Probabilistic Aspects of Computer Science: Probabilistic Automata

Serge Haddad

LMF, ENS Paris-Saclay & CNRS & INRIA

MPRI M1

Plan

1. Presentation

Properties of Stochastic Languages

Decidability Results

An introductive example

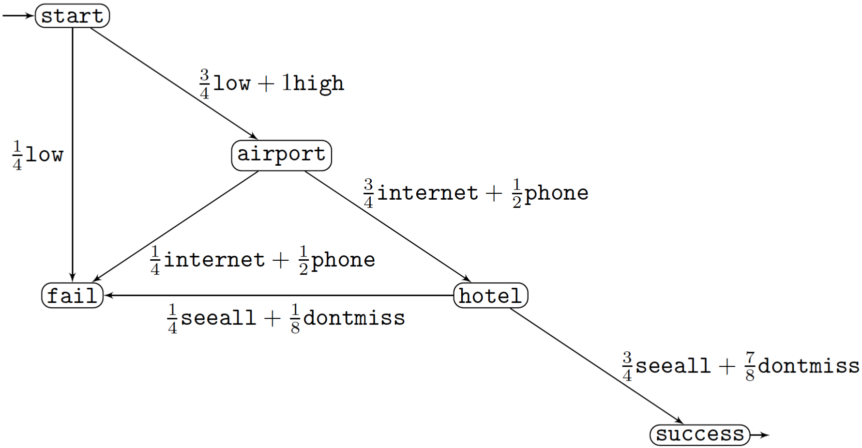
Planning holidays in a foreign country

1. Choosing which train or plane to use;
2. Renting an house or a room in an hotel;
3. Buying tickets for some exhibitions, etc.

Usually these actions must be planned before the holidays.

Thus one looks for an *a priori* optimal policy
that maximizes the probability to *reach* a goal.

Formalisation



The probability of success of lowcost · internet· seeall is $\frac{27}{64}$.

Probabilistic automata

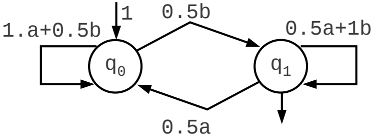
Probabilistic Automata (PA) are a variation of MDP where:

- The set of possible actions is the same in every state.
- There are no rewards.
- There is a subset of final states.

More formally, a PA $\mathcal{A} = (Q, A, \{\mathbf{P}_a\}_{a \in A}, \pi_0, F)$ is defined by:

- Q, the finite set of states;
- A, the finite alphabet;
- For all $a \in A$, $\mathbf{P}_a$, a probability transition matrix over S;
- π_0, the initial distribution over states and $F \subseteq Q$ the final states.

Illustration



- $A = \{a, b\}$;
- $Q = \{q_0, q_1\}$, $F = \{q_1\}$;
- $\pi_0[q_0] = 1$.

An edge from a state to another one is labelled by a vector of transition probabilities indexed by A. The vector is denoted by a formal sum.

For instance, the transition from q_0 to itself is labelled by $1a + 0.5b$ means that:

- when a is chosen in state q_0,
  the probability that the next state is q_0, $\mathbf{P}_a[q_0, q_0]$, is equal to 1.
- when b is chosen in state q_0,
  the probability that the next state is q_0, $\mathbf{P}_b[q_0, q_0]$, is equal to 0.5.

Policies in PA

Words are policies. When some finite word $w \stackrel{\text{def}}{=} a_1 \ldots a_n$ is selected, we are interested in the probability to be in a final state using w as a policy.

Given $\mathcal{A}$ a PA and $w \stackrel{\text{def}}{=} a_1 \ldots a_n \in A^*$ a word, the *acceptance probability* of w by $\mathcal{A}$ is defined by:

$$\mathbf{Pr}_{\mathcal{A}}(w) \stackrel{\text{def}}{=} \sum_{q \in Q} \pi_{\mathbf{0}}[q] \sum_{q' \in F} \left(\prod_{i=1}^{n} \mathbf{P}_{a_i} \right) [q, q']$$

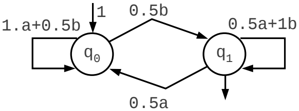
Notation. Given a word $w \stackrel{\text{def}}{=} a_1 \ldots a_n$, the probability matrix $\mathbf{P}_w$ is defined by $\mathbf{P}_w \stackrel{\text{def}}{=} \prod_{i=1}^{n} \mathbf{P}_{a_i}$. In particular $\mathbf{P}_{\varepsilon} = \mathbf{Id}$.

With these notations:

$$\mathbf{Pr}_{\mathcal{A}}(w) = \pi_0 \mathbf{P}_w \mathbf{1}_F^T$$

where $\mathbf{1}_F$ is the indicator vector of subset F.

Illustration



Observe that for all w, $\mathbf{Pr}_{\mathcal{A}}(w) = \mathbf{Pr}(\text{to be in } q_1 \text{ after following policy of } w)$ and $1 - \mathbf{Pr}_{\mathcal{A}}(w) = \mathbf{Pr}(\text{to be in } q_0 \text{ after following policy of } w)$

- $\mathbf{Pr}_{\mathcal{A}}(\varepsilon) = 0$, $\mathbf{Pr}_{\mathcal{A}}(a) = \frac{1}{2}\mathbf{Pr}_{\mathcal{A}}(\varepsilon) = 0$
- $\mathbf{Pr}_{\mathcal{A}}(ab) = \mathbf{Pr}_{\mathcal{A}}(a) + \frac{1}{2}(1 - \mathbf{Pr}_{\mathcal{A}}(a)) = \frac{1}{2}$
- $\mathbf{Pr}_{\mathcal{A}}(abb) = \mathbf{Pr}_{\mathcal{A}}(ab) + \frac{1}{2}(1 - \mathbf{Pr}_{\mathcal{A}}(ab)) = \frac{3}{4}$
- $\mathbf{Pr}_{\mathcal{A}}(abba) = \frac{1}{2}\mathbf{Pr}_{\mathcal{A}}(abb) = \frac{3}{8}$

