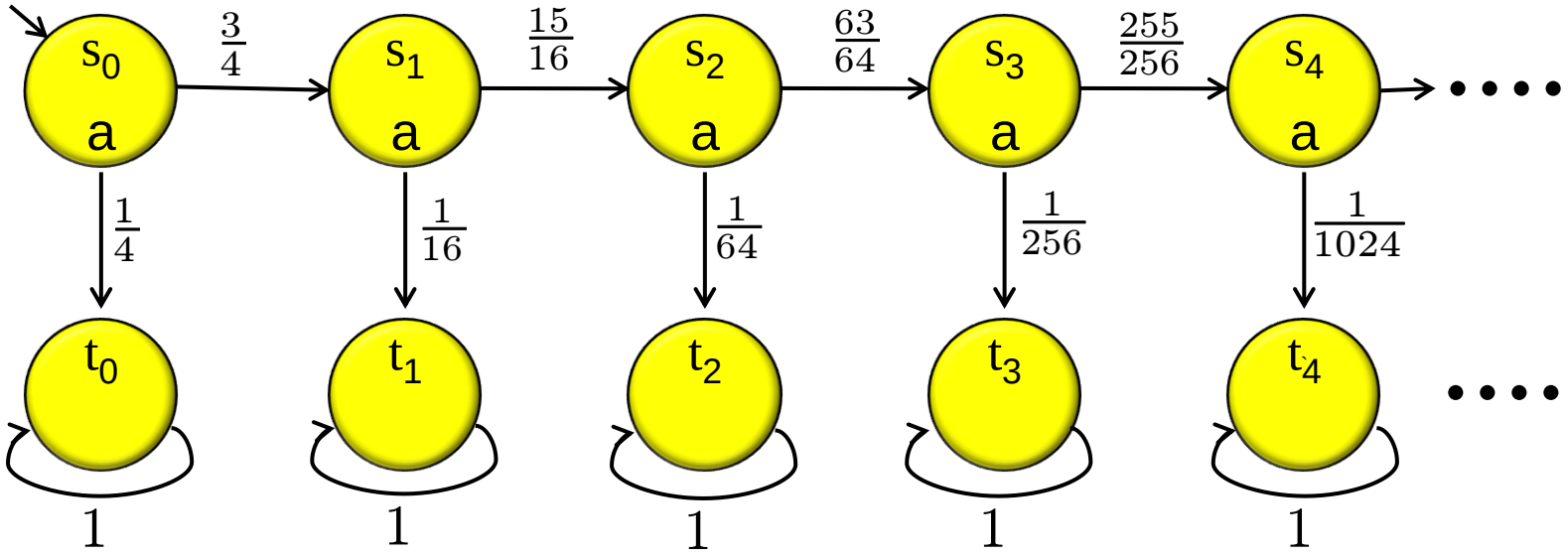
- Presently no unifying framework for reasoning about Markov chains.
- Abstraction is an open problem.
- p-Automata – provide such a framework:
  - Acceptors of Markov chains (as a whole!).
  - Express Markov chain bisimulation class.
  - Express pCTL, pCTL*, future ω-regular extensions.
  - Closed under Boolean operations.
  - Simulation approximates language containment.
  - Complete abstraction framework for pCTL.

Outline of Talk

- Motivation and introduction.
- Markov chains and pCTL
- p-Automata.
- First results.
- Conclusions.

