Supervisory Control of Probabilistic Discrete Event Systems

Vera Pantelic, Steven Postma, and Mark Lawford

Abstract

This paper considers supervisory control of probabilistic discrete event systems (PDES). PDESs are modeled as generators of probabilistic languages. The supervisory control problem considered is to find, if possible, a supervisor under whose control the behaviour of a plant is identical to a given probabilistic specification. The probabilistic supervisors we employ are a generalization of the deterministic ones previously employed in the literature. At any state, the supervisor enables/disables events with certain probabilities. Necessary and sufficient conditions for the existence of such a supervisor, and an algorithm for its computation are presented.

Software Quality Research Laboratory

SQRL Report No. 21

Authors are with the SQRL, Department of Computing and Sofware, Faculty of Engineering, McMaster University, 1280 Main Street West, Hamilton ON, L8S 4K1 (E-mail: pantelv at mcmaster dot ca, steven.m.postma at gmail dot com, lawford at mcmaster dot ca).

I. Introduction

THE supervisory control theory of discrete event systems was developed in the seminal work of Ramadge and Wonham [1]. A supervisor (controller) controls a plant by enabling/disabling controllable events based on the observation of the previous behaviour of the plant. The supervisory control problem considered is to supervise the plant so it generates a given specification language. In order to model stochastic behaviour of the plant, many models of stochastic behavior of discrete event systems have been proposed; for example, Markov chains [2], Rabin's probabilistic automata [3], stochastic Petri nets [4]. We follow the theory of probabilistic DES that was developed in [5], [6] using an algebraic approach. A stochastic discrete event systems is represented as an automaton with transitions labeled with probabilities. The probabilities of all the events in a certain state add up to at most one. With Rabin's probabilistic automata [3], on the other hand, the sum of the probabilities of an event at a state is one. Also, unlike the Markov chains [2], the emphasis of the approach of [5], [6] is on event traces rather than state traces.

The control of different models of stochastic discrete event systems has been investigated in [7], [8], etc. Rabin's probabilistic automata are used in [7] as the underlying model, while [8] uses Markov chains. In [9], the model of [5], [6] is adopted and deterministic supervisors for DES are generalized to *probabilistic supervisors*. The control method of *random disablement* is used: after observing a string s, the probabilistic supervisor enables an event σ with a certain probability. The necessary and sufficient conditions for the existence of a supervisor for a class of nonterminating DESs are given and reduce to checking whether certain linear equalities and inequalities hold. Further, the paper describes a way to transform a terminating PDES into the nonterminating PDES, so that the presented results can be applied.

The work of [10] builds upon [9] by giving a formal proof of the necessity and sufficiency of the conditions presented in [9]. It also gives an algorithm for the calculation of the supervisor. A supervisory control framework for stochastic discrete event systems that was developed in [11] represents a special case of the control introduced in [9]. Deterministic control is considered where controllable events are disabled dynamically, so that the probabilities of their execution become zero, and the probabilities of the occurrence of other events proportionally increase. The control objective considered is to construct a supervisor such that the controlled plant does not execute specified illegal traces, and the probabilities of occurrences of the events in the system are greater than or equal to specified values. The paper gives a necessary and sufficient condition for the existence of a supervisor. Further, in [12], a technique to compute a maximally permissive supervisor on-line is given. In [13], [14], the same model of [5], [6] is used. The requirements specification is given by weights assigned to states and the control goal is, roughly speaking, to reach the states with more weight (more desired states) more often. A deterministic control is synthesized for a given requirements specification so that a measure based on the specification and probabilities of the plant is optimized.

This paper merges the works of [9] and [10]. The notation used in [10] is rather different than the classical notation introduced in the seminal work of [1] and used in [9]. We will use this classical notation. Further, a more detailed literature review is presented. Some of the proofs from [10] have been reworked and a new, more general example for the computation of a supervisor is used. We also consider a special case when only controllable events can occur in a plant. This case has not been solved in any of the previous work.

Section II introduces PDES modeled as generators of probabilistic languages. The technique of random disablement is presented in Section III. The probabilistic supervisory problem is stated in Section IV. This section also introduces the main results of [9] and [10]. These results apply to nonterminating PDESs. Section V gives the formal proof of the main result. Section VI extends the presented results to terminating PDESs. Finally, Section VII concludes with avenues for future work.

II. Preliminaries

A probabilistic DES (PDES) is modeled as a probabilistic generator $G = (Q, \Sigma, \delta, q_0, Q_m, p)$ [9], where Q is a nonempty set of states, Σ is an 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, $Q_m \subseteq Q$ is the set of marking states, which represent the completed tasks, and $p : Q \times \Sigma \to [0, 1]$ is the statewise event probability distribution. The state transition function is traditionally extended by induction on the length of strings to $\delta : Q \times \Sigma^* \to Q$ (for $q \in Q, s \in \Sigma^*, \sigma \in \Sigma$):

$$\delta(q, \epsilon) = q, \text{ and } \delta(q, s\sigma) = \delta(\delta(q, s), \sigma),$$

whenever $\delta(q, s)$ and $\delta(\delta(q, s), \sigma)$ are both defined. 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 $q \in Q$, $\sum_{\sigma\in\Sigma} p(q, \sigma) = 1$. The probabilistic generator G is terminating if there is at least one reachable state $q \in Q$ such that $\sum_{\sigma\in\Sigma} p(q, \sigma) < 1$. Upon entering state q, the probability that the system terminates at that state is $1 - \sum_{\sigma\in\Sigma} p(q, \sigma)$. Throughout the sequel, unless stated otherwise, we will be considering nonterminating generators.

The language $L(G)$ generated by a probabilistic DES automaton $G = (Q, \Sigma, \delta, q_0, Q_m, p)$ is given by $L(G) = \{s \in \Sigma^* \mid \delta(q_0, s)!\}$. The marked language of G is given by $L_m(G) = \{s \in \Sigma^* \mid \delta(q_0, s) \in Q_m\}$, whereas 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 $s \in L(G)$.

III. Control of PDES

As in classical supervisory theory, the set Σ is partitioned into Σ_c and Σ_u, the sets of controllable and uncontrollable events, respectively. Given a specification language $E \subseteq \Sigma^*$ and generator G of probabilistic language $L_p(G)$ representing a plant, the goal is to find a supervisor V such that the language generated by the plant under supervision, $L_p(V/G)$, is equal to E. A classical, deterministic supervisor can only disable controllable events $\sigma \in \Sigma_c$. It can be defined using a function $V : L(G) \times \Sigma \to \{0, 1\}$:

$$(\forall s \in L(G))(\forall \sigma \in \Sigma)V(s, \sigma) = \begin{cases} 1 & \text{if } \sigma \in \Sigma_u \text{ or } s\sigma \in E \\ 0 & \text{otherwise} \end{cases}$$

We now explore the limited effect a classical supervisor can have on a PDES. Figure 1 shows two PDESs: the first one, G, represents a plant, and the second one, G_2, is a requirements specification. Controllable events are marked with a bar on their edges. A number next to an event represents the

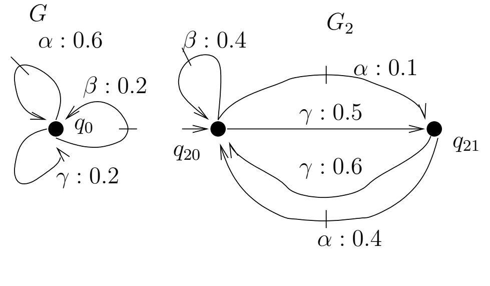

Fig. 1. Plant G and requirements specification G_2

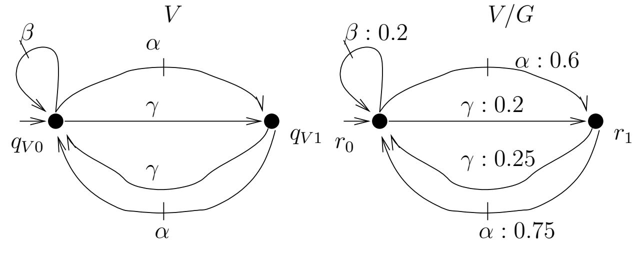

Fig. 2. Deterministic supervisor V and controlled plant V/G

probability distribution of that event. G has alphabet $\Sigma = \{\alpha, \beta, \gamma\}$ and is nonterminating. The event γ is uncontrollable, and, therefore, always enabled. We also make an important assumption about the behaviour of a supervisor: *After an event is disabled, the probabilities of the remaining enabled events proportionally increase*. The question we want to answer is: Is there a deterministic supervisor V such that $L_p(V/G) = L_p(G_2)$?

