LTS minimisation, population-, reward-semantics

Tanja

IST

Notation: $[n]$ for set $\{1, \ldots, n\}$.

Problem 1. (equivalence) Given M_1 and M_2, is $L(M_1) = L(M_2)$?

Problem 2. (minimisation) Given M_1 with n_1 states, and $n_2 < n_1$, is there M_2 with n_2 states such that $L(M_1) = L(M_2)$?

Definition 1. Probabilistic automaton (PA) is given by

- $[n]$ set of states,
- $[m] = \Sigma$ the set of labels,
- $\pi_0 \in [0, 1]^n$ the initial distribution (row notation),
- for each label $\lambda \in [m]$, a transition probability matrix $\mathbf{P}_\lambda \in [0, 1]^{n \times n}$ - hence a family of transition probability matrices,
- $\mu \in \{0, 1\}^{[n]}$, the accepting vector (column notation), where the 1-entries are the accepting states.

Extend notation with $\mathbf{P}_\epsilon = \mathbf{I}_n$ and $\mathbf{P}_{\lambda w} := \mathbf{P}_\lambda \mathbf{P}_w$. The language of a PA $\mathcal{A}$ is $\mathcal{L}_\mathcal{A} : \Sigma^* \to [0, 1]$, given by $\mathsf{P}_\mathcal{A}(w) := \pi_0 \mathbf{P}_w \mu$.

The equivalence check for two PA's can be done in polynomial time [6, 1], and minimisation can be done in PSPACE, by reduction from the existential theory of the reals [4, 2], and is shown to be NP-hard [2], by a reduction from the restricted hypercube problem.

In case that the PA is not initialised, minimisation requires checking the forward bisimulation [3], that can be computed in polynomial time.

Det. finite automata (DFA) are a special case of PA's, when the distributions defined with $\mathbf{P}_\lambda$ are Dirac functions, and the minimisation can be done in polynomial time (survey [7], most known algorithms are Hopcroft's partition refinement and Brzozowski elegant construction which works also for NFA).

Weighted automata are a generalisation of PA's when $\mathbf{P}_\lambda$ is not necessarily stochastic, and the minimisation is also polynomial [2].

1 PROBLEM FORMULATION

We are interested in labelled Markov chains (LMC), a special case of PA's with one non-accepting, sink state $n \in [n]$, when for each state $i \in [n]$, there is exactly one label $\lambda \in \Sigma$ for which $\mathbf{P}_\lambda(i, n) = 0$ and for all other labels $\lambda' \neq \lambda$, $\mathbf{P}_{\lambda'}(i, n) = 1$. The accepting vector assigns weight 1 to each accepting state.

LMC's are also Markov chains where each state bears a unique label from the alphabet Σ, and the traces are naturally defined.

We are interested in the problems of equivalence and minimisation for the standard trace semantics and several relaxed notions of semantics - transient semantics, weighted transient semantics and function transient semantics.

Definition 2. A labelled transition system (LMC) is given by

- $[n]$ set of states,
- $[m] = \Sigma$ the set of labels,
- $\pi_0 \in [0,1]^n$ the initial distribution,
- the labelling matrix $\mathbf{V} \in \{0,1\}^{n\times m}$
- the transition probability matrix $\mathbf{P} \in [0,1]^{n\times n}$,
- the label-weight vector $\mu : \Sigma \to \mathbb{R}$.

Then, the state-trace semantics gives the probability of observing a finite sequence of states $i_0, \ldots, i_k \in [n]$:

$$\Delta(\pi_0, i_0 \ldots i_k) := \pi_0(i_0) \prod_{j=0}^{k-1} P_{i_j, i_{j+1}},$$

and the classical, label-trace semantics is given by a language $L : \Sigma^* \to [0,1]$, such that for $w \in \Sigma^k$,

$$L(w) = \Delta(\pi_0, w) = \sum_{..} \Delta(\pi_0, i_1 \ldots, i_k).$$

. We are also interested in relaxed semantics:

- $L_0 : \mathbb{N} \times \Sigma \to [0,1]$, s.t. $L_0(k,a) = \sum_{w \text{ ends with } a} L(\pi_0, w)$ [1].
- $R_0 : \mathbb{N} \times \Sigma \to \mathbb{R}$, s.t. $R_0(k,a) = L_0(k,a)\mu(a)$.
- F_0 - class of functions for each label

2 SOME KNOWN AND EXTENDED RESULTS

2.1 EQUIVALENCE

We first show that the termination results about equivalence and minimisation translate from PA to LMC: equivalence wrt. L, L_0 and R_0 semantics can be checked in polynomial time (L is shown in [1], we modify the algorithm so to deal with the relaxed semantics, and use Cayley-Hamilton thm.).

