INT. J. CONTROL, 2004, VOL. 77, NO. 12, 1083–1100

Optimal supervisory control of finite state automata

ASOK RAY*, JINBO FU and CONSTANTINO LAGOA

This paper presents optimal supervisory control of dynamical systems that can be represented by deterministic finite state automaton (DFSA) models. The performance index for the optimal policy is obtained by combining a measure of the supervised plant language with (possible) penalty on disabling of controllable events. The signed real measure quantifies the behaviour of controlled sublanguages based on a state transition cost matrix and a characteristic vector as reported in earlier publications. Synthesis of the optimal control policy requires at most n iterations, where n is the number of states of the DFSA model generated from the unsupervised plant language. The computational complexity of the optimal control synthesis is polynomial in n. Syntheses of the control algorithms are illustrated with two application examples.

1. Introduction

In a seminal paper, Ramadge and Wonham (1987) pioneered the concept of discrete-event supervisory control of finite-state automata (equivalently, regular languages). A plant, supervised under a control policy, is represented by a sublanguage of the unsupervised plant language, which could be different under different supervisors if they are constrained to satisfy dissimilar specifications. Such a set of supervised plant sublanguages are, in general, partially ordered; it is necessary to establish a quantitative measure for total ordering of their respective performance. To address this issue, Wang and Ray (2004), and their colleagues (Ray and Phoha 2003, Surana and Ray 2004) have developed a signed measure of regular languages.

Optimal control of regular languages has been proposed by several researchers based on different assumptions. Some of these researchers have attempted to quantify the controller performance using different types of cost assigned to the individual events. Passino and Antsaklis (1989) proposed path costs associated with state transitions and hence optimal control of a discrete event system is equivalent to following the shortest path on the graph representing the uncontrolled system. Kumar and Garg (1995) made use of the concept of payoff and control costs that are incurred only once regardless of the number of times the system visits the state associated with the cost. Consequently, the resulting cost is not a function of the dynamic behavior of the plant. Brave and Heymann (1993) introduced the concept of optimal attractors in discrete-event control. Sengupta and Lafortune (1998) used control cost in addition to the path cost in optimization of the performance index for trade-off between finding the shortest path and reducing the control cost. Although costs were assigned to the events, no distinction was made for events generated at (or leading to) different states that could be 'good' or 'bad'. These optimal control strategies have addressed performance enhancement of discrete-event control systems without a quantitative measure of languages.

Recently, Fu *et al.* (2004) have proposed a state-based method for optimal control of regular languages by selectively disabling controllable events so that the resulting optimal policy can be realized as a controllable supervisor. The performance index of the optimal policy is a signed real measure of the supervised sublanguage, which is expressed in terms of a cost matrix and a characteristic vector (Ray and Phoha 2003, Surana and Ray 2004), but it does not assign any additional penalty for event disabling. In a follow-up publication, Fu *et al.* (2003 b) extended their earlier work on optimal control to include the cost of (controllable) event disabling. The rationale is that, without the event disabling cost, an optimal supervisor makes the best trade-off between reaching good states and avoiding bad states, and achieves optimal performance in terms of the language measure of the supervised plant. However, another supervisor that has a slightly inferior performance relative to the above optimal controller may only require disabling of fewer or some other controllable events, which is much less difficult to achieve. Therefore, with due consideration to event disabling, the second controller might be preferable.

The work, reported in this paper, augments and consolidates the theory and applications of optimal supervisory control of regular languages, which have been reported in an informal structure in previous publications (Fu *et al.* 2003 b, 2004). The performance index for the optimal control policy proposed in this paper is obtained by combining a real signed measure of the

Received 1 October 2003. Revised and accepted 22 July 2004.

* Author for correspondence. e-mail: axr2@psu.edu

The Pennsylvania State University, University Park, PA 16802, USA.

