Use of a Metric in Supervisory Control of Probabilistic Discrete Event Systems

Vera Pantelic and Mark Lawford

SQRL, Department of Computing and Sofware, Faculty of Engineering, McMaster University, Hamilton, ON, Canada L8S 4K1 (e-mail: pantelv at mcmaster dot ca, lawford at mcmaster dot ca)

Abstract: This work represents a natural extension of our work on optimal probabilistic supervisory control of probabilistic discrete event systems (PDESs). In that work, a pseudometric on the initial states of two probabilistic generators that represent probabilistic systems is used to measure the distance between the two systems. The pseudometric is given a fixed point characterization. This paper gives a logical characterization of the same pseudometric such that the distance between two systems is measured by a formula that distinguishes between the systems the most. A trace characterization of the pseudometric is then derived from the logical characterization. Further, the solution of the problem of approximation of a given probabilistic generator with another generator of a prespecified structure is suggested such that the new model is as close as possible to the original one in the pseudometric. The significance of the approximation is then discussed.

Keywords: Supervisory control, stochastic systems, discrete event systems, metrics.

1. INTRODUCTION

A supervisory control framework for PDESs was proposed in Lawford and Wonham (1993). PDESs are modeled as probabilistic generators inspired by Garg et al. (1999). Further, deterministic supervisors for DES are generalized to *probabilistic supervisors*: after observing a string s, the probabilistic supervisor enables an event σ with a certain probability. The supervisory control problem considered in Lawford and Wonham (1993) is to find, if possible, a supervisor under whose control the behaviour of a plant is identical to a given probabilistic specification. Further, Lawford and Wonham (1993) show that a plant under probabilistic control can generate a much larger class of probabilistic languages than deterministic control, and give the necessary and sufficient conditions for the existence of a supervisor for a class of PDESs. A formal proof of the necessity and sufficiency of the conditions and an algorithm for the calculation of the supervisor, if it exists, are presented in Pantelic et al. (2009). Analogous to a problem in classical supervisory control theory, it can happen that, given a plant to be controlled and a probabilistic specification language, no probabilistic supervisor exists such that the plant under control generates the prespecified probabilistic language. In this case, when the exact solution is not achievable, a designer tries to find a supervisor such that the plant generates the behavior closest to the desired behaviour (Pantelic and Lawford (2009)). Therefore, the supervisor is synthesized by minimizing the distance between the achievable probabilistic behavior of the plant under control and the probabilistic behaviour of the requirement. The distance is measured by a pseudometric on the states of probabilistic transition systems which was introduced in Deng et al. (2006). The pseudometric is characterized as the greatest fixed point of a function. For reactive systems, the work of Deng et al. (2006) is closely related to Desharnais et al. (2002), Desharnais et al. (2004), van Breugel and Worrell (2001a), van Breugel and Worrell (2001b, 2005).

The pseudometric of Deng et al. (2006) is slightly modified in Pantelic and Lawford (2009). The pseudometric intuitively matches a notion of the distance between PDESs, accounting for all differences between corresponding transition probabilities. Furthermore, as it is suggested for a large class of systems, the pseudometric allows for an extension of the work to e.g., nondeterministic systems. Also, there is a simple algorithm to compute distances in this pseudometric for our generative, deterministic model (Pantelic and Lawford (2009)).

This paper further motivates the choice of the pseudometric in the solution of the control problem of Pantelic and Lawford (2009) by characterizing the pseudometric using a real-valued logic along the lines of Desharnais et al. (2002). However, the logic itself is different than that of Desharnais et al. (2002) as our models are generative. Also, the main part of the characterization proof is, to the best of our knowledge, novel. The idea of logical characterization is that the distance between two systems is measured by a logical formula that distinguishes between the systems the most. Further, in this paper, this logical characterization is used as useful information on how the probabilistic traces of the systems are related.

In the control theory of PDESs, Chattopadhyay and Ray (2008) introduce a pseudometric in a symbolic pattern recognition application to measure the distance between the original model and the one with prespecified structure, where the latter has the same long term distribution over the states as the original one. In this paper, the problem of a similar probabilistic model transformation (probabilistic model fitting) is discussed in our setting. A probabilistic generator is approximated by another one with a prespecified structure such that the distance between the two is as small as possible in the pseudometric of Pantelic and Lawford (2009). The significance of the fitting, especially with respect to the control algorithm of Pantelic and Lawford (2009) is then discussed.

In Section 2, the probabilistic control of PDESs is reviewed. Section 3 presents the logical characterization of the pseudometric. The trace characterization that stems from the logical one is presented in Section 4. The probabilistic model fitting problem, its solution, and its applications in control theory are introduced in Section 5. Section 6 concludes with avenues for future work.

In this paper, the proofs are omitted due to space restrictions. They can be found in Pantelic (2009).

2. PRELIMINARIES

In this section, PDESs modeled as generators of probabilistic languages are presented. Then, the problem of optimal supervisory control of PDESs, and the pseudometric used in the solution of the problem are introduced (for more details, see Pantelic and Lawford (2009)).