We first consider the case when the PDES G is in the state q_0 and the PDES G_2 is in the state q_{20}. The required probabilities of all the events in the state of q_{20} are nonzero. Therefore, the deterministic supervisor V should enable all (controllable) events (state q_{V_0} of DES V in Figure 2). Hence, the probabilities of events in the controlled plant remain unchanged (see state r_0 of the controlled plant V/G in Figure 2).

Next, after an odd number of α and γ events (PDES G is in state q_0 and PDES G_2 is in state q_{21}), the supervisor should disable β. When V disables only β, the plant can choose between α and γ. The probabilities of these events occuring in the resulting system are increased proportional to their initial probabilities. Therefore, the probability of α occuring in state r_1 of the controlled plant is equal to:

$$P(\alpha|\sigma \in \{\gamma, \alpha\}) = \frac{p(q_0, \alpha)}{p(q_0, \alpha) + p(q_0, \gamma)} = \frac{0.6}{0.6 + 0.2} = 0.75$$

Similarly, the probability of γ ocurring is 0.25.

Therefore, although the requirement was met nonprobabilistically (meaning $L(V/G) = L(G_2)$), it is obvious that there is no deterministic control such that $L_p(V/G) = L_p(G_2)$. The example illustrates that application of deterministic supervisors to PDES results in a rather limited class of probabilistic languages. Hence, applying a deterministic supervisor to a PDES might be unacceptable for a designer.

We now generalize deterministic supervisors for DES to *probabilistic supervisors*. The control technique used is called *random disablement*. Instead of deterministically enabling or disabling controllable events, probabilistic supervisors enable them with certain probabilities. This means that, upon reaching a certain state q, the control pattern is chosen according to supervisor's probability distributions of controllable events. Consequently, the controller does not always enable the same events when in the state q.

For a PDES $G = (Q, \Sigma, \delta, q_0, Q_m, p)$, a *probabilistic supervisor* is a function $V_p : L(G) \times \Sigma \to [0, 1]$ such that

$$(\forall s \in L(G))(\forall \sigma \in \Sigma)V_p(s, \sigma) = \begin{cases} 1 & \text{if } \sigma \in \Sigma_u \\ x_{s,\sigma} & \text{otherwise, where } x_{s,\sigma} \in [0, 1] \end{cases}$$

Therefore, after observing a string s, the supervisor enables the event σ with a probability $V_p(s, \sigma)$. After a set of controllable events to be enabled, Θ, has been decided upon (uncontrollable events are always enabled), the system acts as if supervised by a deterministic supervisor. An example of a probabilistic supervisor is given in Figure 3. Note that the probabilities of all the events that can execute in a state of

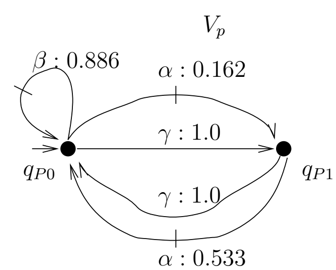

Fig. 3. Probabilistic supervisor V_p

this generator do not, in general, add up to 1. This is because those are not the probabilities of events occurring, but rather being enabled.

What is the probability that an event α will occur in a plant G under the control of probabilistic supervisor V_p when the string $s \in L(G)$ has been observed? First, the control pattern is chosen according to controllable event probabilities of the supervisor, and then, under that pattern, the plant makes a choice according to its events probabilities. For a state q, we define *the set of possible events* to be

$Pos(q) := \{\sigma \in \Sigma | p(q,\sigma) > 0\}$. Then, the probability that the event $\alpha \in \Sigma$ will occur after string s has been observed (let q be the state of the plant after s) is equal to:

$$P(\alpha \in V_p/G|s) = \sum_{\Theta \in \mathcal{P}(Pos(q)\cap\Sigma_c)} P(\alpha | V_p \text{ enables } \Theta \text{ after } s) \cdot P(V_p \text{ enables } \Theta | s) \tag{1}$$

where

$$\begin{aligned} P(\alpha | V_p \text{ enables } \Theta \text{ after } s) &= \begin{cases} \dfrac{p(q,\alpha)}{\sum\limits_{\sigma\in\Theta\cup\Sigma_u} p(q,\sigma)} & \text{if } \alpha \in \Theta \cup \Sigma_u \\ 0 & \text{otherwise} \end{cases} \\ P(V_p \text{ enables } \Theta | s) &= \prod_{\sigma\in\Theta} V_p(s,\sigma) \cdot \prod_{\sigma\in(Pos(q)\cap\Sigma_c)\setminus\Theta} (1 - V_p(s,\sigma)) \end{aligned} \tag{2}$$

For the probabilistic controller from Figure 3, the possible enablement patterns in state q_{P0} are $\Theta = \emptyset$, $\Theta = \{\alpha\}$, $\Theta = \{\beta\}$, and $\Theta = \{\alpha,\beta\}$. Let $P_\sigma(s)$ be the probabillity of σ occurring in the plant under control after the string s is observed, $P_\sigma(s) = P(\sigma \in V_p/G|s)$. If we apply (1) to the controller from Figure 3 for a string s such that the controller is at state q_{P0}, then :

$$\begin{aligned} P_\alpha(s) &= p(q_0,\alpha)V_p(s,\alpha)V_p(s,\beta) + P(\alpha|\sigma \in \{\gamma,\alpha\})V_p(s,\alpha)(1 - V_p(s,\beta)) \\ &= 0.6(0.162)(0.886) + 0.75(0.162)(1 - 0.886) \\ &= 0.1 \end{aligned} \tag{3}$$

$$\begin{aligned} P_\beta(s) &= p(q_0,\beta)V_p(s,\alpha)V_p(s,\beta) + P(\beta|\sigma \in \{\gamma,\beta\})(1 - V_p(s,\alpha))V_p(s,\beta) \\ &= 0.2(0.162)(0.886) + 0.5(1 - 0.162)(0.886) \\ &= 0.4 \end{aligned} \tag{4}$$

$$\begin{aligned} P_\gamma(s) &= p(q_0,\gamma)V_p(s,\alpha)V_p(s,\beta) + P(\gamma|\sigma \in \{\gamma,\alpha\})V_p(s,\alpha)(1 - V_p(s,\beta)) \\ &\quad + P(\gamma|\sigma \in \{\gamma,\beta\})(1 - V_p(s,\alpha))V_p(s,\beta) \\ &\quad + P(\gamma|\sigma \in \{\gamma\})(1 - V_p(s,\alpha))(1 - V_p(s,\beta)) \\ &= 0.2(0.162)(0.886) + 0.25(0.162)(1 - 0.886) \\ &\quad + 0.5(1 - 0.162)(0.886) + 1.0(1 - 0.162)(1 - 0.886) \\ &= 0.5 \end{aligned} \tag{5}$$

Similarly, for a string t that corresponds to the supervisor in Figure 3 being at state q_{P1}, we have $P_\alpha(t) = 0.4$, $P_\beta(t) = 0$ and $P_\gamma(t) = 0.6$. Therefore, the plant under probabilistic control succeeds in generating the probabilistic language $L_p(G_2)$ whereas a deterministic controller failed. However, in the general case, for a given plant, there might not exist a probabilistic supervisor for a given probabilistic specification language. In the next section, we will explore the conditions under which a probabilistic supervisor exists.

IV. Probabilistic Supervisory Control Problem

A. Problem Statement

Our goal is to match the behaviour of the controlled plant with a given probabilistic specification language. We call this problem the *Probabilistic Supervisory Control Problem*. More formally:

Given a plant PDES G_1 and a specification PDES G_2, find, if possible, a probabilistic supervisor V_p such that $L_p(V_p/G_1) = L_p(G_2)$.

Let $x_s \in [0,1]^{Pos(q)\cap\Sigma_c}$ be the control input after string s has been observed, and we will write $x_{s,\sigma}$ instead of $x_s(\sigma)$, $\sigma \in Pos(q) \cap \Sigma_c$.

First, we note that the ratio between the probabilities of two uncontrollable events in an uncontrolled plant should remain the same in the plant under control. Here is the informal reasoning for this claim. Let $\alpha, \beta \in \Sigma_u$. We compare the values of $P(\alpha \in V/G|s)$ and $P(\beta \in V/G|s)$ when calculated using (1). For a control policy Θ, the value of $P(V_p \text{ enables } \Theta|s)$ is the same for both α and β. Also, the factor $1/\sum_{\sigma\in\Theta\cup\Sigma_u} p(s,\sigma)$ is the same for both events. Therefore, the only distinguishing factor is $p(s,\sigma)$, which is a constant. The formal proof of this claim will be presented in Section V.