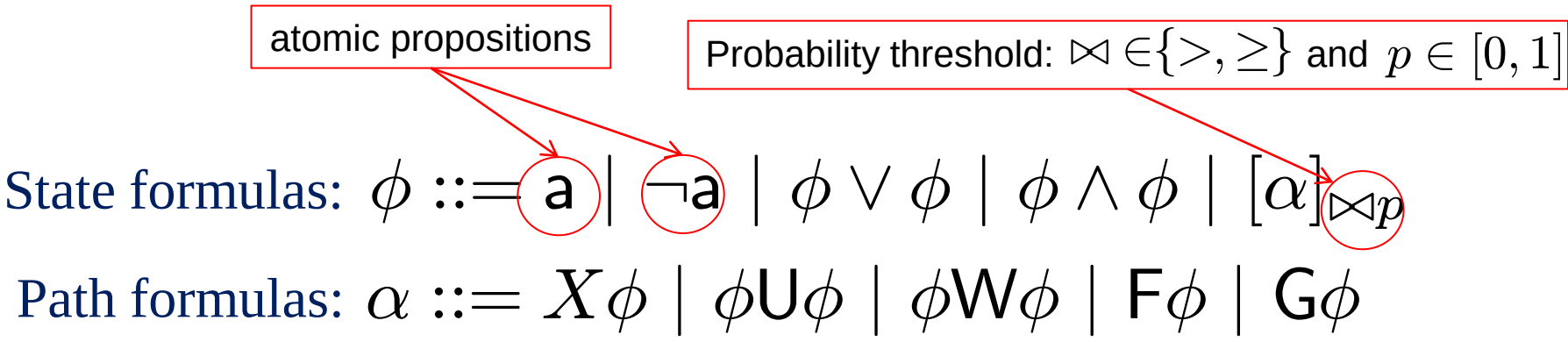
Markov Chains and pCTL


1/2
1/2
1/2
1/2
1/2
1/2
1/2
1/2
1/2
1/2
1/2
1/2

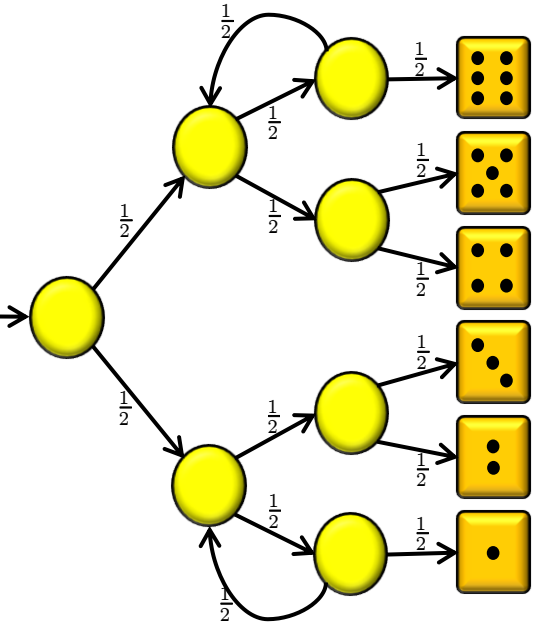

s_0
a
s_1
a
s_2
a
s_3
a
s_4
a
$\frac{3}{4}$
$\frac{15}{16}$
$\frac{63}{64}$
$\frac{255}{256}$
$\frac{1}{4}$
$\frac{1}{16}$
$\frac{1}{64}$
$\frac{1}{256}$
$\frac{1}{1024}$
t_0
t_1
t_2
t_3
t_4
1
1
1
1
1

pCTL

pCTL is the de-facto standard for reasoning about Markov chains.