Theorem 1. The condition on initial distributions which ensure equivalence is equivalent to the following relations on the PA's constructed from LMC as shown

[1] the subscript 0 refers to that the 'memory' in terms of the sequence of labels that are remembered is 0

in [1] (labels are moved from states to all the outgoing edges, all states except the sink state are accepting):

for trace equivalence ([1]),

$$\rho_1 \sim \rho_2 \text{ iff } \left(\forall \sigma \in \Sigma . \rho_1 \nu_F^1 = \rho_2 \nu_F^2 \text{ and } \rho_1 M_\sigma^1 \sim \rho_2 M_\sigma^2\right),$$

which is not weaker than

$$\rho_1 \sim_1 \rho_2 \text{ iff } \left(\forall \sigma \in \Sigma . \rho_1 M_\sigma^1 \nu_F^1 = \rho_2 M_\sigma^2 \nu_F^2 \text{ and } \rho_1 T_1 \sim_1 \rho_2 T_2\right),$$

which is not weaker than transient-reward equivalence:

$$\rho_1 \sim_2 \rho_2 \text{ iff } \left(\forall \sigma \in \Sigma . \rho_1 M_\sigma^1 w_F^1 = \rho_2 M_\sigma^2 w_F^2 \text{ and } \rho_1 T_1 \sim_2 \rho_2 T_2\right).$$

Proof. The proof for relation $\sim$ and language L can be found in [1]. The base condition is that the probability mass in the states bearing the same label must be the same between the two systems. Moreover, whichever letter is processed, the same condition must hold. This is sufficient for equivalence, because it ensures that after processing any sequence of letters, the condition holds.

For the relation $\sim_1$, the probability mass in the states bearing the same label must be the same between the two systems. However, the condition should hold after processing in time, independently of which letter was processed. Similarly, for the relation $\sim_2$, only the base condition changes.

Theorem 2. There is a polynomial-time algorithm for deciding the equivalence problem wrt. L, L_0 and R_0 (L is done in [1]).

2.2 MINIMISATION: DECIDABILITY

The minimisation wrt. each of the semantics can be reduced to the existential theory of the reals.

Proposition 1. The minimality problem wrt. $\sim$, $\sim_1$ and $\sim_2$ can be reduced to $ExTh(\mathbb{R})$ ($\sim$ is done in [4, 2]).

Lemma 1. If LMC M_1 and M_2 are L_0 equivalent up to time step $n+m-1$, then $L_0(M_1) = L_0(M_2)$.

Proof. Write down the initial vector as $\pi = [x_1, \ldots, x_n, -y_1, \ldots, -y_m]$, matrix $A(k) = [A_1^k, 0; 0, A_2^k]$, projection matrix $\Pi = [\Pi_1, 0; 0, \Pi_2]$, for $k \in \mathbb{N}$. Then, by Cayley-Hamilton theorem, A^{n+m} is a linear combination $A^{n+m} = \mathcal{L}(A^{n+m-1}, \ldots, A, I) = \rho_{n+m-1} A^{n+m-1} + \ldots + \rho_1 A + \rho_0 I$. So, if for all $k \leq n+m-1$, $\pi A(k) \Pi = 0$, then also for $k = n+m$, we have $\pi A(n+m) \Pi = \pi \mathcal{L}(A^{n+m-1}, \ldots, A, I) \Pi = 0$ and by induction for all $k \in \mathbb{N}$, $\pi A(k) \Pi = 0$.

Remark 1. For $n_2 = m = |\Sigma|$, and if the chain underlying M_1 has a unique stationary distribution π, the minimality can be resolved by a polynomial algorithm.

Proof. This observation follows from [5]. The problem of whether M_1 and M_2 are equivalent, in this case, is the problem of whether the projection of M_1 with respect to lumping all the states with a same label is also a Markov chain. This is exactly the problem which the authors of [5] address, and they practically check the equivalence between the quotient $M_1/\sim$, constructed effectively based on the unique stationary distribution of M_1. As the equivalence is decided in polynomial time, the result follows.