Further, we consider controllable events. We slightly abuse the notation for $P_\sigma(s)$ in order to explicitly relate $P_\alpha(s)$ and $P_\beta(s)$ of (3) and (4) to supervisor probabilities, $x_{s,\alpha}$ and $x_{s,\beta}$. If we apply (1) to the plant G from Figure 1, then:

$$\begin{aligned} P_\alpha(x_{s,\alpha}, x_{s,\beta}) &= 0.6x_{s,\alpha}x_{s,\beta} + 0.75x_{s,\alpha}(1-x_{s,\beta}) \\ &= 0.75x_{s,\alpha} - 0.15x_{s,\alpha}x_{s,\beta} \quad (6) \\ P_\beta(x_{s,\alpha}, x_{s,\beta}) &= 0.2x_{s,\alpha}x_{s,\beta} + 0.5(1-x_{s,\alpha})x_{s,\beta} \\ &= 0.5x_{s,\beta} - 0.3x_{s,\alpha}x_{s,\beta} \quad (7) \end{aligned}$$

Our goal is to find constraints on P_α and P_β such that (6) and (7) are satisfied, and $(x_{s,\alpha}, x_{s,\beta}) \in [0,1]\times[0,1]$. To map this region from the $(x_{s,\alpha}, x_{s,\beta})$ to the (P_α, P_β) plane (as shown at Figure 4), we

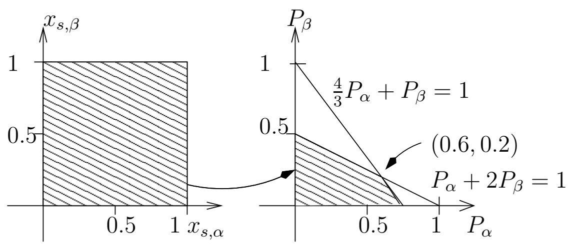

Fig. 4. Mapping from $(x_{s,\alpha}, x_{s,\beta})$ to (P_α, P_β) plane

use the following logic. If $x_{s,\alpha}$ is equal to 0, then, according to (6) and (7), $P_\alpha = 0$ (α is disabled), and $P_\beta = 0.5x_{s,\beta}$. So, in this case, $P_\alpha = 0$ and $P_\beta \in [0, 0.5]$. Similarly, when $x_{s,\beta} = 0$, then $P_\beta = 0$ and $P_\alpha \in [0, 0.75]$. For $x_{s,\alpha} = 1$, we solve for $x_{s,\beta}$ in one of (6) or (7) and substitute it into other equation to get $\frac{4}{3}P_\alpha + P_\beta = 1$. Similarly, for $x_{s,\beta} = 1$, we get $P_\alpha + 2P_\beta = 1$. Those two lines intersect at the point $(0.6, 0.2)$, that corresponds to the probabilities of α and β in the original, uncontrolled system.

There exists another way to derive those bounds. Since G is nonterminating and the controller never disables all the events, then

$$P_\alpha + P_\beta + P_\gamma = 1. \quad (8)$$

Since γ is uncontrollable, then $V_p(s,\gamma) = 1$. Let's consider a case when $x_{s,\alpha} = 1$. This means that α is effectively uncontrollable, so:

$$\frac{P_\gamma}{P_\alpha} = \frac{p(q,\gamma)}{p(q,\alpha)} = \frac{1}{3}$$

Also, as $x_{s,\alpha}$ decreases, $P(\alpha)$ decreases too. Therefore:

$$P_\gamma \geq \frac{1}{3}P_\alpha$$

with equality holding when $x_{s,\alpha} = 1$. We plug this back into (8) to get $\frac{4}{3}P_\alpha + P_\beta \leq 1$. Similarly, if we assume that $x_{s,\beta} = 1$, we get $P_\alpha + 2P_\beta \leq 1$.

B. Main Result

In the previous subsection we gave the intuition behind the conditions for the existence of the probabilistic supervisor that were first presented in [9]. We now present the conditions formally together with an algorithm for the computation of the supervisor which is the main result of [10]. We incorporated the special case ($Pos(q) \cap \Sigma_u = \emptyset$) into the theorems.

For notational convenience, instead of $x(\sigma)$ ($x(\sigma) \in \mathbb{R}, \sigma \in \Sigma$), we will write x_σ.

Theorem 1. *Let $G_1 = (Q, \Sigma, \delta_1, q_0, Q_m, p_1)$ and $G_2 = (R, \Sigma, \delta_2, r_0, R_m, p_2)$ be two nonterminating PDESs with disjoint state sets Q and R. There exists a probabilistic supervisor V_p such that $L_p(V_p/G_1) = L_p(G_2)$ iff for all $s \in L(G_2)$ there exists $q \in Q$ such that $\delta_1(q_0, s) = q$ and, letting $r = \delta_2(r_0, s)$, the following two conditions hold:*

(i) $Pos(q) \cap \Sigma_u = Pos(r) \cap \Sigma_u$, *and for all* $\sigma \in Pos(q) \cap \Sigma_u$,

$$\frac{p_1(q,\sigma)}{\sum\limits_{\alpha\in\Sigma_u} p_1(q,\alpha)} = \frac{p_2(r,\sigma)}{\sum\limits_{\alpha\in\Sigma_u} p_2(r,\alpha)}$$

(ii) $Pos(r) \cap \Sigma_c \subseteq Pos(q) \cap \Sigma_c$, *and, if* $Pos(q) \cap \Sigma_u \neq \emptyset$, *then for all* $\sigma \in Pos(q) \cap \Sigma_c$,

$$\frac{p_2(r,\sigma)}{p_1(q,\sigma)} \sum_{\alpha\in\Sigma_u} p_1(q,\alpha) + \sum_{\alpha\in Pos(q)\cap\Sigma_c} p_2(r,\alpha) \leq 1.$$

Conditions (i) and (ii) together are necessary and sufficient for the existence of a probabilistic supervisor solving the PSCP. The first part of both conditions corresponds to controllability as used in classical supervisory theory (namely, the condition $Pos(q)\cap\Sigma_u = Pos(r)\cap\Sigma_u$ of (i), and $Pos(r)\cap\Sigma_c \subseteq Pos(q)\cap\Sigma_c$ of (ii)). The remaining equations and inequalities correspond to the conditions for probability matching.

Theorem 2. *Assume that the conditions (i) and (ii) of Theorem 1 are satisfied. Let $\Gamma = Pos(q) \cap \Sigma_c$ if $Pos(q) \cap \Sigma_u \neq \emptyset$, and $\Gamma = (Pos(q) \cap \Sigma_c)\backslash\{\gamma\}$ otherwise, where $\gamma \in Pos(q)$ is such that for every $\sigma \in Pos(q)$, $\frac{p_2(r,\gamma)}{p_1(q,\gamma)} \geq \frac{p_2(r,\sigma)}{p_1(q,\sigma)}$ is satisfied. Let $x_s^0 \in [0,1]^\Gamma$ and $f : \mathbb{R}^\Gamma \to \mathbb{R}^\Gamma$. For $x_s^0 = 0$, the sequence*

$$x_s^{k+1} = f(x_s^k), \qquad k = 0, 1, \ldots, \text{ where}$$

$$f_\sigma(x_s) = \frac{p_2(r,\sigma)}{p_1(q,\sigma) h_\sigma(x_s)}, \sigma \in \Gamma \text{ and}$$

$$h_\sigma(x_s) = \sum_{\Theta\in\mathcal{P}(\Gamma\backslash\{\sigma\})} \frac{1}{1 - \sum\limits_{\alpha\in\Theta} p_1(q,\alpha)} \prod_{\alpha\in\Theta} (1 - x_{s,\alpha}) \prod_{\alpha\in\Gamma\backslash\{\sigma\}\backslash\Theta} x_{s,\alpha}$$

converges to the control input x_s^* (i.e., $V_p(s,\sigma) = x_{s,\sigma}^*$ for $\sigma \in \Gamma$).