More generally, the following recursive equations hold:

$$\mathbf{Pr}_{\mathcal{A}}(wa) = \frac{1}{2}\mathbf{Pr}_{\mathcal{A}}(w) \text{ and } \mathbf{Pr}_{\mathcal{A}}(wb) = \frac{1}{2}(1 + \mathbf{Pr}_{\mathcal{A}}(w))$$

from which one can derive an explicit expression of the acceptance probability:

$$\mathbf{Pr}_{\mathcal{A}}(a_1 \dots a_n) = \sum_{i=1}^{n} 2^{i-n-1} \cdot \mathbf{1}_{a_i=b}$$

Which word maximizes the acceptance probability?

Stochastic languages

We are interested in "useful" policies.

This directly leads to the introduction of *stochastic languages*. Let:

- $\mathcal{A}$ be a probabilistic automaton;
- $\theta \in [0,1]$ be a *threshold*;
- $\bowtie \in \{<, \leq, >, \geq, =, \neq\}$ be a comparison operator.

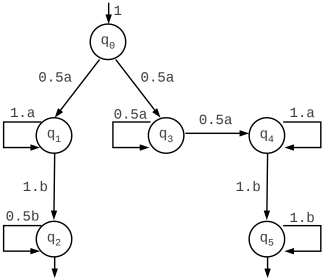
Then $L_{\bowtie\theta}(\mathcal{A})$ is defined by:

$$L_{\bowtie\theta}(\mathcal{A}) = \{w \in A^* \mid \mathbf{Pr}_{\mathcal{A}}(w) \bowtie \theta\}$$

For expressiveness and decidability issues, one also needs the following definitions.

- A *rational* PA is a PA with probability distributions over $\mathbb{Q}^Q$.
- A *rational stochastic language* is a stochastic language specified by a rational PA and a rational threshold.

Counting with PA



(a succinct representation with an omitted absorbing rejecting state)

Any word z different from $a^m b^n$ with $m > 0, n > 0$ cannot be accepted.

Let $w \stackrel{\text{def}}{=} a^m b^n$ with $m > 0, n > 0$. w can be accepted by:

- a path q_0, q_1^m, q_2^n with probability $\frac{1}{2^n}$;
- or by a family of paths q_0, q_3^r, q_4^s, q_5^n with $0 < r, s$ and $r + s = m$ with cumulated probability $\frac{1}{2} - \frac{1}{2^m}$.

Summing, one obtains: $\frac{1}{2} + \frac{1}{2^n} - \frac{1}{2^m}$.

Thus: $\mathcal{L}_{=0.5}(\mathcal{A}) = \{a^n b^n \mid n > 0\}$

Plan

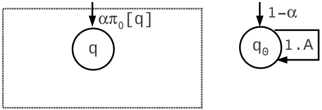
Expressiveness problems

Provide a minimal set of comparison operators and thresholds.

Position the stochastic languages w.r.t. the Chomsky hierarchy.

Study the closure properties of the stochastic languages.

A single threshold is enough



The value α depends on $\theta \neq \frac{1}{2}$ in the following way:

- If $\theta > \frac{1}{2}$ then $q_0 \notin F$ and $\alpha \stackrel{\text{def}}{=} \frac{1}{2\theta}$ so that for all $w \in A^*$,
$$\mathbf{Pr}_{\mathcal{A}'}(w) = \frac{1}{2\theta}\mathbf{Pr}_{\mathcal{A}}(w)$$
Thus $w \in L_{\bowtie \frac{1}{2}}(\mathcal{A}')$ iff $w \in L_{\bowtie \theta}(\mathcal{A})$.

- If $\theta < \frac{1}{2}$ then $q_0 \in F$ and $\alpha \stackrel{\text{def}}{=} \frac{1}{2(1-\theta)}$ so that for all $w \in A^*$,
$$\mathbf{Pr}_{\mathcal{A}'}(w) = \frac{1-2\theta+\mathbf{Pr}_{\mathcal{A}}(w)}{2(1-\theta)}$$
Thus $w \in L_{\bowtie \frac{1}{2}}(\mathcal{A}')$ iff $w \in L_{\bowtie \theta}(\mathcal{A})$.

Getting rid of (dis)equality

Given a PA $\mathcal{A}$, we build $\mathcal{A}'$ as follows.

- The set of states $Q' \stackrel{\text{def}}{=} Q \times Q$;
- $\mathbf{P}'_a[(q_1, q_2), (q'_1, q'_2)] \stackrel{\text{def}}{=} \mathbf{P}_a[q_1, q'_1]\mathbf{P}_a[q_2, q'_2]$;
- $\pi'_0[q_1, q_2] \stackrel{\text{def}}{=} \pi_0[q_1]\pi_0[q_2]$ and $F' \stackrel{\text{def}}{=} F \times (Q \setminus F)$.