2.1 Modeling PDES

The probabilistic DES (PDES) can be modeled as a probabilistic generator $G = (Q, \Sigma, \delta, q_0, p)$, where Q is the nonempty finite set of states, Σ is a finite alphabet whose elements we will refer to as event labels, $\delta : Q \times \Sigma \to Q$ is the (partial) transition function, $q_0 \in Q$ is the initial state, and $p : Q \times \Sigma \to [0, 1]$ is the statewise event probability distribution (Lawford and Wonham (1993)). The results to be presented are for prefix closed probabilistic specification languages; hence the lack of marking states in the definition of a probabilistic generator. The transition function is traditionally extended by induction on the length of strings to $\delta : Q \times \Sigma^* \to Q$ in a natural way. For a state q, and a string s, the expression $\delta(q, s)!$ will denote that δ is defined for the string s in the state q.

The probability that the event $\sigma \in \Sigma$ is going to occur at the state $q \in Q$ is $p(q, \sigma)$. For the generator G to be well-defined, (i) $p(q, \sigma) = 0$ should hold if and only if $\delta(q, \sigma)$ is undefined, and (ii) $\forall q \; \sum_{\sigma\in\Sigma} p(q, \sigma) \leq 1$. The probabilistic generator G is nonterminating if, for every reachable state $q \in Q$, $\sum_{\sigma\in\Sigma} p(q, \sigma) = 1$. Conversely, G is terminating if there is at least one reachable state $q \in Q$ such that $\sum_{\sigma\in\Sigma} p(q, \sigma) < 1$. The probability that the system terminates at state q is $1 - \sum_{\sigma\in\Sigma} p(q, \sigma)$. Throughout the sequel, we will consider nonterminating generators. A terminating plant can easily be transformed into a nonterminating one using the technique described in Lawford and Wonham (1993).

The language $L(G)$ generated by a probabilistic DES generator $G = (Q, \Sigma, \delta, q_0, p)$ is $L(G) = \{s \in \Sigma^* \,|\, \delta(q_0, s)!\}$. The probabilistic language generated by G is defined as:

$$L_p(G)(\epsilon) = 1$$

$$L_p(G)(s\sigma) = \begin{cases} L_p(G)(s) \cdot p(\delta(q_0, s), \sigma), & \text{if } \delta(q_0, s)! \\ 0, & \text{otherwise.} \end{cases}$$

Informally, $L_p(G)(s)$ is the probability that the string s is executed in G. Also, $L_p(G)(s) > 0$ iff $s \in L(G)$.

For each state $q \in Q$, function $\rho_q : \Sigma \times Q \to [0, 1]$ is defined such that for any $q' \in Q$, $\sigma \in \Sigma$, we have $\rho_q(\sigma, q') = p(q, \sigma)$ if $q' = \delta(q, \sigma)$, and 0 otherwise. The function ρ_q is a probability distribution on the set $\Sigma \times Q$. Also, for a state q, *the set of possible events* is defined to be $Pos(q) := \{\sigma \in \Sigma | \delta(q, \sigma)!\}$.

2.2 Control Problem

As usual, the event set Σ is partitioned into disjoint sets Σ_c and Σ_u, the sets of controllable and uncontrollable events, respectively. Deterministic supervisors for DES are generalized to *probabilistic supervisors*. Instead of deterministically enabling or disabling controllable events, probabilistic supervisors enable them with certain probabilities.

Optimal Probabilistic Supervisory Control Problem (OPSCP): Let $G_1 = (Q, \Sigma, \delta_1, q_0, p_1)$ be a plant PDES, and let $G_2 = (R, \Sigma, \delta_2, r_0, p_2)$ be a requirements specification PDES. The plant G_1 under the control of the supervisor V_p will be denoted V_p/G_1. If there is no probabilistic supervisor V_p such that $L_p(V_p/G_1) = L_p(G_2)$ (i.e., the conditions of Lawford and Wonham (1993) fail), find V_p such that

(1) $L(V_p/G_1) \subseteq L(G_2)$ and supervisor V_p is maximally permissive in the nonprobabilistic sense (i.e., $L(V_p/G_1)$ is the supremal controllable sublanguage of $L(G_2)$ with the respect to G_1).
(2) The probabilistic behaviour of the controlled plant is "as close as possible" to the probabilistic behaviour of the requirements specification, now restricted to supremal controllable sublanguage of $L(G_2)$ with the respect to G_1.

The solution to the problem is given in Pantelic and Lawford (2009). The supremal controllable sublanguage of $L(G_2)$ with the respect to G_1 is generated as the maximally permissive behaviour of the controlled plant. Then, the distance (in the pseudometric to be described in Section 2.3) between the generators representing the controlled plant, and the probabilistic requirement (now restricted to the supremal controllable sublanguage) is minimized such that the probabilistic controllability conditions of Lawford and Wonham (1993) are satisfied. An iterative algorithm is given to approximate the probabilities of the controlled plant. As the underlying graphs of the two generators are isomorphic, the distance is minimized by minimizing the distance between each pair of isomorphic states. The algorithm is linear in the number of states of both the plant and the requirement generator, and polynomial in the number of events.