International Journal of Control ISSN 0020–7179 print/ISSN 1366–5820 online
http://www.tandf.co.uk/journals
DOI: 10.1080/00207170420002737662

supervised plant language with the cost of disabled event(s). Starting with the (regular) language of an unsupervised plant automaton, the optimal control policy makes a trade-off between the measure of the supervised sublanguage and the associated event disabling cost to achieve the best performance. Like any other optimization procedure, it is possible to choose different performance indices to arrive at different optimal policies for discrete event supervisory control. Nevertheless, usage of the language measure provides a systematic procedure for precise comparative evaluation of different supervisors so that the optimal control policy(ies) can be conclusively identified.

The derived theoretical results are presented with formal proofs and are supported by two application examples. The first application is on supervisory control of a twin-engine aircraft for its safe and reliable operation. The second application is on decision and control of a multiprocessor decoding system for efficient operation. The major contribution of this paper is conceptualization, formulation and illustration of a quantitative method for analysis and synthesis of optimal supervisory control policies for plant dynamics that can be captured by regular languages. From the above perspectives, the performance index for the optimal control policy proposed in this paper is obtained by combining the measure of the supervised plant language with the cost of disabled event(s).

This paper is organized in six sections including the present one. Section 2 briefly reviews the previous work on language measure. Section 3 presents the optimal control policy without the event disabling cost. Section 4 modifies the performance index to include the event disabling cost and formulates the algorithm of the optimal control policy with event disabling cost as an extension of § 3. Section 5 presents two application examples to illustrate the concepts of optimal control without and with event disabling cost. The paper is summarized and concluded in § 6 along with recommendations for future work.

2. Brief review of language measure

This section briefly reviews the concept of signed real measure of regular languages (Ray and Phoha 2003, Surana and Ray 2004). Let the plant behaviour be modelled as a deterministic finite state automaton (DFSA)

$$G_i \equiv (Q, \Sigma, \delta, q_i, Q_m) \tag{1}$$

where Q is the finite set of states with $|Q| = n$, and $q_i \in Q$ is the initial state; Σ is the (finite) alphabet of events with $|\Sigma| = m$; the Kleene closure of Σ is denoted as Σ^* that is the set of all finite-length strings of events including the empty string ε; the (possibly partial) function $\delta : Q \times \Sigma \to Q$ represents state transitions and $\delta^* : Q \times \Sigma^* \to Q$ is an extension of δ; and $Q_m \subseteq Q$ is the set of marked (i.e. accepted) states.

Definition 1: The language $L(G_i)$ generated by a DFSA G initialized at the state $q_i \in Q$ is defined as

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

Definition 2: The language $L_m(G_i)$ marked by a DFSA G_i initialized at the state $q_i \in Q$ is defined as

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

The language $L(G_i)$ is partitioned as the non-marked and the marked languages, $L^o(G_i) \equiv L(G_i) - L_m(G_i)$ and $L_m(G_i)$, consisting of event strings that, starting from $q_i \in Q$, terminate at one of the non-marked states in $Q - Q_m$ and one of the marked states in Q_m, respectively. The set Q_m is further partitioned into Q_m^+ and Q_m^-, where Q_m^+ contains all *good* marked states that are desired to be terminated on and Q_m^- contains all *bad* marked states that one may not want to terminate on, although it may not always be possible to avoid the bad states while attempting to reach the good states. The marked language $L_m(G_i)$ is further partitioned into $L_m^+(G_i)$ and $L_m^-(G_i)$ consisting of good and bad strings that, starting from q_i, terminate on Q_m^+ and Q_m^-, respectively.

A signed real measure $\mu : 2^{\Sigma^*} \to \mathrm{R} \equiv (-\infty, \infty)$ is constructed for quantitative evaluation of every event string $s \in \Sigma^*$. The language $L(G_i)$ is decomposed into null, i.e. $L^o(G_i)$, positive, i.e. $L_m^+(G_i)$, and negative, i.e. $L_m^-(G_i)$ sublanguages.