Once a word w is selected,
the two components of the DES behave independently and so:

$$\mathbf{Pr}_{\mathcal{A}'}(w) = \mathbf{Pr}_{\mathcal{A}}(w)(1 - \mathbf{Pr}_{\mathcal{A}}(w))$$

Consequently $\mathbf{Pr}_{\mathcal{A}'}(w) \leq \frac{1}{4}$ with equality iff $\mathbf{Pr}_{\mathcal{A}}(w) = \frac{1}{2}$. Thus:

$$L_{\geq \frac{1}{4}}(\mathcal{A}') = L_{=\frac{1}{2}}(\mathcal{A})$$

Getting rid of “lower (or equal) than”

Given a PA $\mathcal{A}$, we build $\mathcal{A}'$ by complementing the final states. Then:

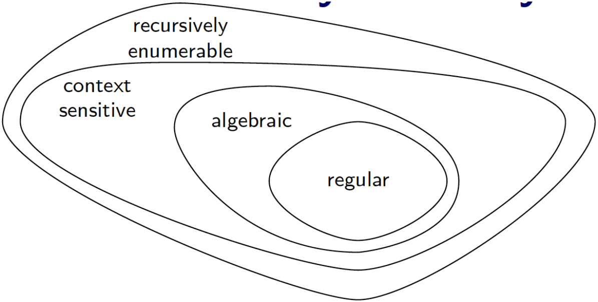
$$\mathbf{Pr}_{\mathcal{A}'}(w) = 1 - \mathbf{Pr}_{\mathcal{A}}(w)$$

And so:

$$L_{\geq\theta}(\mathcal{A}') = L_{<\theta}(\mathcal{A})$$

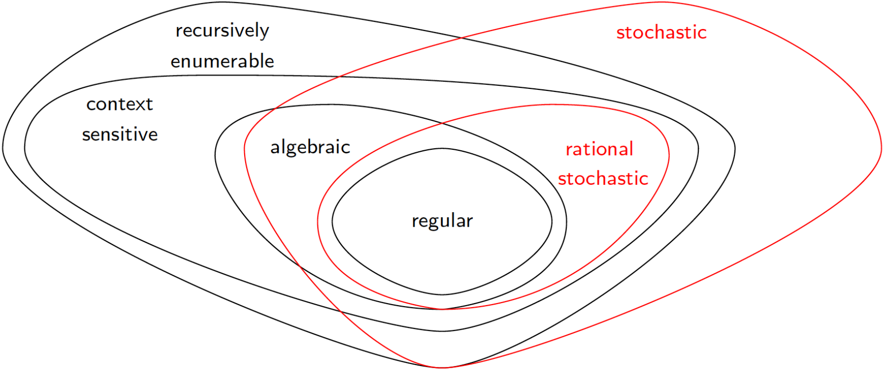
$$L_{>\theta}(\mathcal{A}') = L_{\leq\theta}(\mathcal{A})$$

The Chomsky hierarchy

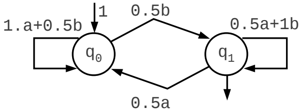


Class	Grammar	Device
Regular language	$L \to aR \vert a \vert \varepsilon$ with $L, R \in \Delta$, $a \in \Sigma$	Finite automaton
Algebraic language	$L \to R_1 \dots R_n$ with $L \in \Delta$ and $R_i \in \Delta \cup \Sigma$	Stack automaton
Context-sensitive language	$L_1 \dots L_m \to R_1 \dots R_n$ $m \leq n$, $(S \to \varepsilon)$ with $L_i, R_j \in \Delta \cup \Sigma$	Non determ. Turing machine performing in linear space
Recursively enumerable language	$L_1 \dots L_m \to R_1 \dots R_n$ avec $L_i, R_j \in \Delta \cup \Sigma$	Turing machine

Revisiting the Chomsky hierarchy



Non recursively enumerable languages



Define $v_a \stackrel{\text{def}}{=} 0$ and $v_b \stackrel{\text{def}}{=} 1$.

The acceptance probability of $w_1 \ldots w_n$ is the binary number $0.v_{w_n} \ldots v_{w_1}$.

So $\mathcal{L}_{>\theta}(\mathcal{A})$ is the set of representations of numbers (with finite binary development) greater than θ.

Thus given $0 \leq \theta < \theta' \leq 1$,

$$\mathcal{L}_{>\theta'}(\mathcal{A}) \subsetneq \mathcal{L}_{>\theta}(\mathcal{A})$$

So there is an uncountable number of stochastic languages implying that "most" of them are non recursively enumerable.

This result does not hold for rational stochastic languages.

Regular versus stochastic languages

A deterministic automaton is a stochastic automaton with probabilities in $\{0, 1\}$.

Thus regular languages are stochastic languages.

The language $\{a^n b^n \mid n > 0\}$ is a rational stochastic non regular language.

Non stochastic context-free languages (1)

$$L \stackrel{\text{def}}{=} \{a^{n_1} b a^{n_2} b \ldots a^{n_k} b a^* \mid \exists i > 1 \; n_i = n_1\}$$

is a non stochastic context-free language.

Proof.

L is context-free. Use a non deterministic automaton with a counter.

- With a counter one counts n_1 the number of a's until the first occurrence of b.
- Then one guesses an occurrence of b and decrements the counter by the occurrences of a until the next occurrence of b.
- If the counter is zero the word is accepted.

Assume that (1) $L = L_{>\theta}(\mathcal{A})$ or (2) $L = L_{\geq\theta}(\mathcal{A})$.

Let $\sum_{i=0}^{n} c_i X^i$ be the minimal polynomial of $\mathbf{P}_a$.

Since 1 is an eigenvalue of $\mathbf{P}_a$, one gets $\sum_{i=0}^{n} c_i = 0$
and there are positive and negative coefficients.

By definition, $\sum_{i=0}^{n} c_i \mathbf{P}_{a^i} = 0$ and so for any word w, $\sum_{i=0}^{n} c_i \mathbf{P}_{a^i w} = 0$.