V. Formal Proof

If there exists a probabilistic supervisor V_p such that $L_p(V_p/G_1) = L_p(G_2)$, then $L(G_2) \subseteq L(G_1)$. Therefore, let $s \in L(G_2)$ and assume there exists $q \in Q$ such that $q = \delta_1(q_0, s)$, and $r = \delta_2(r_0, s)$. For notational convenience, whenever obvious from the context, we will omit the symbols for strings and states so that, e.g., instead of $p_1(q,\sigma)$, we shall write $p_{1,\sigma}$, and instead of $p_2(r,\sigma)$, we shall write $p_{2,\sigma}$. We will use small Greek letters to denote events and capital Greek letters for sets of events. To denote sets whose elements are not necessarily events, capital Roman letters will be used, and small Roman letters will denote functions. Also, we assume the set difference operation to be left-associative.

Further, without loss of generality, we assume that at the state q, not all the possible events are controllable, that is $\sum_{\sigma\in\Sigma_c} p_{1,\sigma} < 1$. This assumption is safe since if $p_{1,\sigma} = 0$ for all $\sigma \in \Sigma_u$, then the PSCP reduces to the PSCP with only controllable events which can be transformed into a problem with exactly one uncontrollable event (we will discuss this further in Section V-B). Note that in the case of at least one possible uncontrollable event, we have $\Gamma = Pos(q) \cap \Sigma_c$.

After a string $s \in L(G_2)$ has been observed, the supervisory problem is effectively the problem of finding the control input vector $x \in [0,1]^\Gamma$ such that $P(x) = p_2$, where $P : [0,1]^\Gamma \to [0,1]^\Sigma$ and P_σ is defined as $P_\sigma = P(\sigma \text{ in } V_p/G|s)$ given by Equation 1, for all $\sigma \in \Sigma$.

A. Proof

Lemma 1. *Let $x \in \mathbb{R}^\Psi$ and $\Psi \subseteq \Gamma$. Then,* $\sum\limits_{\Phi\subseteq\Psi} \prod\limits_{\sigma\in\Phi} x_\sigma \prod\limits_{\sigma\in\Psi\setminus\Phi} (1-x_\sigma) = 1.$

Proof: We prove the lemma by the induction on the size of Ψ. Let $k = |\Psi|$. For $k = 1$, the identity is satisfied. Assume that it holds for $k = n-1$. We now prove that it holds for $k = n$. Let $\sigma \in \Gamma$, but $\sigma \nsubseteq \Psi$. Then:

$$\begin{aligned}
\sum_{\Phi\subseteq\Psi\cup\{\sigma\}} \prod_{\alpha\in\Phi} x_\alpha \prod_{\alpha\in(\Psi\cup\{\sigma\})\setminus\Phi} (1-x_\alpha) &= \sum_{\substack{\Phi\subseteq\Psi\cup\{\sigma\}\\ \sigma\in\Phi}} \prod_{\alpha\in\Phi} x_\alpha \prod_{\alpha\in(\Psi\cup\{\sigma\})\setminus\Phi} (1-x_\alpha) + \sum_{\substack{\Phi\subseteq\Psi\cup\{\sigma\}\\ \sigma\notin\Phi}} \prod_{\alpha\in\Phi} x_\alpha \prod_{\alpha\in(\Psi\cup\{\sigma\})\setminus\Phi} (1-x_\alpha) \\
&= x_\sigma \sum_{\Phi\subseteq\Psi} \prod_{\alpha\in\Phi} x_\alpha \prod_{\alpha\in\Psi\setminus\Phi} (1-x_\alpha) + (1-x_\sigma) \sum_{\Phi\subseteq\Psi} \prod_{\alpha\in\Phi} x_\alpha \prod_{\alpha\in\Psi\setminus\Phi} (1-x_\alpha) \\
&= 1
\end{aligned}$$

■

Lemma 2. *Let $x \in [0,1]^\Gamma$. Then, $P_\sigma(x) = p_{1,\sigma} x_\sigma h_\sigma(x)$ for every $\sigma \in \Sigma$, where $h_\sigma : \mathbb{R}^\Gamma \to \mathbb{R}$ is given by*

$$h_\sigma(x) = \sum_{\Theta\in\mathcal{P}(\Gamma\setminus\{\sigma\})} \frac{1}{1-\sum\limits_{\alpha\in\Theta} p_{1,\alpha}} \prod_{\alpha\in\Theta} (1-x_\alpha) \prod_{\alpha\in\Gamma\setminus\{\sigma\}\setminus\Theta} x_\alpha \tag{9}$$

Proof: Let $x \in [0,1]^\Gamma$. For $\sigma \in \Sigma$, Equation 1 can be equivalently expressed as:

$$P_\sigma(x) = \sum_{\Theta\in\mathcal{P}(\Gamma\setminus\{\sigma\})} \frac{p_{1,\sigma}}{\sum\limits_{\alpha\in\Theta\cup\{\sigma\}\cup\Sigma_u} p_{1,\alpha}} \prod_{\alpha\in\Theta\cup\{\sigma\}} x_\alpha \prod_{\alpha\in\Gamma\setminus(\Theta\cup\{\sigma\})} (1-x_\alpha)$$

If we apply the substitution $\Omega = \Gamma\setminus(\Theta\cup\{\sigma\})$ to the previous equation, it becomes

$$P_\sigma(x) = \sum_{\Omega\in\mathcal{P}(\Gamma\setminus\{\sigma\})} \frac{p_{1,\sigma}}{1-\sum\limits_{\alpha\in\Omega} p_{1,\alpha}} \prod_{\alpha\in\Omega} (1-x_\alpha) \prod_{\alpha\in\Gamma\setminus\Omega} x_\alpha$$

The previous equation is well-defined as, for any $\Omega \in \mathcal{P}(\Gamma\setminus\{\sigma\})$, we have $1-\sum_{\alpha\in\Omega} p_{1,\alpha} > 0$ since we assumed that there is at least one event $\alpha \in \Sigma_u$ such that $p_{1,\alpha} > 0$. Therefore, for $\sigma \in \Sigma_c$, we have:

$$P_\sigma(x) = \sum_{\Omega\in\mathcal{P}(\Gamma\setminus\{\sigma\})} \frac{p_{1,\sigma}}{1-\sum\limits_{\alpha\in\Omega} p_{1,\alpha}} \prod_{\alpha\in\Omega} (1-x_\alpha) \cdot \left(x_\sigma \prod_{\alpha\in\Gamma\setminus\{\sigma\}\setminus\Omega} x_\alpha \right) = p_{1,\sigma} x_\sigma h_\sigma(x)$$

For $\sigma \in \Sigma_u$, since $\sigma \notin \Gamma$ and $x_\sigma = 1$, then:

$$P_\sigma(x) = \sum_{\Omega\in\mathcal{P}(\Gamma\setminus\{\sigma\})} \frac{p_{1,\sigma}}{1-\sum\limits_{\alpha\in\Omega} p_{1,\alpha}} \prod_{\alpha\in\Omega} (1-x_\alpha) \prod_{\alpha\in\Gamma\setminus\{\sigma\}\setminus\Omega} x_\alpha = p_{1,\sigma} x_\sigma h_\sigma(x)$$

. ■

Next, we introduce the following order on $\mathbb{R}^\Gamma$. For $x, y \in R^\Gamma$, $x \leq y$ iff for all $\sigma \in \Gamma$, $x_\sigma \leq y_\sigma$. Let $f : (X, \leq) \to (Y, \leq)$ be a mapping between posets. This mapping is monotone if whenever $x \leq y$, then $f(x) \leq f(y)$; it is antitone if whenever $x \leq y$, then $f(x) \geq f(y)$. Also, for $a \in \mathbb{R}$, let $\bar{a}$ denote $x \in R^\Gamma$ such that $x_\sigma = a$.

Lemma 3. *Let $\Delta \subseteq \Gamma$ and $l : \mathcal{P}(\Gamma) \to \mathbb{R}$ be positive and monotone. The function $f_\Delta : \mathbb{R}^\Gamma \to \mathbb{R}$ given by* $f_\Delta(x) = \sum\limits_{\Phi\in\mathcal{P}(\Delta)} l(\Phi) \prod\limits_{\sigma\in\Phi} (1-x_\sigma) \prod\limits_{\sigma\in\Delta\setminus\Phi} x_\sigma$ *is positive and antitone on* $[0,1]^\Gamma$.