For example, the chains in Fig. **??**a) are trace equivalent, and therefore they are also transient-equivalent and reward-equivalent. The chain in Fig. **??**a) is trace equivalent but not bisimilar in a classical sense, in Fig. **??**b) is transient equivalent for initial distribution equal to the stationary one $(1/3, 2/3)$ and $(1/3, 1/3, 1/3)$, but these distributions are not trace equivalent; In Fig. **??**c) is transient reward equivalent but not transient equivalent. In the last picture, we show two automata where labels determine a function over the output variable. The three state automaton can be transformed into an equivalent two-state automaton, and that is the minimal one, in case the set of functions to consider is pre-determined (in this concrete case does not allow a function $p' = p + 1/2$).

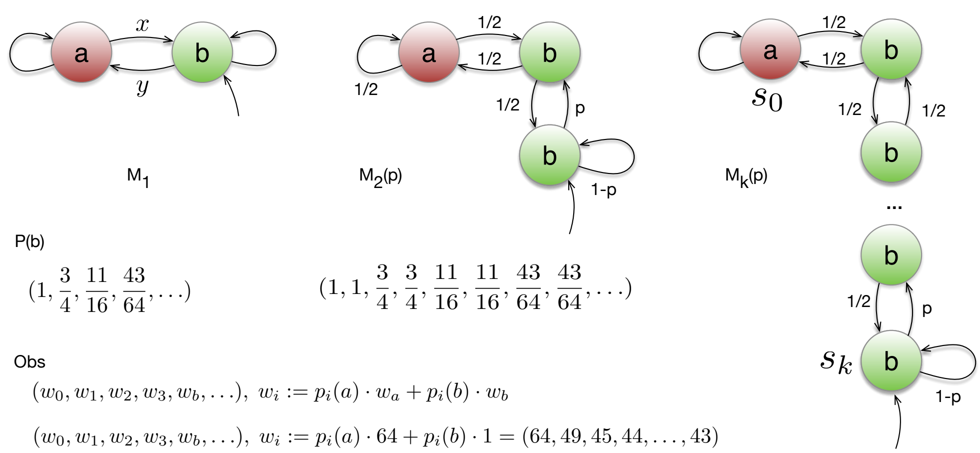
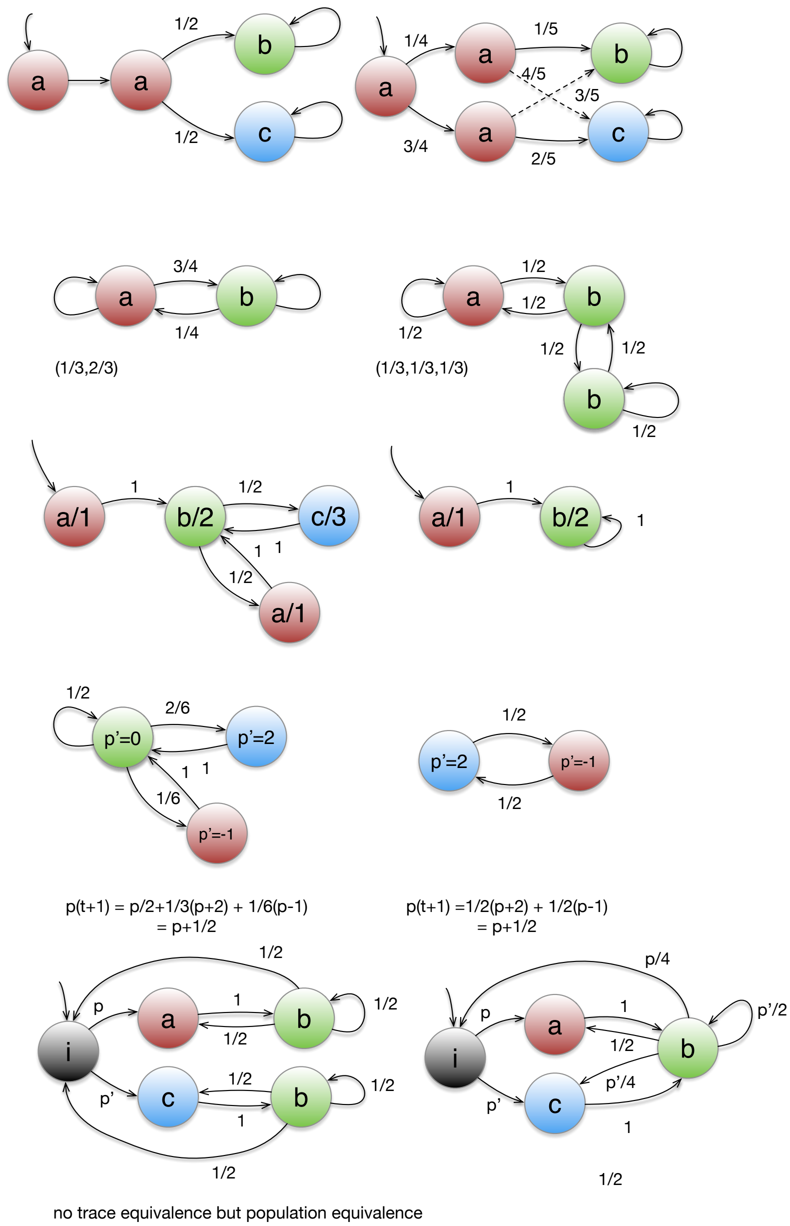


