International Journal of Control
Vol. 83, No. 3, March 2010, 457–483

Optimal control of infinite horizon partially observable decision processes modelled as generators of probabilistic regular languages

Ishanu Chattopadhyay* and Asok Ray

Department of Mechanical Engineering, The Pennsylvania State University, University Park, PA, USA

(*Received 22 December 2008; final version received 20 July 2009*)

Decision processes with incomplete state feedback have been traditionally modelled as partially observable Markov decision processes. In this article, we present an alternative formulation based on probabilistic regular languages. The proposed approach generalises the recently reported work on language measure theoretic optimal control for perfectly observable situations and shows that such a framework is far more computationally tractable to the classical alternative. In particular, we show that the infinite horizon decision problem under partial observation, modelled in the proposed framework, is λ-approximable and, in general, is not harder to solve compared to the fully observable case. The approach is illustrated via two simple examples.

Keywords: POMDP; formal language theory; partial observation; language measure; discrete event systems

1. Introduction and motivation

Planning under uncertainty is one of the oldest and most studied problems in research literature pertaining to automated decision making and artificial intelligence. Often, the central objective is to sequentially choose control actions for one or more agents interacting with the operating environment such that some associated reward function is maximised for a pre-specified finite future (finite horizon problems) or for all possible futures (infinite horizon problems). The control problem becomes immensely difficult if the effect of such control actions is not perfectly observable to the controller. Absence of perfect observability also makes it hard to take optimal corrective actions in response to uncontrollable exogenous disturbances from the interacting environment. Such scenarios are of immense practical significance; the loss of sensors and communication links often cannot be avoided in modern multi-component and often non-collocated engineered systems. Under these circumstances, an event may conceivably be observable at one plant state while being unobservable at another; event observability may even become dependent on the history of event occurrences. Among the various mathematical formalisms studied to model and solve such control problems, Markov decision processes (MDPs) have received significant attention. A brief overview of the current state-of-art in MDP-based decision theoretic planning is necessary to place this work in appropriate context.

1.1 *Markov decision processes*

MDP models (Puterman 1990; White 1993) extend the classical planning framework (Mcallester and Rosenblitt 1991; Penberthy and Weld 1992; Peng and Williams 1993; Kushmerick, Hanks, and Weld 1995) to accommodate uncertain effects of agent actions with the associated control algorithms attempting to maximise the expected reward and is capable, in theory, of handling realistic decision scenarios arising in operations research, optimal control theory and, more recently, autonomous mission planning in probabilistic robotics (Atrash and Koenig 2001). In brief, an MDP consists of states and actions with a set of action-specific probability transition matrices allowing one to compute the distribution over model states resulting from the execution of a particular action sequence. Thus, the endstate resulting from an action is not known uniquely a priori. However, the agent is assumed to occupy one and only one state at any given time, which is correctly observed, once the action sequence is complete. Furthermore, each state is associated with a reward value and the performance of a controlled MDP is the integrated reward over specified operation time (which can be infinite). A partially observable Markov decision process (POMDP) is a generalisation of MDPs which assumes actions to be nondeterministic as in an MDP, but relaxes the assumption of perfect knowledge of the current model state.

A policy for an MDP is a mapping from the set of states to the set of actions. If both sets are assumed to

*Corresponding author. Email: ixc128@psu.edu

ISSN 0020–7179 print/ISSN 1366–5820 online

DOI: 10.1080/00207170903193458
http://www.informaworld.com

be finite, the number of possible mappings is also finite implying that an optimal policy can be found by conducting search over this finite set. In a POMDP, on the other hand, the current state can be estimated only as a distribution over underlying model states as a function of operation and observation history. The space of all such estimations or *belief states* is a continuous space although the underlying model has only a finite number of states. In contrast to MDPs, a POMDP policy is a mapping from the belief space to the set of actions implying that the computation of optimal policy demands a search over a continuum making the problem drastically more difficult to solve.

1.2 *Negative results pertaining to POMDP solution*

As stated above, an optimal solution to a POMDP is a policy which specifies actions to execute in response to state feedback with the objective of maximising performance. Policies may be *deterministic* with a single action specified at each belief state or *stochastic* which specify an allowable choice of actions at each state. Policies can also be categorised as *stationary*, *time-dependent* or *history-dependent*; stationary policies depend only on the current belief state, time-dependent policies may vary with the operation time and history-dependent policies vary with the state history. The current state-of-art in POMDP solution algorithms (Zhang and Golin 2001; Cassandra 1998) are all variations of Sondick's original work (Sondik 1978) on value iteration based on dynamic programming (DP). Value iterations, in general, are required to solve a large number of linear programs at each DP update and consequently suffer from exponential worst case complexity. Given that it is hard to find an optimal policy, it is natural to try to seek the one that is *good* enough. Ideally, one would be reasonably satisfied to have an algorithm guaranteed to be fast which produces a policy that is reasonably close (λ-approximation) to the optimal solution. Unfortunately, the existence of such algorithms is unlikely or, in some cases, impossible. Complexity results show that POMDP solutions are nonapproximable (Burago, de Rougemont, and Slissenko 1996; Madani, Hanks, and Condon 1999; Lusena, Goldsmith, and Mundhenk 2001) with the above-stated guarantee existing in general only if certain complexity classes collapse. For example, the optimal stationary policy for POMDPs of finite-state space can be λ-approximated if and only if $P = NP$. Table 1 reproduced from Lusena et al. (2001) summarises the known complexity results in this context. Thus, finding the history-dependent optimal policy for even a finite horizon POMDP is PSPACE-complete. Since this is a broader problem class than NP, the result suggests that POMDP problems are even harder than NP-complete problems. Clearly, infinite horizon POMDPs can be no easier to solve than finite horizon POMDPs. In spite of the recent development of new exact and approximate algorithms to efficiently compute optimal solutions (Cassandra 1998) and machine-learning approaches to cope with uncertainty (Hansen 1998), the most efficient algorithms to date are able to compute near optimal solutions only for POMDPs of relatively small state spaces.

Table 1. λ-approximability of optimal POMDP solutions.

Policy	Horizon	Approximability
Stationary	K	Not unless P=NP
Time-dependent	K	Not unless P=NP
History-dependent	K	Not unless P=PSPACE
Stationary	∞	Not unless P=NP
Time-dependent	∞	Uncomputable

1.3 *Probabilistic regular language based models*

This work investigates decision-theoretic planning under partial observation in a framework distinct from the MDP philosophy (Figure 1). Decision processes are modelled as probabilistic finite-state automata (PFSA) which act as generators of probabilistic regular languages (Chattopadhyay and Ray 2008b).

Note: It is important to note that the PFSA model used in this article is conceptually very different from the notion of probabilistic automata introduced by Rabin (1963), Paz (1971), etc., and essentially follows the formulation of p-language theoretic analysis first reported by Garg (1992a, 1992b).

The key differences between the MDP framework and PFSA-based modelling (Figure 1) can be enumerated briefly as follows:

(1) In both MDP and PFSA formalisms, we have the notion of states. The notion of actions in the former is analogous to that of events in the latter. However, unlike actions in the MDP framework, which can be executed at will (if defined at the current state), the generation of events in the context of PFSA models is probabilistic. Also, such events are categorised as being controllable or uncontrollable. A controllable event can be 'disabled' so that the state change due to the generation of that particular event is inhibited; uncontrollable

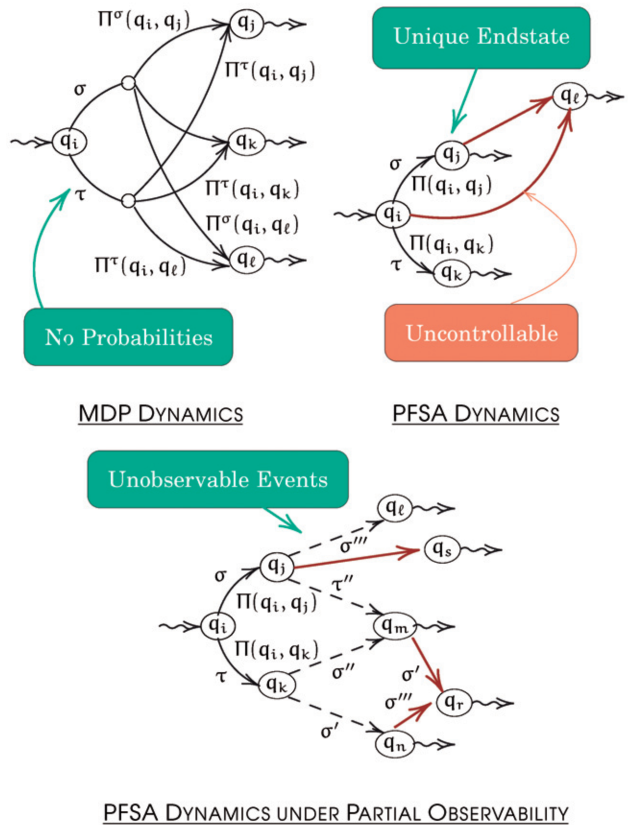

Figure 1. Comparison of modelling semantics for MDPs and PFSA.

events, on the other hand, cannot be disabled in this sense.

(2) For an MDP, given a state and an action selected for execution, we can only compute the probability distribution over model states resulting from the action; although the agent ends up in an unique state due to execution of the chosen action, this endstate cannot be determined a priori. For a PFSA, on the other hand, given a state, we only know the probability of occurrence of each alphabet symbol as the next to-be generated event each of which causes a transition to an a priori known unique endstate; however, the next state is still uncertain due to the possible execution of uncontrollable events defined at the current state. Thus, both the formalisms aim to capture the uncertain effects of agent decisions; albeit via different mechanisms.

(3) Transition probabilities in MDPs are, in general, functions of both the current state and the action executed; i.e. there are m transition probability matrices where m is the cardinality of the set of actions. PFSA models, on the other hand, have only one transition probability matrix computed from the state-based event generation probabilities.

(4) It is clear that MDPs emphasise states and state-sequences; while PFSA models emphasise events and event-sequences. For example, in POMDPs, the observations are states; while those in the observability model for PFSAs (as adopted in this article) are events.

(5) In other words, partial observability in MDP directly results in not knowing the current state; in PFSA models partial observability results in not knowing transpired events which as an effect causes confusion in the determination of the current state.

This article presents an efficient algorithm for computing the history-dependent (Lusena et al. 2001) optimal supervision policy for infinite horizon decision problems modelled in the PFSA framework. The key tool used is the recently reported concept of a rigorous language measure for probabilistic finite-state language generators (Chattopadhyay and Ray 2006). This is a generalisation of the work on language measure-theoretic optimal control for the fully observable case (Chattopadhyay and Ray 2007) and we show in this article that the partially observable scenario is not harder to solve in this modelling framework.

The rest of this article is organised in five additional sections and two brief appendices. Section 2 introduces the preliminary concepts and relevant results from the reported literature. Section 3 presents an online implementation of the language measure-theoretic supervision policy for perfectly observable plants which lays the framework for the subsequent development of the proposed optimal control policy for partially observable systems in Section 4. The theoretical development is verified and validated in two simulated examples in Section 5. This article is summarised and concluded in Section 6 with recommendations for future work.

2. Preliminary concepts and related work

This section presents the formal definition of the PFSA model and summarises the concept of signed real measure of regular languages; the details are reported in Ray (2005), Ray, Phoha, and Phoha (2005) and Chattopadhyay and Ray (2006b). Also, we briefly review the computation of the unique maximally permissive optimal control policy for PFSA (Chattopadhyay and Ray 2007b) via the maximisation of the language measure. In the sequel, this measure-theoretic approach will be generalised to address partially observable cases and is thus critical to the development presented in this article.

2.1 *The PFSA model*

Let $G_i = (Q, \Sigma, \delta, q_i, Q_m)$ be a finite-state automaton model that encodes all possible evolutions of the discrete-event dynamics of a physical plant, where $Q = \{q_k : k \in \mathcal{I}_Q\}$ is the index set of states and $\mathcal{I}_Q \equiv \{1, 2, \ldots, n\}$ is the index set of states; the automaton starts with the initial state q_i; the alphabet of events is $\Sigma = \{\sigma_k : k \in \mathcal{I}_\Sigma\}$, having $\Sigma \bigcap \mathcal{I}_Q = \emptyset$ and $\mathcal{I}_\Sigma \equiv \{1, 2, \ldots, \ell\}$ is the index set of events; $\delta : Q \times \Sigma \to Q$ is the (possibly partial) function of state transitions; and $Q_m \equiv \{q_{m_1}, q_{m_2}, \ldots, q_{m_l}\} \subseteq Q$ is the set of marked (i.e. accepted) states with $q_{m_k} = q_j$ for some $j \in \mathcal{I}_Q$. Let Σ^* be the Kleene closure of Σ, i.e. the set of all finite-length strings made of the events belonging to Σ as well as the empty string ϵ that is viewed as the identity of the monoid Σ^* under the operation of string concatenation, i.e. $\epsilon s = s = s\epsilon$. The state transition map δ is recursively extended to its reflexive and transitive closure $\delta : Q \times \Sigma^* \to Q$ by defining

$$\forall q_j \in Q, \quad \delta(q_j, \epsilon) = q_j \tag{1a}$$

$$\forall q_j \in Q, \sigma \in \Sigma, s \in \Sigma^\star, \quad \delta(q_i, \sigma s) = \delta(\delta(q_i, \sigma), s) \tag{1b}$$

Definition 2.1: The language $L(q_i)$ generated by a deterministic finite-state automata (DFSA) G initialised at the state $q_i \in Q$ is defined as:

$$L(q_i) = \{s \in \Sigma^* \mid \delta^*(q_i, s) \in Q\} \tag{2}$$

The language $L_m(q_i)$ marked by the DFSA G initialised at the state $q_i \in Q$ is defined as

$$L_m(q_i) = \{s \in \Sigma^* \mid \delta^*(q_i, s) \in Q_m\} \tag{3}$$

Definition 2.2: For every $q_j \in Q$, let $L(q_i, q_j)$ denote the set of all strings that, starting from the state q_i, terminate at the state q_j, i.e.

$$L_{i,j} = \{s \in \Sigma^* \mid \delta^*(q_i, s) = q_j \in Q\} \tag{4}$$

To complete the specification of a PFSA, we need to specify the event generation probabilities and the state characteristic weight vector, which we define next.

Definition 2.3: The event generation probabilities are specified by the function $\tilde{\pi} : Q \times \Sigma^\star \to [0, 1]$ such that $\forall q_j \in Q$, $\forall \sigma_k \in \Sigma$, $\forall s \in \Sigma^\star$,

(1) $\tilde{\pi}(q_j, \sigma_k) \stackrel{\triangle}{=} \tilde{\pi}_{jk} \in [0, 1)$; $\sum_k \tilde{\pi}_{jk} = 1 - \theta$, with $\theta \in (0, 1)$;
(2) $\tilde{\pi}(q_j, \sigma) = 0$ if $\delta(q_j, \sigma)$ is undefined; $\tilde{\pi}(q_j, \epsilon) = 1$;
(3) $\tilde{\pi}(q_j, \sigma_k s) = \tilde{\pi}(q_j, \sigma_k)\ \tilde{\pi}(\delta(q_j, \sigma_k), s)$.

Notation 2.1: The $n \times \ell$ event cost matrix $\widetilde{\Pi}$ is defined as: $\widetilde{\Pi}|_{ij} = \tilde{\pi}(q_i, \sigma_j)$

Definition 2.4: The state transition probability $\pi : Q \times Q \to [0, 1)$, of the DFSA G_i is defined as follows:

$$\forall q_i, q_j \in Q, \quad \pi_{ij} = \sum_{\sigma \in \Sigma \text{ s.t. } \delta(q_i, \sigma) = q_j} \tilde{\pi}(q_i, \sigma) \tag{5}$$

Notation 2.2: The $n \times n$ state transition probability matrix Π is defined as $\Pi|_{ij} = \pi(q_i, q_j)$

The set Q_m of marked states is partitioned into Q_m^+ and Q_m^-, i.e. $Q_m = Q_m^+ \cup Q_m^-$ and $Q_m^+ \cap Q_m^- = \emptyset$, where Q_m^+ contains all *good* marked states that we desire to reach, and Q_m^- contains all *bad* marked states that we want to avoid, although it may not always be possible to completely avoid the *bad* states while attempting to reach the *good* states. To characterise this, each marked state is assigned a real value based on the designer's perception of its impact on the system performance.

Definition 2.5: The characteristic function $\chi : Q \to [-1, 1]$ that assigns a signed real weight to state-based sublanguages $L(q_i, q)$ is defined as

$$\forall q \in Q, \quad \chi(q) \in \begin{cases} [-1, 0), & q \in Q_m^- \\ \{0\}, & q \notin Q_m \\ (0, 1], & q \in Q_m^+ \end{cases} \tag{6}$$

The state weighting vector, denoted by $\chi = [\chi_1\ \chi_2\ \ldots\ \chi_n]^T$, where $\chi_j \equiv \chi(q_j)\ \forall j \in \mathcal{I}_Q$, is called the χ-vector. The j-th element χ_j of χ-vector is the weight assigned to the corresponding terminal state q_j.

Remark 2.1: The state characteristic function $\chi : Q \to [-1, 1]$ or equivalently the characteristic vector $\boldsymbol{\chi}$ is analogous to the notion of the reward function in MDP analysis. However, unlike MDP models, where the reward (or penalty) is put on individual state-based actions, in our model, the characteristic is put on the state itself. The similarity of the two notions is clarified by noting that just as MDP performance can be evaluated as the total reward garnered as actions are executed sequentially, the performance of a PFSA can be computed by summing the characteristics of the states visited due to transpired event sequences.

Plant models considered in this article are DFSA (plant) with well-defined event occurrence *probabilities*. In other words, the occurrence of events is probabilistic, but the state at which the plant ends up, *given a particular event has occurred*, is deterministic. No emphasis is laid on the initial state of the plant, i.e. we allow for the fact that the plant may start from any state. Furthermore, having defined the characteristic state weight vector χ, it is not necessary to specify the set of marked states, because if $\chi_i = 0$, then q_i is not marked and if $\chi_i \neq 0$, then q_i is marked.

Definition 2.6 (Control philosophy): If $q_i \xrightarrow{\sigma} q_k$ and the event σ is disabled at state q_i, then the supervisory action is to prevent the plant from making a transition to the state q_k, by forcing it to stay at the original state q_i. Thus, disabling any transition σ at a given state q results in deletion of original transition and the appearance of self-loop $\delta(q, \sigma) = q$ with the occurrence probability of σ from the state q remaining unchanged in the supervised and unsupervised plants.

Definition 2.7 (Controllable transitions): For a given plant, transitions that can be disabled in the sense of Definition 2.6 are defined to be *controllable* transitions. The set of controllable transitions in a plant is denoted by $\mathscr{C}$. *Note that controllability is state-based.*

It follows that plant models can be specified by the following sextuplet:

$$G = (Q, \Sigma, \delta, \widetilde{\Pi}, \chi, \mathscr{C}) \tag{7}$$

2.2 *Formal language measure for terminating plants*

The formal language measure is first defined for terminating plants (Garg 1992b) with sub-stochastic event generation probabilities, i.e. the event generation probabilities at each state summing to strictly less than unity. In general, the marked language $L_m(q_i)$ consists of both good and bad event strings that, starting from the initial state q_i, lead to Q_m^+ and Q_m^-, respectively. Any event string belonging to the language $L^0(q_i) = L(q_i) - L_m(q_i)$ leads to one of the non-marked states belonging to $Q - Q_m$ and L^0 does not contain any one of the good or bad strings. Based on the equivalence classes defined in the Myhill–Nerode Theorem (Hopcroft, Motwani, and Ullman 2001), the regular languages $L(q_i)$ and $L_m(q_i)$ can be expressed as

$$L(q_i) = \bigcup_{q_k \in Q} L_{i,k} \tag{8}$$

$$L_m(q_i) = \bigcup_{q_k \in Q_m} L_{i,k} = L_m^+ \cup L_m^- \tag{9}$$

where the sublanguage $L_{i,k} \subseteq L(q_i)$ having the initial state q_i is uniquely labelled by the terminal state q_k, $k \in \mathcal{I}_Q$ and $L_{i,j} \cap L_{i,k} = \emptyset \ \forall j \neq k$; and $L_m^+ \equiv \bigcup_{q_k \in Q_m^+} L_{i,k}$ and $L_m^- \equiv \bigcup_{q_k \in Q_m^-} L_{i,k}$ are good and bad sublanguages of $L_m(q_i)$, respectively. Then, $L^0 = \bigcup_{q_k \notin Q_m} L_{i,k}$ and $L(q_i) = L^0 \cup L_m^+ \cup L_m^-$.

