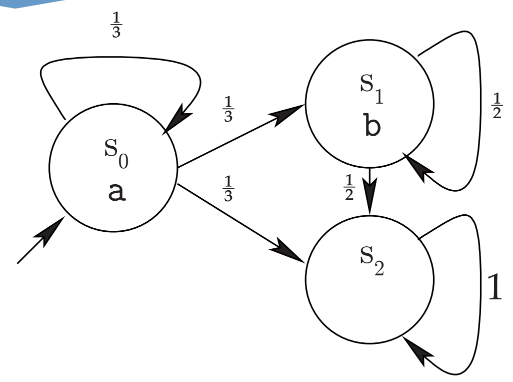
p Automata

Approach to Probabilistic Verification

- p-automata read an entire Markov chain as input and either accept or reject it.
- Their definition combines the combinatorial structure of alternating automata with the ability to quantify probabilities of regular sets of paths.
- Two probabilistic quantifiers: one tallies probabilities of immediate next locations (reminiscent of the X operator in PCTL); the other measures the probabilities of regular path sets.
- Transitions are positive Boolean formulas with an extended base set, combining states q with threshold obligations: $[\![q]\!]_{\geq 0.5}$ says that the path set represented by q has probability ≥ 0.5.
- A probabilistic separation operator $*$ decomposes the witness path set for a probability threshold into disjoint subsets: $*([\![q_1]\!]_{\geq p1},[\![q_2]\!]_{\geq p2})$ says that the path set determined by state q_i has probability at least p_i for $i=1,2$; and that the sets measured by these probabilities are disjoint. (Think "disjoint and".)

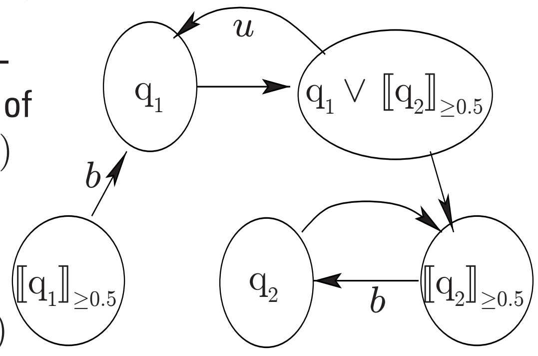
Markov Chains

A finitely branching, countable labeled Markov chain M over set of propositions $\mathbb{A}$ is a tuple $\langle S,P,s^{in},L\rangle$, where S is a countable set of locations, P a stochastic matrix, s^{in} initial location, and $L(s)$ the set of propositions true in location s.



p-Automata

A p-automaton $\mathcal{A}$ is a tuple $\langle \Sigma,Q,\delta,\phi^{in},\alpha\rangle$, where Σ is a finite input alphabet, Q is a set of states, δ: $Q\times\Sigma \to \mathcal{B}^+(Q\cup[\![Q]\!])$ the transition function, ϕ^{in} the initial condition, $\alpha\subseteq Q$ an acceptance condition, and $[\![Q]\!] =\{[\![q_i]\!]_{\geq pi},*([\![q_1]\!]_{\geq p1},\ldots,[\![q_n]\!]_{\geq pn}) \mid q_i\in Q,\ p_i\in[1,0],n\in\mathbb{N}\}$.



Example

Let $\mathcal{A} = \langle \mathcal{P}(\{\mathtt{a},\mathtt{b}\}),\{q_1,q_2\},\delta,[\![q_1]\!]_{\geq 0.5},\{q_2\}\rangle$ be a p-automaton with δ as follows (and as in the graph above):

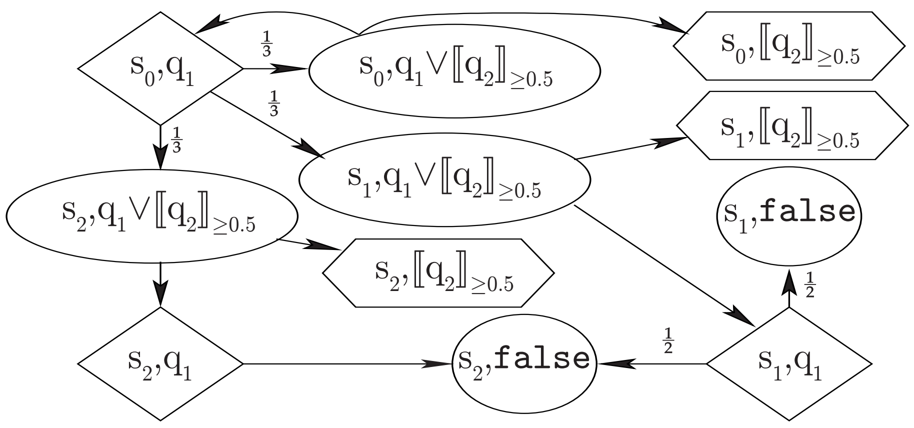
$\delta(q_1,\{\mathtt{a},\mathtt{b}\}) = \delta(q_1,\{\mathtt{a}\}) = q_1 \vee [\![q_2]\!]_{\geq 0.5}$
$\delta(q_2,\{\mathtt{a},\mathtt{b}\}) = \delta(q_2,\{\mathtt{b}\}) = [\![q_2]\!]_{\geq 0.5}$
$\delta(q_1,\{\}) = \delta(q_1,\{\mathtt{b}\}) = \delta(q_2,\{\mathtt{a}\}) = \delta(q_2,\{\}) = \mathtt{false}$