Examples

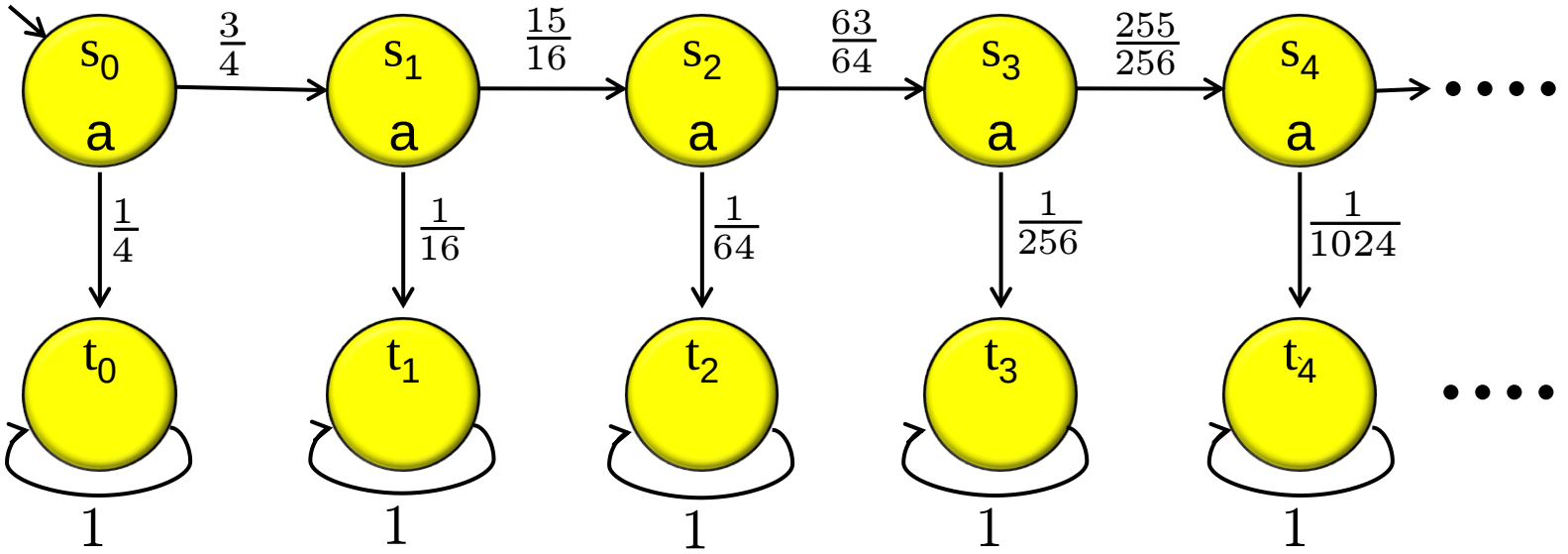


$[\mathtt{Fsome\_num}]_{\geq 1}$

$[\mathtt{Fnum\_4}]_{\geq \frac{1}{6}} \wedge [\mathtt{Fnum\_not\_4}]_{\geq \frac{5}{6}}$

No Finite Model

$[\mathsf{G}\,(\mathsf{a} \wedge [\mathsf{F}\neg\mathsf{a}]_{>0})]_{>0}$



p-Automata

p-Automata

- Motivated by alternating tree automata and pCTL:
  - Include existential and universal choices.
  - Include quantification over probability of path sets.
- Combine path measure and regular path sets.
- Two types of transitions:
  - Unbounded – part of regular path measure.
  - Bounded – measure the probability

Definition

p-Automaton is $A = \langle \Sigma, Q, \delta, \varphi^{in}, \alpha \rangle$, where

- Σ – finite input alphabet.
- Q – set of states (not necessarily finite).
- $\delta : Q \times \Sigma \to B^+(Q \cup [\![Q]\!])$ transition function.
- $\varphi^{in} \in B^+(Q \cup [\![Q]\!])$ – initial condition.
- α – acceptance condition.