A signed real measure $\mu^i : 2^{L(q_i)} \to \mathbb{R} \equiv (-\infty, +\infty)$ is constructed on the σ-algebra $2^{L(q_i)}$ for any $i \in \mathcal{I}_Q$ and interested readers are referred to Ray (2005) and Ray et al. (2005) for the details of measure-theoretic definitions and results. With the choice of this σ-algebra, every singleton set made of an event string $s \in L(q_i)$ is a measurable set. By Hahn Decomposition Theorem (Rudin 1988), each of these measurable sets qualifies itself to have a numerical value based on the above state-based decomposition of $L(q_i)$ into L^0(null), L^+(positive) and L^-(negative) sublanguages.

Definition 2.8: Let $\omega \in L(q_i, q_j) \subseteq 2^{L(q_i)}$. The signed real measure μ^i of every singleton string set ω is defined as

$$\mu^i(\{\omega\}) = \tilde{\pi}(q_i, \omega)\chi(q_j) \tag{10}$$

The signed real measure of a sublanguage $L_{i,j} \subseteq L(q_i)$ is defined as

$$\mu_{i,j} = \mu^i(L(q_i, q_j)) = \left(\sum_{\omega \in L(q_i, q_j)} \tilde{\pi}(q_i, \omega) \right) \chi_j \tag{11}$$

Therefore, the signed real measure of the language of a DFSA G_i initialised at $q_i \in Q$, is defined as

$$\mu_i = \mu^i(L(q_i)) = \sum_{j \in \mathcal{I}_Q} \mu^i(L_{i,j}) \tag{12}$$

It is shown in Ray (2005) and Ray et al. (2005) that the language measure in Equation (12) can be expressed as

$$\mu_i = \sum_{j \in \mathcal{I}_Q} \pi_{ij}\mu_j + \chi_i \tag{13}$$

The language measure vector, denoted as $\mu = [\mu_1 \ \mu_2 \ \ldots \ \mu_n]^T$, is called the μ-vector. In a vector form, Equation (13) becomes

$$\boldsymbol{\mu} = \Pi\boldsymbol{\mu} + \boldsymbol{\chi} \tag{14}$$

whose solution is given by

$$\boldsymbol{\mu} = (\mathbb{I} - \Pi)^{-1}\boldsymbol{\chi} \tag{15}$$

The inverse in Equation (15) exists for terminating plant models (Garg 1992a, 1992b) because Π is a contraction operator (Ray 2005a; Ray et al. 2005) due to the strict inequality $\sum_j \pi_{ij} < 1$. The residual $\theta_i = 1 - \sum_j \pi_{ij}$ is referred to as the termination probability for state $q_i \in Q$. We extend the analysis to non-terminating plants (Garg 1992a, 1992b) with stochastic transition probability matrices (i.e. with $\theta_i = 0$, $\forall q_i \in Q$) by renormalising the language measure (Chattopadhyay and Ray 2006) with respect to the uniform termination probability of a limiting terminating model as described further.

Let $\widetilde{\Pi}$ and Π be the stochastic event generation and transition probability matrices for a non-terminating plant $G_i = (Q, \Sigma, \delta, q_i, Q_m)$. We consider the terminating plant $G_i(\theta)$ with the same DFSA structure $(Q, \Sigma, \delta, q_i, Q_m)$ such that the event generation probability matrix is given by $(1-\theta)\widetilde{\Pi}$ with $\theta \in (0,1)$ implying that the state transition probability matrix is $(1-\theta)\Pi$.

Definition 2.9 (Renormalised measure)**:** The renormalised measure $\nu_\theta^i : 2^{L(q_i)} \to [-1,1]$ for the θ-parameterised terminating plant $G_i(\theta)$ is defined as:

$$\forall \omega \in L(q_i),\ \nu_\theta^i(\{\omega\}) = \theta \mu^i(\{\omega\}) \tag{16}$$

The corresponding matrix form is given by

$$\mathbf{v}_\theta = \theta\, \boldsymbol{\mu} = \theta\, [I - (1-\theta)\Pi]^{-1} \boldsymbol{\chi} \quad \text{with } \theta \in (0,1) \tag{17}$$

We note that the vector representation allows for the following notational simplification:

$$\nu_\theta^i(L(q_i)) = \mathbf{v}_\theta|_i \tag{18}$$

The renormalised measure for the non-terminating plant G_i is defined to be $\lim_{\theta\to 0^+} \nu_\theta^i$.

The following results are retained for the sake of completeness. Complete proofs can be found in Chattopadhyay (2006) and Chattopadhyay and Ray (2006b).

Proposition 2.1: *The limiting measure vector* $\mathbf{v}_0 \triangleq \lim_{\theta\to 0^+} \mathbf{v}_\theta$ *exists and* $\|\mathbf{v}_0\|_\infty \leq 1$.

Proposition 2.2: *Let* Π *be the stochastic transition matrix of a non-terminating PFSA* (Garg 1992a, 1992b). *Then, as the parameter* $\theta \to 0^+$, *the limiting measure vector is obtained as*: $\mathbf{v}_0 = \mathcal{C}(\Pi)\boldsymbol{\chi}$ *where the matrix operator* $\mathcal{C} \triangleq \lim_{k\to\infty} \frac{1}{k}\sum_{j=0}^{k-1} \Pi^j$ *is the Cesaro limit* (Bapat and Raghavan 1997; Berman and Plemmons 1979) *of the stochastic transition matrix* Π.

Corollary 2.1 (to Proposition 2.2)**:** *The expression* $\mathcal{C}(\Pi)\mathbf{v}_\theta$ *is independent of* θ. *Specifically, the following identity holds for all* $\theta \in (0,1)$:

$$\mathcal{C}(\Pi)\mathbf{v}_\theta = \mathcal{C}(\Pi)\boldsymbol{\chi} \tag{19}$$

Notation 2.3: The linearly independent orthogonal set $\{v^i \in \mathbb{R}^{\text{Card}(Q)} : v^i_j = \delta_{ij}\}$ is denoted as $\mathcal{B}$ where δ_{ij} denotes the Krönecker delta function. We note that there is a one-to-one onto mapping between the states $q_i \in Q$ and the elements of $\mathcal{B}$, namely:

$$q_i \mapsto \alpha \iff \alpha_k = \begin{cases} 1 & \text{if } k = i \\ 0 & \text{otherwise} \end{cases} \tag{20}$$

Definition 2.10: For any non-zero vector $v \in \mathbb{R}^{\text{CARD}(Q)}$, the normalising function $\mathscr{N} : \mathbb{R}^{\text{CARD}(Q)} \backslash \mathbf{0} \to \mathbb{R}^{\text{CARD}(Q)}$ is defined as $\mathscr{N}(v) = \frac{v}{\sum_i v_i}$.

2.3 *The optimal supervision problem: formulation and solution*

A supervisor disables a subset of the set $\mathscr{C}$ of controllable transitions and hence there is a bijection between the set of all possible supervision policies and the power set $2^{\mathscr{C}}$. That is, there exist $2^{|\mathscr{C}|}$ possible supervisors and each supervisor is uniquely identifiable with a subset of $\mathscr{C}$ and the corresponding language measure $\mathbf{v}_\theta$ allows a quantitative comparison of different policies.

Definition 2.11: For an unsupervised plant $G = (Q, \Sigma, \delta, \widetilde{\Pi}, \chi, \mathscr{C})$, let $G^\dagger$ and $G^\ddagger$ be the supervised plants with sets of disabled transitions, $\mathscr{D}^\dagger \subseteq \mathscr{C}$ and $\mathscr{D}^\ddagger \subseteq \mathscr{C}$, respectively, whose measures are $\mathbf{v}^\dagger$ and $\mathbf{v}^\ddagger$. Then, the supervisor that disables $\mathscr{D}^\dagger$ is defined to be superior to the supervisor that disables $\mathscr{D}^\ddagger$ if $\mathbf{v}^\dagger \geqq_{\text{ELEMENTWISE}} \mathbf{v}^\ddagger$ and strictly superior if $\mathbf{v}^\dagger >_{\text{ELEMENTWISE}} \mathbf{v}^\ddagger$.

Definition 2.12 (Optimal supervision problem)**:** Given a (non-terminating) plant $G = (Q, \Sigma, \delta, \widetilde{\Pi}, \chi, \mathscr{C})$, the problem is to compute a supervisor that disables a subset $\mathscr{D}^\star \subseteq \mathscr{C}$, such that $\forall \mathscr{D}^\dagger \subseteq \mathscr{C}$, $\mathbf{v}^\star \geqq_{\text{ELEMENTWISE}} \mathbf{v}^\dagger$ where $\mathbf{v}^\star$ and $\mathbf{v}^\dagger$ are the measure vectors of the supervised plants $G^\star$ and $G^\dagger$ under $\mathscr{D}^\star$ and $\mathscr{D}^\dagger$, respectively.

Remark 2.2: The solution to the optimal supervision problem is obtained in Chattopadhyay and Ray (2007) and Chattopadhyay and Ray (2007b) by designing an optimal policy for a *terminating* plant (Garg 1992a, 1992b) with a substochastic transition probability matrix $(1-\theta)\widetilde{\Pi}$ with $\theta \in (0,1)$. To ensure that the computed optimal policy coincides with the one for $\theta = 0$, the suggested algorithm chooses a *small* value for θ in each iteration step of the design algorithm. However, choosing θ too small may cause numerical problems in convergence. Algorithm B.2 (Appendix B) computes the critical lower bound $\theta_\star$ (i.e. how small a θ is actually required). In conjunction with Algorithm B.2, the optimal supervision problem is solved by using of Algorithm B.1 for a generic *PFSA* as reported in Chattopadhyay (2007) and Chattopadhyay and Ray (2007b).

The following results in Proposition 2.3 are critical to development in the sequel and hence are presented here without proof. The complete proofs are available in Chattopadhyay (2007) and Chattopadhyay and Ray (2007b).

Proposition 2.3

(1) (*Monotonicity*) *Let* $\mathbf{v}^{[k]}$ *be the language measure vector computed in the kth iteration of Algorithm* B.1. *The measure vectors computed by the*

algorithm form an elementwise non-decreasing sequence, i.e. $\mathbf{v}^{[k+1]} \geqq_{\text{ELEMENTWISE}} \mathbf{v}^{[k]}\ \forall k$.

(2) (*Effectiveness*) *Algorithm* B.1 *is an effective procedure* (Hopcroft et al. 2001), *i.e. it is guaranteed to terminate.*

(3) (*Optimality*) *The supervision policy computed by Algorithm* B.1 *is optimal in the sense of Definition* 2.12.

(4) (*Uniqueness*) *Given an unsupervised plant* G, *the optimal supervisor* $G^\star$, *computed by Algorithm* B.1, *is unique in the sense that it is maximally permissive among all possible supervision policies with optimal performance. That is, if* $\mathscr{D}^\star$ *and* $\mathscr{D}^\dagger$ *are the disabled transition sets, and* $\mathbf{v}^\star$ *and* $\mathbf{v}^\dagger$ *are the language measure vectors for* $G^\star$ *and an arbitrarily supervised plant* $G^\dagger$, *respectively, then* $\mathbf{v}^\star \equiv_{\text{ELEMENTWISE}} \mathbf{v}^\dagger \Rightarrow \mathscr{D}^\star \subset \mathscr{D}^\dagger \subseteq \mathscr{C}$.

Definition 2.13: Following Remark 2.2, we note that Algorithm B.2 computes a lower bound for the critical termination probability for each iteration of Algorithm B.1 such that the disabling/enabling decisions for the terminating plant coincide with the given non-terminating model. We define

$$\theta_{\min} = \min_k \theta_\star^{[k]} \tag{21}$$

where $\theta_\star^{[k]}$ is the termination probability computed by Algorithm B.2 in the kth iteration of Algorithm B.1.

Definition 2.14: If G and $G^\star$ are the unsupervised and optimally supervised PFSA, respectively, then we denote the renormalised measure of the terminating plant $G^\star(\theta_{\min})$ as $\nu_\star^i : 2^{L(q_i)} \to [-1, 1]$ (Definition 2.9). Hence, in vector notation we have

$$\mathbf{v}_\star = \mathbf{v}_{\theta_{\min}} = \theta_{\min}[I - (1-\theta_{\min})\Pi^\star]^{-1}\boldsymbol{\chi} \tag{22}$$

where $\Pi^\star$ is the transition probability matrix of the supervised plant $G^\star$.

Remark 2.3: Referring to Algorithm B.1, it is noted that $\mathbf{v}_\star = \nu^{[k]}$ where K is the total number of iterations for Algorithm B.1.

2.4 *The partial observability model*

The observation model used in this article is defined by the so-called unobservability maps developed in Chattopadhyay and Ray (2007a) as a generalisation of natural projections in discrete event systems. It is important to mention that while some authors refer to unobservability as the case where no transitions are observable in the system, we use the terms 'unobservable' and 'partially observable' interchangeably in the sequel. The relevant concepts developed in Chattopadhyay and Ray (2007a) are enumerated in this section for the sake of completeness.

2.4.1 *Assumptions and notations*

We make two key assumptions:

- The unobservability situation in the model is specified by a bounded memory unobservability map $\mathfrak{p}$ which is available to the supervisor.
- Unobservable transitions are uncontrollable.

Definition 2.15: An unobservability map $\mathfrak{p} : Q \times \Sigma^\star \to \Sigma^\star$ for a given model $G = (Q, \Sigma, \delta, \widetilde{\Pi}, \boldsymbol{\chi}, \mathscr{C})$ is defined recursively as follows: $\forall q_i \in Q$, $\sigma_j \in \Sigma$ and $\sigma_j\omega \in L(q_i)$,

$$\mathfrak{p}(q_i, \sigma_j) = \begin{cases} \epsilon, & \text{if } \sigma_j \text{ is unobservable from } q_i \\ \sigma_j, & \text{otherwise} \end{cases} \tag{23a}$$

$$\mathfrak{p}(q_i, \sigma_j\omega) = \mathfrak{p}(q_i, \sigma_j)\mathfrak{p}(\delta(q_i, \sigma), \omega) \tag{23b}$$

We can indicate transitions to be unobservable in the graph for the automaton $G = (Q, \Sigma, \delta, \widetilde{\Pi}, \chi, \mathscr{C})$ as *unobservable* and this would suffice for a complete specification of the unobservability map acting on the plant. The assumption of bounded memory of the unobservability maps imply that although we may need to unfold the automaton graph to unambiguously indicate the unobservable transitions, there exists a finite unfolding that suffices for our purpose. Such unobservability maps were referred to as *regular* in Chattopadhyay and Ray (2007a).

Remark 2.4: The unobservability maps considered in this article are state-based as opposed to being event-based observability considered in Ramadge and Wonham (1987).

Definition 2.16: A string $\omega \in \Sigma^\star$ is called unobservable at the supervisory level if at least one of the events in ω is unobservable, i.e. $\mathfrak{p}(q_i, \omega) \neq \omega$ Similarly, a string $\omega \in \Sigma^\star$ is called completely unobservable if each of the events in ω is unobservable, i.e. $\mathfrak{p}(q_i, \omega) = \epsilon$. Also, if there are no unobservable strings, we denote the unobservability map $\mathfrak{p}$ as trivial.

The subsequent analysis requires the notion of the phantom automaton introduced in Chattopadhyay and Ray (2006a). The following definition is included for the sake of completion.

Definition 2.17: Given a model $G = (Q, \Sigma, \delta, \widetilde{\Pi}, \boldsymbol{\chi}, \mathscr{C})$ and an unobservability map $\mathfrak{p}$, the phantom automaton $\mathscr{P}(G) = (Q, \Sigma, \mathscr{P}(\delta), \mathscr{P}(\widetilde{\Pi}), \boldsymbol{\chi}, \mathscr{P}(\mathscr{C}))$ is defined as follows:

$$\mathscr{P}(\delta)(q_i,\sigma_j) = \begin{cases} \delta(q_i,\sigma_j) & \text{if } \mathfrak{p}(q_i,\sigma_j) = \epsilon \\ \text{Undefined} & \text{otherwise} \end{cases} \tag{24a}$$

$$\mathscr{P}(\widetilde{\Pi})(q_i,\sigma_j) = \begin{cases} \widetilde{\Pi}(q_i,\sigma_j) & \text{if } \mathfrak{p}(q_i,\sigma_j) = \epsilon \\ 0 & \text{otherwise} \end{cases} \tag{24b}$$

$$\mathscr{P}(\mathscr{C}) = \varnothing \tag{24c}$$

Remark 2.5: The phantom automata in the sense of Definition 2.17 is a finite-state machine description of the language of completely unobservable strings resulting from the unobservability map $\mathfrak{p}$ acting on the model $G = (Q, \Sigma, \delta, \widetilde{\Pi}, \boldsymbol{\chi}, \mathscr{C})$. Note that Equation (24c) is a consequence of the assumption that unobservable transitions are uncontrollable. Thus, no transition in the phantom automaton is controllable.

Algorithm B.3 (Appendix B) computes the transition probability matrix for the phantom automaton of a given plant G under a specified unobservability map $\mathfrak{p}$ by deleting all observable transitions from G.

2.4.2 *The Petri net observer*

For a given model $G = (Q, \Sigma, \delta, \widetilde{\Pi}, \boldsymbol{\chi}, \mathscr{C})$ and a non-trivial unobservability map $\mathfrak{p}$, it is, in general, impossible to pinpoint the current state from an observed event sequence at the supervisory level. However, it is possible to estimate the set of plausible states from a knowledge of the phantom automaton $\mathscr{P}(G)$.

Definition 2.18 (Instantaneous state description)**:** For a given plant $G_0 = (Q, \Sigma, \delta, \widetilde{\Pi}, \boldsymbol{\chi}, \mathscr{C})$ initialised at state $q_0 \in Q$ and a non-trivial unobservability map $\mathfrak{p}$, the instantaneous state description is defined to be the image of an observed event sequence $\omega \in \Sigma^\star$ under the map $\overline{Q} : \mathfrak{p}(L(G_0)) \longrightarrow 2^Q$ as follows:

$$\overline{Q}(\omega) = \left\{ q_j \in Q : \exists s \in \Sigma^\star \text{ s.t. } \delta(q_0, s) = q_j \bigwedge \mathfrak{p}(q_0, s) = \omega \right\}$$

Remark 2.6: Note that for a trivial unobservability map $\mathfrak{p}$ with $\forall \omega \in \Sigma^\star$, $\mathfrak{p}(\omega) = \omega$, we have $\overline{Q}(\omega) = \delta(q_0, \omega)$ where q_0 is the initial state of the plant.

