London Hopper Colloquium 2017

Research Spotlight Competition

Should I Take the Umbrella?

Probabilistic Languages in Everyday Life

Claudia Cauli

PhD 1st Year

University of Leicester

LE1 7RH, Leicester, UK

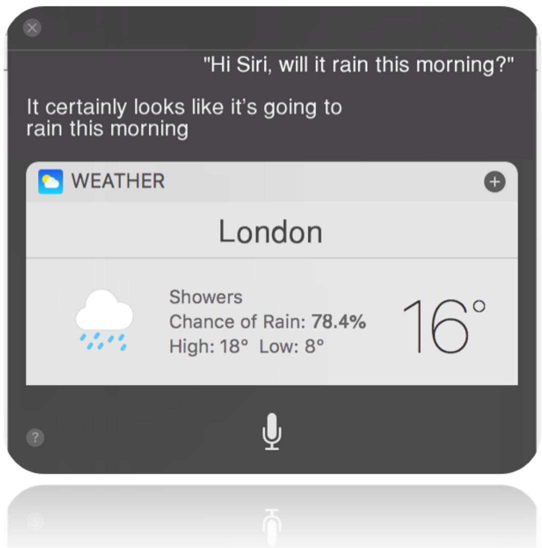
joint work with Dr Nir Piterman

cc488@le.ac.uk

http://claudiacauli.com

Motivation — *The quest for Formal Methods for Probabilistic Systems*

- Probabilistic systems are modelled as *Markov Chains*.
- Their properties are expressed using *Probabilistic Languages*.
- Properties are then *checked* against the model.



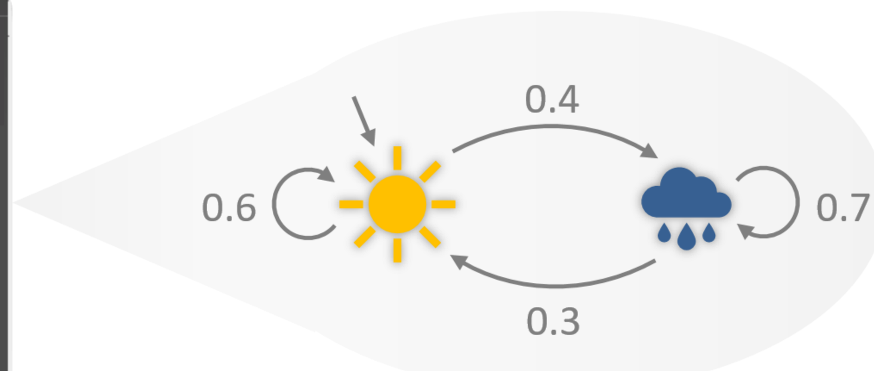


Fig 1. *A Markov chain that models hourly weather variations.*

Related Problems

- To *specify properties* of probabilistic models → Probabilistic Languages?
- To *increase* the range of possible properties → Expressive Power?
- To *verify* such properties in an efficient way → Verification Approach?

We investigate the *Automata-Theoretic verification approach for Probabilistic Logics* and prove the equivalence of two formalisms: p-Automata and μ^p-Calculus.

Approach *The analysis and study of μ^p-Calculus and p-Automata*

- Two classes of *probabilistic languages*: *Logics* & *Automata*.
- Automata-Theoretic approach to verification: *Logics are handled as Automata*.
- To be used interchangeably, logics and automata must be: *Equivalent*.

μ^p-Calculus

[1] Set of *formulas* built up from:

- ***p, ¬p*** e.g. *"rain", "not rain", "sun"* etc.
- **∧ / ∨** *"and" / "or"*
- **O** *"in the next step"*
- **μ** *"finitely many times"*
- **[…]$_x$** *"with probability x…"*

$$[\ \bigcirc\ rain\ ]_{\geq 0.5}$$

"With probability ≥0.5, in the next hour rain"

Fig 2. *An example of* ***μ^p-Calculus*** *formula. Is it true on the Markov chain of Fig.1?*

p-Automata

[2] *Graph-like structures* defined by:

- **Alphabet** *Symbols in input*
- **States** *Vertices in the graph*
- **Transitions** *Edges linking vertices*
- **Initial Condition** *Combination of states*
- **Acceptance** *States w/ min even number*

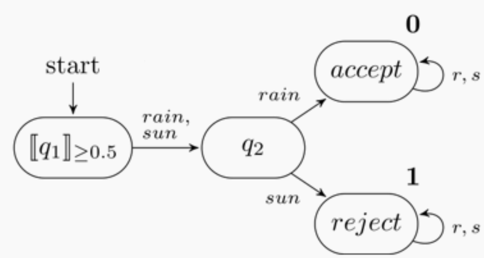


Fig 3. *A graph representation of a* ***p-Automaton****.*

[1] Castro P., Kilmurray C., and Piterman N., Tractable Probabilistic μ-Calculus that Expresses Probabilistic Temporal Logics. In *32nd Symposium on Theoretical Aspects of Computer Science (LIPIcs).* Schloss Dagstuhl. 2015.

[2] Chatterjee K. and Piterman N., Obligation Blackwell Games and p-Automata. *Journal of Symbolic Logic.* 2017. *To Appear*

Results μ^p-Calculus and p-Automata have the same Expressive Power

- *A framework to convert from μ^p-Calculus to p-Automata and backwards.*

Theorem 1. [3] *For every μ^p-calculus formula we can construct a p-automaton that accepts exactly those Markov chains that satisfy the formula.*

$$\mu X.\ rain \vee [\ \bigcirc X\ ]_{>0}$$

"With probability >0, eventually rain"

Fig 4. ***μ^p-Calculus*** *formula that either reads 'rain' or repeats itself. Since μ means finitely many repetitions, 'rain' must be read eventually.*

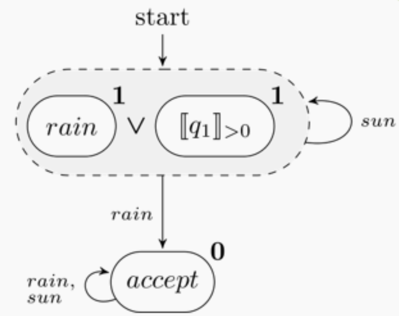


Fig 5. *A* ***p-Automaton*** *that either reads 'rain' or checks again. Since the topmost states have an odd number, 'rain' must be seen eventually.*

Theorem 2. [3] *For every p-automaton we can construct a μ^p-calculus formula satisfied in exactly those Markov chains accepted by the automaton.*

Future Work

- Different Models (e.g. *Markov Processes, Interval Markov Chains, Constraint Markov Chains*)
- More expressive language (Additional Operators: *Concurrency, Probabilistic and/or*)

London Hopper Colloquium 2017
Research Spotlight Competition

[3] Cauli C. and Piterman N., Equivalence of μ^p-Calculus and p-Automata. In *Implementation and Application of Automata – 22nd International Conference, CIAA 2017, Proceedings*. 2017. *To Appear*