Non stochastic context-free languages (2)

Proof (continued).

Let $Pos = \{i \mid 0 \leq i \leq n \wedge c_i > 0\}$ and $NonPos = \{i \mid 0 \leq i \leq n \wedge c_i \leq 0\}$.

Write Pos as $\{i_1, \ldots, i_k\}$.

Choose $w \overset{\text{def}}{=} ba^{i_1}b \ldots ba^{i_k}b$.

Case $L = L_{>\theta}(\mathcal{A})$. Let $0 \leq i \leq n$, by definition of L,

$$\pi_0 \mathbf{P}_{a^i w} \mathbf{1}_F^T > \theta \text{ iff } i \in \{i_1, \ldots, i_k\}$$

So:

$0 = \sum_{i=0}^{n} c_i \pi_0 \mathbf{P}_{a^i w} \mathbf{1}_F^T = \sum_{i \in Pos} c_i \pi_0 \mathbf{P}_{a^i w} \mathbf{1}_F^T + \sum_{i \in NonPos} c_i \pi_0 \mathbf{P}_{a^i w} \mathbf{1}_F^T$
$> (\sum_{i \in Pos} c_i)\theta + (\sum_{i \in NonPos} c_i)\theta = (\sum_{i=0}^{n} c_i)\theta = 0$

leading to a contradiction.

Case $L = L_{\geq\theta}(\mathcal{A})$. Let $0 \leq i \leq n$, by definition of L,

$$\pi_0 \mathbf{P}_{a^i w} \mathbf{1}_F^T \geq \theta \text{ iff } i \in \{i_1, \ldots, i_k\}$$

So: $0 = \sum_{i=0}^{n} c_i \pi_0 \mathbf{P}_{a^i w} \mathbf{1}_F^T = \sum_{i \in Pos} c_i \pi_0 \mathbf{P}_{a^i w} \mathbf{1}_F^T + \sum_{i \in NonPos} c_i \pi_0 \mathbf{P}_{a^i w} \mathbf{1}_F^T$
$> (\sum_{i \in Pos} c_i)\theta + (\sum_{i \in NonPos} c_i)\theta = (\sum_{i=0}^{n} c_i)\theta = 0$

leading to a contradiction.

Non context-free stochastic languages (1)

$$L \stackrel{\text{def}}{=} \{a^n b^n c^n \mid n > 0\}$$

is a non context-free rational stochastic language.

Proof.

Using Ogden's lemma it can be easily proved that L is not context-free.

Observe that $L = L_1 \cap L_2$ with $L_1 \stackrel{\text{def}}{=} \{a^n b^n c^+ \mid n > 0\}$ and $L_2 \stackrel{\text{def}}{=} \{a^+ b^n c^n \mid n > 0\}$.

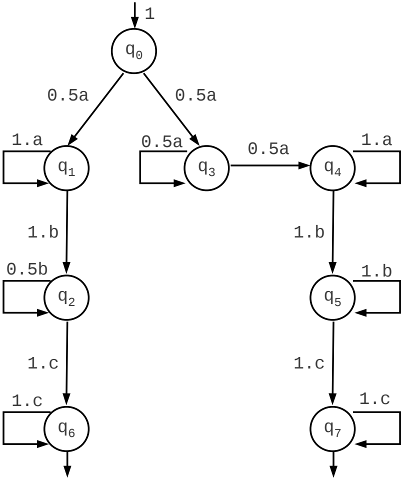
So we prove that:

- for $i \in \{1, 2\}$, $L_i = L_{=\frac{1}{2}}(\mathcal{A}_i)$ for some $\mathcal{A}_i$
- the family of languages $\{L = L_{=\frac{1}{2}}(\mathcal{A})\}_{\mathcal{A}}$ is closed under intersection.

Non context-free stochastic languages (2)

Proof (continued).

$$L_{=\frac{1}{2}}(\mathcal{A}) = \{a^n b^n c^+ \mid n > 0\}$$



Non context-free stochastic languages (3)

Proof (ended).

Let $L_{=\frac{1}{2}}(\mathcal{A}_1)$ and $L_{=\frac{1}{2}}(\mathcal{A}_2)$ be two arbitrary languages.

Using the previous construction, let $\mathcal{A}'_1$ and $\mathcal{A}'_2$ be automata such that:

- For any word w, $\mathbf{Pr}_{\mathcal{A}'_i}(w) \leq \frac{1}{4}$;
- $L_{=\frac{1}{2}}(\mathcal{A}_i) = L_{=\frac{1}{4}}(\mathcal{A}'_i)$.

One builds $\mathcal{A}$ as follows:

- The set of states $Q \stackrel{\text{def}}{=} Q'_1 \times Q'_2$;
- $\mathbf{P}_a[(q_1, q_2), (q'_1, q'_2)] \stackrel{\text{def}}{=} (\mathbf{P}'_1)_a[q_1, q'_1](\mathbf{P}'_2)_a[q_2, q'_2]$;
- $\pi'_0[q_1, q_2] \stackrel{\text{def}}{=} \pi_{1,0}[q_1]\pi_{2,0}[q_2]$ and $F \stackrel{\text{def}}{=} F'_1 \times F'_2$.

By construction, $\mathbf{Pr}_{\mathcal{A}}(w) = \mathbf{Pr}_{\mathcal{A}'_1}(w)\mathbf{Pr}_{\mathcal{A}'_2}(w)$.