The instantaneous state description $\overline{Q}(\omega)$ can be estimated on-line by constructing a Petri net observer with flush-out arcs (Moody and Antsaklis 1998; Gribaudo, Sereno, Horvath, and Bobbio 2001). The advantage of using a Petri net description is the compactness of representation and the simplicity of the on-line execution algorithm that we present next. Our preference of a Petri net description over a subset construction for finite-state machines is motivated by the following: the Petri net formalism is natural, due to its ability to model transitions of the type $q_1 \rightarrow |\begin{smallmatrix} \nearrow q_2 \\ \searrow q_3 \end{smallmatrix}$, which reflects the condition 'the plant can possibly be in states q_2 or q_3 after an observed transition from q_1'. One can avoid introducing an exponentially large number of 'combined states' of the form $[q_2, q_3]$ as involved in the subset construction and more importantly preserve the state description of the underlying plant. Flush-out arcs were introduced by Gribaudo et al. (2001) in the context of fluid stochastic Petri nets. We apply this notion to ordinary nets with similar meaning: a flush-out arc is connected to a labelled transition, which, on firing, removes a token from the input place (if the arc weight is one). Instantaneous descriptions can be computed on-line efficiently due to the following result:

Proposition 2.4

(1) *Algorithm B.4 has polynomial complexity.*
(2) *Once the Petri net observer has been computed offline, the current possible states for any observed sequence can be computed by executing Algorithm B.5 online.*

Proof: Proof is given in Chattopadhyay and Ray (2007a). □

3. Online implementation of measure-theoretic optimal control under perfect observation

This section devises an online implementation scheme for the language measure-theoretic optimal control algorithm which will be later extended to handle plants with non-trivial unobservability maps. Formally, a supervision policy S for a given plant $G = (Q, \Sigma, \delta, \widetilde{\Pi}, \chi, \mathscr{C})$ specifies the control in the terms of disabled controllable transitions at each state $q_i \in Q$, i.e. $S = (G, \phi)$ where

$$\phi : Q \longrightarrow \{0,1\}^{\text{Card}\,(\Sigma)} \tag{25}$$

The map ϕ is referred in the literature as the state feedback map (Ramadge and Wonham 1987) and it specifies the set of disabled transitions as follows: if at state $q_i \in Q$ and events σ_{i_1}, σ_{i_r} are disabled by the particular supervision policy, then $\phi(q_i)$ is a binary sequence on $\{0, 1\}$ of length equal to the cardinality of the event alphabet Σ such that

$$\phi(q_i) = \big[\, 0 \;\; \cdots \;\; \underset{\downarrow i_1\textbf{th element}}{1} \;\; \cdots \;\; 0 \;\; \cdots \;\; 0 \;\; \underset{\downarrow i_r\textbf{th element}}{1} \;\; \cdots \,\big]$$

Remark 3.1: If it is possible to partition the alphabet Σ as $\Sigma = \Sigma^c \bigsqcup \Sigma^{uc}$, where Σ^c is the set of controllable transitions and Σ^{uc} is the set of uncontrollable transitions, then it suffices to consider ϕ as a map $\phi : Q \rightarrow \{0, 1\}^{\text{Card}(\Sigma^c)}$. However, since we consider controllability to be state dependent (i.e. the possibility that an event is controllable if generated at a state q_i and uncontrollable if generated at some other state q_j), such a partitioning scheme is not feasible.

Under perfect observation, a computed supervisor (G, ϕ) responds to the report of a generated event as follows:

- The current state of the plant model is computed as $q_{\text{current}} = \delta(q_{\text{last}}, \sigma)$, where σ is the reported event and q_{last} is the state of the plant model before the event is reported.
- All events specified by $\phi(q_{\text{current}})$ is disabled.

Note that such an approach requires the supervisor to remember $\phi(q_i) \forall q_i \in Q$, which is equivalent to keeping in memory an $n \times m$ matrix, where n is the number of plant states and m is the cardinality of the event alphabet. We show that there is a alternative simpler implementation.

Lemma 3.1: *For a given finite-state plant* $G = (Q, \Sigma, \delta, \widetilde{\Pi}, \chi, \mathscr{C})$ *and the corresponding optimal language measure* $\boldsymbol{\nu}_\star$, *the pair* $(G, \boldsymbol{\nu}_\star)$ *completely specifies the optimal supervision policy.*

Proof: The optimal configuration $G^\star$ is characterised as follows (Chattopadhyay 2006; Chattopadhyay and Ray 2007a):

- if for states $q_i, q_j \in Q$, $\boldsymbol{\nu}_\star\big|_i > \boldsymbol{\nu}_\star\big|_j$, then all controllable transitions $q_i \underset{q_i}{\rightarrow} q_j$ are disabled.
- if for states $q_i, q_j \in Q$, $\boldsymbol{\nu}_\star\big|_i \leqq \boldsymbol{\nu}_\star\big|_j$, then all controllable transitions $q_i \underset{q_i}{\rightarrow} q_j$ are enabled.

It follows that if the supervisor has access to the unsupervised plant model G and the language measure vector $\boldsymbol{\nu}_\star$, then the optimal policy can be implemented by the following procedure:

(1) Compute the current state of the plant model as $q_{\text{current}} = \delta(q_{\text{last}}, \sigma)$, where σ is the reported event and q_{old} is the state of the plant model before the event is reported. Let $q_{\text{current}} = q_i$.
(2) Disable all controllable transitions $q_i \underset{\sigma_j}{\rightarrow} q_k$ if $\boldsymbol{\nu}_\star\big|_i > \boldsymbol{\nu}_\star\big|_k$ for all $q_k \in Q$.

This completes the proof. The procedure is summarised in Algorithm 3.1. □

The approach given in Lemma 3.1 is important from the perspective that it forms the intuitive basis for extending the optimal control algorithm derived under the assumption of perfect observation to situations where one or more transitions are unobservable at the supervisory level.

Algorithm 3.1: Online Implementation of Optimal Control

```
input  : G = (Q, Σ, δ, Π̃, χ, 𝒞) ,p, Initial state q0
output: Optimal Control Actions
begin
    Compute G^opt by G --(𝒜_O)--> G^opt;
    Set θ★★ = min θ★; /* Min. θ★ for all iterations */
    Set μ = μ^{G^opt_θ★★};
    Set q_current = q0;                /* initial state */
    while true do                      /* Infinite Loop */
        Observe event σj;  /* Perfect Observation */
        Compute q_current = δ(q_current, σj);
        for k = 1 to m do    /* m = Cardinality of Σ */
            Compute q_next = δ(q_current, σk);
            if (q_current, σk, q_next) ∈ 𝒞 then      /* If q_Test == qj then μ(q_Test) = μj */
                if μ(q_Test) ≧ μ(q_current) then     /* If q_current == qi then μ(q_current) = μi */
                    Disable σk;
                endif
            else
                Enable σk;
            endif
        endfor
    endw
end
```

4. Optimal control under non-trivial unobservability

This section makes use of the unobservability analysis presented in Section 2.4 to derive a modified online-implementable control algorithm for partially observable probabilistic finite-state plant models.

4.1 *The fraction net observer*

In Section 2.4, the notion of instantaneous description of was introduced as a map $\overline{Q} : p(L(G_i)) \longrightarrow 2^Q$ from the set of observed event traces to the power set of the state set Q, such that given an observed event trace ω, $\overline{Q}(\omega) \subseteq Q$ is the set of states that the underlying deterministic finite state plant can possibly occupy at the given instant. We constructed a Petri net observer (Algorithm B.4) and showed that the instantaneous description can be computed online with polynomial complexity. However, for a plant modelled by a probabilistic regular language, the knowledge of the event occurrence probabilities allows us to compute not only the set of possible current states (i.e. the instantaneous description) but also the probabilistic cost of ending up in each state in the instantaneous description. To achieve this objective, we modify the Petri net observer introduced in Section 2.4.2 by assigning (possibly) fractional weights computed as functions of the event occurrence probabilities to the input arcs. The output arcs are still given unity weights. In the sequel, the Petri net observer with possibly fractional arc weights is referred to as the fraction net observer (FNO). First we need to formalise the notation for the FNO.

Definition 4.1: Given a finite-state terminating plant model $G(\theta) = (Q, \Sigma, \delta, (1-\theta)\widetilde{\Pi}, \chi, \mathscr{C})$, and an unobservability map $\mathfrak{p}$, the FNO, denoted as $\mathscr{F}_{(G_\theta, \mathfrak{p})}$, is a labelled Petri net $(Q, \Sigma, A^{\mathcal{I}}, A^{\mathcal{O}}, w^{\mathcal{I}}, x^0)$ with fractional arc weights and possibly fractional markings, where Q is the set of places, Σ is the event label alphabet, $A^{\mathcal{I}} \subseteqq Q \times \Sigma \times Q$ and $A^{\mathcal{O}} \subseteqq Q \times \Sigma$ are the sets of input and output arcs, $w^{\mathcal{I}}$ is the input weight assignment function and $x^0 \in \mathcal{B}$ (Notation 2.3) is the initial marking. The output arcs are defined to have unity weights.

The algorithmic construction of an FNO is derived next. We assume that the Petri net observer has already been computed (by Algorithm B.4) with Q the set of places, Σ the set of transition labels, $A^{\mathcal{I}} \subseteqq Q \times \Sigma \times Q$ the set of input arcs and $A^{\mathcal{O}} \subseteqq Q \times \Sigma$ the set of output arcs.

Definition 4.2: The input weight assigning function $w^{\mathcal{I}}: A^{\mathcal{I}} \to (0, \infty)$ for the FNO is defined as

$$\forall q_i \in Q, \quad \forall \sigma_j \in \Sigma, \quad \forall q_k \in Q,$$
$$\delta(q_i, \sigma_j) = q_\ell \Longrightarrow w^{\mathcal{I}}(q_i, \sigma_j, q_k) = \sum_{\substack{\omega \in \Sigma^\star \text{ s.t.} \\ \delta^\star(q_\ell, \omega) = q_k \bigwedge p(q_\ell, \omega) = \epsilon}} (1-\theta)^{|\omega|} \tilde{\pi}(q_\ell, \omega)$$

where $\delta: Q \times \Sigma \to Q$ is the transition map of the underlying DFSA and $\mathfrak{p}$ is the given unobservability map and $\tilde{\pi}$ is the event cost (i.e. the occurrence probability) function (Ray 2005). It follows that the weight on an input arc from transition σ_j (having an output arc from place q_i) to place q_k is the sum of the total conditional probabilities of all completely unobservable paths by which the underlying plant can reach the state q_k from state q_ℓ where $q_\ell = \delta(q_i, \sigma_j)$.

Computation of the input arc weights for the FNO requires the notion of the phantom automaton (Definition 2.17). The computation of the arc weights for the FNO is summarised in Algorithm 4.1.

Proposition 4.1: *Given a Petri net observer* $(Q, \Sigma, A^{\mathcal{I}}, A^{\mathcal{O}})$, *the event occurrence probability matrix* $\widetilde{\pi}$ *and the transition probability matrix for the phantom automaton* $\mathscr{P}(\Pi)$, *Algorithm* 4.1 *computes the arc weights for the FNO as stated in Definition* 4.2.

Proof: Algorithm 4.1 employs the following identity to compute input arc weights:

$$\forall q_i \in Q, \;\; \forall \sigma_j \in \Sigma, \;\; \forall q_k \in Q,$$
$$w^{\mathcal{I}}(q_i, \sigma_j, q_k) = \begin{cases} \left[\mathbb{I} - (1-\theta)\mathscr{P}(\Pi)\right]^{-1}\Big|_{\ell k} & \text{if } (q_i, \sigma_j, q_k) \in A^{\mathcal{I}} \wedge \delta(q_i, \sigma_j) = q_\ell \\ 0, & \text{otherwise} \end{cases}$$

Algorithm 4.1: Computation of Arc Weights for FNO

```
input  : Petri Net Observer (Q, Σ, A^I, A^O), Event Occurrence
         probability Matrix π̃, 𝒫(Π)
output: w^I, w^O
begin
    /* Computing Weights for Input Arcs                */
    for i = 1 to n do
        for j = 1 to m do
            for k = 1 to n do
                if (q_i, σ_j, q_k) ∈ A^I then
                    Compute q_ℓ = δ(q_i, σ_j);
                    w^I(q_i, σ_j, q_k) = [I − 𝒫(Π)]^{-1}|_{ℓk};
                endif
            endfor
        endfor
    endfor
end
```

which follows from the following argument. Assume that for the given unobservability map $\mathfrak{p}$, $G^{\mathscr{P}}$ is the phantom automaton for the underlying plant G. We observe that the measure of the language of all strings initiating from state q_ℓ and terminating at state q_k in the phantom automaton $G^{\mathscr{P}}$ is given by $\left[\mathbb{I} - \mathscr{P}(\Pi)\right]^{-1}\big|_{\ell k}$. Since every string generated by the phantom automaton is completely unobservable (in the sense of Definition 2.17), we conclude

$$\left[\mathbb{I} - (1-\theta)\mathscr{P}(\Pi)\right]^{-1}\Big|_{\ell k} = \sum_{\substack{\omega \in \Sigma^\star \text{ s.t.} \\ \delta^\star(q_\ell, \omega) = q_k \bigwedge p(q_\ell, \omega) = \epsilon}} (1-\theta)^{|\omega|} \tilde{\pi}(q_\ell, \omega) \tag{26}$$

This completes the proof. □

In Section 2.4.2, we presented Algorithm B.5 to compute the instantaneous state description $\overline{Q}(\omega)$ online without referring to the transition probabilities. The approach consisted of firing all enabled transitions (in the Petri net observer) labelled by σ_j on observing the event σ_j in the underlying plant. The set of possible current states then consisted of all states which corresponded to places with one or more tokens. For the FNO, we use a slightly different approach which involves the computation of a set of event-indexed state transition matrices.

Definition 4.3: For an FNO $(Q, \Sigma, A^{\mathcal{I}}, A^{\mathcal{O}}, w^{\mathcal{I}}, x^0)$, the set of event-indexed state transition matrices $\mathbf{\Gamma} = \{\Gamma^{\sigma_j}: \sigma_j \in \Sigma\}$ is a set of m matrices each of dimension $n \times n$ (where m is the cardinality of the event alphabet Σ and n is the number of places), such that on observing event σ_j in the underlying plant, the updated marking $x^{[k+1]}$ for the FNO (due to firing of all enabled σ_j-labelled transitions in the net) can be obtained from the existing marking $x^{[k]}$ as follows:

$$x^{[k+1]} = x^{[k]}\Gamma^{\sigma_j} \tag{27}$$

The procedure for computing $\mathbf{\Gamma}$ is presented in Algorithm 4.2. Note that the only inputs to the

```
Algorithm 4.2: Derivation of Transition Matrices Γ^σj
   input : 𝒫(Π), δ, 𝔭
   output: Γ^σj ∀σj ∈ Σ
1  begin
2    for j ∈ {1, ··· , m} do      /* m = No. of events */
3      for i ∈ {1, ··· , n} do    /* n = No. of places */
4        if δ(qi, σj) is undefined OR 𝔭(qi, σj) = ε then
5          Set i^th row of Γ^j = [0, ··· ,0]^T;
6        else
7          Compute r = δ(qi, σj) ;
8          Set i^th row of Γ^j = r^th row of [𝕀 − 𝒫(Π)]^−1;
9        endif
10     endfor
11   endfor
12 end
```

algorithm are the transition matrix for the phantom automaton, the unobservability map $\mathfrak{p}$ and the transition map for the underlying plant model. The next proposition shows that the algorithm is correct.

Proposition 4.2: *Algorithm* 4.2 *correctly computes the set of event-indexed transition matrices* $\Gamma = \{\Gamma^{\sigma}{}_j: \sigma_j \in \Sigma\}$ *for a given FNO* $(Q, \Sigma, A^{\mathcal{I}}, w^{\mathcal{I}}, x^0)$ *in the sense stated in Definition* 4.3.

Proof: Let the current marking of the FNO specified as $(Q, \Sigma, A^{\mathcal{I}}, A^{\mathcal{O}}, w^{\mathcal{O}}, w^{\mathcal{I}})$ be denoted by $x^{[k]}$ where $x^{[k]} \in [0, \infty)^n$ with $n = \text{Card}(Q)$. Assume that event $\sigma_j \in \Sigma$ is observed in the underlying plant model. To obtain the updated marking of the FNO, we need to fire all transitions labelled by σ_j in the FNO. Since the graph of the FNO is identical with the graph of the Petri net observer constructed by Algorithm B.4, it follows that if $\delta(q_i, \sigma_j)$ is undefined or the event σ_j is unobservable from the state q_i in the underlying plant, then there is a flush-out arc to a transition labelled σ_j from the place q_i in the graph of the FNO. This implies that the content of place q_i will be flushed out and hence will not contribute to any place in the updated marking $x^{[k+1]}$, i.e.

$$x_i^{[k]} \Gamma_{i\ell}^{\sigma_j} = 0 \forall \quad i \in \{1, \ldots, n\} \tag{28}$$

implying that the ith column of the matrix Γ^{σ_j} is $[0, \ldots, 0]^T$. This justifies Line 5 of Algorithm 4.2. If σ_j is defined and observable from the state q_i in the underlying plant, then we note that the contents of the place q_i end up in all places $q_\ell \in Q$ such that there exists an input arc (q_i, σ_j, q_ℓ) in the FNO. Moreover, the contribution to the place q_ℓ coming from place q_i is weighted by $w^{\mathcal{I}}(q_i, \sigma_j, q_\ell)$. Denote this contribution by $c_{i\ell}$. Then we have

$$\begin{aligned} c_{i\ell} &= w^{\mathcal{I}}(q_i, \sigma_j, q_\ell) x_i^{[k]} \\ \Longrightarrow \sum_i c_{i\ell} &= \sum_i w^{\mathcal{I}}(q_i, \sigma_j, q_\ell) x_i^{[k]} \\ \Longrightarrow x_\ell^{[k+1]} &= \sum_i w^{\mathcal{I}}(q_i, \sigma_j, q_\ell) x_i^{[k]} \end{aligned} \tag{29}$$

Note that $\sum_i c_{i\ell} = x_\ell^{[k+1]}$ since contributions from all places to q_ℓ sum to the value of the updated marking in the place q_ℓ. Recalling from Proposition 4.1 that

$$w^{\mathcal{I}}(q_i, \sigma_j, q_\ell) = \left[\mathbb{I} - (1-\theta)\mathscr{P}(\Pi) \right]^{-1} \Big|_{r\ell} \tag{30}$$

where $q_r = \delta(q_i, \sigma_j)$ in the underlying plant, the result follows. □

Proposition 4.2 allows an alternate computation of the instantaneous state description. We assume that the initial state of the underlying plant is known and hence the initial marking for the FNO is assigned as follows:

$$x_i^{[0]} = \begin{cases} 1 & \text{if } q_i \text{ is the initial state} \\ 0 & \text{otherwise} \end{cases} \tag{31}$$

It is important to note that since the underlying plant is a DFSA having only one initial state, the initial marking of the FNO has only one place with value 1 and all remaining places are empty. It follows from Proposition 4.2 that for a given initial marking $x^{[0]}$ of the FNO, the marking after observing a string $\omega = \sigma_{r_1} \cdots \sigma_{r_k}$ where $\sigma_j \in \Sigma$ is obtained as:

$$x^{[k]} = x^{[0]} \prod_{j=r_1}^{j=r_k} \Gamma^{\sigma_j} \tag{32}$$

Referring to the notation for instantaneous description introduced in Definition 2.18, we have

$$\overline{Q}(\omega) = \left\{ q_i \in Q : x_i^{[|\omega|]} > 0 \right\} \tag{33}$$

Remark 4.1: We observe that to solve the state determinacy problem, we only need to know if the individual marking values are non-zero. The specific values of the entries in the marking $x^{[k]}$, however, allow us to estimate the cost of occupying individual states in the instantaneous description $\overline{Q}(\omega)$.