Proof: First, we find the derivative of the function $f_\Delta(x)$ with respect to x_α, for $\alpha \in \Sigma$. When $\alpha \notin \Delta$, f_Δ does not depend on x_α and $\frac{\partial f_R}{\partial x_\alpha}(x) = 0$. For the case when $\sigma \in \Delta$,

$$\begin{aligned} f_\Delta(x) &= \sum_{\substack{\Phi \in \mathcal{P}(\Delta) \\ \alpha \in \Phi}} l(\Phi) \prod_{\sigma \in \Phi} (1 - x_\sigma) \prod_{\sigma \in \Delta \setminus \Phi} x_\sigma + \sum_{\substack{\Phi \in \mathcal{P}(\Delta) \\ \alpha \notin \Phi}} l(\Phi) \prod_{\sigma \in \Phi} (1 - x_\sigma) \prod_{\sigma \in \Delta \setminus \Phi} x_\sigma \\ &= \sum_{\Phi \in \mathcal{P}(\Delta \setminus \{\alpha\})} \left(l(\Phi \cup \{\alpha\})(1 - x_\alpha) + l(\Phi) x_\alpha \right) \prod_{\sigma \in \Phi} (1 - x_\sigma) \prod_{\sigma \in \Delta \setminus \{\alpha\} \setminus \Phi} x_\sigma \end{aligned}$$

and so

$$\frac{\partial f_\Delta}{\partial x_\alpha}(x) = (-1) \cdot \sum_{\Phi \in \mathcal{P}(\Delta \setminus \{\alpha\})} \left(l(\Phi \cup \{\alpha\}) - l(\Phi) \right) \prod_{\sigma \in \Phi} (1 - x_\sigma) \prod_{\sigma \in \Delta \setminus \{\alpha\} \setminus \Phi} x_\sigma$$

which is always non-positive on $x \in [0,1]^\Gamma$ since l is monotone. Therefore, f_Δ is antitone on $[0,1]^\Gamma$ and, consequently, $f_\Delta(x) \geq f_\Delta(\bar{1}) = l(\emptyset) > 0$. Hence, f_Δ is positive on $[0,1]^\Gamma$. ■

Lemma 4. *The functions $h_\sigma(x)$, $\sigma \in \Sigma$, as defined in Lemma 2 are positive antitone, and such that $x_\sigma h_\sigma(x) \leq h_\gamma(x)$ on $[0,1]^\Gamma$ for all $\sigma \in \Sigma_c, \gamma \in \Sigma_u$.*

Proof: Let $\Delta \subseteq \Gamma$ be a nonempty set. Let $l : \mathcal{P}(\Delta) \to \mathbb{R}$ be given by $l(\Delta) = \frac{1}{1 - \sum\limits_{\sigma \in \Delta} p_{1,\sigma}}$. Let $\Phi, \Lambda \in \mathcal{P}(\Delta)$ be such that $\Phi \subseteq \Lambda$. Since $0 \leq \sum_{\sigma \in \Phi} p_{1,\sigma} \leq \sum_{\sigma \in \Lambda} p_{1,\sigma} < 1$, l is monotone. Therefore, according to Lemma 3, h_σ are positive antitone. We now prove that for all $\sigma \in \Sigma_c, \gamma \in \Sigma_u$ we have $x_\sigma h_\sigma(x) \leq h_\gamma(x)$:

$$\begin{aligned} x_\sigma h_\sigma(x) &= x_\sigma \sum_{\Theta \in \mathcal{P}(\Gamma \setminus \{\sigma\})} l(\Theta) \prod_{\alpha \in \Theta} (1 - x_\alpha) \prod_{\alpha \in \Gamma \setminus \{\sigma\} \setminus \Theta} x_\alpha = \sum_{\Theta \in \mathcal{P}(\Gamma \setminus \{\sigma\})} l(\Theta) \prod_{\alpha \in \Theta} (1 - x_\alpha) \prod_{\alpha \in \Gamma \setminus \Theta} x_\alpha \\ &\leq \sum_{\Theta \in \mathcal{P}(\Gamma)} l(\Theta) \prod_{\alpha \in \Theta} (1 - x_\alpha) \prod_{\alpha \in \Gamma \setminus \Theta} x_\alpha \\ &= \sum_{\Theta \in \mathcal{P}(\Gamma \setminus \{\gamma\})} l(\Theta) \prod_{\alpha \in \Theta} (1 - x_\alpha) \prod_{\alpha \in \Gamma \setminus \Theta \setminus \{\gamma\}} x_\alpha, \quad \text{for any } \gamma \in \Sigma_u, \text{ since } \gamma \notin \Gamma \\ &= h_\gamma(x), \quad \text{for any } \gamma \in \Sigma_u. \end{aligned}$$

■

Let $\{x^k\}$ be a sequence of real numbers, and $x \in \mathbb{R}$. We will write $x^k \uparrow x$ iff $x^k \leq x^{k+1}$ for all $k \in \mathbb{N}$, and $x^k \to x$ as $k \to \infty$. The lemma gives sufficient conditions for the existence of a fixpoint of the function f.

Lemma 5. *Let $f : D \to \mathbb{R}^\Gamma$ be a monotone function on $D \subseteq \mathbb{R}^\Gamma$, $a_0, b_0 \in \mathbb{R}$, such that $a_0 \leq b_0$, $[a_0, b_0]^\Gamma \subseteq D$ and $\bar{a}_0 \leq f(\bar{a}_0)$. Assume that for every $x \in [a_0, b_0]^\Gamma$ such that $x \leq f(x)$, we have $f(x) \leq \bar{b}_0$. Then, the sequence $\{x^k\}$ given by*

$$x^0 = \bar{a}_0, \quad x^{k+1} = f(x^k), \quad k = 0, 1, \ldots$$

exists and is such that $x^k \uparrow x^*$ for some $x^* \in [a_0, b_0]^\Gamma$. If, furthermore, f is lower continuous ($\lim\limits_{x^k \uparrow x} f(x^k) = f(x)$), then $x^* = f(x^*)$.

Proof:

We first use induction to show that the sequence $\{x^k\}$ is a monotone chain contained in $[a_0, b_0]^\Gamma$. Because of the assumptions, $\bar{a}_0 \leq f(\bar{a}_0) \leq \bar{b}_0$, so that the basis step is true. Assume that $\{x^i\}$ for $i \leq k$ forms a monotone chain in $[a_0, b_0]^\Gamma$. Since x^k is in $[a_0, b_0]^\Gamma$, $x^{k+1} = f(x^k)$ is defined. By the induction hypothesis $x^{k-1} \leq x^k$ holds. Hence, since f is monotone, $f(x^{k-1}) \leq f(x^k)$ also holds; consequently,

$x^k \leq x^{k+1}$. Since $\bar{a}_0 \leq x^k$, then $\bar{a}_0 \leq x^{k+1}$. Also, since $f(x^k) \leq \bar{b}_0$, then $x^{k+1} \leq \bar{b}_0$. Therefore, for $i \leq k+1$, $\{x^i\}$ is a monotone chain contained in $[a_0, b_0]^\Gamma$.

Since $\{x^k\}$ is monotone and has a finite upper bound $\bar{b}_0$, it converges to a point $x^* \leq \bar{b}_0$. Since f is lower continuous, from $x^k \uparrow x^*$ follows $x^{k+1} \to f(x^*)$; therefore $x^* = f(x^*)$. ■

The following theorem presents necessary and sufficient conditions for controllable events' probabilities to be assignable to given probabilities. Further, if those conditions are satisfied, the fixpoint algorithm to calculate the control input is given.

Theorem 3. *Assume that $Pos(r) \cap \Sigma_c \subseteq \Gamma$, and, for every $\sigma \in \Gamma$:*

$$\frac{p_{2,\sigma}}{p_{1,\sigma}} \sum_{\alpha \in \Sigma_u} p_{1,\alpha} + \sum_{\alpha \in \Gamma} p_{2,\alpha} \leq 1 \tag{10}$$

Then, the sequence $\{x^k\}$ given by

$$x^0 = 0, \quad x^{k+1} = f(x^k) \quad k = 0, 1, \ldots \qquad \textit{where} \quad f_\sigma(x) = \frac{p_{2,\sigma}}{p_{1,\sigma} h_\sigma(x)}, \sigma \in \Gamma \tag{11}$$

exists and is such that $x^k \uparrow x^*$ for some $x^* \in [0,1]^\Gamma$. Furthermore, $P_\sigma(x^*) = p_{2,\sigma}$ for all $\sigma \in \Sigma_c$. Conversely, for any $x \in [0,1]^\Gamma$, if $p_{2,\sigma} \triangleq P_\sigma(x)$ for all $\sigma \in \Sigma_c$, then $Pos(r) \cap \Sigma_c \subseteq \Gamma$ and Equation 10 holds.