So for all word w, $\mathbf{Pr}_{\mathcal{A}}(w) \leq \frac{1}{16}$ and $\mathbf{Pr}_{\mathcal{A}}(w) = \frac{1}{16}$ iff $\mathbf{Pr}_{\mathcal{A}'_1}(w) = \mathbf{Pr}_{\mathcal{A}'_2}(w) = \frac{1}{4}$.

Consequently,

$$L_{=\frac{1}{16}}(\mathcal{A}) = L_{=\frac{1}{2}}(\mathcal{A}_1) \cap L_{=\frac{1}{2}}(\mathcal{A}_2)$$

Inclusion in context-sensitive languages

The class of rational stochastic languages is strictly included in the class of context-sensitive languages.

Proof.

Context-sensitive languages are the languages for which membership checking can be performed by a non deterministic procedure in linear space.

A deterministic procedure in linear space (far from being optimal)

Pre-computation in constant space.

- Compute the l.c.m., say b, of denominators of θ, items of matrices $\{\mathbf{P}_a\}_{a\in A}$, and items of vector π_0.
- Build the integer matrices $\mathbf{P}'_a \overset{\text{def}}{=} b\mathbf{P}_a$ and vector $\pi'_0 \overset{\text{def}}{=} b\pi_0$.

For word $w \overset{\text{def}}{=} a_1 \dots a_n$, the problem becomes $\pi'_0(\prod_{i=1}^{n} \mathbf{P}'_{a_i})\mathbf{1}_F^T \bowtie \theta b^{n+1}$?

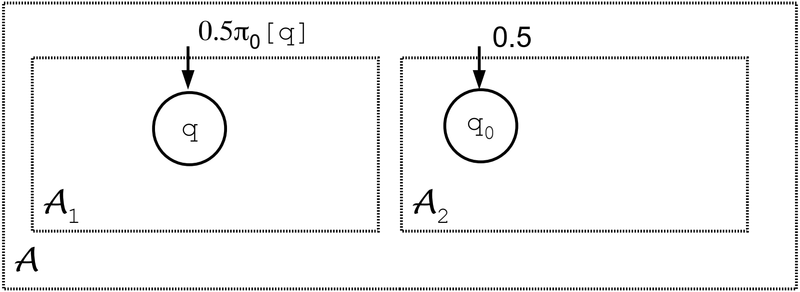
- Compute θb^{n+1} in space $O(n)$.
- Compute $\mathbf{v} \overset{\text{def}}{=} \pi'_0(\prod_{i=1}^{n} \mathbf{P}'_{a_i})$ by initializing $\mathbf{v}$ to π'_0 and then iteratively multiply it by $\mathbf{P}'_{a_i}$. The vectors are bounded by b^{n+1}. So this is performed in space $O(n)$.
- The sum and comparison are also done in space $O(n)$.

Operations with regular languages

The family of (rational) stochastic languages is closed under intersection and union with regular languages.

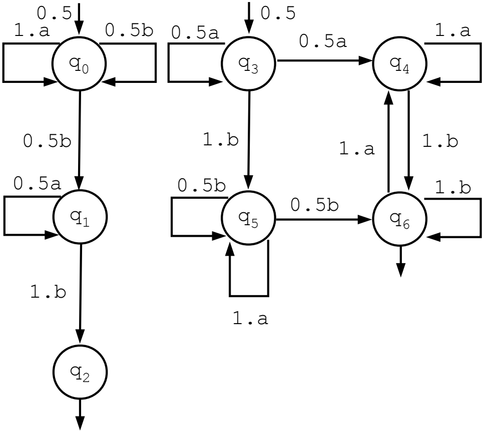
Proof.

Let $L_{\bowtie\theta}(\mathcal{A}_1)$ be a (rational) stochastic language (with $\bowtie \in \{>, \geq\}$) and $L_{=1}(\mathcal{A}_2)$ be a regular language.



$L_{\bowtie\frac{\theta}{2}}(\mathcal{A}) = L_{\bowtie\theta}(\mathcal{A}_1) \cup L_{=1}(\mathcal{A}_2)$ and $L_{\bowtie\frac{1+\theta}{2}}(\mathcal{A}) = L_{\bowtie\theta}(\mathcal{A}_1) \cap L_{=1}(\mathcal{A}_2)$

A stochastic language



$$L_{=\frac{1}{2}}(\mathcal{A}) = \{a^{m_1}b \dots ba^{m_k}b \mid 1 < k \wedge m_1 = m_k\}$$

since $\mathbf{Pr}_{\mathcal{A}}(a^{m_1}b \dots ba^{m_k}b) = \frac{1}{2}\left(\left(\frac{1}{2}\right)^{k+m_k-1} + 1 - \left(\frac{1}{2}\right)^{k+m_1-1}\right)$

Concatenation

The family of (rational) stochastic languages is not closed under concatenation with a regular language.

Proof.

Let $L \stackrel{\text{def}}{=} \{a^{m_1}b \dots ba^{m_k}b \mid 1 < k \wedge m_1 = m_k\}$ be the previous stochastic language.

Then $LA^* = \{a^{m_1}ba^{m_2}b \dots a^{m_k}ba^* \mid \exists i > 1 \; m_i = m_1\}$ which is not a stochastic language.

Iteration

The family of (rational) stochastic languages is not closed under Kleene star.

Proof.

Let $L \stackrel{\text{def}}{=} \{a^{m_1}b \dots ba^{m_k}b \mid 1 < k \wedge m_1 = m_k\}$ be the previous stochastic language.
Assume that $L^* = L_{\bowtie\theta}(\mathcal{A})$ with $\bowtie \in \{>, \geq\}$.