4.2 *State entanglement due to partial observability*

The markings of the FNO $\mathscr{F}_{(G_\theta, \mathfrak{p})}$ for the plant $G(\theta) = (Q, \Sigma, \delta, (1-\theta)\widetilde{\Pi}, \chi, \mathscr{C})$ in the case of perfect observation is of the following form:

$\forall k \in \mathbb{N}$, $x^{[k]} = [0 \cdots 0\ 1\ 0 \cdots 0]^T$, i.e. $x^{[k]} \in \mathcal{B}$ (Notation 2.3)

It follows that for a perfectly observable system, $\mathcal{B}$ is an enumeration of the state set Q in the sense $x_i^{[k]} = 1$ implies that the current state is $q_i \in Q$. Under a non-trivial unobservability map $\mathfrak{p}$, the set of all possible FNO markings proliferates and we can interpret $x^{[k]}$ after the kth observation instance as the current states of the observed dynamics. This follows from the fact that no previous knowledge beyond that of the current FNO marking $x^{[k]}$ is required to define the future evolution of $x^{[k]}$. The effect of partial observation can then be interpreted as adding new states to the model with each new state a linear combination of the underlying states enumerated in $\mathcal{B}$.

Drawing an analogy with the phenomenon of state entanglement in quantum mechanics, we refer to $\mathcal{B}$ as the set of *pure* states; while all other occupancy estimates that may appear are referred to as *mixed* or *entangled* states. Even for a finite state plant model, the cardinality of the set of all possible entangled states is not guaranteed to be finite.

Lemma 4.1: *Let $\mathscr{F}_{(G_\theta,\mathfrak{p})}$ with initial marking $x^{[0]} \in \mathcal{B}$ be the FNO for the underlying terminating plant $G(\theta) = (Q, \Sigma, \delta, (1-\theta)\widetilde{\Pi}, \chi, \mathscr{C})$ with uniform termination probability θ. Then for any observed string $\omega = \sigma_{r_1} \cdots \sigma_{r_s}$ of length $s \in \mathbb{N}$ with $\sigma_{r_j} \in \Sigma$ $\forall r_j \in \{1, \ldots, k\}$, the occupancy estimate $x^{[k]}$, after the occurrence of the* kth *observable transition, satisfies*:

$$x^{[k]} \in \left[0, \frac{1}{\theta}\right]^{\textsc{Card}(\Sigma)} \setminus \mathbf{0} \tag{34a}$$

Proof: Let the initial marking $x^{[0]} \in \mathcal{B}$ be given by

$$\begin{matrix} [0 \cdots & 1 & \cdots 0] \\ (i\text{th element}) & \uparrow & \end{matrix} \tag{35}$$

Elementwise non-negativity of $x^{[k]}$ for all $k \in \mathbb{N}$ follows from the fact that $x^{[0]} \in \mathcal{B}$ is elementwise non-negative and each Γ^σ is a non-negative matrix for all $\sigma \in \Sigma$. We also need to show that $x^{[k]}$ cannot be the zero vector. The argument is as follows: assume that if possible $x^{[\ell]}\Gamma^\sigma = \mathbf{0}$ where $x^{[\ell]} \neq \mathbf{0}$ and $\sigma \in \Sigma$ is the current observed event. It follows from the construction of the transition matrices that $\forall q_i \in Q, x_i^{[\ell]} \neq 0$ implies that either $\delta(q_i, \sigma)$ is undefined or $\mathfrak{p}(q_i, \sigma) = \varepsilon$. In either case, it is impossible to observe the event σ with the current occupancy estimate $x^{[\ell]}$, which is a contradiction. Finally, we need to prove the elementwise upper bound of $\frac{1}{\theta}$ on $x^{[k]}$. We note that that $x_j^{[k]}$ is the sum of the total conditional probabilities of all strings $u \in \Sigma^\star$ initiating from state $q_i \in Q$ (since $\forall j, x_j^{[0]} = \delta_{ij}$) that terminate on the state $q_j \in Q$ and satisfy

$$\mathfrak{p}(u) = \omega \tag{36}$$

It follows that $x_j^{[k]} \leqq x^{[0]}[\mathbb{I} - (1-\theta)\Pi]^{-1}\big|_j$ since the right-hand side is the sum of conditional probabilities of all strings that go to q_j from q_i, irrespective of the observability. Hence we conclude:

$$\|x^{[k]}\|_\infty \leqq \|x^{[0]}[\mathbb{I} - (1-\theta)\Pi]^{-1}\|_\infty \leqq 1 \times \frac{1}{\theta}$$

which completes the proof. □

Remark 4.2: It follows from Lemma 4.1 that the entangled states belong to a compact subset of $\mathbb{R}^{\textsc{Card}(Q)}$.

Definition 4.4 (Entangled state set)**:** For a given $G = (Q, \Sigma, \delta, \widetilde{\Pi}, \chi, \mathscr{C})$ and $\mathfrak{p}$, the entangled state set $Q_{\mathscr{F}} \subset \mathbb{R}^{\textsc{Card}}(Q) \setminus \mathbf{0}$ is the set of all possible markings of the FNO initiated at any of the pure states $x^{[0]} \in \mathcal{B}$.

4.3 *An illustrative example of state entanglement*

We consider the plant model as presented in the left-hand plate of Figure 2. The finite-state plant model with the unobservable transition (marked in red dashed) along with the constructed Petri net observer is shown in Figure 2. The event occurrence probabilities assumed are shown in Table 2 and the transition probability matrix $\mathbf{P}$ is shown in Table 3. Given $\theta = 0.01$, we apply Algorithm B.3 to obtain:

$$\left[\mathbb{I} - (1-\theta)\mathscr{P}(\Pi)\right]^{-1} = \begin{bmatrix} 1 & 0.2 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{37}$$

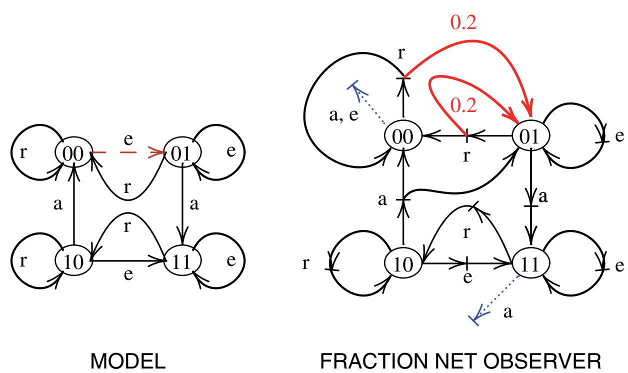

Figure 2. Underlying plant and Petri net observer. Available in colour online.

Table 2. Event occurrence probabilities.

	e	r	a
00	0.2	0.8	0
01	0.2	0.5	0.3
11	0.6	0.4	0
10	0.3	0.5	0.2

Table 3. Transition probability matrix Π.

	00	01	11	10
00	0.8	0.2	0	0
01	0.5	0.2	0.3	0
11	0	0	0.6	0.4
10	0.2	0	0.3	0.5

The arc weights are then computed for the FNO and the result is shown in the right-hand plate of Figure 2. Note that the arcs in bold red are the ones with fractional weights in this case and all other arc weights are unity. The set of transitions matrices $\mathbf{\Gamma}$ are now computed from Algorithm 4.2 as

$$\Gamma^e = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \mathbf{1} & 0 & 0 \\ 0 & 0 & \mathbf{1} & \mathbf{1} \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad \Gamma^r = \begin{bmatrix} \mathbf{1} & \mathbf{1} & 0 & 0 \\ \mathbf{0.2} & \mathbf{0.2} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{1} & \mathbf{1} \end{bmatrix}$$

$$\Gamma^a = \begin{bmatrix} 0 & 0 & 0 & \mathbf{1} \\ 0 & 0 & 0 & \mathbf{0.2} \\ 0 & \mathbf{1} & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

We consider three different observation sequences rr, re, ra assuming that the initial state in the underlying plant is 00 in each case (i.e. the initial marking of the FNO is given by $\alpha^0 = [1\ 0\ 0\ 0]^T$. The final markings (i.e. the entangled states) are given by

$$\alpha\Gamma^r\Gamma^r = \begin{bmatrix} 1.20 \\ 0.24 \\ 0 \\ 0 \end{bmatrix}, \alpha\Gamma^r\Gamma^e = \begin{bmatrix} 0 \\ 0.2 \\ 0 \\ 0 \end{bmatrix}, \alpha\Gamma^r\Gamma^a = \begin{bmatrix} 0 \\ 0 \\ 0.2 \\ 0 \end{bmatrix} \tag{38}$$

Note that while in the case of the Petri net observer, we could only say that $\overline{Q}(rr) = \{q_1, q_2\}$, for the FNO, we have an estimate of the cost of occupying each state (1.2 and 0.24, respectively, for the first case).

Next we consider a slightly modified underlying plant with the event occurrence probabilities as tabulated in Table 4. The modified plant (denoted as Model 2) is shown in the right-hand plate of Figure 3. The two models are simulated with the initial pure state set to [0 0 1 0] in each case. We note that the number of entangled states in the course of simulated operation more than doubles from 106 for Model 1 to 215 for Model 2 (Figure 4). In the simulation, entangled state vectors were distinguished with a tolerance of 10^{-10} on the max norm.

Table 4. Event occurrence probabilities for Model 2.

	e	r	a
00	0.2	0.79	0.01 ←
01	0.2	0.5	0.3
11	0.6	0.39	0.01 ←
10	0.3	0.5	0.2

4.4 *Maximisation of integrated instantaneous measure*

Definition 4.5 Instantaneous characteristic**:** Given a plant $G(\theta) = (Q, \Sigma, \delta, (1-\theta)\widetilde{\Pi}, \chi, \mathscr{C})$, the instantaneous characteristic $\hat{\chi}(t)$ is defined as a function of plant operation time $t \in [0, \infty)$ as follows:

$$\hat{\chi}(t) = \boldsymbol{\chi}\big|_i \tag{39}$$

where $q_i \in Q$ is the state occupied at time t

Definition 4.6 Instantaneous measure for perfectly observable plants**:** Given a plant $G(\theta) = (Q, \Sigma, \delta, (1-\theta), \widetilde{\Pi}, \chi, \mathscr{C})$, the instantaneous measure $(\hat{\nu}_\theta(t))$ is defined as a function of plant operation time $t \in [0, \infty)$ as follows:

$$\hat{\nu}_\theta(t) = \langle \alpha(t), \mathbf{v}_\theta \rangle \tag{40}$$

where $\alpha \in \mathcal{B}$ corresponds to the state that G is observed to occupy at time t (refer to Equation (20)) and $\mathbf{v}_\theta$ is the renormalised language measure vector for the underlying plant G with uniform termination probability θ.

Next, we show that the optimal control algorithms presented in Section 3, for perfectly observable

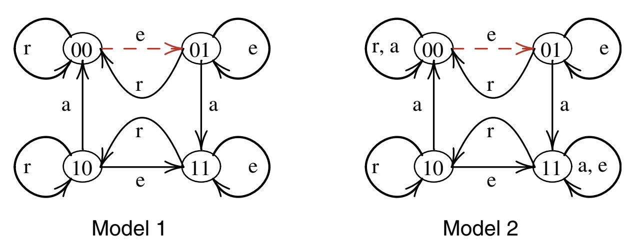

Figure 3. Underlying models to illustrate effect of unobservability on the cardinality of the entangled state set.

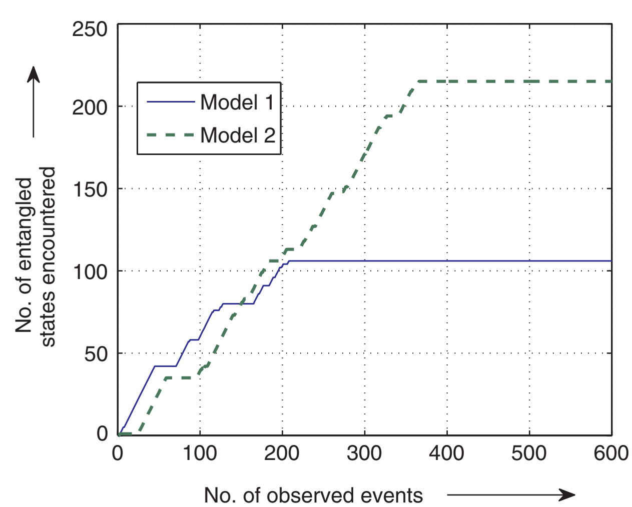

Figure 4. Total number of distinct entangled states encountered as a function of the number of observation ticks, i.e. the number of observed events.

situations, can be interpreted as maximising the expectation of the time-integrated instantaneous measure for the finite-state plant model under consideration (Figure 5).

Proposition 4.3: For the unsupervised plant $G = (Q, \Sigma, \delta, \widetilde{\Pi}, \chi, \mathscr{C})$ with all transitions observable at the supervisory level, let $G^\star$ be the optimally supervised plant and $G^\#$ be obtained by arbitrarily disabling controllable transitions. Denoting the instantaneous measures for $G^\star$ and $G^\#$ by $\hat{\nu}^\star_\theta(t)$ and $\hat{\nu}^\#_\theta(t)$ for some uniform termination probability $\theta \in (0, 1)$ respectively, we have

$$\mathbf{E}\left(\int_0^t \hat{\nu}^\star_\theta(\tau)\mathrm{d}\tau\right) \geqq \mathbf{E}\left(\int_0^t \hat{\nu}^\#_\theta(\tau)\mathrm{d}\tau\right) \forall t \in [0, \infty), \forall \theta \in (0, 1) \tag{41}$$

where t is the plant operation time and $\mathbf{E}(\cdot)$ denotes the expected value of the expression within braces.

Proof: Assume that the stochastic transition probability matrix for an arbitrary finite-state plant model be denoted by Π and denote the Cesaro limit as: $\mathcal{C} = \lim_{k\to\infty} \frac{1}{k}\sum_{j=0}^{k-1} \Pi^j$. Denoting the final stable state probability vector as p^i, where the plant is assumed to initiate operation in state q_i, we claim that $p^i_j = \mathcal{C}_{ij}$ which follows immediately from noting that if the initiating state is q_i then

$$(p^i)^T = \begin{bmatrix} 0\cdots & 0 & 1\cdots & 0 \end{bmatrix} \lim_{k\to\infty} \frac{1}{k}\sum_{j=0}^{k-1} \Pi^j$$
$$\uparrow i\text{th element}$$

i.e. $(p^i)^T$ is the ith row of $\mathcal{C}(\Pi)$. Hence, we have

$$\mathbf{E}\left(\int_0^t \hat{\chi}(\tau)\mathrm{d}\tau\right) = \int_0^t \mathbf{E}(\hat{\chi}(\tau))\mathrm{d}\tau = t\langle p^i, \boldsymbol{\chi}\rangle = t\mathbf{v}_0\big|_i$$
$$(\text{Note} : \theta = 0)$$

where finite number of states guarantee that the expectation operator and the integral can be exchanged. Recalling that optimal supervision elementwise maximises the language measure vector $\mathbf{v}_0$, we conclude that

$$\mathbf{E}\left(\int_0^t \hat{\chi}^\star(\tau)\mathrm{d}\tau\right) \geqq \mathbf{E}\left(\int_0^t \hat{\chi}^\#(\tau)\mathrm{d}\tau\right) \forall t \in [0, \infty) \tag{42}$$

where the $\hat{\chi}(t)$ for the plant configurations $G^\star$ and $G^\#$ is denoted as $\hat{\chi}^\star$ and $\hat{\chi}^\#$, respectively. Noting that the construction of the Petri net observer (Algorithm B.4) implies that in the case of perfect observation, each transition leads to exactly one place, we conclude that the instantaneous measure is given by

$$\hat{\nu}_\theta(t) = \mathbf{v}_\theta\big|_i \text{ where the current state at time } t \text{ is } q_i \tag{43}$$

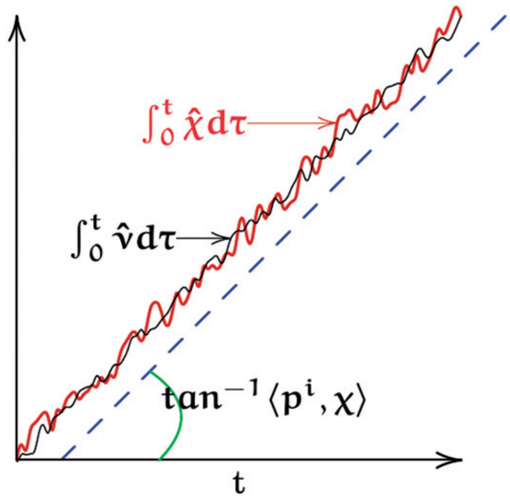

Figure 5. Time integrals of instantaneous measure and instantaneous characteristic vs. operation time.

Furthermore, we recall from Corollary 2.1 that

$$\mathcal{C}\mathbf{v}_\theta = \mathcal{C}\boldsymbol{\chi} \Longrightarrow \mathbf{E}(\hat{\nu}_\theta(t)) = \mathbf{E}(\hat{\chi}(t)) \forall t \in [0, \infty) \tag{44}$$

which leads to the following argument:

$$\begin{aligned}
&\mathbf{E}\left(\int_0^t \hat{\chi}^\star(\tau)\mathrm{d}\tau\right) \geqq \mathbf{E}\left(\int_0^t \hat{\chi}^\#(\tau)\mathrm{d}\tau\right) \forall t \in [0, \infty) \\
&\Longrightarrow \int_0^t \mathbf{E}\big(\hat{\chi}^\star(\tau)\big)\mathrm{d}\tau \geqq \int_0^t \mathbf{E}\big(\hat{\chi}^\#(\tau)\big)\mathrm{d}\tau \; \forall t \in [0, \infty) \\
&\Longrightarrow \int_0^t \mathbf{E}\big(\hat{\nu}^\star_\theta(\tau)\big)\mathrm{d}\tau \geqq \int_0^t \mathbf{E}\big(\hat{\nu}^\#_\theta(\tau)\big)\mathrm{d}\tau \; \forall t \in [0, \infty), \forall \theta \in (0, 1) \\
&\Longrightarrow \mathbf{E}\left(\int_0^t \hat{\nu}^\star_\theta(\tau)\mathrm{d}\tau\right) \geqq \mathbf{E}\left(\int_0^t \hat{\nu}^\#_\theta(\tau)\mathrm{d}\tau\right) \forall t \in [0, \infty), \forall \theta \in (0, 1)
\end{aligned}$$

This completes the proof. □

Next we formalise a procedure of implementing an optimal supervision policy from a knowledge of the optimal language measure vector for the underlying plant.

4.5 *The optimal control algorithm*

For any finite-state underlying plant $G(\theta) = (Q, \Sigma, \delta, (1-\theta)\widetilde{\Pi}, \chi, \mathscr{C})$ and a specified unobservability map $\mathfrak{p}$, it is possible to define a probabilistic transition system as a possibly infinite-state generalisation of PFSA which we denote as the entangled transition system corresponding to the underlying plant and the specified unobservability map. In defining the entangled transition system (Definition 4.7), we use a similar formalism as stated in Section 2.1, with the exception of dropping the last argument for controllability specification in Equation (7). Controllability needs to handled separately to address the issues of partial controllability arising as a result of partial observation.