Proof: First, we show that f is defined on $[0,1]^\Gamma$ and monotone. By Lemma 3, h_σ is positive and antitone on $[0,1]^\Gamma$. Therefore, f_σ is positive and monotone on $[0,1]^\Gamma$, and $f(\bar{0}) \geq \bar{0}$. We show that whenever $x \leq f(x)$, then $f(x) \leq \bar{1}$. For $x \in [0,1]^\Gamma$ assume that $x \leq f(x)$. Then $P_\sigma(x) = p_{1,\sigma} x_\sigma h_\sigma(x) \leq p_{2,\sigma}$ for $\sigma \in \Gamma$. For $\sigma \in \Gamma$ let

$$l_\sigma = \frac{1 - \sum\limits_{\alpha \in \Gamma \setminus \{\sigma\}} p_{1,\alpha}}{p_{1,\sigma}}.$$

Note that l_σ is well-defined since $p_{1,\sigma} > 0$.

$$\begin{aligned}
l_\sigma p_{1,\sigma} h_\sigma(x) &= \sum_{\Omega \in \mathcal{P}(\Gamma \setminus \{\sigma\})} \left(1 - \frac{\sum\limits_{\alpha \in \Gamma \setminus \{\sigma\} \setminus \Omega} p_{1,\alpha}}{1 - \sum\limits_{\alpha \in \Omega} p_{1,\alpha}} \right) \prod_{\alpha \in \Omega} (1 - x_\alpha) \prod_{\alpha \in \Gamma \setminus \Omega} x_\alpha \\
&= 1 - \sum_{\Omega \in \mathcal{P}(\Gamma \setminus \{\sigma\})} \sum_{\alpha \in \Gamma \setminus \{\sigma\} \setminus \Omega} \frac{p_{1,\alpha}}{1 - \sum\limits_{\alpha \in \Omega} p_{1,\alpha}} \prod_{\alpha \in \Omega} (1 - x_\alpha) \prod_{\alpha \in \Gamma \setminus \{\sigma\} \setminus \Omega} x_\alpha \qquad \text{(from Lemma 1)} \\
&\geq 1 - \sum_{\Omega \in \mathcal{P}(\Gamma \setminus \{\sigma\})} \sum_{\alpha \in \Gamma \setminus \{\sigma\} \setminus \Omega} \left(\frac{p_{1,\alpha}}{1 - \sum\limits_{\alpha \in \Omega} p_{1,\alpha}} x_\sigma + \frac{p_{1,\alpha}}{1 - p_{1,\sigma} - \sum\limits_{\alpha \in \Omega} p_{1,\alpha}} (1 - x_\sigma) \right) \prod_{\alpha \in \Omega} (1 - x_\alpha) \prod_{\alpha \in \Gamma \setminus \{\sigma\} \setminus \Omega} x_\alpha \\
&\left(\text{since } \frac{p_\alpha}{1 - \sum\limits_{\alpha \in \Omega} p_\alpha} x_\sigma + \frac{p_\alpha}{1 - p_\alpha - \sum\limits_{\alpha \in \Omega} p_\alpha} (1 - x_\sigma) \geq \frac{p_\alpha}{1 - \sum\limits_{\alpha \in \Omega} p_\alpha} \right) \\
&= 1 - \sum_{\Omega \in \mathcal{P}(\Gamma \setminus \{\sigma\})} \sum_{\alpha \in \Gamma \setminus \{\sigma\} \setminus \Omega} \frac{p_{1,\alpha}}{1 - \sum\limits_{\alpha \in \Omega} p_{1,\alpha}} \prod_{\alpha \in \Omega} (1 - x_\alpha) \prod_{\alpha \in \Gamma \setminus \Omega} x_\alpha - \\
&\qquad - \sum_{\substack{\Omega \in \mathcal{P}(\Gamma) \\ \sigma \in \Omega}} \sum_{\alpha \in \Gamma \setminus \{\sigma\} \setminus \Omega} \frac{p_{1,\alpha}}{1 - \sum\limits_{\alpha \in \Omega} p_{1,\alpha}} \prod_{\alpha \in \Omega} (1 - x_\alpha) \prod_{\alpha \in \Gamma \setminus \Omega} x_\alpha \\
&= 1 - \sum_{\Omega \in \mathcal{P}(\Gamma)} \sum_{\alpha \in \Gamma \setminus \{\sigma\} \setminus \Omega} \frac{p_{1,\alpha}}{1 - \sum\limits_{\alpha \in \Omega} p_{1,\alpha}} \prod_{\alpha \in \Omega} (1 - x_\alpha) \prod_{\alpha \in \Gamma \setminus \Omega} x_\alpha
\end{aligned}$$

$$= 1 - \sum_{\alpha \in \Gamma \backslash \{\sigma\}} \sum_{\Omega \in \mathcal{P}(\Gamma \backslash \{\alpha\})} \frac{p_{1,\alpha}}{1 - \sum\limits_{\alpha \in \Omega} p_{1,\alpha}} \prod_{\alpha \in \Omega} (1 - x_\alpha) \prod_{\alpha \in \Gamma \backslash \Omega} x_\alpha$$

(Suppose $\Omega \in \mathcal{P}(\Gamma)$ and $\alpha \in \Gamma \backslash \{\sigma\} \backslash \Omega$. Then $\alpha \in \Gamma \backslash \{\sigma\}$. Since $\alpha \notin \Omega, \Omega \backslash \{\alpha\} = \Omega$, so $\Omega \in \mathcal{P}(\Gamma \backslash \{\alpha\})$)

$$= 1 - \sum_{\alpha \in \Gamma \backslash \{\sigma\}} P_\alpha(x) \qquad \text{(from Lemma 2)}$$

$$\geq 1 - \sum_{\alpha \in \Gamma \backslash \{\sigma\}} p_{2,\alpha} \qquad \text{(since } P_\sigma \leq p_{2,\sigma} \text{ for } \alpha \in \Gamma)$$

Therefore,

$$f_\sigma(x) = \frac{p_{2,\sigma}}{p_{1,\sigma} h_\sigma(x)} = \frac{l_\sigma p_{2,\sigma}}{l_\sigma p_{1,\sigma} h_\sigma(x)} \leq \frac{l_\sigma p_{2,\sigma}}{1 - \sum\limits_{\alpha \in \Gamma \backslash \{\sigma\}} p_{2,\alpha}} \leq 1$$

Using Lemma 5, the sequence $x^0 = \bar{0}$ and $x^{k+1} = f(x^k)$ for $k = 0, 1, \ldots,$ exists and is such that $x^k \uparrow x^*$ for some $x^* \in [0,1]^\Gamma$. Since f is continuous on $[0,1]^\Gamma$, it follows that $f(x^*) = x^*$, or equivalently that $P_\sigma(x^*) = p_{2,\sigma}$ for $\sigma \in \Gamma$. Also, since $Pos(r) \cap \Sigma_c \subseteq \Gamma$, then for $\sigma \in \Sigma_c$, but $\sigma \notin \Gamma$, we have $P_\sigma(x^*) = p_{1,\sigma} x^*_\sigma h_\sigma = 0 = p_{2,\sigma}$. Therefore, $P_\sigma(x^*) = p_{2,\sigma}$ for $\sigma \in \Sigma_c$.

We now prove the necessity of the condition. Suppose that there is $x \in [0,1]^\Gamma$ such that $P_\sigma(x) = p_{2,\sigma}$ for any $\sigma \in \Sigma_c$. Then, for $\sigma \notin \Gamma$ we have $P_\sigma(x) = 0 = p_{2,\sigma}$. Therefore, $Pos(r) \cap \Sigma_c \subseteq \Gamma$. Next, we note that for any $\sigma, \alpha \in \Sigma_u$ we have that $h_\sigma = h_\alpha$. We therefore introduce the function $g : \mathbb{R}^\Gamma \to \mathbb{R}$ such that $g(x) = h_\sigma(x)$ for any $\sigma \in \Sigma_u$. Then, since our generators are nonterminating,

$$1 = \sum_{\alpha \in \Sigma_u} P_\alpha(x) + \sum_{\alpha \in \Sigma_c} P_\alpha(x) = \sum_{\alpha \in \Sigma_u} p_{1,\alpha} x_\alpha h_\alpha(x) + \sum_{\alpha \in \Sigma_c} p_{2,\alpha} = g(x) \sum_{\alpha \in \Sigma_u} p_{1,\alpha} + \sum_{\alpha \in \Gamma} p_{2,\alpha}. \tag{12}$$

Since $p_{2,\sigma} = p_{1,\sigma} x_\sigma h_\sigma(x)$ and, according to Lemma 4, $x_\sigma h_\sigma(x) \leq g(x)$, it follows that $g(x) \geq \frac{p_{2,\sigma}}{p_{1,\sigma}}$. By plugging this inequality into Equation 12, we get the condition (10). ∎

So far we have only considered the conditions under which the probabilities of controllable events can be assigned to specified probabilities. We will now consider uncontrollable events as well.