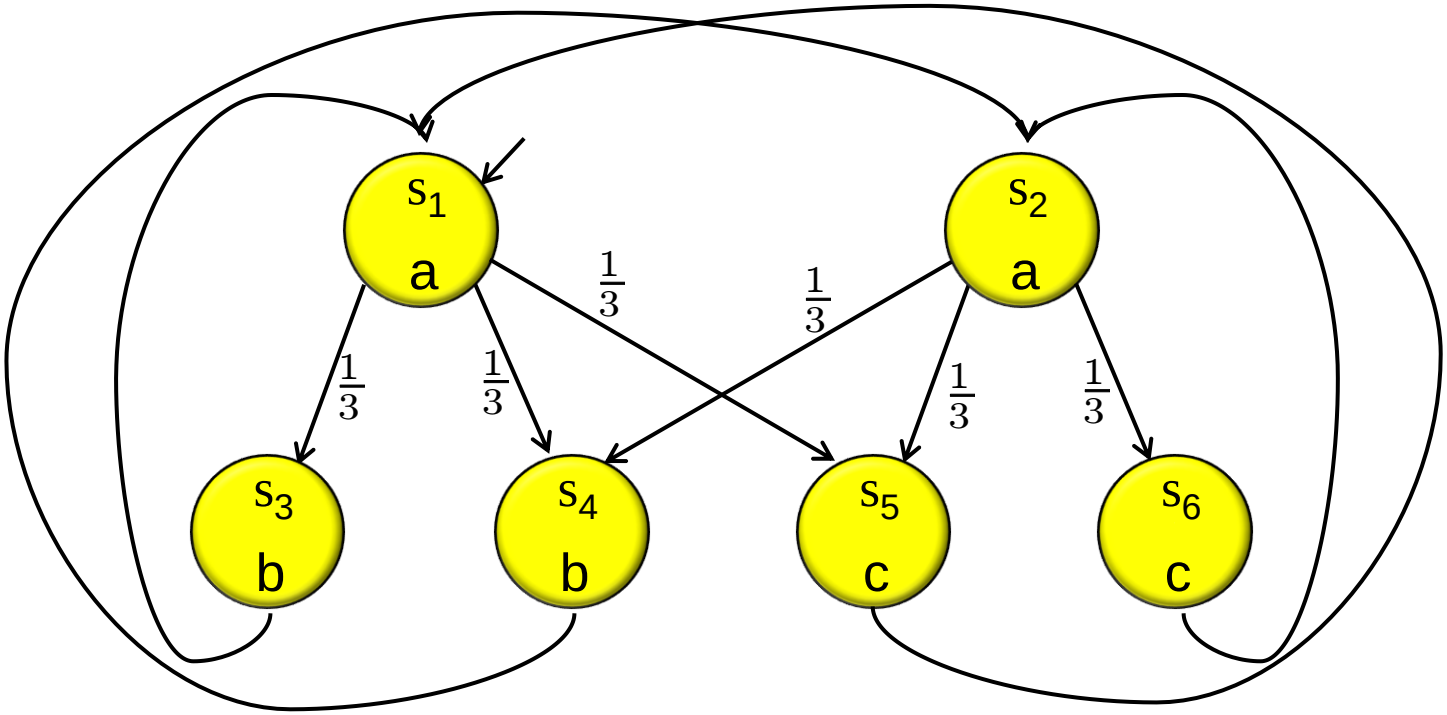
$B^{+}(Q \cup [\![Q]\!])$

- Boolean connectives: existential and universal choice.
- $[\![q]\!]_{\bowtie p}$ holds in location s if measure of paths that start in s and satisfy q is $\bowtie p$
- $*([\![q_1]\!]_{\geq p_1}, [\![q_2]\!]_{> p_2})$ is
  - Paths that satisfy q_1 have probability at least p_1.
  - Paths that satisfy q_2 have probability greater than p_2.
  - The sets supplying probability are *immediately* disjoint (a-la separation logic …).
- $\circledast([\![q_1]\!]_{> p_1}, [\![q_2]\!]_{\geq p_2})$ is dual.

Example

$$A_1 = \langle \{a, b, c\}, \{s_1, \ldots, s_6\}, \delta_1, [\![s_1]\!]_{\geq 1}, \{s_1, \ldots, s_6\} \rangle$$