Definition 4.7 (Entangled transition system)**:** For a given plant $G(\theta) = (Q, \Sigma, \delta, (1-\theta)\widetilde{\Pi}, \chi, \mathscr{C})$ and an unobservability map $\mathfrak{p}$, the entangled transition system $\mathscr{E}_{(G,\mathfrak{p})} = (Q_{\mathscr{F}}, \Sigma, \Delta, \tilde{\pi}_{\mathscr{E}}, \chi_{\mathscr{E}})$ is defined as:

(1) The transition map $\Delta: Q_{\mathscr{F}} \times \Sigma^\star \to Q_{\mathscr{F}}$ is defined as:

$$\forall \alpha \in Q_{\mathscr{F}},\ \Delta(\alpha, \omega) = \alpha \prod_{\sigma_1}^{\sigma_m} \Gamma^{\sigma_i} \text{ where } \omega = \sigma_1 \cdots \sigma_m$$

(2) The event generation probabilities $\tilde{\pi}_{\mathscr{E}}: Q_{\mathscr{F}} \times \Sigma^\star \to [0, 1]$ are specified as:

$$\tilde{\pi}_{\mathscr{E}}(\alpha, \sigma) = \sum_{i=1}^{i=\text{CARD}(Q)} (1-\theta)\mathcal{N}(\alpha_i)\tilde{\pi}(q_i, \sigma)$$

(3) The characteristic function $\chi_{\mathcal{E}}$: $Q_{\mathscr{F}} \to [-1, 1]$ is defined as: $\chi_{\mathcal{E}}(\alpha) = \langle \alpha, \boldsymbol{\chi} \rangle$

Remark 4.3: The definition of $\tilde{\pi}_{\mathscr{E}}$ is consistent in the sense:

$$\forall \alpha \in Q_{\mathscr{F}},\ \sum_{\sigma \in \Sigma} \tilde{\pi}_{\mathscr{E}}(\alpha, \sigma) == \sum_i \mathcal{N}(\alpha_i)(1-\theta) = 1-\theta$$

implying that if $Q_{\mathscr{F}}$ is finite then $\mathscr{E}_{(G,\mathfrak{p})}$ is a perfectly observable terminating model with uniform termination probability θ.

Proposition 4.4: *The renormalised language measure $\nu_\theta^{\mathscr{E}}(\alpha)$ for the state $\alpha \in Q_{\mathscr{F}}$ of the entangled transition system $\mathscr{E}_{(G,\mathfrak{p})} = (Q_{\mathscr{F}}, \Sigma, \Delta, \tilde{\pi}_{\mathscr{E}}, \chi_{\mathscr{E}})$ can be computed as follows*:

$$\nu_\theta^{\mathscr{E}}(\alpha) = \langle \alpha, \mathbf{v}_\theta \rangle \tag{45}$$

where $\mathbf{v}_\theta$ is the language measure vector for the underlying terminating plant $G(\theta) = (Q, \Sigma, \delta, (1-\theta)\widetilde{\Pi}, \chi, \mathscr{C})$ with uniform termination probability θ.

Proof: We first compute the measure of the pure states $\mathcal{B} \subset Q_{\mathscr{F}}$ of $\mathscr{E}_{(G,\mathfrak{p})} = (Q_{\mathscr{F}}, \Sigma, \Delta, \tilde{\pi}_{\mathscr{E}}, \chi_{\mathscr{E}})$ denoted by the vector $\mathbf{v}_\theta^{\mathscr{E}}$. Since every string generated by the phantom automaton is completely unobservable, it follows that the measure of the empty string ε from any state $\alpha \in \mathcal{B}$ is given by $\alpha[\mathbb{I} - (1-\theta)\mathfrak{p}(\Pi)]^{-1}\boldsymbol{\chi}$. Let α correspond to the state $q_i \in Q$ in the underlying plant. Then the measure of the set of all strings generated from $\alpha \in \mathcal{B}$ having at least one observable transition in the underlying plant is given by

$$\sum_j (1-\theta)\left[\mathbb{I} - (1-\theta)\mathscr{P}(\Pi)\right]^{-1}\left(\Pi - \mathscr{P}(\Pi)\right)\Big|_{ij}\{\mathbf{v}_\theta^{\mathscr{E}}\}_j \tag{46}$$

which is simply the measure of the set of all strings of the form $\omega_1\sigma\omega_2$ where $\mathfrak{p}(\omega_1\sigma\omega_2) = \sigma\mathfrak{p}(\omega_2)$. It therefore follows from the additivity of measures that

$$\begin{aligned}
\mathbf{v}_\theta^{\mathscr{E}} &= (1-\theta)\left[\mathbb{I} - (1-\theta)\mathscr{P}(\Pi)\right]^{-1}\left(\Pi - \mathscr{P}(\Pi)\right)\mathbf{v}_\theta^{\mathscr{E}} \\
&\quad + \left[\mathbb{I} - (1-\theta)\mathscr{P}(\Pi)\right]^{-1}\boldsymbol{\chi} \\
\Rightarrow \mathbf{v}_\theta^{\mathscr{E}} &= \left[\mathbb{I} - (1-\theta)\left[\mathbb{I} - (1-\theta)\mathscr{P}(\Pi)\right]^{-1}\left(\Pi - \mathscr{P}(\Pi)\right)\right]^{-1} \\
&\quad \times \left[\mathbb{I} - (1-\theta)\mathscr{P}(\Pi)\right]^{-1}\boldsymbol{\chi} \\
\Rightarrow \mathbf{v}_\theta^{\mathscr{E}} &= \left[\mathbb{I} - (1-\theta)\Pi\right]^{-1}\boldsymbol{\chi} = \mathbf{v}_\theta
\end{aligned} \tag{47}$$

which implies that for any pure state $\alpha \in \mathcal{B}$, we have $\nu_\theta^{\mathscr{E}}(\alpha) = \langle \alpha, \mathbf{v}_\theta \rangle$. The general result then follows from the following linear relation arising from the definitions of $\tilde{\pi}_{\mathscr{E}}$ and $\chi\mathscr{E}$:

$$\forall \alpha \in \mathcal{B},\quad \forall k \in \mathbb{R},\quad \nu_\theta^{\mathscr{E}}(k\alpha) = k\nu_\theta^{\mathscr{E}}(\alpha) \tag{48}$$

This completes the proof. □

Definition 4.8 (Instantaneous characteristic for entangled transition systems)**:** Given an underlying plant $G(\theta) = (Q, \Sigma, \delta, (1-\theta)\widetilde{\Pi}, \chi, \mathscr{C})$ and an unobservability map $\mathfrak{p}$, the instantaneous characteristic $\hat{\chi}_{\mathscr{E}}(t)$ for the corresponding entangled transition system $\mathscr{E}_{(G,\mathfrak{p})} = (Q_{\mathscr{F}}, \Sigma, \Delta, \tilde{\pi}_{\mathscr{E}}, \chi_{\mathscr{E}})$ is defined as a function of plant operation time $t \in [0, \infty)$ as follows:

$$\hat{\chi}_{\mathscr{E}}(t) = \langle \alpha(t), \boldsymbol{\chi} \rangle \tag{49}$$

where $\alpha(t)$ is the entangled state occupied at time t

Definition 4.9 (Instantaneous measure for partially observable plants)**:** Given an underlying plant $G(\theta) = (Q, \Sigma, \delta, (1-\theta)\widetilde{\Pi}, \chi, \mathscr{C})$ and an unobservability map $\mathfrak{p}$, the instantaneous measure $(\hat{\nu}_\theta(t))$ is defined as a function of plant operation time $t \in [0, \infty)$ as follows:

$$\hat{\nu}_\theta(t) = \langle \alpha(t), \mathbf{v}_\theta^{\mathscr{E}} \rangle \tag{50}$$

where $\alpha \in Q_{\mathcal{E}}$ is the entangled state at time t and $\mathbf{v}_\theta^{\mathscr{E}}$ is the renormalised language measure vector for the corresponding entangled transition system $\mathscr{E}_{(G,\mathfrak{p})} = (Q_{\mathscr{F}}, \Sigma, \Delta, \tilde{\pi}_{\mathscr{E}}, \chi_{\mathscr{E}})$.

Corollary 4.1 (Corollary to Proposition 4.4)**:** *For a given plant $G(\theta) = (Q, \Sigma, \delta, (1-\theta)\widetilde{\Pi}, \chi, \mathscr{C})$ and an unobservability map $\mathfrak{p}$, the instantaneous measure $\hat{\nu}_\theta : [0, \infty) \to [-1, 1]$ is given by*

$$\hat{\nu}_\theta(t) = \langle \alpha(t), \mathbf{v}_\theta \rangle \tag{51}$$

where $\alpha(t)$ is the current state of the entangled transition system $\mathscr{E}_{(G,\mathfrak{p})} = (Q_{\mathscr{F}}, \Sigma, \Delta, \tilde{\pi}_{\mathscr{E}}, \chi_{\mathscr{E}})$ at time t and $\mathbf{v}_\theta$ is the language measure vector for the underlying plant G.

Proof: Follows from Definitions 4.9, 4.7 and Proposition 4.4. □

Proposition 4.4 has a crucial consequence. It follows that elementwise maximisation of the measure vector $\mathbf{v}_\theta$ for the underlying plant automatically maximises the measures of each of the entangled states irrespective of the particular unobservability map $\mathfrak{p}$. This allows us to directly formulate the optimal supervision policy for cases where the cardinality of the entangled state set is finite. However, before we embark upon the construction of such policies, we need to address the controllability issues arising due to state entanglement. We note that for a given entangled state $\alpha \in Q_{\mathscr{F}} \backslash \mathcal{B}$, an event $\sigma \in \Sigma$ may be controllable from some but not all of the states $q_i \in Q$ that satisfy $\alpha_i > 0$. Thus, the notion of controllability introduced in Definition 2.7 needs to be generalised and *disabling of a transition $\sigma \in \Sigma$ from an entangled state can still change the current state*. We formalise the analysis by defining a set of event-indexed disabled transition matrices by suitably modifying Γ^σ as follows.

Definition 4.10: For a given plant $G = (Q, \Sigma, \delta, \widetilde{\Pi}, \chi, \mathscr{C})$, the event indexed disabled transition matrices $\Gamma^\sigma_{\mathcal{D}}$ is defined as

$$\Gamma^\sigma_{\mathcal{D}}\big|_{ij} = \begin{cases} \delta_{ij}, & \text{if } \sigma \text{ is controllable at } q_i \text{ and } \mathfrak{p}(q_i, \sigma) = \sigma \\ \Gamma^\sigma_{ij}, & \text{otherwise} \end{cases}$$

Evolution of the current entangled state α to α' due to the firing of the disabled transition $\sigma \in \Sigma$ is then computed as:

$$\alpha' = \alpha \Gamma^\sigma_{\mathcal{D}} \tag{52}$$

Remark 4.4: If an event $\sigma \in \Sigma$ is uncontrollable at every state $q_i \in Q$, then $\Gamma^\sigma_{\mathcal{D}} = \Gamma^\sigma$. On the other hand, if event σ is always controllable (and hence by our assumption always observable), then we have $\Gamma^\sigma_{\mathcal{D}} = \mathbb{I}$. In general, we have $\Gamma^\sigma_{\mathcal{D}} \neq \Gamma^\sigma \neq \mathbb{I}$.

Proposition 4.3 shows that optimal supervision in the case of perfect observation yields a policy that maximises the time-integral of the instantaneous measure. We now outline a procedure (Algorithm 4.3) to maximise $\int_0^t \hat{\nu}_\theta(\tau)\mathrm{d}\tau$ when the underlying plant has a non-trivial unobservability map.

Lemma 4.2: *Let the following condition be satisfied for a plant $G = (Q, \Sigma, \delta, \widetilde{\Pi}, \chi, \mathscr{C})$ and an unobservability map $\mathfrak{p}$:*

$$\textsc{Card}\,(Q_{\mathscr{F}}) < \infty \tag{53}$$

Algorithm 4.3: Optimal Control under Partial Observation (Preliminary Procedure For Illustration)

input : $G = (Q, \Sigma, \delta, \widetilde{\Pi}, \chi, \mathscr{C})$,$\mathfrak{p}$, Initial State q_0 for G
begin
  while *true* **do** /* Infinite Loop */
    Compute the optimal measure vector $\mathbf{v}_\star$ for G
    Set the current entangled state to $\alpha = q_0[\mathbb{I} - \mathscr{P}(\Pi)]^{-1}$
    if *current entangled state is* α **then**
      for $\sigma \in \Sigma$ **do**
        if $\langle \alpha\Gamma^\sigma, \mathbf{v}_\star\rangle < \langle \alpha\Gamma^\sigma_{\mathscr{D}}, \mathbf{v}_\star\rangle$ **then**
          Disable σ
        endif
      endfor
    endif
    Observe next event $\sigma \in \Sigma$
    if σ *is enabled* **then**
      Update the entangled state to $\alpha\Gamma^\sigma$
    else
      Update the entangled state to $\alpha\Gamma^\sigma_{\mathscr{D}}$
    endif
  endw
end

Then the control actions generated by Algorithm 4.3 is optimal in the sense that

$$\mathbf{E}\left(\int_0^t \hat{\nu}^\star_\theta(\tau)\mathrm{d}\tau\right) \geqq \mathbf{E}\left(\int_0^t \hat{\nu}^\#_\theta(\tau)\mathrm{d}\tau\right) \ \forall t \in [0,\infty), \forall \theta \in (0,1) \tag{54}$$

where $\hat{\nu}^\star_\theta(t)$ and $\hat{\nu}^\#_\theta(t)$ are the instantaneous measures at time t for control actions generated by Algorithm 4.3 and an arbitrary policy respectively.

Proof

Case 1: First, we consider the case where the following condition is true:

$$\forall \sigma \in \Sigma,\ \left(\Gamma^\sigma_{\mathscr{D}} = \Gamma^\sigma\right) \bigvee \left(\forall \alpha \in Q_{\mathscr{F}},\ \alpha\Gamma^\sigma_{\mathscr{D}} = \alpha\right) \tag{55}$$

which can be paraphrased as follows:

> Each event is either uncontrollable at every state $q \in Q$ in the underlying plant $G(\theta) = (Q, \Sigma, \delta, (1-\theta)\widetilde{\Pi}, \chi, \mathscr{C})$ or is controllable at every state at which it is observable.

We note that the entangled transition system qualifies as a perfectly observable probabilistic finite state machine (Remark 4.3) since the unobservability effects have been eliminated by introducing the entangled states. If the above condition stated in Equation (53) is true, then no generalisation of the notion of event controllability in $\mathscr{E}_{(G,\mathfrak{p})} = (Q_{\mathscr{F}}, \Sigma, \Delta, \tilde{\pi}_{\mathscr{E}}, \chi_{\mathscr{E}})$ is required (Definition 4.10). Under this assumption, the claim of the lemma then follows from Lemma 3.1 by noting that Algorithm 4.3 under the above assumption reduces to the procedure stated in Algorithm 3.1 when we view the entangled system as a perfectly observable PFSA model.

Case 2: Next we consider the general scenario where the condition in Equation (53) is relaxed. We note that the key to the online implementation result in stated Lemma 3.1 is the monotonicity lemma proved in Chattopadhyay and Ray (2007b), which states that for any given terminating plant $G(\theta) = (Q, \Sigma, \delta, (1-\theta)\widetilde{\Pi}, \chi, \mathscr{C})$ with uniform termination probability θ, the following iteration sequence elementwise increases the measure vector monotonically:

(1) Compute $\mathbf{v}_\theta$
(2) If $\mathbf{v}_\theta|_i < \mathbf{v}_\theta|_j$, then disable all events $q_i \xrightarrow{\sigma} q_j$, otherwise enable all events $q_i \xrightarrow{\sigma} q_j$
(3) Go to step 1.

The proof of the monotonicity lemma (Chattopadhyay and Ray 2007b) assumes that 'disabling' $q_i \xrightarrow{\sigma} q_j$ replaces it with a self loop at state q_i labelled σ with the same generation probability, i.e. $\widetilde{\Pi}(q_i, \sigma)$ remains unchanged. Now if there exists a $\sigma \in \Sigma$ with $\Gamma^\sigma_{\mathscr{D}} \neq \mathbb{I}$, then we need to consider the fact that on disabling σ, the new transition is no longer a self loop, but ends up in some other state $q_k \in Q$. Under this more general situation, we claim that Algorithm 4.3 is true; or in other words, we claim that the following procedure elementwise increases the measure vector monotonically:

(1) Compute $\mathbf{v}_\theta$
(2) Let $q_i \xrightarrow{\sigma} q_j$ (if enabled) and $q_i \xrightarrow{\sigma} q_k$ (if disabled)
(3) If $\mathbf{v}_\theta|_j < \mathbf{v}_\theta|_k$, then disable $q_i \xrightarrow{\sigma} q_j$, otherwise enable $q_i \xrightarrow{\sigma} q_j$
(4) Go to step 1

which is guaranteed by Proposition A.1 in Appendix A. Convergence of this iterative process and the optimality of the resulting supervision policy in the sense of Definition 2.12 can be worked out exactly on similar lines as shown in Chattopadhyay and Ray (2007b). This completes the proof. □

In order to extend the result of Lemma 4.2 to the general case where the cardinality of the entangled state set can be infinite, we need to introduce a sequence of finite-state approximations to the potentially infinite-state entangled transition system. This would allow us to work out the above extension as a natural consequence of continuity arguments. The finite-state approximations are parameterised by $\eta \in (0, 1]$ which approaches 0 from above as we derive closer and closer approximations. The formal definition of such an η-quantised approximation for $\mathscr{E}_{(G,\mathfrak{p})} = (Q_{\mathscr{F}}, \Sigma, \Delta, \tilde{\pi}_{\mathscr{E}}, \chi_{\mathscr{E}})$ is stated next.

Definition 4.11 (η-quantised approximation)**:** For a plant $G(\theta) = (Q, \Sigma, \delta, (1-\theta)\widetilde{\Pi}, \chi, \mathscr{C})$, an unobservability map $\mathfrak{p}$ and a given $\eta \in (0, 1]$, a probabilistic finite-state machine $\mathscr{E}^\eta_{(G,\mathfrak{p})} = (Q^\eta_{\mathscr{F}}, \Sigma, \Delta^\eta, \tilde{\pi}_{\mathscr{E}}, \chi_{\mathscr{E}})$ qualifies as an η-quantised approximation of the corresponding entangled transition system $\mathscr{E}_{(G,\mathfrak{p})} = (Q_{\mathscr{F}}, \Sigma, \Delta, \tilde{\pi}_{\mathscr{E}}, \chi_{\mathscr{E}})$ if

$$\Delta^\eta(\alpha, \omega) = \zeta_\eta(\Delta(\alpha, \omega)) \tag{56}$$

where $\zeta_\eta : [0, \frac{1}{\theta}]^{\mathrm{CARD}(Q)} \to Q^\eta_{\mathscr{F}}$ is a quantisation map satisfying:

$$\mathrm{CARD}\,(Q^\eta_{\mathscr{F}}) < \infty \tag{57a}$$

$$\forall \alpha \in \mathcal{B}, \quad \zeta_\eta(\alpha) = \alpha \tag{57b}$$

$$\forall \alpha \in Q_{\mathscr{F}}, \; \|\zeta_\eta(\alpha) - \alpha\|_\infty \leqq \eta \tag{57c}$$

where $\|\cdot\|_\infty$ is the standard max norm. Furthermore, we denote the language measure of the state $\alpha \in Q^\eta_{\mathscr{F}}$ as $\nu^\eta_\theta(\alpha)$ and the measure vector for the pure states $\alpha \in \mathcal{B}$ is denoted as $\mathbf{v}^\eta_\theta$.