2.3 The pseudometric

Probabilistic bisimulation as commonly used to define an equivalence relation between probabilistic systems is not a robust relation: two states of probabilistic systems are bisimilar if and only if they have the same transitions with exactly the same probabilities to states in the same equivalence classes. As a more flexible way to compare probabilistic systems, a notion of *pseudometric* is introduced. A *pseudometric on a set of states* Q is a function $d : Q \times Q \to \mathbb{R}$ that defines a distance between two elements of Q, and satisfies the following conditions: $d(x, y) \geq 0$, $d(x, x) = 0$, $d(x, y) = d(y, x)$, and $d(x, z) \leq d(x, y) + d(y, z)$, for any $x, y, z \in Q$. If all distances are not greater than 1, the pseudometric is *1-bounded*. In the sequel, the terms metric and pseudometric will be used interchangeably.

The work of Deng et al. (2006) introduces a pseudometric on states for a large class of probabilistic automata, including reactive and generative probabilistic automata. The pseudometric is based on the Kantorovich metric on distributions. Two states are at distance 0 in this metric if and only if they are probabilistic bisimilar. Here, the metric is presented only for probabilistic generators.

Let $G = (Q, \Sigma, \delta, q_0, p)$ be a nonterminating PDES, where $Q = \{q_0, q_1, \ldots q_{N-1}\}$. First, in Desharnais et al. (2002) and Deng et al. (2006), the class $\mathcal{M}$ of 1-bounded pseudometrics on states is defined with the ordering ($d_1, d_2 \in \mathcal{M}$)

$$d_1 \preceq d_2 \text{ if } \forall q_q, q_r \; d_1(q_q, q_r) \geq d_2(q_q, q_r). \tag{1}$$

Further, it is proved that $(\mathcal{M}, \preceq)$ is a complete lattice.

Next, let $d \in \mathcal{M}$, and let the constant $e \in (0, 1]$ be a *discount factor* that determines the degree to which the difference in the probabilities of farther transitions is discounted: the smaller the value of e, the greater the discount on future transitions. Let $q_q, q_r \in Q$, and let ρ_{q_q} and ρ_{q_r} be the distributions on $\Sigma \times Q$ induced by the states q_q and q_r, respectively. Next, let $i(q_q, \sigma) = i$ such that $q_i = \delta(q_q, \sigma)$ if $\delta(q_q, \sigma)!$, and $i(q_q, \sigma) = 0$, otherwise. Similarly, $j(q_r, \sigma) = j$ such that $q_j = \delta(q_r, \sigma)$ if $\delta(q_r, \sigma)!$, and $j(q_r, \sigma) = 0$, otherwise. For readability purposes, we will write i instead of $i(q_q, \sigma)$, and j instead of $j(q_r, \sigma)$. Further, we will write $\rho_{\sigma,i}$ instead of $\rho_{q_q}(\sigma, q_i)$, and, similarly, $\rho'_{\sigma,j}$ instead of $\rho_{q_r}(\sigma, q_j)$. Then, the pseudometric on states d_{fp} is given as the greatest fixed-point of the function $\mathcal{D}$ on $\mathcal{M}$, that, in the special case of probabilistic generators, can be shown to be (see Pantelic and Lawford (2009)):

$$\begin{aligned}
&\mathcal{D}(d)(q_q, q_r) \\
&\quad = \sum_{\sigma \in \Sigma} max(\rho_{\sigma,i} - \rho'_{\sigma,j} + e\rho'_{\sigma,j} d(q_i, q_j), e\rho_{\sigma,i} d(q_i, q_j)) \\
&\quad = \sum_{\sigma \in \{\sigma \in \Sigma | \rho_{\sigma,i} \geq \rho'_{\sigma,j}\}} \left(\rho_{\sigma,i} - \rho'_{\sigma,j} + e\rho'_{\sigma,j} d(q_i, q_j)\right) \\
&\quad + \sum_{\sigma \in \{\sigma \in \Sigma | \rho_{\sigma,i} < \rho'_{\sigma,j}\}} e\rho_{\sigma,i} d(q_i, q_j)
\end{aligned} \tag{2}$$

We arbitrarily choose $i(q_q, \sigma)$ to be 0 (similarly for $j(q_r, \sigma)$) when $\delta(q_q, \sigma)$ is not defined although we could have chosen any other $i \in \{1, \ldots, N-1\}$. This is because when $\delta(q_q, \sigma)!$ does not hold, then $\rho_{\sigma, i(q_q, \sigma)} = 0$ for any $i(q_q, \sigma) \in \{0, \ldots, N-1\}$.

The distances between the states in d_{fp} are larger by the factor $1/e$ than the distances in metric defined in Deng et al. (2006). This has been done so that the distances are in the range $[0, 1]$, instead of $[0, e]$.