Lemma 6. *There exists $x \in [0,1]^\Gamma$ such that $P_\sigma(x) = p_{2,\sigma}$ for every $\sigma \in \Sigma$ iff $Pos(r) \cap \Sigma_c \subseteq \Gamma$, $Pos(q) \cap \Sigma_u = Pos(r) \cap \Sigma_u$, (10) holds for every $\sigma \in \Gamma$, and for every $\sigma \in Pos(q) \cap \Sigma_u$*

$$\frac{p_{1,\sigma}}{\sum\limits_{\alpha \in \Sigma_u} p_{1,\alpha}} = \frac{p_{2,\sigma}}{\sum\limits_{\alpha \in \Sigma_u} p_{2,\alpha}} \tag{13}$$

Proof: Assume that $Pos(r) \cap \Sigma_c \subseteq \Gamma$, $Pos(q) \cap \Sigma_u = Pos(r) \cap \Sigma_u$, (10) holds for $\sigma \in \Gamma$, and (13) holds for $\sigma \in Pos(q) \cap \Sigma_u$. Then, according to Theorem 3, for all $\sigma \in \Sigma_c$, we have $P_\sigma(x) = p_{2,\sigma}$, and, according to Lemma 2, for all $\sigma \in \Sigma_u$, $P_\sigma(x) = p_{1,\sigma} x_\sigma h_\sigma(x)$, that is $P_\sigma(x) = p_{1,\sigma} g(x)$. Therefore, for all $\sigma \in \Sigma_u$:

$$P_\sigma(x) = p_{1,\sigma} g(x) = p_{1,\sigma} \cdot \frac{\sum\limits_{\alpha \in \Sigma_u} p_{1,\alpha} g(x)}{\sum\limits_{\alpha \in \Sigma_u} p_{1,\alpha}} = p_{1,\sigma} \cdot \frac{\sum\limits_{\alpha \in \Sigma_u} P_\alpha(x)}{\sum\limits_{\alpha \in \Sigma_u} p_{1,\alpha}} = p_{1,\sigma} \cdot \frac{1 - \sum\limits_{\alpha \in \Sigma_c} P_\alpha(x)}{\sum\limits_{\alpha \in \Sigma_u} p_{1,\alpha}} = p_{1,\sigma} \cdot \frac{\sum\limits_{\alpha \in \Sigma_u} p_{2,\alpha}}{\sum\limits_{\alpha \in \Sigma_u} p_{1,\alpha}} = p_{2,\sigma}$$

For the converse, assume that there is a supervisory controller x such that $P_\sigma(x) = p_{2,\sigma}$ for every $\sigma \in \Sigma$. Then, according to Theorem 3, $Pos(r) \cap \Sigma_c \subseteq \Gamma$ and (10) holds. If $p_{1,\sigma} = 0$, then $P_\sigma(x) = p_{1,\sigma} g(x) = 0 = p_{2,\sigma}$. Also, if $p_{2,\sigma} = 0 = P_\sigma(x) = p_{1,\sigma} g(x)$, then $p_{1,\sigma} = 0$ (since $g(x) \neq 0$). Therefore, $Pos(q) \cap \Sigma_u = Pos(r) \cap \Sigma_u$. Then, for $\sigma \in Pos(q) \cap \Sigma_u$:

$$\frac{p_{2,\sigma}}{\sum\limits_{\alpha \in \Sigma_u} p_{2,\alpha}} = \frac{P_\sigma(x)}{\sum\limits_{\alpha \in \Sigma_u} p_{2,\alpha}} = \frac{p_{1,\sigma}}{\sum\limits_{\alpha \in \Sigma_u} p_{2,\alpha}} g(x) = \frac{p_{1,\sigma}}{\sum\limits_{\alpha \in \Sigma_u} p_{2,\alpha}} \cdot \frac{\sum\limits_{\alpha \in \Sigma_u} p_{1,\alpha} g(x)}{\sum\limits_{\alpha \in \Sigma_u} p_{1,\alpha}}$$

$$= \frac{p_{1,\sigma}}{\sum\limits_{\alpha \in \Sigma_u} p_{2,\alpha}} \cdot \frac{\sum\limits_{\alpha \in \Sigma_u} P_\alpha(x)}{\sum\limits_{\alpha \in \Sigma_u} p_{1,\alpha}} = \frac{p_{1,\sigma}}{\sum\limits_{\alpha \in \Sigma_u} p_{2,\alpha}} \cdot \frac{\sum\limits_{\alpha \in \Sigma_u} p_{2,\alpha}}{\sum\limits_{\alpha \in \Sigma_u} p_{1,\alpha}} = \frac{p_{1,\alpha}}{\sum\limits_{\alpha \in \Sigma_u} p_{1,\alpha}}$$

∎

B. Special Case: $Pos(q) \cap \Sigma_u = \emptyset$

We now address the issue mentioned in Section V: in a certain state, only controllable events can happen in the plant. Then, a probabilistic supervisor can disable them all which would cause termination. However, as we consider nonterminating generators, this is not allowed to happen. An elegant solution is to always enable one event: this event effectively becomes uncontrollable and the problem reduces to the one already proved. We now show that, if an event γ with the maximal ratio $p_{2,\gamma}/p_{1,\gamma}$ is chosen, then the conditions (10) and (13) are satisfied.

Formally, let $Pos(q) \cap \Sigma_u = \emptyset$. Then, only for this local problem, we declare event $\gamma \in Pos(q) \cap \Sigma_c$ to be uncontrollable. Then, $\Gamma = (Pos(q) \cap \Sigma_c) \backslash \{\gamma\}$. The condition (13) is trivially satisfied. The left hand side of the condition in (10), for $\sigma \in \Gamma$, becomes:

$$\begin{aligned}\frac{p_{2,\sigma}}{p_{1,\sigma}} p_{1,\gamma} + \sum_{\alpha \in \Gamma} p_{2,\alpha} &= \frac{p_{2,\sigma}}{p_{1,\sigma}} p_{1,\gamma} - p_{2,\gamma} + \sum_{\alpha \in \Gamma} p_{2,\alpha} + p_{2,\gamma} \\ &= \frac{p_{2,\sigma}}{p_{1,\sigma}} p_{1,\gamma} - p_{2,\gamma} + \sum_{\alpha \in Pos(q) \cap \Sigma_c} p_{2,\alpha} = \frac{p_{2,\sigma}}{p_{1,\sigma}} p_{1,\gamma} - p_{2,\gamma} + 1\end{aligned}$$

Then, the condition (10) becomes:

$$\frac{p_{2,\sigma}}{p_{1,\sigma}} p_{1,\gamma} - p_{2,\gamma} \leq 0,$$

which is equivalent to

$$\frac{p_{2,\sigma}}{p_{1,\sigma}} - \frac{p_{2,\gamma}}{p_{1,\gamma}} \leq 0.$$

If the event γ is one with the maximal ratio $p_{2,\gamma}/p_{1,\gamma}$, it is obvious that the condition is satisfied for any $\sigma \in \Gamma$ and the algorithm for computation of control input from Theorem 3 can be applied to this special case, with γ considered an uncontrollable event.