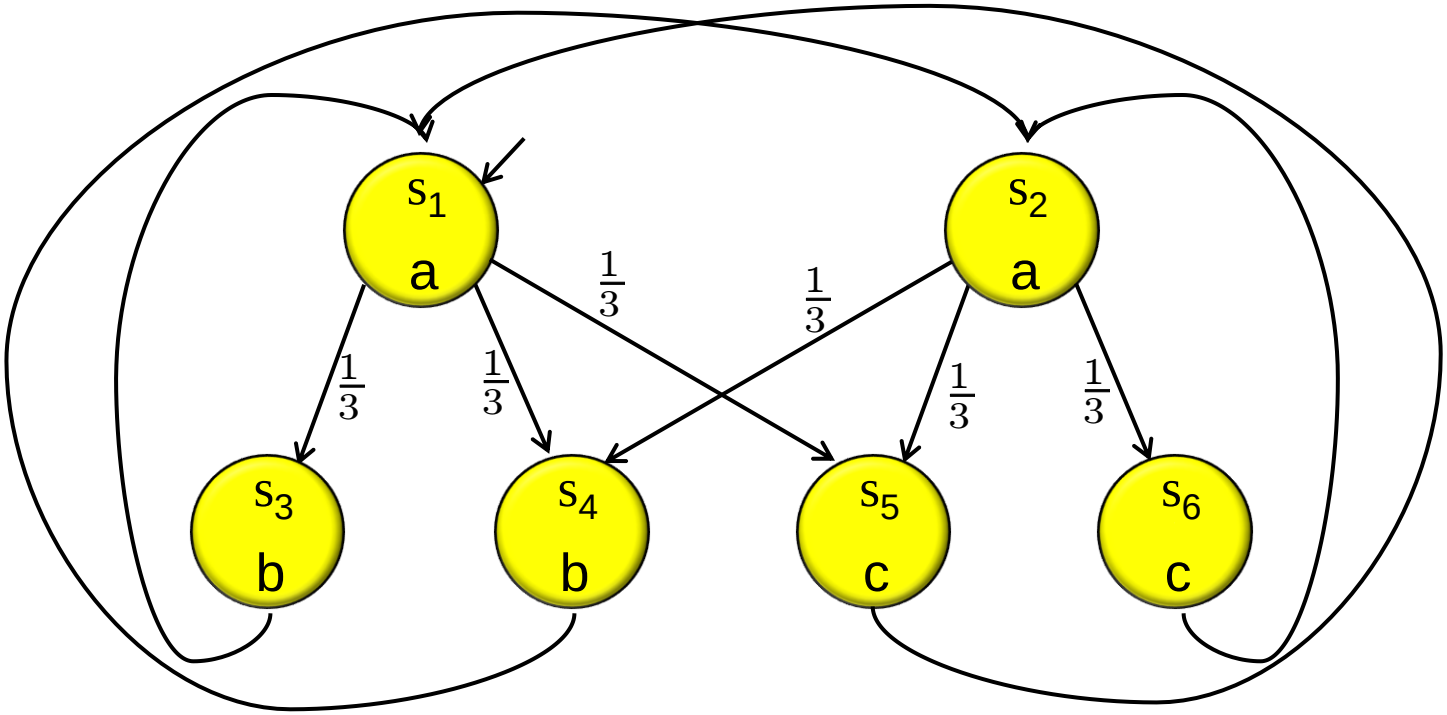
$$\begin{array}{lcl} \delta_1(s_1, a) & = & [\![s_3]\!]_{\geq \frac{1}{3}} \wedge [\![s_4]\!]_{\geq \frac{1}{3}} \wedge [\![s_5]\!]_{\geq \frac{1}{3}} \\ \delta_1(s_2, a) & = & [\![s_4]\!]_{\geq \frac{1}{3}} \wedge [\![s_5]\!]_{\geq \frac{1}{3}} \wedge [\![s_6]\!]_{\geq \frac{1}{3}} \\ \delta_1(s_3, b) & = & \delta_1(s_5, c) = [\![s_1]\!]_{\geq 1} \\ \delta_1(s_4, b) & = & \delta_1(s_6, c) = [\![s_2]\!]_{\geq 1} \end{array}$$



Example

$$A_2 = \langle \{a, b, c\}, \{s_1, \ldots, s_6\}, \delta_1, [\![s_1]\!]_{\geq 1}, \{s_1, \ldots, s_6\} \rangle$$

$$\begin{array}{rcl}
\delta_2(s_1, a) & = & *([\![s_3]\!]_{\geq \frac{1}{3}}, [\![s_4]\!]_{\geq \frac{1}{3}}, [\![s_5]\!]_{\geq \frac{1}{3}}) \\
\delta_2(s_2, a) & = & *([\![s_4]\!]_{\geq \frac{1}{3}}, [\![s_5]\!]_{\geq \frac{1}{3}}, [\![s_6]\!]_{\geq \frac{1}{3}}) \\
\delta_2(s_3, b) = \delta_2(s_5, c) & = & [\![s_1]\!]_{\geq 1} \\
\delta_2(s_4, b) = \delta_2(s_6, c) & = & [\![s_2]\!]_{\geq 1}
\end{array}$$



Acceptance Games

- Given a p-automaton *A* and an input structure *M* we want to construct a game such that player 1 wins iff *A* accepts *M*.
- Existential and universal choice handled in standard way.
- Two new things:
  - Systems are probabilistic – use stochastic games.
  - Star and bounded transitions – player 1 commits to values it can achieve.
- Structural Restrictions.