The pseudometric d_{fp} is defined on the states of a single PDES. The distance between two PDESs (with disjoint sets of states) is the distance between their initial states in a new PDES that represents the union of the PDESs defined in a natural way (Pantelic and Lawford (2009)). In the sequel, the union will not be formalized as it does not change the distance between the states.

Remark 1. According to Tarski's fixed point theorem, the greatest fixed point of function $\mathcal{D}$ can be reached through an iterative process that starts from the greatest element. As the number of transitions from a state of a probabilistic generator is finite, the greatest fixed point of the function $\mathcal{D}$ is reached after at most ω iterations (Deng et al. (2006), Desharnais et al. (2002)) (equivalently, the closure ordinal of $\mathcal{D}$ is ω). Therefore, the metric d_{fp} can be reached through the following iterative process.

Definition 1. The distance function d^0_{fp} is defined as $d^0_{fp} = 0$, and the distance function $d^{n+1}_{fp}, n \in \mathbb{N}$, is given as $d^{n+1}_{fp} = \mathcal{D}(d^n_{fp})$, where $\mathcal{D}$ is given in (2).

3. LOGICAL CHARACTERIZATION

The aforementioned metric has been given a fixed point characterization. This section presents a logical characterization of the same metric. The logic used is real-valued so that it can handle probabilities. If the systems are probabilistic bisimilar, there should not be a formula that distinguishes between the systems. The distance between the states in the metric is achieved through a formula that distinguishes between the systems the most.

As before, let $G = (Q, \Sigma, \delta, q_0, p)$ be a nonterminating generator, where $Q = \{q_0, q_1, \ldots q_{N-1}\}$.

Definition 2. Given an alphabet Σ, the logic $\mathcal{L}$ is defined as follows:

$$\phi ::= \mathbf{1} \mid \langle\sigma\rangle\phi \mid \bigvee_{\sigma \in \Theta} \langle\sigma\rangle\phi \mid 1 - \phi \mid \phi \ominus p,$$

where p is a rational number in $[0, 1]$, $\sigma \in \Sigma$, and $\Theta \subseteq \Sigma$.

The formula ϕ evaluated at a state $q \in Q$, $\phi(q)$, is a measure of how much ϕ is satisfied at the state. The semantics of the logic $\mathcal{L}$ is given next.

Definition 3. Let $q \in Q$, and ρ_q be the probability distribution on $\Sigma \times Q$ induced by state q. Let $\phi \in \mathcal{L}$, and $\psi : \Sigma \to \mathcal{L}$. The notation ψ_σ will be used for $\psi(\sigma)$, $\sigma \in \Sigma$. Then:

$$\begin{aligned}
\mathbf{1}(q) &= 1 \\
\langle\sigma\rangle\phi(q) &= e\rho_q(\sigma, q_{i(q,\sigma)})\phi(q_{i(q,\sigma)}) \\
\bigvee_{\sigma \in \Theta} \langle\sigma\rangle\psi_\sigma(q) &= \sum_{\sigma \in \Theta} e\rho_q(\sigma, q_{i(q,\sigma)})\psi_\sigma(q_{i(q,\sigma)}) \\
(1 - \phi)(q) &= 1 - \phi(q) \\
(\phi \ominus p)(q) &= max(\phi(q) - p, 0)
\end{aligned}$$

where $\sigma \in \Sigma$, and, as before, $i(q, \sigma) = i$ such that $q_i = \delta(q, \sigma)$ if $\delta(q, \sigma)!$, and $i(q, \sigma) = 0$, otherwise.

The presented logic represents a probabilistic modification of Hennessy-Milner logic (Hennessy and Milner (1985)). The formula $\mathbf{1}$ corresponds to the constant *true*, $\langle\sigma\rangle\phi$ is the next operator, $1 - \phi$ corresponds to negation, and $\phi \ominus p$ provides for the testing of the value of ϕ (Desharnais et al. (2002)). The logic supports only a specific disjunction of form $\bigvee\langle\sigma\rangle\phi$; extending it to $\bigvee\phi$ would require a more complicated formalization not necessary for the main result to be presented.

The metric d_L is defined next. The distance between two states are measured by a formula that differentiates them the most.

Definition 4. For every $q_q, q_r \in Q$, the metric d_L is defined as:

$$d_L(q_q, q_r) = \sup_{\phi \in \mathcal{L}} \{|\phi(q_q) - \phi(q_r)|\}.$$

In this logical setting, the smaller the factor e is, the more discounted the difference is for complex formulae.

An example is given in Figure 1. The states q_0 and r_0 are at the distance $0.25e + 0.75e^2$ in the metric d_L, witnessed by the formula $\phi = \bigvee_{\sigma \in \{\alpha,\beta\}} \langle\sigma\rangle\phi_\sigma$, where $\phi_\alpha = 1 - \langle\gamma\rangle\mathbf{1}$, and $\phi_\beta = \langle\tau\rangle\mathbf{1}$. Further, states q_1 and r_1 (also, q_1 and r_2) are at the distance e as witnessed by the formula $\phi = \langle\tau\rangle\mathbf{1}$.