C. Example

We now present the calculation of a probabilistic supervisor for the example from Figure 1, where $\Sigma_c = \{\alpha, \beta\}$, and $\Sigma_u = \{\gamma\}$. The case when the string $s \in L(G_2)$ has been observed such that G is at the state q_0, and G_2 is at q_{20} will be presented in detail. Again, for notational convenience, we shall write $p_{1,\sigma}$ instead of $p_1(q_0, \sigma)$, and $p_{2,\sigma}$ instead of $p_2(q_{20}, \sigma)$ where $\sigma \in \Sigma$. Let $p_1 = (p_{1,\alpha}, p_{1,\beta}, p_{1,\gamma})$, and $p_2 = (p_{2,\alpha}, p_{2,\beta}, p_{2,\gamma})$. From Figure 1, it follows $p_1 = (0.6, 0.2, 0.2), p_2 = (0.1, 0.4, 0.5)$. First we check if there exists a control input $x_s = (x_{s,\alpha}, x_{s,\beta})$ such that $P(x_s) = p_2$. The equality of Theorem 1 is trivially satisfied. We then check if the inequalities of Theorem 1 are satisfied: $\frac{1}{6} \cdot p_{1,\gamma} + p_{2,\alpha} + p_{2,\beta} = 0.53 \leq 1, 2p_{1,\gamma} + p_{2,\alpha} + p_{2,\beta} = 0.9 \leq 1$.Therefore, there exists $x_s^* \in [0,1]^{\{\alpha,\beta\}}$ such that $P(x_s^*) = p_2$, and vector x_s^* can be calculated by the fixpoint iteration where $x_s^0 = (0,0)$, $x_s^k = f(x_s^{k-1})$, and

$$f_\alpha(x_s) = \frac{p_{2,\alpha}}{p_{1,\alpha}\left(x_{s,\beta} + \frac{1}{1-p_{1,\beta}}(1 - x_{s,\beta})\right)} \quad \text{and} \quad f_\beta(x_s) = \frac{p_{2,\beta}}{p_{1,2}\left(x_{s,\alpha} + \frac{1}{1-p_{1,\alpha}}(1 - x_{s,\alpha})\right)}$$

After just a few iterations, the sequence $\{x_s^k\}$ converges to $x_s^* = (0.162, 0.886)$. The fixpoint computation is shown on Figure 5.

The calculation of the supervisor after the string $t \in L(G)$ has been observed such that G_2 is in the state q_{21} gives the result $x_t^* = (0.533, 0)$. The supervisor is shown in Figure 3.

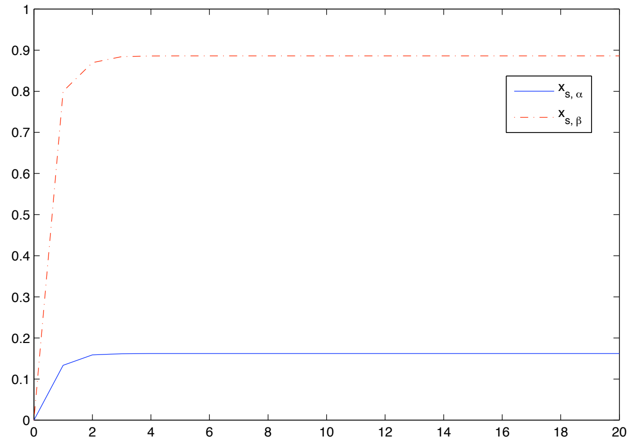

Fig. 5. Fixpoint iteration

VI. Handling terminating PDESs

The results and proofs presented in this paper apply only to nonterminating systems. We now present their extension to terminating systems as introduced in [9]. The terminating PDES $G = (Q, \Sigma, \delta, q_0, Q_m, p)$ extends to nonterminating $G' = (Q \cup \{q_\perp\}, \Sigma \cup \{\sigma_\perp\}, \delta', q_0, Q_m, p')$, where

$$\begin{array}{rcl} p'(q,\sigma) = p(q,\sigma), \sigma \in \Sigma & & \delta'(q,\sigma) = \delta(q,\sigma), \sigma \in \Sigma \\ p'(q,\sigma_\perp) = 1 - \sum_{\sigma\in\Sigma} p(q,\sigma) & \text{and} & \delta'(q,\sigma_\perp) = q_\perp \text{ if } p'(q,\sigma_\perp) > 0 \\ p'(q_\perp,\sigma_\perp) = 1 & & \delta'(q_\perp,\sigma_\perp) = q_\perp \end{array}$$

An example of a terminating generator and its transformation to the nonterminating one is shown in Figure 6. A subtlety of terminating PDES is depicted in Figure 6. Again, in the state q, as depicted in

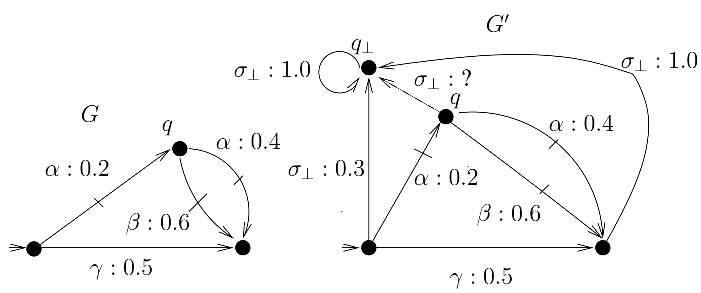

Fig. 6. Terminating PDES G and resulting nonterminating PDES G'

Figure 6, all the events that can occur are controllable. The supervisor can disable them all. The probability of termination in this state is equal to the probability that all the controllable events possible from this state will be disabled. This situation is shown in Figure 6 using a dashed arrow from the state q to $q_\perp$.

VII. Conclusion

In this paper, we presented the solution of the probabilistic supervisory control problem as introduced in [9] and [10]. The control technique used is called random disablement: events are enabled with certain probabilities. The necessary and sufficient conditions for the existence of the solution are introduced and a fixpoint algorithm for computation of supervisor is given. Detailed formal proofs are presented. The results apply to nonterminating PDES. Also, a straightforward extension to handle terminating PDES is presented.

We are currently working on the problem of finding a supervisor such that a closest approximation to a probabilistic specification language is achieved when there does not exist an exact solution to the probabilistic supervisory control problem. Further, probabilistic control with marking, probabilistic control under partial observations, modular probabilistic control etc. would be interesting problems, as well as studying other control objectives (e.g., optimal control, [12]) inside the same framework.

REFERENCES

[1] P. J. Ramadge and W. M. Wonham, "Supervisory control of a class of discrete event processes," *SIAM Journal on Control and Optimization*, vol. 25, no. 1, pp. 206–230, 1987.

[2] C. G. Cassandras, *Discrete Event Systems: Modeling and Performance Analysis*. Homewood, IL, USA: Richadr D. Irwin, Inc., and Aksen Associates, Inc., 1993.

[3] M. O. Rabin, "Probabilistic automata," *Information and Control*, vol. 6, no. 3, pp. 230–245, September 1963. [Online]. Available: citeseer.ist.psu.edu/article/emerson99utilizing.html

[4] M. Molloy, "Performance analysis using stochastic petri nets," *IEEE Transactions on Computers*, vol. C-39, no. 9, pp. 913–917, 1982.

[5] V. Garg, "An algebraic approach to modeling probabilistic discrete event systems," in *Proc. of 31st Conf. Decision and Control*, Tucson, AZ, USA, Dec. 1992, pp. 2348–2353.

[6] ——, "Probabilistic languages for modeling of DEDS," in *Proc. of 26th Conf. on Information Sciences and Systems*, vol. 1, Princeton, NJ, Mar. 1992, pp. 198–203.

[7] H. Mortazavian, "Controlled stochastic languages," in *Proc. of 31st Annual Allerton Conference on Comm., Contr., and Comp.*, 1993, pp. 938–947.

[8] V. S. Borkar, *Topics in controlled Markov chains*. New York: Wiley, 1991.

[9] M. Lawford and W. Wonham, "Supervisory control of probabilistic discrete event systems," in *Proceedings of the 36th Midwest Symposium on Circuits and Systems*, vol. 1. IEEE, Aug. 1993, pp. 327–331.

[10] S. Postma and M. Lawford, "Computation of probabilistic supervisory controllers for model matching," in *Proceedings of Allerton Conference on Communications, Control, and Computing*, 2004.

[11] R. Kumar and V. Garg, "Control of stochastic discrete event systems: Existence," in *Proceedings of 1998 International Workshop on Discrete Event Systems*, Cagliari, Italy, August, 1998, pp. 24–29. [Online]. Available: citeseer.ist.psu.edu/kumar98control.html

[12] ——, "Control of stochastic discrete event systems modeled by probabilistic languages," *IEEE Transactions on Automatic Control*, vol. 46, no. 4, pp. 593–606, Apr. 2001.

[13] Chattopadhyay and A. Ray, "Language-measure-theoretic optimal control of probabilistic finite state systems," in *Proc. of 46th Conf. Decision and Control*, New Orleans, LA, USA, Dec. 2007, pp. 5930–5935.

[14] ——, "Language-measure-theoretic optimal control of probabilistic finite-state systems," *International Journal of Control*, vol. 80, no. 8, pp. 1271–1290, 2007.