Simulation Games

- Given two automata A_1 and A_2, construct a game such that player 1 wins iff $A_1 \preceq A_2$.
- Generalize simulation games by considering star and bounded transitions on the left and on the right.
- For finite p-Automata or p-Automata arising from Markov chains, simulation implies language containment.

$$A \preceq B \Longrightarrow \mathcal{L}(A) \subseteq \mathcal{L}(B)$$

Results

Closures of Languages

- Closure under conjunction and disjunction is standard.
- Closure under complement.
- Language emptiness and language containment are inter-reducible.
- Given two bisimilar Markov chains $M_1 \sim M_2$:

$$M_1 \in \mathcal{L}(A) \text{ iff } M_2 \in \mathcal{L}(A)$$

Embedding Markov Chains

A Markov chain $M = (S, P, L, s^{in})$ is embedded into a p-automaton $A_M = \langle 2^{\mathsf{AP}}, Q, \delta, \varphi^{in}, \alpha \rangle$:

$$
\begin{array}{rcl}
Q & = & \{(s, s') \in S \times S \mid P(s, s') > 0\} \\
\delta((s, s'), L(s)) & = & *(\llbracket (s', s'') \rrbracket_{\geq P(s', s'')} \mid s'' \in \mathsf{succ}(s')) \\
\delta((s, s'), \sigma) & = & \mathbf{f} \qquad\qquad \text{if } \sigma \neq L(s) \\
\varphi^{in} & = & *(\llbracket (s^{in}, s') \rrbracket_{\geq P(s^{in}, s')} \mid P(s^{\mathsf{in}}, s') > 0) \\
\alpha & = & Q
\end{array}
$$

$$M' \in \mathcal{L}(A_M) \text{ iff } M \sim M'$$

Embedding pCTL

- Similar to translation of CTL to tree automata.
- Given a pCTL formula φ over AP construct the p-automaton $A_\varphi = \langle 2^{\mathrm{AP}}, cl(\varphi) \cup \mathrm{AP}, \rho_x, \rho_\epsilon(\varphi), \alpha \rangle$:
- $cl(\varphi)$ is the set of temporal subformulas of φ.
- α includes everything except $\psi_1 \mathsf{U} \psi_2$.
- ρ_x and ρ_ϵ unfold fixpoints and replace $[\cdot]$ by $[\![\cdot]\!]$. For example, $\psi_1 \mathsf{U} \psi_2$ replaced by $(\psi_1 \wedge \mathsf{X}\psi_1 \mathsf{U} \psi_2) \vee \psi_2$

$$M \models \varphi \text{ iff } M \in \mathcal{L}(A_\varphi)$$

Abstraction

- p-Automata abstract Markov chains.
- For every pCTL formula φ and infinite Markov chain M such that $M \models \varphi$ there is a finite p-automaton A such that $A_M \preceq A$ and $A \preceq A_\varphi$.

Conclusions

p-Automata

- Developed a notion of automata that accept Markov chains.
- Defined acceptance and simulation games through stochastic two-player games.
- p-Automata are closed under Boolean operations. Languages closed under bisimulation.
- Can express pCTL and Markov chains.
- Complete abstraction framework for pCTL.

Related Work

- Rabin (probabilistic) automata.
  - Can be thought as linear time probabilistic automata.
  - Define a mapping from words to probability of acceptance.
  - Can define a language by including a threshold.
  - Unrelated to pCTL and model checking.
- Co-algebraic automata.
  - Accept Markov chains.
  - Inherently infinite.
  - Finite model property, hence cannot express pCTL.
- Classical automata.
  - Can be used for linear time model checking.
  - Do not give answers for pCTL.

Future Work

- Decidability of language emptiness.
  - Qualitative (0,1 thresholds).
  - Quantitative.
  - Generalizes open problem of pCTL satisfiability.
- Remove structural restrictions.
  - Define games that generalize stochastic games.
  - Generalize Martin's determinacy result.
- Markov Decision Processes.
- Usage within a CEGAR framework.

Thank you, Amir.