Fig. 1.

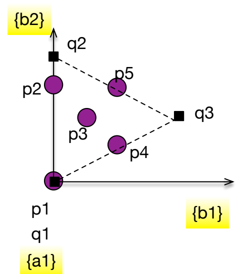
2.3 MINIMISATION: COMPLEXITY

In a special case that the original LMC is irreducible and aperiodic, we know its stationary distribution, and $n_2 = m = |\Sigma|$, the problem is reduced to deciding equivalence between M_1 and the aggregated process $M_1/_\sim$ where all states with the same label are equivalent, and the rates are constructed under assuming the distribution among lumped states that is in agreement with the stationary π.



1/2
1/2
a
a
b
c
1/4
1/5
4/5
3/5
3/4
2/5
3/4
1/4
(1/3,2/3)
1/2
1/2
1/2
1/2
1/2
1/2
(1/3,1/3,1/3)
a/1
b/2
c/3
1
1/2
1
1
1/2
a/1
b/2
1
1
1/2
2/6
p'=0
p'=2
1
1
1/6
p'=-1
1/2
1/2
p'=2
p'=-1
p(t+1) = p/2+1/3(p+2) + 1/6(p-1)
= p+1/2
p(t+1) =1/2(p+2) + 1/2(p-1)
= p+1/2
1/2
p
i
1
1/2
1/2
p'
1/2
1
1/2
1/2
p/4
p'/2
p
1
1/2
p'/4
p'
1
1/2
no trace equivalence but population equivalence

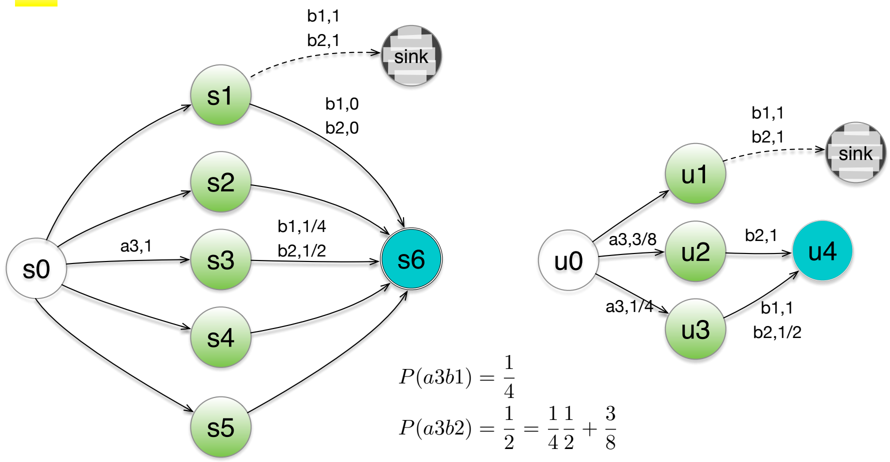
Fig. 2.



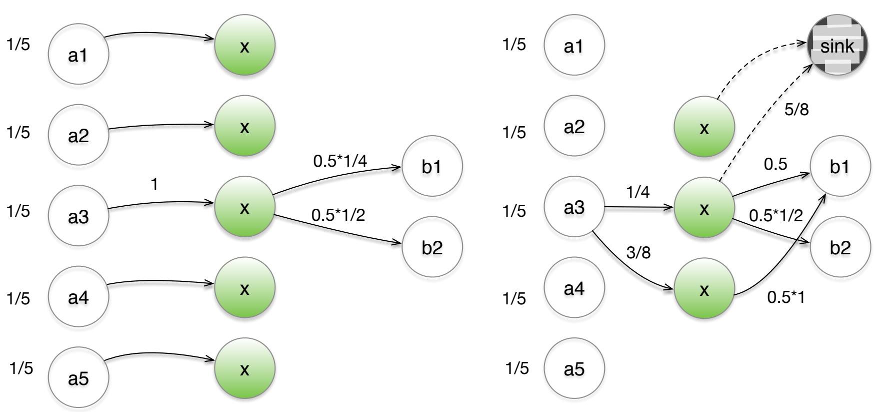
p1	0	0
p2	0	3/4
p3	1/4	1/2
p4	1/2	1/4
p5	1/2	3/4