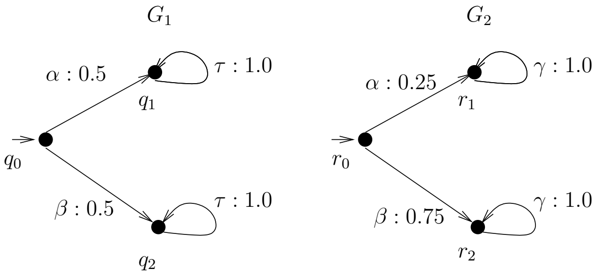

Fig. 1. Example

The goal is to show that the metric d_{fp} is equal to the metric d_L up to constant e.

Lemma 1. Let $q_q, q_r \in Q$. For a function $\psi : \Sigma \to \mathcal{L}$, the shorthand notation ψ_σ will be used for $\psi(\sigma)$. Then:

$$d_L(q_q, q_r) = \sup_{\psi_\sigma \in \mathcal{L}} \left\{ \left| \bigvee_{\sigma\in\Sigma} \langle\sigma\rangle\psi_\sigma(q_q) - \bigvee_{\sigma\in\Sigma} \langle\sigma\rangle\psi_\sigma(q_r) \right| \right\}.$$

The main result relating the two metrics is presented next. It states that d_L and d_{fp} are equal up to constant e.

Theorem 1. $d_L = ed_{fp}$

4. FROM LOGIC TO TRACES

First, $L_p(G)(s)$ is modified to define the *discounted probability of a string* s in G, denoted $P_d(G)(s)$.

Definition 5. Let $P_d(G) : L(G) \to [0, 1]$ be defined as:

$$P_d(G)(\epsilon) = 1$$

$$P_d(G)(s\sigma) = \begin{cases} e \cdot P_d(G)(s) \cdot p(\delta(q_0, s), \sigma), & \text{if } \delta(q_0, s)! \\ 0, & \text{otherwise} \end{cases}$$

where $s \in L(G)$, $\sigma \in \Sigma$. Then, $P_d(G)(s)$ is the *discounted probability of a string* s in G.

Informally, the discounted probability of a string is the probability of occurrence of a string discounted by factor e for every event in the string, $P_d(G)(s) = e^{|s|}L_p(G)$.

Let G_1 and G_2 be two probabilistic generators. An important result states that there is not a string whose discounted probabilities differ more than the distance d_L between corresponding generators.

Theorem 2.

$$d_L(G_1, G_2) \geq \sup_{s\in\Sigma^*} \{|P_d(G_1)(s) - P_d(G_2)(s)|\} \qquad (3)$$

Further, it can be shown that distance in the metric d_L between the two systems is also greater than the difference in discounted probabilities of a set of strings such that none of strings is a substring of another. Let $\Gamma \subseteq \Sigma^*$, such that no string in Γ is a prefix of another string in Γ. Then:

Theorem 3.

$$d_L(G_1, G_2) \geq \sup_{\Gamma\subseteq\Sigma^*} \left\{ \left| \sum_{s\in\Gamma} P_d(G_1)(s) - \sum_{s\in\Gamma} P_d(G_2)(s) \right| \right\}$$

Similarly, the correspondence between the discounted probability of strings and formulae in $\mathcal{L}$ can be made for the remaining formulae of Definition 3. Therefore, the metric measures not only the difference in probabilities of strings in two languages (discounted for their length), but also the difference in discounted probabilities of a certain set of strings, or some more complicated properties of strings, e.g., whether the discounted probability of a string is greater than a prespecified value.

5. PROBABILISTIC MODEL FITTING

First, the problem of probabilistic model fitting is defined, and a solution for the fitting is suggested. Next, its significance in control theory, and especially, with regards to the algorithm of Pantelic and Lawford (2009), is discussed.

5.1 Probabilistic Model Fitting: Problem and Solution

For a probabilistic generator $G = (Q, \Sigma, \delta, q_0, p)$, the (nonprobabilistic) discrete event system (DES) that underlies G will be denoted $G' = (Q, \Sigma, \delta, q_0)$. The synchronous product of two (nonprobabilistic) discrete event systems (DESs) that underlie two PDESs is defined as follows. Let $G_1 = (Q_1, \Sigma, \delta_1, q_{01}, p_1)$ and $G_2 = (Q_2, \Sigma, \delta_2, r_0, p_2)$ be probabilistic generators.

Definition 6. The synchronous product of $G'_1 = (Q_1, \Sigma, \delta_1, q_{01})$ and $G'_2 = (Q_2, \Sigma, \delta_2, r_0)$, denoted $G'_1 \parallel G'_2$, is the reachable sub-DES of DES $G_a = (Q_a, \Sigma, \delta, q_0)$, where $Q_a = Q_1 \times Q_2$, $q_0 = (q_{01}, r_0)$, and, for any $\sigma \in \Sigma$, $q_i \in Q_i$, $i = 1, 2$, it holds that $\delta((q_1, q_2), \sigma) = (\delta_1(q_1, \sigma), \delta_2(q_2, \sigma))$ whenever $\delta_1(q_1, \sigma)!$, and $\delta_2(q_2, \sigma)!$.