We note the following:

(1) For a given $\eta \in (0, 1]$, there may exist uncountably infinite number of distinct probabilistic finite-state machines that qualify as an η-quantised approximation to $\mathscr{E}_{(G,\mathfrak{p})} = (Q_{\mathscr{F}}, \Sigma, \Delta, \tilde{\pi}_{\mathscr{E}}, \chi_{\mathscr{E}})$, i.e. the approximation is not unique.
(2) $\lim_{\eta \to 0^+} \mathscr{E}^\eta_{(G,\mathfrak{p})} = \mathscr{E}_{(G,\mathfrak{p})}$.
(3) The compactness of $[0, \frac{1}{\theta}]^{\mathrm{CARD}(Q)}$ is crucial in the definition.
(4) The set of pure states of $\mathscr{E}_{(G,\mathfrak{p})} = (Q_{\mathscr{F}}, \Sigma, \Delta, \tilde{\pi}_{\mathscr{E}}, \chi_{\mathscr{E}})$ is a subset of $Q^\eta_{\mathscr{F}}$, i.e. $\mathcal{B} \subset Q^\eta_{\mathscr{F}}$.
(5) The measure of an arbitrary state $\alpha \in Q^\eta_{\mathscr{F}}$ is given by $\langle \alpha, \mathbf{v}^\eta_\theta \rangle$.

Lemma 4.3: *The language measure vector $\mathbf{v}^\eta_\theta$ for the set of pure states $\mathcal{B}$ for any η-quantised approximation of $\mathscr{E}_{(G,\mathfrak{p})} = (Q_{\mathscr{F}}, \Sigma, \Delta, \tilde{\pi}_{\mathscr{E}}, \chi_{\mathscr{E}})$ is upper semi-continuous w.r.t. η at $\eta = 0$.*

Proof: Let M_k be a sequence in $\mathbb{R}^{\mathrm{CARD}}(Q)$ such that $M_k|_i$ denotes the measure of the expected state after $k \in \mathbb{N} \cup \{0\}$ observations for the chosen η-quantised approximation $\mathscr{E}^\eta_{(G,\mathfrak{p})}$ beginning from the pure state corresponding to $q_i \in Q$. We note that

$$\sum_{k=0}^{\infty} M_k = \mathbf{v}^\eta_\theta \tag{58}$$

Furthermore, we have

$$M_0 = A\chi^{[0]} \tag{59}$$

where $A = \theta[\mathbb{I} - (1-\theta)\mathscr{P}(\Pi)]^{-1}$ and $\chi^{[0]}$ is the perturbation of the characteristic vector χ due to quantisation, implying that

$$\|M_0 - A\chi\|_\infty \leqq \|A\|_\infty \eta \tag{60}$$

Denoting $B = \left[\mathbb{I} - (1-\theta)\mathscr{P}(\Pi)\right]^{-1}(1-\theta)\big(\Pi - \mathscr{P}(\Pi)\big)$, we note that

$$M_k = B^k \chi^{[k]} \Longrightarrow \|M_k\|_\infty \leqq \|B\|^k_\infty \|A\|_\infty \eta \tag{61}$$

It then follows that we have:

$$\left\|\mathbf{v}_\theta^\eta - \mathbf{v}_\theta\right\|_\infty \leqq \left(\sum_k \|B\|_\infty^k\right)\|A\|_\infty \eta \quad (62)$$

We claim that the following bounds are satisfied:

(1) $\|A\| \leqq 1$
(2) $\sum_k \|B\|_\infty^k \leqq \frac{1}{\theta}$

For the first claim, we note

$$\begin{aligned}\left[\mathbb{I} - (1-\theta)\mathscr{P}(\Pi)\right]^{-1} &= \sum_{k=0}^\infty \theta(1-\theta)^k \mathscr{P}(\Pi)^k \\ &\leqq_{\text{ELEMENTWISE}} \sum_{k=0}^\infty \theta(1-\theta)^k \Pi^k \\ &= \theta\left[\mathbb{I} - (1-\theta)\Pi\right]^{-1}\end{aligned}$$

The result then follows by noting that $\theta[\mathbb{I} - (1-\theta)\Pi]^{-1}$ is a stochastic matrix for all $\theta \in (0,1)$. For the second claim, denoting $\boldsymbol{e} = [1 \dots 1]^T$, we conclude from stochasticity of Π:

$$\begin{aligned}\left(\Pi - \mathscr{P}(\Pi)\right)\boldsymbol{e} &= \left[\mathbb{I} - \mathscr{P}(\Pi)\right]\boldsymbol{e} \\ &= \left[\mathbb{I} - (1-\theta)\mathscr{P}(\Pi)\right]\boldsymbol{e} - \theta\mathscr{P}(\Pi)\boldsymbol{e} \\ &\Rightarrow \left[\mathbb{I} - (1-\theta)\mathscr{P}(\Pi)\right]^{-1}\left(\Pi - \mathscr{P}(\Pi)\right)\boldsymbol{e} \\ &= \boldsymbol{e} - \mathscr{P}(\Pi)\theta\left[\mathbb{I} - (1-\theta)\mathscr{P}(\Pi)\right]^{-1}\boldsymbol{e} \\ &\Rightarrow \frac{1}{1-\theta}B\boldsymbol{e} \\ &= \left\{\mathbb{I} - \theta\left[\mathbb{I} - (1-\theta)\mathscr{P}(\Pi)\right]^{-1}\right\}\boldsymbol{e} + \theta\boldsymbol{e}\end{aligned} \quad (63)$$

Since B is a non-negative matrix, it follows from Equation (63) that:

$$\left\|\frac{1}{1-\theta}B\right\|_\infty = 1 - \min_i \left\{\theta\left[\mathbb{I} - (1-\theta)\mathscr{P}(\Pi)\right]^{-1}\Big|_i\right\} + \theta$$

Noting that $\theta\left[\mathbb{I} - (1-\theta)\mathscr{P}(\Pi)\right]^{-1} = \theta + \theta\sum_{k=1}^\infty ((1-\theta)\mathscr{P}(\Pi))^k$,

$$\left\|\frac{1}{1-\theta}B\right\|_\infty \leqq 1 - \theta + \theta \Rightarrow \left\|\frac{1}{1-\theta}B\right\|_\infty \leqq 1 \Rightarrow \|B\|_\infty \leqq 1-\theta$$

$$\Rightarrow \sum_{k=0}^\infty \|B\|_\infty^k \leqq \frac{1}{1-(1-\theta)} = \frac{1}{\theta} \quad (64)$$

Noting that $\mathbf{v}_\theta^0 = \mathbf{v}_\theta$ and $\theta > 0$, we conclude from Equation (62):

$$\forall \eta > 0, \quad \left\|\mathbf{v}_\theta^\eta - \mathbf{v}_\theta^0\right\|_\infty \leqq \eta\frac{1}{\theta} \quad (65)$$

which implies that $\mathbf{v}_\theta^\eta$ is upper semi-continuous w.r.t. η at $\eta = 0$. This completes the proof. □

Lemma 4.4: *For any plant* $G = (Q, \Sigma, \delta, \widetilde{\Pi}, \chi, \mathscr{C})$ *with an unobservability map* $\mathfrak{p}$: *the control actions generated by Algorithm* 4.3 *is optimal in the sense that*

$$\mathbf{E}\left(\int_0^t \hat{\nu}_\theta^\star(\tau)\mathrm{d}\tau\right) \geqq \mathbf{E}\left(\int_0^t \hat{\nu}_\theta^\#(\tau)\mathrm{d}\tau\right) \; \forall t \in [0,\infty), \forall\theta \in (0,1) \quad (66)$$

where $\hat{\nu}_\theta^\star(t)$ and $\hat{\nu}_\theta^\#(t)$ are the instantaneous measures at time t for control actions generated by Algorithm 4.3 and an arbitrary policy respectively.

Proof: First, we note that it suffices to consider terminating plants $G(\theta) = (Q, \Sigma, \delta, (1-\theta)\widetilde{\Pi}, \chi, \mathscr{C})$ such that $\theta \leqq \theta_{\min}$ (Definition 2.13) for the purpose of defining the optimal supervision policy (Chattopadhyay and Ray 2007b). Algorithm 4.3 specifies the optimal control policy for plants with termination probability θ when the set of entangled states is finite (Lemma 4.2). We claim that the result is true when this finiteness condition stated in Equation (53) is relaxed. The argument is as follows: the optimal control policy as stated in Algorithm 4.3 for finite $Q_{\mathscr{F}}$ can be paraphrased as:

- Maximise language measure for every state offline
- Follow the measure gradient online

Since $\text{CARD}(Q_{\mathscr{F}}^\eta) < \infty$, it follows from Lemma 4.2 that such a policy yields the optimal decisions for an η-quantised approximation of $\mathscr{E}_{(G,\mathfrak{p})} = (Q_{\mathscr{F}}, \Sigma, \Delta, \tilde{\pi}_{\mathscr{E}}, \chi_{\mathscr{E}})$ for any $\eta > 0$. As we approach $\eta = 0$, we note that it follows from continuity that there exists an $\eta_\star > 0$ such that the sequence of disabling decisions does not change for all $\eta \leqq \eta_\star$ implying that the optimally controlled transition sequence is identical for all $\eta \leqq \eta_\star$. Since it is guaranteed by Definition 4.11 that for identical transition sequences, quantised entangled states $\alpha_\eta^{[k]}$ are within η-balls of actual entangled state $\alpha^{[k]}$ after the kth observation, we conclude that

$$\forall k, \forall \eta \in (0, \eta_\star], \quad \left\|\alpha_\eta^{[k]} - \alpha^{[k]}\right\|_\infty \leqq \eta \quad (67)$$

It therefore follows that for any control policy, we have

$$\begin{aligned}&\forall \eta \in (0,\eta_\star], \left|\int_0^t \hat{\nu}_\theta^\eta(\tau)\mathrm{d}\tau - \int_0^t \hat{\nu}_\theta(\tau)\mathrm{d}\tau\right| \\ &\leqq \int_0^t \left|\langle\alpha_\eta^{[k]}, \mathbf{v}_\theta^\eta\rangle - \langle\alpha^{[k]}, \mathbf{v}_\theta\rangle\right|\mathrm{d}\tau \leqq \eta\left(1 + \frac{1}{\theta} + \frac{1}{\theta^2}\right)t\end{aligned} \quad (68)$$

implying that $\int_0^t \hat{\nu}_\theta^\eta(\tau)\mathrm{d}\tau$ is semi-continuous from above at $\eta = 0$ which completes the proof. □

Proposition 4.5: *Algorithm* 4.4 *correctly implements the optimal control policy for an arbitrary finite-state plant* $G = (Q, \Sigma, \delta, \widetilde{\Pi}, \chi, \mathscr{C})$ *with specified unobservability map* $\mathfrak{p}$.

Algorithm 4.4: Optimal Control under Partial Observation (Finalized Version)

input : $G = (Q, \Sigma, \delta, \widetilde{\Pi}, \chi, \mathscr{C})$,p
output: Optimal Control Actions

1 **begin** /* OFFLINE EXECUTION */
2 Compute $\nu_\star$;
3 Set $\theta = \theta_{min}$;
4 Compute $M = [\mathbb{I} - (1-\theta_{min})\mathscr{P}(\Pi)]^{-1}$;
5 **for** $\sigma \in \Sigma$ **do**
6 Compute Γ^σ; /* Algorithm 4.2 */
7 Compute $\Gamma^\sigma_{\mathscr{D}}$;
8 Compute $T^\sigma = [\Gamma^\sigma - \Gamma^\sigma_{\mathscr{D}}]\nu_\star$; /* Column Vector */
9 **endfor**
10 Initialize $\alpha_0 = [0 \cdots 1 \cdots 0]$ (i_0^{th} element) $\uparrow$; /* Init. state: q_{i_0} */
11 Compute $\alpha = \alpha_0 M$; /* For ω s.t. $p(q_i, \omega) = \epsilon$ */
12 **while true do** /* ONLINE EXECUTION */
13 **for** $\sigma \in \Sigma$ **do**
14 **if** $\alpha T^\sigma < 0$ **then**
15 Disable σ; /* Control Action */
16 **endif**
17 **endfor**
18 Observe event σ;
19 **if** σ *is disabled* **then**
20 $\alpha = \mathcal{N}(\alpha\Gamma^\sigma_{\mathscr{D}})$;
21 **else**
22 $\alpha = \mathcal{N}(\alpha\Gamma^\sigma)$;
23 **endif**
24 **endw**
25 **end**

Proof: We first note that Algorithm 4.4 is a detailed restatement of Algorithm 4.3 with the exception of the normalisation step in lines 20 and 22. On account of non-negativity of any entangled state α and the fact $\alpha \neq \mathbf{0}$ (Lemma 4.1), we have:

$$\text{sign}\big(\alpha(\Gamma^\sigma - \Gamma^\sigma_{\mathscr{D}})\big) = \text{sign}\big(\mathcal{N}(\alpha)(\Gamma^\sigma - \Gamma^\sigma_{\mathscr{D}})\big) \tag{69}$$

which verifies the normalisation steps. The result then follows immediately from Lemma 4.4. □

Remark 4.5: The normalisation steps in Algorithm 4.4 serve to mitigate numerical problems. Lemma 4.1 guarantees that the entangled state $\alpha \neq \mathbf{0}$. However, repeated right multiplication by the transition matrices may result in entangled states with norms arbitrarily close to 0 leading to numerical errors in comparing arbitrarily close floating point numbers. Normalisation partially remedies this by ensuring that the entangled states used for the comparisons are sufficiently separated from $\mathbf{0}$. There is, however, still the issue of approximability and even with normalisation, we may need to compare arbitrarily close values. The next proposition addresses this by showing that, in contrast to MDP based models, the optimisation algorithm for PFSA is indeed λ-approximable (Lusena et al. 2001), i.e. deviation from the optimal policy is guaranteed to be small for small errors in value comparisons in Algorithm 4.4. This further implies that the optimisation algorithm is robust under small parametric uncertainties in the model as well as to errors arising from finite precision arithmetic in digital computer implementations.

Proposition 4.6 (Approximability)**:** *In a finite precision implementation of Algorithm* 4.4, *with real numbers distinguished upto* $\lambda > 0$, *i.e.*

$$\forall a, b \in \mathbb{R}, \quad |a - b| \leqq \lambda \Rightarrow a - b \equiv 0 \tag{70}$$

we have $\forall t \in [0, \infty)$, $\forall \theta \in (0, 1)$,

$$0 \leqq \mathbf{E}\left(\int_0^t \hat{\nu}^\star_\theta(\tau)\mathrm{d}\tau\right) - \mathbf{E}\left(\int_0^t \hat{\nu}^\#_\theta(\tau)\mathrm{d}\tau\right) < \lambda \tag{71}$$

where $\hat{\nu}^\star_\theta(t)$ and $\hat{\nu}^\#_\theta(t)$ are the instantaneous measures at time t for the exact (i.e. infinite precision) and approximate (up to λ-precision) implementations of the optimal policy, respectively.

Proof: Let $G(\theta) = (Q, \Sigma, \delta, (1-\theta)\widetilde{\Pi}, \chi, \mathscr{C})$ be the underlying plant. First, we consider the perfectly observable case, i.e. with every transition observable at the supervisory level. Denoting the optimal and approximate measure vectors obtained by Algorithm B.1 as $\mathbf{v}^\star_\theta$ and $\mathbf{v}^\#_\theta$, we claim that

$$\mathbf{v}^\star_\theta - \mathbf{v}^\#_\theta \leqq_{\text{ELEMENTWISE}} \lambda \tag{72}$$

Using the algebraic structure of the Monotonicity Lemma (Chattopadhyay and Ray 2007) (see also Lemma A.1), we obtain

$$\mathbf{v}^\star_\theta - \mathbf{v}^\#_\theta = \theta\big[\mathbb{I} - (1-\theta)\Pi^\star\big]^{-1}(1-\theta)\underbrace{\big[\Pi^\star - \Pi^\#\big]\mathbf{v}^\#_\theta}_{\mathbf{M}}$$

We note that it follows from the exact optimality of $\mathbf{v}^\star_\theta$ that

$$\mathbf{v}^\star_\theta - \mathbf{v}^\#_\theta \geqq_{\text{ELEMENTWISE}} 0 \tag{73}$$

Denoting the ith row of the matrix $\mathbf{M}$ as $\mathbf{M}_i$, we note that M_i is of the form $\sum_{j=1}^{\text{CARD}(Q)} a_j b_j$ where

$$a_j \leqq \Pi_{ij} \tag{74}$$

$$b_j = \big|\mathbf{v}^\#_\theta|_i - \mathbf{v}^\#_\theta|_j\big| \tag{75}$$

We note that the inequality in Equation (74) follows from the fact that event enabling and disabling is a redistribution of the *controllable* part of the unsupervised transition matrix Π. Also, since $\Pi^\#$ was obtained via λ-precision optimisation, we have

$$\big(\mathbf{v}^\#_\theta|_i > \mathbf{v}^\#_\theta|_j\big) \bigwedge q_j \xrightarrow[\text{disabled}]{\sigma} q_i \Rightarrow \big|\mathbf{v}^\#_\theta|_i - \mathbf{v}^\#_\theta|_j\big| \leqq \lambda \tag{76a}$$

$$\big(\mathbf{v}^\#_\theta|_i \leqq \mathbf{v}^\#_\theta|_j\big) \bigwedge q_j \xrightarrow[\text{enabled}]{\sigma} q_i \Rightarrow \big|\mathbf{v}^\#_\theta|_i - \mathbf{v}^\#_\theta|_j\big| \leqq \lambda \tag{76b}$$

It therefore follows from stochasticity of Π that

$$\|M\|_\infty < \|\Pi\|_\infty \lambda = \lambda \tag{77}$$

Hence, noting that $\left\| \theta \left[\mathbb{I} - (1-\theta)\Pi^\star \right]^{-1} \right\|_\infty = 1$, we have

$$\| \mathbf{v}_\theta^\star - \mathbf{v}_\theta^\# \|_\infty \leqq \left\| \theta \left[\mathbb{I} - (1-\theta)\Pi^\star \right]^{-1} \right\|_\infty \times |1-\theta| \times \lambda < \lambda \tag{78}$$

which proves the claim made in Equation (72). It then follows from Lemma 3.1 that for the perfectly observable case we have $\forall t \in [0, \infty)$, $\forall \theta \in (0, 1)$,

$$0 \leqq \mathbf{E}\left(\int_0^t \hat{\nu}_\theta^\star(\tau) \mathrm{d}\tau \right) - \mathbf{E}\left(\int_0^t \hat{\nu}_\theta^\#(\tau) \mathrm{d}\tau \right) < \lambda \tag{79}$$

We recall that for a finite entangled state set $Q_{\mathscr{F}}$, the entangled transition system can be viewed as a perfectly observable terminating plant (Remark 4.3) with possibly partial controllability implying that we must apply the generalised monotonicity lemma (Lemma A.1). Noting that the above argument is almost identically applicable for the generalised monotonicity lemma, it follows that the above result is true for any non-trivial unobservability map on the underlying plant G_θ satisfying $\textsc{Card}(Q_{\mathscr{F}}) < \infty$. The extension to the general case of infinite $Q_{\mathscr{F}}$ then follows from the application of the result to η-approximations of the entangled transition system for $\eta \leqq \eta_\star$ (see Lemma 4.4 for explanation of the bound $\eta_\star$) and recalling the continuity argument stated in Lemma 4.4. This completes the proof. □

The performance of MDP- or POMDP-based models is computed as the total reward garnered by the agent in the course of operation. The analogous notion for PFSA-based modelling is the expected value of integrated instantaneous characteristic $\int_0^t \hat{\chi}(\tau)\mathrm{d}\tau$ (Definition 4.5) as a function of operation time.