Definition 3: The language of all strings that, starting at a state $q_i \in Q$, terminates on a state $q_j \in Q$, is denoted as $L(q_i, q_j)$. That is

$$L(q_i, q_j) \equiv \{s \in L(G_i) : \delta^*(q_i, s) = q_j\}. \tag{4}$$

Definition 4: The characteristic function that assigns a signed real weight to state-partitioned sublanguages $L(q_i, q_j), i = 1, 2, \ldots, n, \quad j = 1, 2, \ldots, n$ is defined as: $\chi : Q \to [-1, 1]$ such that

$$\chi(q_j) \in \begin{cases} [-1, 0) & \text{if } q_j \in Q_m^- \\ \{0\} & \text{if } q_j \notin Q_m \\ (0, 1] & \text{if } q_j \in Q_m^+. \end{cases} \tag{5}$$

Definition 5: The event cost is conditioned on a DFSA state at which the event is generated, and is defined as $\tilde{\pi} : \Sigma^* \times Q \to [0, 1]$ such that $\forall q_j \in Q, \forall \sigma_k \in \Sigma$, $\forall s \in \Sigma^*$:

(1) $\tilde{\pi}[\sigma_k, q_j] \equiv \tilde{\pi}_{jk} \in [0, 1)$; $\quad \sum_k \tilde{\pi}_{jk} < 1$;

(2) $\tilde{\pi}[\sigma, q_j] = 0$ if $\delta(q_j, \sigma)$ is undefined; $\quad \tilde{\pi}[\epsilon, q_j] = 1$;

(3) $\tilde{\pi}[\sigma_k s, q_j] = \tilde{\pi}[\sigma_k, q_j]\, \tilde{\pi}[s, \delta(q_j, \sigma_k)]$.

The event cost matrix is defined as

$$\tilde{\Pi} = \begin{bmatrix} \tilde{\pi}_{11} & \tilde{\pi}_{12} & \dots & \tilde{\pi}_{1m} \\ \tilde{\pi}_{21} & \tilde{\pi}_{22} & \dots & \tilde{\pi}_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{\pi}_{n1} & \tilde{\pi}_{n2} & \dots & \tilde{\pi}_{nm} \end{bmatrix} \tag{6}$$

and is referred to as the $\tilde{\Pi}$-matrix in the sequel.

An application of the induction principle to part (3) in Definition 2.5 shows $\tilde{\pi}[st, q_j] = \tilde{\pi}[s, q_j]\tilde{\pi}[t, \delta^*(q_j, s)]$. The condition $\sum_k \tilde{\pi}_{jk} < 1$ provides a sufficient condition for the existence of the real signed measure as discussed in Surana and Ray (2004) along with additional comments on the physical interpretation of the event cost.

The next task is to formulate a measure of sublanguages of the plant language $L(G_i)$ in terms of the signed characteristic function χ and the non-negative event cost $\tilde{\pi}$.

Definition 6: The signed real measure μ of a singleton string set $\{s\} \subseteq L(q_i, q_j) \subseteq L(G_i) \in 2^{\Sigma^*}$ is defined as

$$\mu(\{s\}) \equiv \tilde{\pi}(s, q_i)\chi(q_j) \quad \forall s \in L(q_i, q_j). \tag{7}$$

The signed real measure of $L(q_i, q_j)$ is defined as:

$$\mu\big(L(q_i, q_j)\big) \equiv \sum_{s \in L(q_i, q_j)} \mu(\{s\}) \tag{8}$$

and the signed real measure of a DFSA G_i, initialized at the state $q_i \in Q$, is denoted as

$$\mu_i \equiv \mu(L(G_i)) = \sum_j \mu\big(L(q_i, q_j)\big). \tag{9}$$

Definition 7: The state transition cost, $\pi : Q \times Q \to [0, 1)$, of the DFSA G_i is defined as

$$\forall q_i, q_j \in Q, \quad \pi_{ij} = \begin{cases} \sum_{\sigma \in \Sigma} \tilde{\pi}[\sigma, q_i], & \text{if } \delta(q_i, \sigma) = q_j \\ 0 & \text{if } \{\delta(q_i, \sigma) = q_j\} = \varnothing. \end{cases} \tag{10}$$

Consequently, the $n \times n$ state transition cost Π-matrix is defined as

$$\Pi = \begin{bmatrix} \pi_{11} & \pi_{12} & \dots & \pi_{1n} \\ \pi_{21} & \pi_{22} & \dots & \pi_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \pi_{n1} & \pi_{n2} & \dots & \pi_{nn} \end{bmatrix}. \tag{11}$$