Note that no minimization is done in the construction of the synchronous product as defined above.

Probabilistic Model Fitting Problem: Let $G_1 = (Q_1, \Sigma, \delta_1, q_{01}, p_1)$ be a probabilistic generator. Given a nonprobabilistic generator $G'_2 = (Q_2, \Sigma, \delta_2, r_0)$ such that $G'_1 \parallel G'_2$ is isomorphic to G'_2, find the statewise event probability distribution p_2 such that probabilistic generator $G_2 = (Q_2, \Sigma, \delta_2, r_0, p_2)$ is as close as possible to G_1 in metric d_{fp}.

The idea of solving the problem is as follows. The generator G_1 is to be modified to make G'_2 isomorphic (identical up to renaming of states) to a subautomaton of G'_1, while the probabilistic language of G_1 is preserved. Then, the distance between G_1 and G_2 is minimized by minimizing the distance between the modified G_1, and G_2. This is allowed as the two distances are the same, since G_1 and its modified version are probabilistic bisimilar:

Theorem 4. Let G_1 and G_2 be two probabilistic generators. Then, if $L_p(G_1) = L_p(G_2)$, then $d_{fp}(G_1, G_2) = 0$.

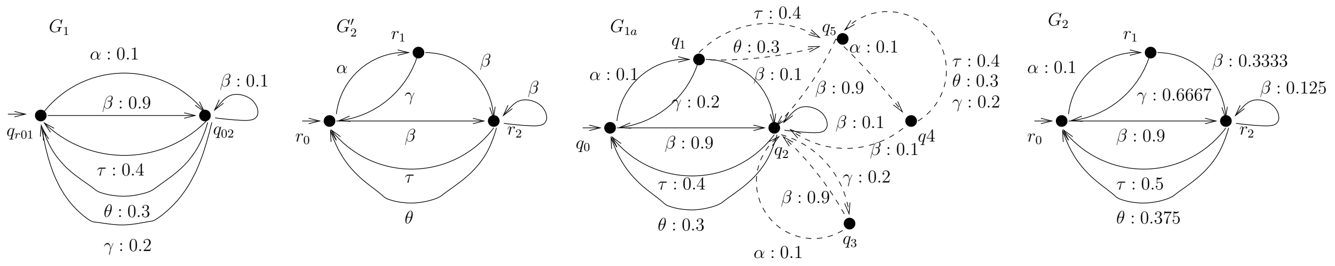

Fig. 2. Model fitting: an example

Next, as previously stated, we seek to represent $L_p(G_1)$ with an automaton G_{1a} such that G'_2 is isomorphic to a subautomaton of G'_{1a}. Figure 2 illustrates an example. The part of G_{1a} drawn by a solid line corresponds to the subautomaton of G'_{1a} isomorphic to G'_2. In general, the automaton G_{1a} will represent a non-minimal realization of $L_p(G_1)$ (in the sense that it might have more states than G_1, but $L_p(G_1) = L_p(G_{1a})$). Generator G_{1a} can be constructed in the following manner.

(1) Self-loop each state of G'_2 with events not possible from that state. Formally, $G'_{2a} = (Q_2, \Sigma, \delta_{2a}, r_0)$, where, for $q \in Q_2$, $\sigma \in \Sigma$:
$$\delta_{2a}(q,\sigma) = \begin{cases} \delta_2(q,\sigma), & \text{if } \delta_2(q,\sigma)! \\ q, & \text{otherwise.} \end{cases}$$
(2) Next, let $G'_{1a} = (Q_{1a}, \Sigma, \delta_{1a}, q_0) = G'_1 \parallel G'_{2a}$.
(3) The probabilistic version of G'_{1a} is $G_{1a} = (Q_{1a}, \Sigma, \delta_{1a}, q_0, p_{1a})$, such that, for all $q \in Q_{1a}$, $\sigma \in \Sigma$ $p_{1a}(q,\sigma) = p_1(r,\sigma)$, where $r = \delta_1(q_{01}, s)$ for any $s \in L(G_{1a})$ such that $q = \delta_{1a}(q_0, s)$.

Lemma 2. $L_p(G_1) = L_p(G_{1a})$.

Now, let $f : Q_2 \to Q_{1a}$ be an embedding (a monomorphism) of G'_2 into G'_{1a}, i.e.:

(1) $f(r_0) = q_0$,
(2) $\forall q \in Q_2 \forall \sigma \in Pos(q) f(\delta_2(q,\sigma)) = \delta_{1a}(f(q),\sigma)$.

The function f always exists and is unique. This fact follows from the construction of G_{1a} and the determinism of generators.