Proposition 4.7 (Performance maximisation)**:** *The optimal control policy stated in Algorithm* 4.4 *maximises infinite horizon performance in the sense of maximising the expected integrated instantaneous state characteristic* (*Definition* 4.5), i.e.

$$\forall t \in [0, \infty),\ \mathbf{E}\left(\int_0^t \hat{\chi}^\star(\tau) \mathrm{d}\tau \right) \geqq \mathbf{E}\left(\int_0^t \hat{\chi}^\#(\tau) \mathrm{d}\tau \right) \tag{80}$$

where the instantaneous characteristic, at time t, for the optimal (*i.e. as defined by Algorithm* 4.4) *and an arbitrary supervision policy is denoted by* $\hat{\chi}^\star(t)$ *and* $\hat{\chi}^\#(t)$, *respectively*.

Proof: We recall that the result is true for the case of perfect observation (Equation (42)). Next we recall from Remark 4.3 that if the unobservability map is non-trivial, but has a finite $Q_{\mathscr{F}}$, then the entangled transition system $\mathscr{E}_{(G,\mathfrak{p})}$ can be viewed as a perfectly observable terminating model with uniform termination probability θ. It therefore follows, that for such cases, we have:

$$\forall t \in [0, \infty),\ \mathbf{E}\left(\int_0^t \hat{\chi}_{\mathscr{E}}^\star(\tau) \mathrm{d}\tau \right) \geqq \mathbf{E}\left(\int_0^t \hat{\chi}_{\mathscr{E}}^\#(\tau) \mathrm{d}\tau \right) \tag{81}$$

We recall from the definition of entangled transition systems (Definition 4.7),

$$\chi_{\mathscr{E}}(t) = \langle \alpha(t), \boldsymbol{\chi} \rangle \tag{82}$$

where $\alpha(t)$ is the entangled state at time t, which in turn implies that we have:

$$\mathbf{E}(\chi_{\mathscr{E}}) = \langle \mathbf{E}(\alpha), \boldsymbol{\chi} \rangle \tag{83}$$

Since $\mathbf{E}(\alpha)|_i$ is the expected sum of conditional probabilities of strings terminating on state q_i of the underlying plant, we conclude that $\mathbf{E}(\alpha)$ is in fact the stationary state probability vector corresponding to the underlying plant. Hence it follows that $\mathbf{E}(\chi_{\mathscr{E}}) = \mathbf{E}(\chi)$ implying that for non-trivial unobservability maps that guarantee $Q_{\mathscr{E}} < \infty$, we have

$$\forall t \in [0, \infty),\ \mathbf{E}\left(\int_0^t \hat{\chi}^\star(\tau) \mathrm{d}\tau \right) \geqq \mathbf{E}\left(\int_0^t \hat{\chi}^\#(\tau) \mathrm{d}\tau \right) \tag{84}$$

The general result for infinite entangled state sets (i.e. for unobservability maps which fail to guarantee $Q_{\mathscr{F}} < \infty$) follows from applying the above result to η-approximations (Definition 4.11) of the entangled transition system and recalling the continuity result of Lemma 4.4. □

4.6 *Computational complexity*

Computation of the supervision policy for an underlying plant with a non-trivial unobservability map requires computation of $\mathbf{v}_\star$ (See Step 2 of Algorithm 4.4), i.e. we need to execute Algorithm B.1 first. It was conjectured and validated via extensive simulation in Chattopadhyay and Ray (2007b) that Algorithm B.1 can be executed with polynomial asymptotic runtime complexity. Noting that each of the remaining steps of Algorithm 4.4 can be executed with worst case complexity of $n \times n$ matrix inversion (where n is the size of the state set Q of the underlying model), we conclude that the overall runtime complexity of proposed supervision algorithm is polynomial in number of underlying model states. Specifically, we have the following result.

Proposition 4.8: *The runtime complexity of the offline portion of Algorithm* 4.4 (*i.e. up to line number* 11) *is same as that of Algorithm* B.1.

Proof: The asymptotic runtime complexity of Algorithm B.1, as shown in Chattopadhyay and Ray (2007b), is $M(n \times n) \times O(I)$ where $M(n \times n)$ is the complexity of $n \times n$ matrix inversion and $O(I)$ is the asymptotic bound on the number of iterations on Algorithm B.1. The proof is completed by noting that the complexity of executing lines 3–11 of Algorithm 4.4 is $M(n \times n)$. □

Remark 4.6: It is immediate that the online portion of Algorithm 4.4 has the runtime complexity of matrix–vector multiplication. It follows that the measure-theoretic optimisation of partially observable plants is no harder to solve that those with perfect observation.

The results of this section establish the following facts:

(1) Decision-theoretic processes modelled in the PFSA framework can be efficiently optimised via maximisation of the corresponding language measure.
(2) The optimisation problem for infinite horizon problems is shown to be λ-approximable, and the solution procedure presented in this article is robust to modelling uncertainties and computational approximations. This is a significant advantage over POMDP-based modelling, as discussed in detail in Section 1.2.

5. Verification in simulation experiments

The theoretical development of the previous sections is next validated on two simple decision problems.

The first example consists of a four state mission execution model. The underlying plant is illustrated in Figure 6. The physical interpretation of the states and events is enumerated in Tables 5 and 6. G is the ground, initial or mission abort state. We assume the mission to be important; hence abort is assigned a negative characteristic value of -1. M represents correct execution and therefore has a positive characteristic of 0.5. The mission moves to state E on encountering possible system faults (event d) from

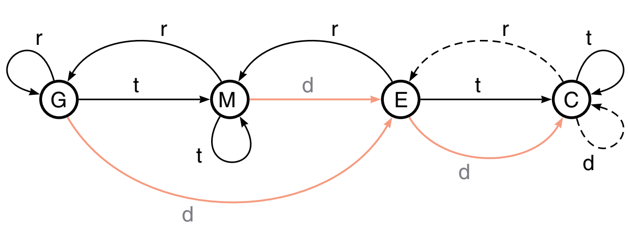

Figure 6. Underlying plant model with four states $Q = \{G, M, E, C\}$ and alphabet $\Sigma = \{t, r, d\}$: unobservable transitions are denoted by dashed arrows ($--\rightarrow$); uncontrollable but observable transitions are shown dimmed ($\leftarrow$).

states G and M. Any further system faults or an attempt to execute the next mission step under such error conditions results in a transition to the critical state C. The only way to correct the situation is to execute fault recovery protocols denoted by r. However, execution of r from the correct mission execution state M results in an abort. Occurrence of system faults d is uncontrollable from every state. Furthermore, under system criticality, we have sensor failure resulting in unobservability of further system faults and success of recovery attempts, i.e. the events d and r are unobservable from state C.

The event occurrence probabilities are tabulated in Table 5. We note that the probabilities of successful execution of mission steps (event t) and damage recovery protocols (event r) are both small under system criticality in state C. Also, comparison of the event probabilities from states M and E reveals that the probability of encountering further errors is higher once some error has already occurred and the probability of successful repair is smaller.

We simulate the controlled execution of the above described mission under the following three strategies:

(1) Null controller: no control enforced.
(2) Optimal control under perfect observation: control enforced using Algorithm 3.1 given that all transitions are observable at the supervisory level.
(3) Optimal control under partial observation: control enforced using Algorithm 4.4 given the above-described unobservability map.

The optimal renormalised measure vector of the system under full observability is computed to be $[-0.0049\ \ -0.0048\ \ -0.0049\ \ -0.0051]^T$. Hence, we observe in Figure 7 that the gradient of the instantaneous measure under perfect observation converges to

Table 5. State descriptions, event occurrence probabilities and characteristic values.

	Physical meaning	t	r	d	χ
G	Ground/abort	0.8	0.05	0.15	-1.00
M	Correct execution	0.5	0.30	0.20	0.50
E	System fault	0.5	0.20	0.30	-0.20
C	System critical	0.1	0.10	0.80	-0.25

Table 6. Event descriptions.

	Physical meaning
t	Execution of next mission step/objective successful
r	Execution of repair/damage recovery protocol
d	System fault encountered

around 0.005. We note that the gradient for the instantaneous measure under partial observation converges close to the former value. The null controller, of course, is the significantly poor.

The performance of the various control strategies are compared based on the expected value of the integrated instantaneous characteristic $\mathbf{E}\left(\int_0^t \hat{\chi}(\tau)\mathrm{d}\tau\right)$. The simulated results are shown in Figure 8. The null controller performs worst and the optimal control strategy under perfect observation performs best. As expected the strategy in which we blindly use the optimal control for perfect observation (Algorithm 3.1) under the given non-trivial unobservability map is exceedingly poor and close-to-best performance is recovered using the optimal control algorithm under partial observation.

The second example is one that originally appeared in the context of POMDPs in Cassandra (1994). The physical specification of the problem is as follows: the player is given a choice between opening one of two closed doors; one has a reward in the room behind it, the other has a tiger. Entering the latter incurs penalty in the form of bodily injury. The player can also choose to *listen* at the doors; and attempt to figure out which room has the tiger. The game resets after each play; and the tiger and the reward are randomly placed in the rooms at the beginning of each such play. Listening at the doors does not enable the player to accurately determine the location of the tiger; it merely makes her odds better. However, listening incurs a penalty; it costs the player if she chooses to listen. The scenario is pictorially illustrated in the top part of Figure 9. We model the physical situation in the PFSA framework as shown in the bottom part of Figure 9.

The PFSA has seven states $Q = \{N, T1, T2, L1, L2, T, A\}$ and eight alphabet symbols $\Sigma = \{s_1, s_2, \ell, t_C, t_I, c_1, c_2, n\}$. The physical meanings of the states and alphabet symbols are enumerated in Tables 7

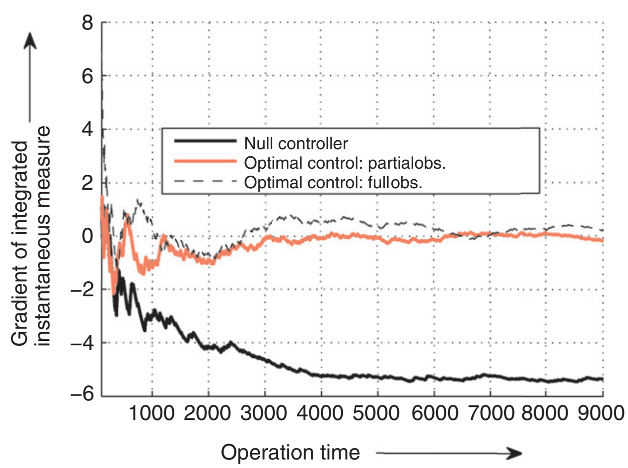

Figure 7. Gradient of integrated instantaneous measure as a function of operation time.

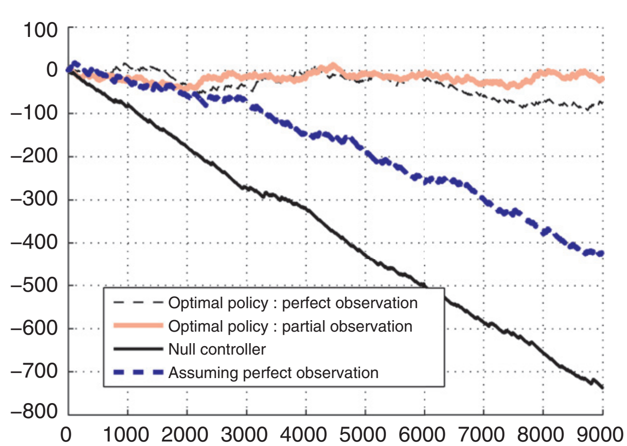

Figure 8. Performance as a function of operation time.

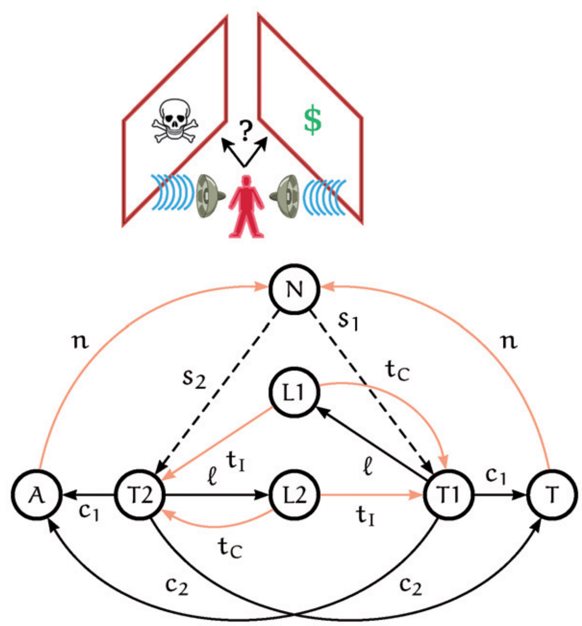

Figure 9. Top: Illustration of the physical scenario, Bottom: Underlying plant model with seven states and eight alphabet symbols: unobservable transitions are denoted by dashed arrows (– – →); uncontrollable but observable transitions are shown dimmed (→).

Table 7. State descriptions.

	Physical meaning
N	Game (Re)Initialise
$T1$	Tiger in 1
$T2$	Tiger in 2
$L1$	Listen: tiger in 1
$L2$	Listen: tiger in 2
T	Tiger chosen
A	Award chosen

and 8, respectively. The characteristic values and the event generation probabilities are tabulated in Table 9. States A and T have characteristics of 1 and -1 to reflect award and bodily injury. The listening states $L1$ and $L2$ also have negative characteristic (-0.75) in accordance with the physical specification. An interesting point is the assignment of negative characteristic to the states $T1$ and $T2$ and this prevents the player from choosing to disable *all* controllable moves from those states. Physically, this precludes the possibility that the player chooses to *not* play at all and sits in either of those states forever; which may turn out to be the optimal course of action if the states $T1$ and $T2$ are not negatively weighted.

Figure 10 illustrates the difference in the event disabling patterns resulting from the different strategies. We note that the optimal controller under perfect observation never disables event ℓ (event no. 3), since the player never needs to *listen* if she already knows which room has the reward. In the case of partial observation, the player decides to selectively listen to improve her odds. Also, note that the optimal policy under partial observation enables events significantly more often as compared to the optimal policy under perfect observation. The game actually proceeds via different routes in the two cases, hence it does not make sense to compare the control decisions after a given number of observation ticks, and the differences in the event disabling patterns must be interpreted only in an overall statistical sense.

Table 8. Event descriptions.

	Physical meaning
s_1	Tiger placed in 1 (unobs.)
s_2	Tiger placed in 2 (unobs.)
ℓ	Choose listen (cont.)
t_c	Correct determination
t_I	Incorrect determination
c_1	Choose 1 (cont.)
c_2	Choose 2 (cont.)
n	Game reset

We compare the simulation results in Figures 11 and 12. We note that in contrast to the first example, the performance obtained for the optimally supervised partially observable case is significantly lower compared to the situation under full observation. This arises from the physical problem at hand; it is clear that it is impossible in this case to have comparable performance in the two cases since the possibility of incorrect choice is significant and cannot be eliminated. The expected entangled state and the stationary probability vector on the underlying model states is compared in Figure 13 as an illustration for the result in Proposition 4.7.

6. Summary, conclusions and future work

In this article we present an alternate framework based on probabilistic finite-state machines (in the sense of Garg (1992a, 1992b)) for modelling partially observable decision problems and establish key advantages of the proposed approach over the current state of art. Namely, we show that the PFSA framework results in approximable problems, i.e. small changes in the model parameters or small numerical errors result in small deviation in the obtained solution. Thus one is guaranteed to obtain near optimal implementations of the proposed supervision algorithm in a computationally efficient manner. This is a significant improvement over the current state of art in POMDP analysis; several negative results exist that imply it is impossible to obtain a near-optimal supervision policy for arbitrary POMDPs in an efficient manner, unless certain complexity classes collapse (see detailed discussion in Section 1.2). The key tool used in this article is the recently reported notion of renormalised measure of probabilistic regular languages. We extend the measure theoretic optimisation technique for perfectly observable PFSA to obtain an online implementable supervision policy for finite-state underlying plants, for which one or more transitions are unobservable at the

Table 9. Event occurrence probabilities and characteristic values.

	χ	s_1	s_2	ℓ	t_C	t_I	c_1	c_2	n
N	0.00	0.5	0.5	0	0	0	0	0	0
$T1$	−0.25	0	0	0.33	0	0	0.33	0.33	0
$T2$	−0.25	0	0	0.33	0	0	0.33	0.33	0
$L1$	−0.75	0	0	0	0.8	0.2	0	0	0
$L2$	−0.75	0	0	0	0.8	0.2	0	0	0
T	−1.00	0	0	0	0	0	0	0	1
A	1.00	0	0	0	0	0	0	0	1

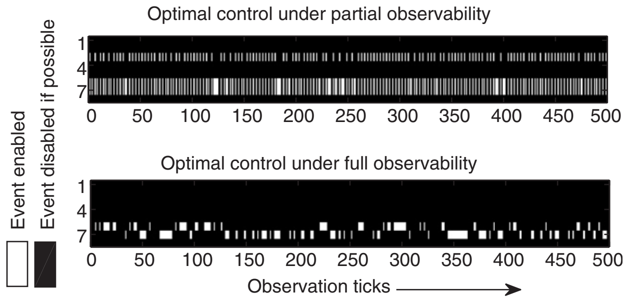

Figure 10. Control maps as a function of observation ticks.

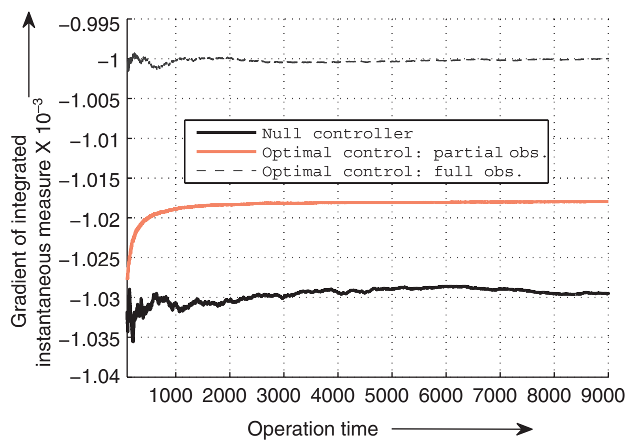

Figure 11. Gradient of integrated instantaneous measure as a function of operation time.

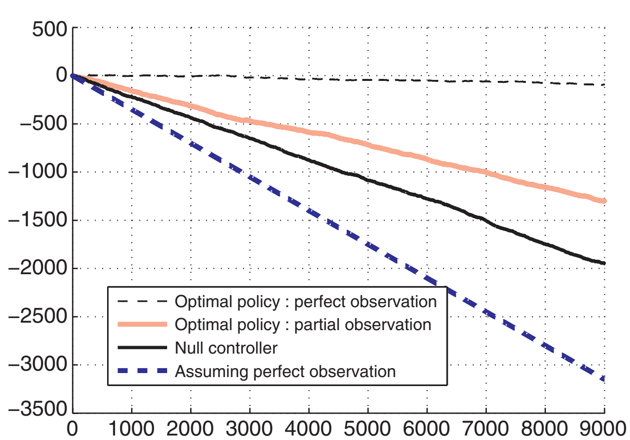

Figure 12. Performance as a function of operation time.

supervisory level. It is further shown that the proposed supervision policy maximises the infinite horizon performance in a sense very similar to that generally used in the POMDP framework; in the latter the optimal policy maximises the total reward garnered by the plant in the course of operation, while in the former, it is shown that the expected value of the integrated instantaneous state characteristic is maximised. Two simple decision problems are included as examples to illustrate the theoretical development.