q1	0	0
q2	0	1
q3	1	1/2

$$p_1 = 1q_1$$
$$p_2 = \frac{3}{4}q_2$$
$$p_3 = \frac{3}{8}q_2 + \frac{1}{4}q_3$$
$$p_4 = \frac{1}{2}q_3$$
$$p_5 = \frac{1}{2}q_3 + \frac{1}{2}q_2$$



$$P(a3b1) = \frac{1}{4}$$
$$P(a3b2) = \frac{1}{2} = \frac{1}{4}\frac{1}{2} + \frac{3}{8}$$



$$P(a3xb1) = 0.5 \cdot \frac{1}{4}$$
$$P(a3xb2) = 0.5 \cdot \frac{1}{4} = 0.5 \cdot (\frac{1}{4}\frac{1}{2} + \frac{3}{8})$$

transient: $(Unif(a_i), 0, 0, 0) \rightarrow (0, 1, 0, 0) \rightarrow (0, 0, \frac{5}{20}, \frac{9}{20})$

Fig. 3.

Remark 2. Rubino/Sericola's works on weak lumpability relate. First, notice that a distribution of a projected process, where a projection is induced by lumping states, is equivalent to a distribution of an LMC, where all states that are lumped are labelled with a same letter, but using different letters for different aggregates. Then, the question whether the projected process is Markov hom. is equivalent to asking if there is a process with $n_2 = |\Sigma|$ states that is equivalent to the initial one.

Theorem 3. *Minimisation for LMC with respect to $\sim$ is NP-hard.*

The proof is inspire from the proof for PA's in [2], we use a reduction from the restricted hypercube problem, shown in Fig. 2.2.

Proposition 2. *Tanja - open: ? Minimisation for LMC with respect to $\sim_1$ can be done in polynomial time.*

Proposition 3. *Tanja - open: ? If the LMC is 'branching' for all labels, then the minimisation with respect to $\sim$ and $\sim_1$ can be done in polynomial time, because it must be either forward or backward bisimulation.*

Notice that the construction used to show the reduction for $\sim_1$ minimisation cannot be used for the transient semantics, because all states labelled with x can be lumped together. The example in Fig. 2.2e) shows two LMC's that are not trace equivalent, but are transient equivalent.

3 RELATION TO THE MARKOV CHAIN AGGREGATION(LUMPING)

Tanja - open: Discuss how Rubino/Sericola's works relate. First, notice that the problem is equivalent to checking the trace equivalence between M_1 and M_2, where each lumped state has a different label. Then, if the chain has a unique stationary, the equivalence must hold also in the stationary, so we can construct the "macro-transitions" in M_2 effectively, if the stationary is known. In each step, the set of distributions that are "allowed" shrinks. What they don't show in Doyen's work are the mathematical properties of the allowed set of distributions, e.g. that the set of 'allowed' distributions is closed under convex combinations.

4 References

References

1. L. Doyen, T. A. Henzinger, and J.-F. Raskin. Equivalence of labeled markov chains. *International journal of foundations of computer science*, 19(03):549–563, 2008.
2. S. Kiefer and B. Wachter. Stability and complexity of minimising probabilistic automata. In *ICALP 2014*. Springer, 2015.
3. K. G. Larsen and A. Skou. Bisimulation through probabilistic testing (preliminary report). In *Proceedings of the 16th ACM SIGPLAN-SIGACT symposium on Principles of programming languages*, pages 344–352. ACM, 1989.
4. P. Mateus, D. Qiu, and L. Li. On the complexity of minimizing probabilistic and quantum automata. *Information and Computation*, 218:36–53, 2012.
5. G. Rubino and B. Sericola. A finite characterization of weak lumpable Markov processes. part I: The discrete time case. *Stochastic processes and their applications*, vol. 45, no 1:115–125, 1993.
6. W.-G. Tzeng. A polynomial-time algorithm for the equivalence of probabilistic automata. *SIAM Journal on Computing*, 21(2):216–227, 1992.
7. B. W. Watson. A taxonomy of finite automata minimization algorithms. 1993.