Without loss of generality, it is assumed that, $Q_{1a} = \{q_0, \ldots, q_{M-1}\}$, $Q_2 = \{r_0, \ldots, r_{N-1}\}$, and $M \geq N > 0$, $d \in \mathcal{M}$, $q \in Q_2$. Next, $i(f(q),\sigma) = i$ such that $q_i = \delta_{1a}(f(q),\sigma)$ if $\delta_{1a}(f(q),\sigma)!$, and $i(f(q),\sigma) = 0$, otherwise. Let $j(q,\sigma) = j$ such that $r_j = \delta_2(q,\sigma)$ if $\delta_2(q,\sigma)!$, and $j(q,\sigma) = 0$, otherwise. For readability purposes, we will write i instead of $i(f(q),\sigma)$, and j instead of $j(q,\sigma)$. The distance between G_{1a} and G_2 is $d_{fp}(q_0, r_0)$. Also, $f(r_0) = q_0$, and

$$\begin{aligned} &\mathcal{D}(d)(f(q),q) \\ &= \sum_{\sigma\in\Sigma} max(\rho_{\sigma,i} - \rho'_{\sigma,j} + e\rho'_{\sigma,j} d(q_i,r_j), e\rho_{\sigma,i} d(q_i,r_j)) \\ &= \sum_{\sigma \in Pos(f(q)) \setminus Pos(q)} \rho_{\sigma,i} \\ &+ \sum_{\sigma\in Pos(q)} max(\rho_{\sigma,i} - \rho'_{\sigma,j} + e\rho'_{\sigma,j} d(f(r_j),r_j), \\ &\qquad e\rho_{\sigma,i} d(f(q_j),q_j)) \qquad (4) \end{aligned}$$
(since $f(r_j) = q_i$, by the definition of f)

where $\rho_{f(q)}$ and ρ_q are the distributions on $\Sigma \times Q$ induced by the states $f(q)$ and q, respectively, and $\rho_{\sigma,i}$ is written instead of $\rho_{f(q)}(\sigma, q_i)$, and, similarly, $\rho'_{\sigma,j}$ instead of $\rho_q(\sigma, r_j)$.

Remark 2. Based on (4), it can be concluded that, for $q \in Q_2$, the distance between state $f(q) \in Q_{1a}$ and state $q \in Q_2$ depends only on distances between $f(t)$ and t, $t \in Q_2$. In Figure 2, the distance between G_{1a} and G_2 depends only on distances between states of pairs (q_0, r_0), (q_1, r_1), and (q_2, r_2); states q_3, q_4, q_5 are irrelevant.

Therefore, in order to calculate the distance between G_{1a} and G_2, only the distances $d_{fp}(f(q),q)$, $q \in Q_2$, are of interest. Hence, the distance between G_{1a} and G_2, for a fixed p_2, can be found by at most ω iterations given in Definition 1, where the domain of d^n_{fp} is restricted to $Q_{1a} \times Q_2$ and only distances between $f(q) \in Q_{1a}$ and $q \in Q_2$ are defined.

This reasoning leads to the solution of the probabilistic model fitting problem as presented next.

Theorem 5. Let $G_1 = (Q_1, \Sigma, \delta_1, q_{01}, p_1)$ be a probabilistic generator. For given $G'_2 = (Q_2, \Sigma, \delta_2, r_0)$ (such that $G'_1 \parallel G'_2$ is isomorphic to G'_2), the statewise event probability distribution p_2 such that $G_2 = (Q_2, \Sigma, \delta_2, r_0, p_2)$ is as close as possible to G_1 in the metric d_{fp} should satisfy, for all $r \in Q_2$, $\sigma \in \Sigma$:
$$p_2(r,\sigma) \geq p_1(q,\sigma) \qquad (5)$$
where $q = \delta_1(q_{01}, s)$ for any $s \in L(G_2)$ such that $r = \delta_2(r_0, s)$.

Therefore, the new model is not unique: as long as the probabilities of the events possible in the new model do not decrease, the new model is as close as possible to the original one. For the example from Figure 2, one of the possible solutions is represented by the rightmost generator of the figure. In another possible solution, the probabilities of occurrence of β and γ at the state r_1 would be 0.2 and 0.8, respectively. Therefore, the fitting can be performed by any redistribution of the probabilities of events that are not possible anymore over the possible ones. Hence, model fitting can accommodate some further requirements on p_2.

5.2 Applications of model fitting

Other than the obvious use of the presented fitting to simplify and reduce the state space of probabilistic systems, the fitting has much more significant control implications.

As mentioned before, it is possible to choose probabilities of events in the new system to a certain extent: as long as they are greater or equal to the original ones. However, some of the further requirements on p_2 cannot be accommodated by Theorem 5 (e.g., an obvious one would be that the probability of an event still possible in the new system should be smaller than in the original system). If the restrictions are given on probabilities of events, statewise, a straightforward modification of the algorithm of Pantelic and Lawford (2009) for $e \in (0,1)$ would suffice.