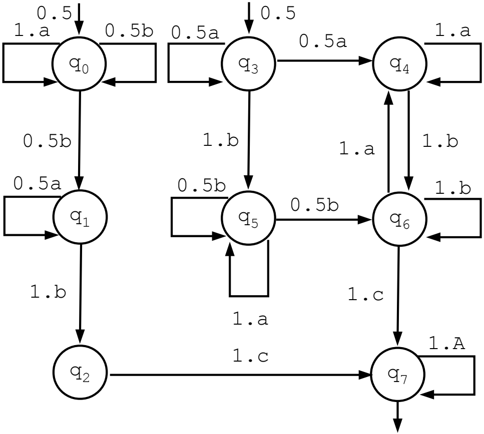
Let $\sum_{i=0}^{n} c_i X^i$ be the minimal polynomial of $\mathbf{P}_a$.
Since 1 is an eigenvalue of $\mathbf{P}_a$, one gets $\sum_{i=0}^{n} c_i = 0$
and there are positive and negative coefficients.

By definition, $\sum_{i=0}^{n} c_i \mathbf{P}_{a^i} = 0$ and so for any word w, $\sum_{i=0}^{n} c_i \mathbf{P}_{a^i w} = 0$.
Let $c_{i_1}, \dots, c_{i_k}$ be the positive coefficients of this polynomial.
Let $w \stackrel{\text{def}}{=} ba^{i_1}b(a^{i_2}b)^2 \dots (a^{i_k}b)^2$. $a^i w \in L^*$ iff $i \in \{i_1, \dots, i_k\}$.

Case $L^* = L_{>\theta}(\mathcal{A})$. Let $0 \leq i \leq n$, $\pi_0 \mathbf{P}_{a^i w} \mathbf{1}_F^T > \theta$ iff $i \in \{i_1, \dots, i_k\}$.
So: $0 = \sum_{i=0}^{n} c_i \pi_0 \mathbf{P}_{a^i w} \mathbf{1}_F^T > (\sum_{i=0}^{n} c_i)\theta = 0$
leading to a contradiction.

Case $L^* = L_{\geq\theta}(\mathcal{A})$. Let $0 \leq i \leq n$, $\pi_0 \mathbf{P}_{a^i w} \mathbf{1}_F^T \geq \theta$ iff $i \in \{i_1, \dots, i_k\}$.
So: $0 = \sum_{i=0}^{n} c_i \pi_0 \mathbf{P}_{a^i w} \mathbf{1}_F^T > (\sum_{i=0}^{n} c_i)\theta = 0$
leading to a contradiction.

A stochastic language



$$L_{=\frac{1}{2}}(\mathcal{A}) = \{a^{m_1}b \dots ba^{m_k}bcA^* \mid 1 < k \wedge m_1 = m_k\}$$

Homomorphism

The family of (rational) stochastic languages is not closed under homomorphism.

Proof.
Let $L \stackrel{\text{def}}{=} \{a^{m_1}b\ldots ba^{m_k}bcA^* \mid 1 < k \wedge m_1 = m_k\}$
be the previous stochastic language.

Define the homomorphism h from A to $A' \stackrel{\text{def}}{=} \{a, b\}$ by:

$$h(a) \stackrel{\text{def}}{=} a \qquad h(b) \stackrel{\text{def}}{=} b \qquad h(c) \stackrel{\text{def}}{=} \varepsilon$$

Then $h(L) = \{a^{m_1}ba^{m_2}b\ldots a^{m_k}ba^* \mid \exists i > 1\ m_i = m_1\}$
which is not a stochastic language.

Plan

Presentation

Properties of Stochastic Languages

3 Decidability Results

Two decision problems

Let $\mathcal{A}$ and $\mathcal{A}'$ be probabilistic automata.

First problem

Are $\mathcal{A}$ and $\mathcal{A}'$ equivalent?

$$\forall w \in A^* \ \mathbf{Pr}_{\mathcal{A}}(w) = \mathbf{Pr}_{\mathcal{A}'}(w)$$

Second problem

Is $L_{\bowtie\theta}(\mathcal{A})$ equal to $L_{\bowtie'\theta'}(\mathcal{A}')$?

For deterministic automata this is the same problem.
It can be solved in polynomial time by a product construction
which provides a witness of non equivalence of size less than $|Q||Q'|$.

Linear algebra recalls

Let $\mathbf{v_0} \in \mathbb{R}^n$ and $\mathbf{v_1}, \ldots, \mathbf{v_k}$ be linearly independent vectors of $\mathbb{R}^n$.

How to check whether $\mathbf{v}_0$ is a linear combination of $\mathbf{v_1}, \ldots, \mathbf{v_k}$?

- Solve in $O(k^3 + n^2)$

$$\begin{pmatrix} \mathbf{v}_1[1] & \ldots & \mathbf{v}_k[1] \\ \ldots & \ldots & \ldots \\ \mathbf{v}_1[n] & \ldots & \mathbf{v}_k[n] \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_k \end{pmatrix} = \begin{pmatrix} \mathbf{v}_0[1] \\ \vdots \\ \mathbf{v}_0[n] \end{pmatrix}$$

- When $\mathbf{v_1}, \ldots, \mathbf{v_k}$ are orthogonal

(i.e. for all $a \neq b$, $\mathbf{v}_a \cdot \mathbf{v}_b \stackrel{\text{def}}{=} \sum_{i=1}^{n} \mathbf{v}_a[i]\mathbf{v}_b[i] = 0$)

Compute in $O(kn)$ the orthogonal projection

$$\mathbf{w}_0 = \sum_{i=1}^{k} \frac{\mathbf{v}_0 \cdot \mathbf{v}_i}{\mathbf{v}_i \cdot \mathbf{v}_i} \mathbf{v}_i$$

Check in $O(n)$ whether $\mathbf{v}_0 = \mathbf{w}_0$.