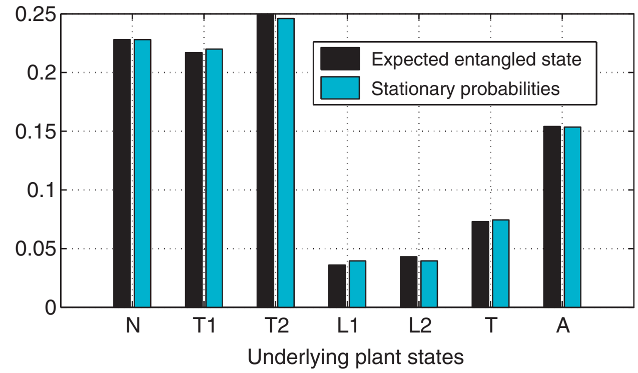

Figure 13. Comparison of expected entangled state with the stationary probability vector on the underlying plant states for the optimal policy under partial observation.

6.1 *Future work*

Future work will address the following areas:

(1) Validation of the proposed algorithm in real-life systems with special emphasis of probabilistic robotics, and detailed comparison with existing techniques: of particular interest would be comparative experimental studies that effectively demonstrate the significant computational advantage of the approach proposed in this article over the POMDP-based state of the art. Carefully designed experiments in robotic mission planning involving sufficiently complex decision problems would be one of the key areas that the authors would pursue in the immediate future. It is expected that the proposed approach would be successful in efficiently solving problem instances that would most probably prove to be intractable to competing techniques.

(2) Generalisation of the proposed technique to handle unobservability maps with unbounded memory, i.e. unobservability maps that result in infinite-state phantom automata: it was shown in Chattopadhyay and Ray (2008a) that for such non-regular unobservability maps, the problem of exactly determining the underlying state is undecidable. However, since the proposed control approach does not need to compute the exact underlying state, the authors are hopeful that an extension to such

general unobservability scenarios might be possible.
(3) Adaptation of the proposed approach to solve finite horizon decision problems which appear to be harder in the proposed framework as compared to the infinite-horizon solution.

Acknowledgements

This work has been supported in part by the US Army Research Laboratory (ARL) and the US Army Research Office (ARO) under Grant No. W911NF-07-1-0376, by the US Office of Naval Research (ONR) under Grant No. N00014-09-1-0688, and by NASA under Cooperative Agreement No. NNX07AK49A. Any opinions, findings and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the sponsoring agencies.

References

Atrash, A., and Koenig, S. (2001), 'Probabilistic Planning for Behavior-based Robots', in *Proceedings of the International FLAIRS conference (FLAIRS)*, pp. 531–535.

Bapat, R.B., and Raghavan, T.E.S. (1997), *Nonnegative Matrices and Applications*, Cambridge: Cambridge University Press.

Berman, A., and Plemmons, R.J. (1979), *Nonnegative Matrices in the Mathematical Science*, New York: Academic Press.

Burago, D., de Rougemont, M., and Slissenko, A. (1996), 'On the Complexity of Partially Observed Markov Decision Processes', *Theoretical Computer Science*, 157, 161–183.

Cassandra, A.R. (1994), *Optimal Policies for Partially Observable Markov Decision Processes*, Providence, RI, USA: Technical report.

Cassandra, A.R. (1998), 'Exact and Approximate Algorithms for Partially Observable Markov Decision Processes', Ph.D. Thesis, Providence, RI, USA: Adviser-Leslie Pack Kaelbling.

Chattopadhyay, I. (2006), 'Quantitative Control of Probabilistic Discrete Event Systems', PhD Dissertation, Department of Mechanical Engineering Pennsylvania State University. http://etda.libraries.psu.edu/theses/approved/WorldWideIndex/ETD-1443.

Chattopadhyay, I., and Ray, A. (2006a), 'A Language Measure for Partially Observed Discrete Event Systems', *International Journal of Control*, 79, 1074–1086.

Chattopadhyay, I., and Ray, A. (2006b), 'Renormalized Measure of Regular Languages', *International Journal of Control*, 79, 1107–1117.

Chattopadhyay, I., and Ray, A. (2007a), 'Generalized Projections in Finite-state Automata & Decidability of State Determinacy', in *American Control Conference (ACC)*, pp. 5664–5669.

Chattopadhyay, I., and Ray, A. (2007b), 'Language-measure-theoretic Optimal Control of Probabilistic Finite-state Systems', *International Journal of Control*, 80, 1271–1290.

Chattopadhyay, I., and Ray, A. (2008a), 'Generalised Projections in Finite-state Automata & Decidability of State Determinacy', *International Journal of Control*, 81, 1626–1644.

Chattopadhyay, I., and Ray, A. (2008b), 'Structural Transformations of Probabilistic Finite-state Machines', *International Journal of Control*, 81, 820–835.

Garg, V.K. (1992a), 'An Algebraic Approach to Modeling Probabilistic Discrete Event Systems', in *Proceedings of 1992 IEEE Conference on Decision and Control*, Tucson, AZ, pp. 2348–2353.

Garg, V.K. (1992b), 'Probabilistic Languages for Modeling of DEDs', in *Proceedings of 1992 IEEE Conference on Information and Sciences*, Princeton, NJ, pp. 198–203.

Gribaudo, M., Sereno, M., Horvath, A., and Bobbio, A. (2001), 'Fluid Stochastic Petri Nets Augmented with Flush-out Arcs: Modelling and Analysis', *Discrete Event Dynamic Systems*, 11(1–2), 97–117.

Hansen, E.A. (1998), 'Finite-memory Control of Partially Observable Systems', Ph.D. Thesis, Amherst, MA, USA: Director- Shlomo Zilberstein.

Hopcroft, J.E., Motwani, R., and Ullman, J.D. (2001), *Introduction to Automata Theory, Languages, and Computation* (2nd ed.), Addison-Wesley.

Kushmerick, N., Hanks, S., and Weld, D.S. (1995), 'An Algorithm for Probabilistic Planning', *Artificial Intelligence*, 76, 239–286.

Lusena, C., Goldsmith, J., and Mundhenk, M. (2001), 'Nonapproximability Results for Partially Observable Markov Decision Processes', *Journal of Artificial Intelligence Research*, 14, 83–103.

Madani, O., Hanks, S., and Condon, A. (1999), 'On the Undecidability of Probabilistic Planning and Infinite-horizon Partially Observable Markov Decision Problems', in *AAAI '99/IAAI '99: Proceedings of the Sixteenth National Conference on Artificial Intelligence and the Eleventh Innovative Applications of Artificial Intelligence Conference Innovative Applications of Artificial Intelligence*, Menlo Park, CA, USA: American Association for Artificial Intelligence, pp. 541–548.

Mcallester, D., and Rosenblitt, D. (1991), 'Systematic Nonlinear Planning', in *Proceedings of the Ninth National Conference on Artificial Intelligence*, pp. 634–639.

Moody, J.O., and Antsaklis, P.J. (1998), *Supervisory Control Ofdiscrete Event Systems using Petri Nets*, Boston: Kluwer Academic.

Paz, A. (1971), *Introduction to Probabilistic Automata (Computer Science and Applied Mathematics)*, Orlando, FL, USA: Academic Press, Inc.

Penberthy, J.S., and Weld, D.S. (1992), *Ucpop: A Sound, Complete, Partial Order Planner for ADL*, San Matco, CA: Morgan Kaufmann, pp. 103–114.

Peng, J., and Williams, R.J. (1993), *Efficient Learning and Planning within the Dyna Framework*, Adaptive Behavior, pp. 437–454.

Puterman, M.L. (1990), 'Markov Decision Processes', in *Handbook of Operations Research* eds. D.P. Heyman, & M.J. Sobel, North Holland Publishers, pp. 331–434.

Rabin, M.O. (1963), 'Probabilistic Automata', *Information and Control*, 6, 230–245.

Ramadge, P.J., and Wonham, W.M. (1987), 'Supervisory Control of a Class of Discrete Event Processes', *SIAM Journal of Control and Optimization*, 25, 206–230.

Ray, A. (2005), 'Signed Real Measure of Regular Languages for Discreteevent Supervisory Control', *International Journal of Control*, 78, 949–967.

Ray, A., Phoha, V., and Phoha, S. (2005), *Quantitative Measure for Discrete Event Supervisory Control*, New York: Springer.

Rudin, W. (1988), *Real and Complex Analysis* (3rd ed.), New York: McGraw Hill.

Sondik, E.J. (1978), 'The Optimal Control of Partially Observable Markov Processes Over the Infinite Horizon: Discounted Costs', *Operations Research*, 26, 282–304.

Stewart, W. (1999), *Computational Probability: Numerical Methods for Computing Stationary Distribution of Finite Irreducible Markov Chains*, New York: Springer.

White, D.J. (1993), *Markov Decision Processes*, Chichester: Wiley.

Zhang, W., and Golin, G.M.J. (2001), *Algorithms for Partially Observable Markov Decision Processes*, Technical report, Hong Kong University of Science and Technology.

Appendix A. Generalised monotonicity lemma

The following proposition is a slight generalisation of the corresponding result reported in Chattopadhyay and Ray (2007b) required to handle cases where the effect of event disabling is not always a self-loop at the current state but produces a pre-specified reconfiguration, e.g.

Disabling $q_i \xrightarrow{\sigma} q_j$ *results in* $q_i \xrightarrow{\sigma} q_k$

Note that for every state $q_i \in Q$, it is pre-specified where each event σ will terminate on been disabled. This generalisation is critical to address the partial controllability issues arising from partial observation at the supervisory level.

Proposition A.1 (Monotonicity)**:** Let $G(\theta) = (Q, \Sigma, \delta, (1-\theta)\widetilde{\Pi}, \chi, \mathscr{C})$ be reconfigured to $G^{\#}_{\theta} = (Q, \Sigma, \delta^{\#}, (1-\theta)\widetilde{\Pi}^{\#}, \boldsymbol{\chi}, \mathscr{C})$ as follows: $\forall i,\, j,\, k \in \{1, 2, \ldots, n\}$, the $(i, j)^{th}$ element $\Pi^{\#}_{ij}$ and the $(i, k)th$ element $\Pi^{\#}_{ik}$ of $\Pi^{\#}$ are obtained as

$$\left.\begin{aligned}\Pi^{\#}_{ij} &= \Pi_{ij} + \beta_{ij}\\ \Pi^{\#}_{ik} &= \Pi_{ik} - \beta_{ij}\end{aligned}\right\} \text{ if } \mu_j > \mu_k \text{ with } \beta_{ij} > 0$$
$$\left.\begin{aligned}\Pi^{\#}_{ij} &= \Pi_{ij}\\ \Pi^{\#}_{ik} &= \Pi_{ik}\end{aligned}\right\} \text{ if } \mu_j = \mu_k \tag{85}$$
$$\left.\begin{aligned}\Pi^{\#}_{ij} &= \Pi_{ij} - \beta_{ij}\\ \Pi^{\#}_{ik} &= \Pi_{ik} + \beta_{ij}\end{aligned}\right\} \text{ if } \mu_j < \mu_k \text{ with } \beta_{ij} > 0$$

Then for the respective measure vectors be $\mathbf{v}_\theta$ and $\mathbf{v}^{\#}_{\theta}$,

$$\mathbf{v}^{\#}_{\theta} \geqq_{\text{ELEMENTWISE}} \mathbf{v}_\theta \quad \forall \theta \in (0, 1) \tag{86}$$

with equality holding if and only if $\Pi^{\#} = \Pi$.

Proof: From the definition of renormalised measure (Definition 2.9), we have

$$\begin{aligned}\mathbf{v}^{\#}_{\theta} - \mathbf{v}_\theta &= \theta\big[I - (1-\theta)\Pi^{\#}\big]^{-1} - \theta[I - (1-\theta)\Pi]^{-1}\boldsymbol{\chi}\\ &= \big[I - (1-\theta)\Pi^{\#}\big]^{-1}(1-\theta)\big(\Pi^{\#} - \Pi\big)\mathbf{v}_\theta\end{aligned}$$

Defining the matrix $\Delta \triangleq \Pi^{\#} - \Pi$, and the ith row of Δ as Δ_i, it follows that

$$\Delta_i \mathbf{v}_\theta = \sum_j \Delta_{ij} \mathbf{v}_\theta|_j = \sum_j \beta_{ij}\Gamma_{ij} \tag{87}$$

where

$$\Gamma_{ij} = \begin{cases} (\mathbf{v}_\theta|_k - \mathbf{v}_\theta|_j) & \text{if } \mathbf{v}_\theta|_k > \mathbf{v}_\theta|_j\\ 0 & \text{if } \mathbf{v}_\theta|_k = \mathbf{v}_\theta|_j \Longrightarrow \Gamma_{ij} \geqq 0\ \forall i, j\\ (\mathbf{v}_\theta|_j - \mathbf{v}_\theta|_k) & \text{if } \mathbf{v}_\theta|_k < \mathbf{v}_\theta|_j \end{cases}$$

Since $\forall j$, $\sum_{i=1}^{n} \Pi_{ij} = \sum_{i=1}^{n} \Pi^{\#}_{ij} = 1$, it follows from non-negativity of Π, that $[I - (1-\theta)\Pi^{\#}]^{-1} \geqq_{\text{ELEMENTWISE}} \mathbf{0}$. Since $\beta_{ij} > 0\backslash, \forall i, j$, it follows that $\Delta_i \mathbf{v}_\theta \geq 0\ \forall i \Rightarrow \mathbf{v}^{\#}_{\theta} \geqq_{\text{ELEMENTWISE}} \mathbf{v}_\theta$. For $\mathbf{v}_\theta|_j \neq 0$ and Δ as defined above, $\Delta_i \mathbf{v}_\theta = 0$ if and only if $\Delta = 0$. Then, $\Pi^{\#} = \Pi$ and $\mathbf{v}^{\#}_{\theta} = \mathbf{v}_\theta$. □

Appendix B. Pertinent algorithms for measure-theoretic control

This section enumerates the pertinent algorithms for computing the optimal supervision policy for a perfectly observable plant $G = (Q, \Sigma, \delta, \widetilde{\Pi}, \chi, \mathscr{C})$. For the proof of correctness, the reader is referred to Chattopadhyay and Ray (2007b).

In Algorithm B.2, we use the following notation:

$$M_0 = \Big[\mathbb{I} - \mathbf{P} + \mathcal{C}(\Pi)\Big]^{-1}, \quad M_1 = \left[\mathbb{I} - \Big[\mathbb{I} - \mathbf{P} + \mathcal{C}(\Pi)\Big]^{-1}\right]$$

$$M_2 = \inf_{\alpha \neq 0} \left|\left|\big[\mathbb{I} - \mathbf{P} + \alpha\mathscr{P}\big]^{-1}\right|\right|_\infty$$

Also, as defined earlier, $\mathcal{C}(\Pi)$ is the stable probability distribution which can be computed using methods reported widely in the literature (Stewart 1999).

Algorithm B.1: Computation of Optimal Supervisor

```
input : P, χ, 𝒞
output: Optimal set of disabled transitions 𝒟*
begin
  Set 𝒟[0] = ∅, Π̃[0] = Π̃, θ*[0] = 0.99, k = 1;
  while (Terminate == false) do
    Compute θ*[k];                              /* Algorithm B.2 */
    Set Π̃[k] = (1−θ*[k]) / (1−θ*[k−1]) Π̃[k−1];
    Compute v[k];
    for j = 1 to n do
      for i = 1 to n do
        Disable all controllable q_i –σ→ q_j s.t. v_j[k] < v_i[k];
        Enable all controllable q_i –σ→ q_j s.t. v_j[k] ≧ v_i[k];
    Collect all disabled transitions in 𝒟[k];
    if 𝒟[k] == 𝒟[k−1] then
      Terminate = true;
    else
      k = k + 1;
  𝒟* = 𝒟[k];                                   /* Optimal disabling set */
end
```

Algorithm B.2: Computation of the Critical Lower Bound $\theta_\star$

```
input  : $\mathbf{P}$, $\chi$
output: $\theta_\star$
begin
    Set $\theta_\star = 1$, $\theta_{curr} = 0$, Compute $\mathcal{C}(\Pi)$ , $M_0$ , $M_1$, $M_2$;
    for $j = 1$ to $n$ do
        for $i = 1$ to $n$ do
            if $(\mathcal{C}(\Pi)\chi)_i - (\mathcal{C}(\Pi)\chi)_j \neq 0$ then
                $\theta_{curr} = \frac{1}{8M_2} | (\mathcal{C}(\Pi)\chi)_i - (\mathcal{C}(\Pi)\chi)_j |$
            else
                for $r = 0$ to $n$ do
                    if $(M_0\chi)_i \neq (M_0\chi)_j$ then
                        Break;
                    else
                        if $(M_0 M_1^r \chi)_i \neq (M_0 M_1^r \chi)_j$ then
                            Break;
                if $r == 0$ then
                    $\theta_{curr} = \frac{|\{(M_0 - \mathcal{C}(\Pi))\chi\}_i - \{(M_0 - \mathcal{C}(\Pi))\chi\}_j|}{8M_2}$;
                else
                    if $r > 0$ AND $r \leqslant n$ then
                        $\theta_{curr} = \frac{|(M_0 M_1 \chi)_i - (M_0 M_1 \chi)_j|}{2^{r+3} M_2}$ ;
                    else
                        $\theta_{curr} = 1$ ;
            $\theta_\star = \min(\theta_\star, \theta_{curr})$ ;
end
```

Algorithm B.3: Computation of Phantom Automaton

```
input  : $Q$, $\Sigma$, $\tilde{\pi}$, Unobservability map $\mathfrak{p}$
output: $\mathscr{P}(\Pi)$
begin
    Set $\tilde{\pi}^{\mathscr{P}} = \tilde{\pi}$;
    for $i = 1$ to $n$ do
        for $j = 1$ to $m$ do
            if $\mathfrak{p}(q_i, \sigma_j) = \sigma_j$ then
                $\tilde{\pi}^{\mathscr{P}}_{ij} = 0$;          /* Delete transition */
    for $i = 1$ to $n$ do
        for $j = 1$ to $n$ do
            $\mathscr{P}(\Pi)_{ij} = \sum_{k:\delta(q_i,\sigma_k)=q_j} \tilde{\pi}^{\mathscr{P}}_{ik}$;
end
```

Algorithm B.4: Petri Net observer

```
input  : $\langle G, p \rangle$
output: Petri net observer
begin
    I. Create a place $q_j$ for each state $q_j$ in $\langle G, p \rangle$;
    II. The set of transition labels is $\Sigma$;
    for each observable transition $q_j \xrightarrow[\sigma]{} q_k$ in $\langle G, p \rangle$ do
        I. Set the initial state in $\langle G, p \rangle$ to $q_k$;
        II. Compute $\overline{Q}(\epsilon)$;
        III. Add a transition labelled $\sigma$ from the place $q_j$ with
        output arcs to all places $q_l \in \overline{Q}(\epsilon)$;
    for each place $q_j$ in the net do
        for each event $\sigma \in \Sigma$ do
            if there is no transition with label $\sigma$ from $q_j$ then
                I. Add a flush-out arc with label $\sigma$ from $q_j$
end
```

Algorithm B.5: Online computation of possible states

```
input  : Petri net observer, Observed sequence $\omega = \tau_1 \tau_2 \ldots \tau_r$
output: $\overline{Q}(\omega)$
begin
    I. Compute the initial marking for the observer as follows:
        a. Compute $\overline{Q}(\epsilon)$;
        b. Put a token in each place $q_j \in \overline{Q}(\epsilon)$;
    for $j = 1$ to $r$ do
        I. Fire all enabled transitions labeled $\tau_j$;
        for each place $q_j$ in the observer do
            if number of tokens in $q_j > 0$ then
                I. Normalize the number of tokens in $q_j$ to 1.
    II. $\overline{Q}(\omega) = \{q_j \mid q_j$ has one token$\}$ ;
end
```