Wang and Ray (2004), and Surana and Ray (2004) have shown that the measure $\mu_i \equiv \mu(L(G_i))$ of the language $L(G_i)$, with the initial state q_i, can be expressed as: $\mu_i = \sum_j \pi_{ij}\mu_j + \chi_i$ where $\chi_i \equiv \chi(q_i)$. Equivalently, in vector notation: $\bar{\mu} = \Pi\bar{\mu} + \bar{\chi}$ where the measure vector $\bar{\mu} \equiv [\mu_1\ \mu_2\ \cdots\ \mu_n]^{\mathrm{T}}$ and the characteristic vector $\bar{\chi} \equiv [\chi_1\ \chi_2\ \cdots\ \chi_n]^{\mathrm{T}}$. From the perspective of constructing an optimal control policy, salient properties of the state transition cost matrix Π are delineated below.

Property 1: *Following Definitions* 4 *and* 5, *there exists* $\theta \in (0, 1)$ *such that the induced infinity norm* $\|\Pi\|_\infty \equiv \max_i \sum_j \pi_{ij} = 1 - \theta$. *The matrix operator* $[I - \Pi]$ *is invertible implying that the inverse* $[I - \Pi]^{-1}$ *is a bounded linear operator* (*Naylor and Sell 1982*) *with its induced infinity norm* $\|[I - \Pi]^{-1}\|_\infty \le \theta^{-1}$. *Therefore, the language measure vector can be expressed as* $\bar{\mu} = [I - \Pi]^{-1}\bar{\chi}$, *where* $\bar{\mu} \in \mathbb{R}^n$, *and computational complexity* (*Surana and Ray 2004*) *of the measure is* $O(n^3)$.

Property 2: *The matrix operator* $[I - \Pi]^{-1} \ge 0$ *elementwise. By Taylor series expansion,* $[I - \Pi]^{-1} = \sum_{k=0}^{\infty} [\Pi]^k$ *and* $[\Pi]^k \ge 0$ *because* $\Pi \ge 0$.

Property 3: *The determinant* $\mathrm{Det}\,[I - \Pi]$ *is real positive because the eigenvalues of the real matrix* $[I - \Pi]$ *appear as real or complex conjugates and they have positive real parts. Hence, the product of all eigenvalues of* $[I - \Pi]$ *is real positive.*

Property 4: *An affine operator* $T : \mathbb{R}^n \to \mathbb{R}^n$ *can be defined as*: $T\bar{\nu} = \Pi\bar{\nu} + \bar{\chi}$ *for any arbitrary* $\nu \in \mathbb{R}^n$. *As* Π *is a contraction,* T *is also a contraction. Since* $\mathbb{R}^n$ *is a Banach space, there exists a unique fixed point* (*Naylor and Sell 1982*) *of* T, i.e. *the measure vector* $\bar{\mu}$ *satisfying the condition* $T\bar{\mu} = \bar{\mu}$. *Therefore, the language measure vector* $\bar{\mu}$ *is uniquely determined as* $\bar{\mu} = [I - \Pi]^{-1}\bar{\chi}$, *which can be interpreted as the unique fixed point of the contraction operator* Π.

3. Optimal control without event disabling cost

This section presents the theoretical foundations of the optimal supervisory control of deterministic finite state automata (DFSA) plants by selectively disabling controllable events so that the resulting optimal policy can be realized as a controllable supervisor (Fu *et al.* 2004). The plant model is first modified to satisfy the specified operational constraints, if any; this model is referred to as the *unsupervised or open loop* plant in the sequel. Then, starting with the (regular) language of the unsupervised plant, the optimal policy maximizes the performance of the controlled sublanguage of the supervised plant without any further constraints. The performance index of the optimal policy is a signed real measure of the supervised sublanguage, described in § 2, which is expressed in terms of a state transition cost matrix Π and a characteristic vector $\bar{\chi}$, but it does not assign any additional penalty for event disabling.