Principles of equivalence checking

Enumeration of words

Looking for a counter-example whose length is increasing starting with word ε.

A stack

Managing a stack of words w in order to find counter-examples aw for all $a \in A$.

For efficiency purposes, the stack contains tuples $(\mathbf{P}_w \mathbf{1}_F, \mathbf{P}'_w \mathbf{1}_{F'}, w)$.

An orthogonal family for restricting the enumeration

Gen is a set of independent orthogonal vectors of $\mathbb{R}^{Q \cup Q'}$.

If w is not a counter-example, check if $\mathbf{v} \stackrel{\text{def}}{=} (\mathbf{P}_w \mathbf{1}_F, \mathbf{P}'_w \mathbf{1}_{F'})$ is generated by Gen.

- producing $\mathbf{v}'$ the orthogonal projection of $\mathbf{v}$ on subspace spanned by Gen;
- comparing $\mathbf{v}'$ to $\mathbf{v}$.

If $\mathbf{v}' \neq \mathbf{v}$ then:

- w is added to the stack;
- $\mathbf{v} - \mathbf{v}'$ is added to Gen.

The algorithm

If $\pi_0 \cdot \mathbf{1}_F \neq \pi'_0 \cdot \mathbf{1}_{F'}$ **then return**$(\mathrm{false}, \varepsilon)$

$Gen \leftarrow \{(\mathbf{1}_F, \mathbf{1}_{F'})\}$; $\mathbf{Push}(Stack, (\mathbf{1}_F, \mathbf{1}_{F'}, \varepsilon))$

Repeat

  $(\mathbf{v}, \mathbf{v}', w) \leftarrow \mathbf{Pop}(Stack)$

  For $a \in A$ **do**

    $\mathbf{z} \leftarrow \mathbf{P}_a \mathbf{v}$; $\mathbf{z}' \leftarrow \mathbf{P}'_a \mathbf{v}'$

    If $\pi_0 \cdot \mathbf{z} \neq \pi'_0 \cdot \mathbf{z}'$ **then return**(false, aw)

    $\mathbf{y} \leftarrow \mathbf{0}$; $\mathbf{y}' \leftarrow \mathbf{0}$

    For $(\mathbf{x}, \mathbf{x}') \in Gen$ **do**

      $\mathbf{y} \leftarrow \mathbf{y} + \frac{\mathbf{z}\cdot\mathbf{x}}{\mathbf{x}\cdot\mathbf{x}}\mathbf{x}$

      $\mathbf{y}' \leftarrow \mathbf{y}' + \frac{\mathbf{z}'\cdot\mathbf{x}'}{\mathbf{x}'\cdot\mathbf{x}'}\mathbf{x}'$

    If $(\mathbf{z}, \mathbf{z}') \neq (\mathbf{y}, \mathbf{y}')$ **then**

      $\mathbf{Push}(Stack, (\mathbf{z}, \mathbf{z}', aw))$

      $Gen \leftarrow Gen \cup \{(\mathbf{z} - \mathbf{y}, \mathbf{z}' - \mathbf{y}')\}$

Until $\mathbf{IsEmpty}(Stack)$

return(true)

Complexity

Time complexity

An item is pushed on the stack iff an item is added to Gen.

There can be no more than $|Q| + |Q'|$ items in Gen.

So there are at most $|Q| + |Q'|$ iterations of the external loop.

The index of the first inner loop ranges over A
while the index of the most inner loop ranges over Gen.

The operations inside the most inner loop are done in $O(|Q| + |Q'|)$.

This leads to an overall time complexity of $O((|Q| + |Q'|)^3|A|)$.

Length of witnesses

In addition, the length of the witness is at most $|Q| + |Q'|$.
(also valid for deterministic automata)

Correctness

Assume that the automata are not equivalent and that the algorithm returns **true**.

Let u be a non examined word such that $\mathbf{Pr}_{\mathcal{A}}(u) \neq \mathbf{Pr}_{\mathcal{A}'}(u)$.

Let $u \stackrel{\text{def}}{=} w'w$ with $w(\neq u)$ the greatest suffix examined by the algorithm.

Among such words u, pick one word such that $|w'|$ is minimal.

Claim. There exists w'' that has been inserted in the stack before w such that $\mathbf{Pr}_{\mathcal{A}}(w'w'') \neq \mathbf{Pr}_{\mathcal{A}'}(w'w'')$.

Let $Gen = \{w_1, \ldots, w_k\}$ when examining w, there exist $\lambda_1, \ldots, \lambda_k$ such that:

So: $\mathbf{P}_w \mathbf{1}_F = \sum_{i=1}^{k} \lambda_i \mathbf{P}_{w_i} \mathbf{1}_F$ and $\mathbf{P}'_w \mathbf{1}_{F'} = \sum_{i=1}^{k} \lambda_i \mathbf{P}'_{w_i} \mathbf{1}_{F'}$

$$\mathbf{Pr}_{\mathcal{A}}(w'w) \stackrel{\text{def}}{=} \pi_0 \mathbf{P}_{w'} \mathbf{P}_w \mathbf{1}_F = \sum_{i=1}^{k} \lambda_i \pi_0 \mathbf{P}_{w'} \mathbf{P}_{w_i} \mathbf{1}_F = \sum_{i=1}^{k} \lambda_i \mathbf{Pr}_{\mathcal{A}}(w'w_i)$$

Similarly: $\mathbf{Pr}_{\mathcal{A}'}(w'w) = \sum_{i=1}^{k} \lambda_i \mathbf{Pr}_{\mathcal{A}'}(w'w_i)$

So there exists i, with $\mathbf{Pr}_{\mathcal{A}}(w'w_i) \neq \mathbf{Pr}_{\mathcal{A}'}(w'w_i)$.