Further, in the solution of the optimal supervisory problem of Pantelic and Lawford (2009), in order for the first, maximal permissiveness requirement as presented in Section 2.2 to be satisfied, the supremal controllable sublanguage of $L(G_2)$ with the respect to G_1 is generated. Then, the distance between the controlled plant, and the probabilistic requirement now restricted to the supremal controllable sublanguage, with normalized probabilities, is minimized. Intuitively, after satisfying the nonprobabilistic requirement, and before the probabilistic part is handled, it makes sense for a designer to modify the original requirement so that its nonprobabilistic behaviour matches the one achievable. Then, the probabilities are revised accordingly: probabilities of the events inadmissible for not satisfying the nonprobabilistic requirement are redistributed over the admissible ones. In Pantelic and Lawford (2009), the redistribution is such that the probability of an event in the new system is proportional to its original probability. Theorem 5 proves that this normalization is justified in a strict mathematical sense, as the new model that is normalized is as close as possible to the original one in the metric d_{fp}. However, a revised specification is going to be at a minimal distance from the original one as long as the probabilities of remaining events are greater than or equal to the original ones: a designer has a freedom to choose how to redistribute the probabilities over the events that are still possible.

Further, the transformation of G_1 into G_{1a} presented here can be used in a modification of the algorithm of Pantelic and Lawford (2009) to handle the control problem presented in Section 2.2 with requirement 2) changed so that the distance between the controlled plant and the unmodified requirement is minimized. More precisely, the probabilistic language $L_p(G_2)$ of the requirements specification G_2 can be exactly represented by a probabilistic generator G_{2a} with nonprobabilistic automaton G'_{2a} that has a subautomaton that is isomorphic to the automaton G_s representing the supremal controllable sublanguage of the controlled plant (see Section 2.2). Then, using the reasoning of Remark 2, the distance between the requirement and the controlled plant depends only on the distances between isomorphic states of the subautomaton of G'_{2a} and G_s. Hence, the algorithm of Pantelic and Lawford (2009) can be modified to minimize the distance between the two systems under the probabilistic controllability conditions of Lawford and Wonham (1993).

6. CONCLUSIONS

A metric was used in a control algorithm to measure distance between two probabilistic generators in Pantelic and Lawford (2009). This paper gives a logical characterization of the metric that offers better insight into the core of the metric from both logic and language standpoints. Further, the metric is used in the probabilistic model fitting problem: a probabilistic language is represented using a specified automaton structure such that that the new representation is as close as possible to the original one. Probabilistic model fitting has significant implications with respect to the work of Pantelic and Lawford (2009).

REFERENCES

Chattopadhyay, I. and Ray, A. (2008). Structural transformations of probabilistic finite state machines. *International Journal of Control*, 81(5), 820–835.

Deng, Y., Chothia, T., Palamidessi, C., and Pang, J. (2006). Metrics for action-labelled quantitative transition systems. *Electronic Notes in Theoretical Computer Science*, 153(2), 79–96. Also appeared in *Proceedings of the 3rd Workshop on Quantitative Aspects of Programming Languages*.

Desharnais, J., Gupta, V., Jagadeesan, R., and Panangaden, P. (2004). Metrics for labelled Markov processes. *Theoretical Computer Science*, 318(3), 323–354.

Desharnais, J., Jagadeesan, R., Gupta, V., and Panangaden, P. (2002). The metric analogue of weak bisimulation for probabilistic processes. In *LICS '02: Proceedings of the 17th Annual IEEE Symposium on Logic in Computer Science*, 413–422. IEEE Computer Society, Washington, DC, USA.

Garg, V., Kumar, R., and Marcus, S. (1999). Probabilistic language formalism for stochastic discrete event systems. *IEEE Transactions on Automatic Control*, 44(2), 280–293.

Hennessy, M. and Milner, R. (1985). Algebraic laws for nondeterminism and concurrency. *Journal of the ACM*, 32(1), 137–161.

Lawford, M. and Wonham, W. (1993). Supervisory control of probabilistic discrete event systems. In *Proceedings of the 36th IEEE Midwest Symposium on Circuits and Systems*, volume 1, 327–331. IEEE.

Pantelic, V. (2009). *Probabilistic supervisory control of probabilistic discrete event systems*. Ph.D. thesis. In preparation.

Pantelic, V. and Lawford, M. (2009). Towards optimal supervisory control of probabilistic discrete event systems. In *DCDS 2009: Proceedings of 2nd IFAC Workshop on Dependable Control of Discrete Systems*, 85–90. Bari, Italy.

Pantelic, V., Postma, S., and Lawford, M. (2009). Probabilistic supervisory control of probabilistic discrete event systems. *IEEE Transactions on Automatic Control*, 54(8), 2013–2018.

van Breugel, F. and Worrell, J. (2001a). An algorithm for quantitative verification of probabilistic transition systems. In K.G. Larsen and M. Nielsen (eds.), *CONCUR*, volume 2154 of *Lecture Notes in Computer Science*, 336–350. Springer.

van Breugel, F. and Worrell, J. (2001b). Towards quantitative verification of probabilistic transition systems. In F. Orejas, P.G. Spirakis, and J. van Leeuwen (eds.), *ICALP*, volume 2076 of *Lecture Notes in Computer Science*, 421–432. Springer.

van Breugel, F. and Worrell, J. (2005). A behavioural pseudometric for probabilistic transition systems. *Theoretical Computer Science*, 331(1), 115–142.