Let $\mathcal{S} \equiv \{S^0, S^1, \dots, S^N\}$ be the finite set of all supervisory control policies that selectively disable controllable events of the unsupervised plant DFSA G and can be realized as regular languages. Denoting $\Pi^k \equiv \Pi(S^k)$, $k \in \{1, 2, \dots, N\}$, the supervisor S^0 is the null controller (i.e. no event is disabled) implying that $L(S^0/G) = L(G)$. The controller cost matrix $\Pi(S^0) = \Pi^0 \equiv \Pi^{\text{plant}}$ that is the Π-matrix of the unsupervised plant automaton G. For a supervisor S^i, $i \in \{1, 2, \ldots, N\}$, the control policy selectively disables certain controllable events at specific state(s); consequently, the corresponding elements of the $\tilde{\Pi}$-matrix (see Definition 4) become zero. Therefore, the inequalities, $\pi_{ij} \geq 0$ and $\sum_j \pi_{ij} < 1$, hold and $L(S^k/G) \subseteq L(G)$ $\forall S^k \in \mathcal{S}$. The language measure vector of a supervised plant $L(S^k/G)$ is expressed as

$$\bar{\mu}^k \equiv [I - \Pi^k]^{-1} \bar{\chi} \tag{12}$$

where the jth element of the vector $\bar{\mu}^k$ is denoted as μ_j^k. In the sequel, $\bar{\mu}^*$ is chosen to be the performance measure for the optimal control policy without event disabling cost.

Proposition 1: *Let j be such that $\mu_j^k = \min_{\ell \in \{1,2,\ldots,n\}} \mu_\ell^k$. If $\mu_j^k \leq 0$, then $\chi_j \leq 0$; and if $\mu_j^k < 0$, then $\chi_j < 0$.*

Proof: The DFSA satisfies the identity $\mu_j^k = \sum_{\ell \in \{1,2,\ldots,n\}} \pi_{j\ell}^k \mu_\ell^k + \chi_j$ that leads to the inequality $\mu_j^k \geq (\sum_\ell \pi_{j\ell}^k)\mu_j^k + \chi_j \Rightarrow (1 - \sum_\ell \pi_{j\ell}^k)\mu_j^k \geq \chi_j$. The proof follows from $(1 - \sum_\ell \pi_{j\ell}^k) > 0$ (see Definitions 5 and 7). □

Corollary 1: *Let $\mu_j^k = \max_{\ell \in \{1,2,\ldots,n\}} \mu_\ell^k$. If $\mu_j^k \geq 0$, then $\chi_j \geq 0$ and if $\mu_j^k > 0$, then $\chi_j > 0$.*

Proof: The proof is similar to that of Proposition 1. □

Proposition 2: *Given $\Pi(S^k) = \Pi^k$ and $\mu^k \equiv [I - \Pi^k]^{-1} \bar{\chi}$, let Π^{k+1} be generated from Π^k for $k \geq 0$ by disabling or re-enabling the appropriate controllable events as follows: $\forall i, j \in \{1, 2, \ldots, n\}$, ijth element of Π^{k+1} is modified as*

$$\pi_{ij}^{k+1} \begin{cases} \geq \pi_{ij}^k & \text{if } \mu_j^k > 0 \\ = \pi_{ij}^k & \text{if } \mu_j^k = 0 \\ \leq \pi_{ij}^k & \text{if } \mu_j^k < 0 \end{cases} \tag{13}$$

and $\Pi^k \leq \Pi^0$ $\forall k$. Then, $\bar{\mu}^{k+1} \geq \bar{\mu}^k$ elementwise and equality holds if and only if $\Pi^{k+1} = \Pi^k$.

Proof: It follows from the the properties of the measure vector $\bar{\mu}$ that

$$\begin{aligned} \bar{\mu}^{k+1} - \bar{\mu}^k &= \left(\left[I - \Pi^{k+1}\right]^{-1} - \left[I - \Pi^k\right]^{-1}\right)\bar{\chi} \\ &= \left[I - \Pi^{k+1}\right]^{-1}\left([I - \Pi^k] - [I - \Pi^{k+1}]\right) \\ &\quad \times \left[I - \Pi^k\right]^{-1} \bar{\chi} \\ &= \left[I - \Pi^{k+1}\right]^{-1}\left[\Pi^{k+1} - \Pi^k\right]\bar{\mu}^k \end{aligned}$$