Term $[\![q_2]\!]_{\geq 0.5}$ represents the recursive property ϕ, that atomic proposition $\mathtt{b}$ holds at the location presently read by q_2 and that ϕ will hold with probability at least 0.5 in the next locations. State q_1 asserts that it is possible to get to a location that satisfies $[\![q_2]\!]_{\geq 0.5}$ along a path that satisfies atomic proposition $\mathtt{a}$. The initial condition $[\![q_1]\!]_{\geq 0.5}$ means the set of paths satisfying $\mathtt{a}U\phi$ has probability at least 0.5.

Games for acceptance & simulation

Acceptance $M\in\mathcal{L}(\mathcal{A})$ and simulation $\mathcal{A}\leq\mathcal{B}$ can be decided through a series of stochastic games and games. (EXPTIME in the sizes of $\mathcal{A}$ and M. Some conditions on $\mathcal{A}$ and $\mathcal{B}$ for simulation.)

Example The stochastic game $G_{M,(\!(q_1)\!)}$ for the SCC $(\!(q_1)\!)$ depicts stochastic configurations as diamond and configurations from other SCCs as hexagons (with the hexagon labeled $(s_1,[\![q_2]\!]_{\geq 0.5})$ having value 1 and all others having value 0). As none of the configurations are accepting, P_0 can only win by reaching optimal hexagons. Hexagon $(s_1,[\![q_2]\!]_{\geq 0.5})$ has value 1 and is the optimal choice for P_0 from configuration $(s_1,q_1\vee[\![q_2]\!]_{\geq 0.5})$. As $(s_2,q_1\vee[\![q_2]\!]_{\geq 0.5})$ has value 0, the value for P_0 of diamond configuration (s_1,q_1) is 0.5. Initial configuration $(s_0,[\![q_1]\!]_{\geq 0.5})$ is a trivial bounded SCC; its value equals 1 as $\frac{1}{3}\mathrm{val}(s_0,q_1\vee[\![q_2]\!]_{\geq 0.5}) + \frac{1}{3}\mathrm{val}(s_1,q_1\vee[\![q_2]\!]_{\geq 0.5}) + \frac{1}{3}\mathrm{val}(s_2,q_1\vee[\![q_2]\!]_{\geq 0.5})$ is 0.5. Thus $M\in \mathcal{L}(\mathcal{A})$.



Properties

- p-automata are closed under Boolean operations.
- The language of a p-automaton is closed under bisimulation.
- Markov chain M can be embedded as a p-automaton accepting the language of Markov chains that are bisimilar to M.
- PCTL formula ϕ can be expressed as language $\mathcal{L}(\mathcal{A})$, and PCTL model checking can be reduced to deciding the acceptance of Markov chains by p-automata $\mathcal{A}$. The complexity of the acceptance game then matches that of model checking.
- Language containment and emptiness are equi-solvable.
- Simulation between p-automata that stem from Markov chains or PCTL formulas is decidable in EXPTIME and under-approximates language containment.

Conclusions

- p-automata are a complete abstraction framework for PCTL: if an infinite Markov chain M satisfies a PCTL formula ϕ, there is a *finite* p-automaton that abstracts M and whose language is contained in that of the p-automaton for ϕ.
- Emptiness, universality, and containment of p-automata seem tightly related to the open problem of decidability of PCTL satisfiability.

Full paper p-Automata: New Foundations for Discrete-Time Probabilistic Verification. To appear in Proc. of QEST 2010.

Imperial College London

Michael Huth, Nir Piterman, Daniel Wagner
{m.huth, nir.piterman, d.wagner}@imperial.ac.uk
Department of Computing
Imperial College London