Let $w' \stackrel{\text{def}}{=} w'''a$. aw_i is examined by the algorithm.

So the word $u' \stackrel{\text{def}}{=} w'w_i$ has a decomposition $u' \stackrel{\text{def}}{=} z'z$ where z the greatest suffix examined by the algorithm has for suffix aw_i. So $|z'| < |w'|$: a contradiction.

Undecidability of the equality problem

Given $\mathcal{A}$ a rational stochastic automaton,
the question $L_{=\frac{1}{2}}(\mathcal{A}) = \{\varepsilon\}$? is undecidable.

Proof.

By reduction of the undecidable Post correspondence problem (PCP):
Given an alphabet A and two morphisms φ_1, φ_2 from A to $\{0,1\}^+$,
does there exist a word $w \in A^+$ such that $\varphi_1(w) = \varphi_2(w)$?

Already undecidable for a restriction where the images of letters lie in $(10+11)^+$.
Inserting a 1 before each letter of images reduces the former problem to the latter.

A word $w \stackrel{\text{def}}{=} a_1 \dots a_n \in (10+11)^+$ defines a value $val(w) \in [0,1]$ by:

$$val(w) \stackrel{\text{def}}{=} \sum_{i=1}^{n} \frac{a_i}{2^{n+1-i}}$$

Since every word starts with a 1, $val(w) = val(w')$ implies $w = w'$.

Reduction of PCP

For $w \in A^+$ and $i \in \{1,2\}$, define $val_i(w) \stackrel{\text{def}}{=} val(\varphi_i(w))$.

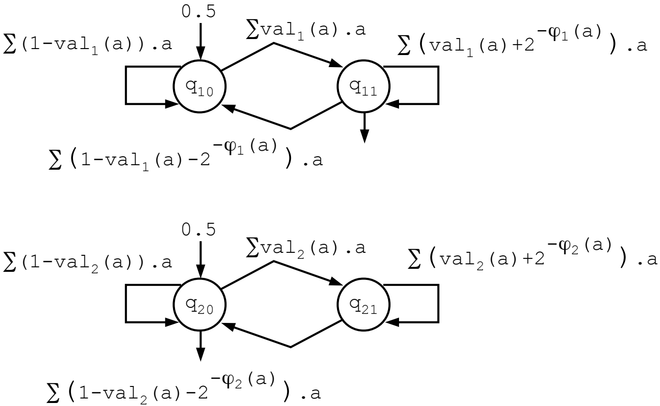
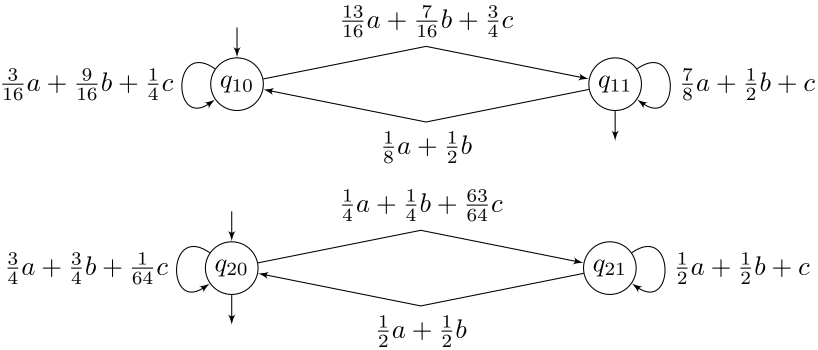


Illustration of the reduction

A	a	b	c
φ_1	$(1)0(1)1$	$(1)0(1)0$	$(1)1$
φ_2	$(1)0$	$(1)0$	$(1)1(1)1(1)1$

A	a	b	c
val_1	$\frac{13}{16}$	$\frac{7}{16}$	$\frac{3}{4}$
val_2	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{63}{64}$



Correctness of the reduction

The recurrence equation:

$$\mathbf{1}_{q_{i0}} \mathbf{P}_{wa} \mathbf{1}_{q_{i1}}^T = \mathbf{1}_{q_{i0}} \mathbf{P}_{w} \mathbf{1}_{q_{i1}}^T (val_i(a) + 2^{-|\varphi_i(a)|}) + (1 - \mathbf{1}_{q_{i0}} \mathbf{P}_{w} \mathbf{1}_{q_{i1}}^T) val_i(a)$$
$$= val_i(a) + 2^{-|\varphi_i(a)|} \mathbf{1}_{q_{i0}} \mathbf{P}_{w} \mathbf{1}_{q_{i1}}^T$$

By induction we obtain that for all $w \stackrel{\text{def}}{=} a_1 \dots a_n$:

$$\mathbf{1}_{q_{i0}} \mathbf{P}_{w} \mathbf{1}_{q_{i1}}^T = \sum_{j=1}^{n} val_i(a_j) 2^{-\sum_{j<k\leq n} |\varphi_i(a_k)|} = val_i(w)$$

So for $w \in A^+$: $\mathbf{Pr}_{\mathcal{A}}(w) = \frac{1}{2}(val_1(w) + 1 - val_2(w))$.

Thus $w \in L_{=\frac{1}{2}}(\mathcal{A})$ iff $val(\varphi_1(w)) = val(\varphi_2(w))$ implying that $\varphi_1(w) = \varphi_2(w)$.