Defining the matrix $\Delta^k \equiv \Pi^{k+1} - \Pi^k$, let the jth column of Δ^k be denoted as Δ_j^k. Then, $\Delta_j^k \leq 0$ if $\mu_j^k < 0$ and $\Delta_j^k \geq 0$ if $\mu_j^k \geq 0$, and the remaining columns of Δ^k are zero vectors. This implies that: $\Delta^k \bar{\mu}^k = \sum_j \Delta_j^k \mu_j^k \geq 0$. Since $\Pi^k \leq \Pi^0$ $\forall k$, $[I - \Pi^{k+1}]^{-1} \geq 0$ elementwise. Then, it follows that $[I - \Pi^{k+1}]^{-1} \Delta^k \bar{\mu}^k \geq 0 \Rightarrow \bar{\mu}^{k+1} \geq \bar{\mu}^k$. For $\mu_j^k \neq 0$ and Δ^k as defined above, $\Delta^k \bar{\mu}^k = 0$ if and only if $\Delta^k = 0$. Then, $\Pi^{k+1} = \Pi^k$ and $\bar{\mu}^{k+1} = \bar{\mu}^k$. □

Corollary 2: *For a given state q_j, let $\mu_j^k < 0$ and Π^{k+1} be generated from Π^k by disabling controllable events that lead to the state q_j. Then, $\mu_j^{k+1} < 0$.*

Proof: Since only jth column of $[I - \Pi^{k+1}]$ is different from that of $[I - \Pi^k]$ and the remaining columns are the same, the jth row of the cofactor matrix of $[I - \Pi^{k+1}]$ is the same as that of the cofactor matrix of $[I - \Pi^k]$. Therefore

$$\operatorname{Det}[I - \Pi^{k+1}]\mu_j^{k+1} = \operatorname{Det}[I - \Pi^k]\mu_j^k$$

Since both determinants are real positive by Property 5 of the Π-matrix, μ_j^k and μ_j^{k+1} have the same sign. □

In Proposition 2, some elements of the jth column of Π^k are decreased (or increased) by disabling (or re-enabling) controllable events that lead to the states q_j for which $\mu_j^k < 0$ (or $\mu_j^k \geq 0$). Next it is shown that an optimal supervisor can be achieved to yield best performance in terms of the language measure.

Proposition 3: *Iterations of event disabling and re-enabling lead to a cost matrix Π^* that is optimal in the sense of maximizing the performance vector $\bar{\mu}^* \equiv [I - \Pi^*]^{-1} \bar{\chi}$ elementwise.*

Proof: Let us consider an arbitrary cost matrix $\tilde{\Pi} \leq \Pi^0$ and $\tilde{\bar{\mu}} \equiv [I - \tilde{\Pi}]^{-1} \bar{\chi}$. It will be shown that $\tilde{\bar{\mu}} \leq \bar{\mu}^*$. Let us rearrange the elements of the $\bar{\mu}^*$-vector such that

$$\bar{\mu}^* = [\underbrace{\mu_1^* \quad \cdots \quad \mu_\ell^*}_{\geq 0} \mid \underbrace{\mu_{\ell+1}^* \quad \cdots \quad \mu_n^*}_{<0}]^{\mathrm{T}}$$

and the cost matrices $\tilde{\Pi}$ and Π^* are also rearranged in the order in which the $\bar{\mu}^*$-vector is arranged.

According to Proposition 2, no controllable event leading to states q_k, $k = 1, 2, \ldots, \ell$, is disabled and all controllable events leading to states q_k, $k = \ell + 1, \ell + 2, \ldots, n$, are disabled. Therefore, the elements in the first ℓ columns of Π^* are the same as those of the Π^0 and only the elements in the last $(n - \ell)$ columns are decreased to the maximum permissible extent by disabling all controllable events. In contrast, the columns of $\tilde{\Pi}$ are reduced by an arbitrary choice. Therefore, defining $\Delta\Pi^* \equiv [\tilde{\Pi} - \Pi^*]$, the first ℓ columns of $\Delta\Pi \leq 0$ and the last $(n - \ell)$ columns of $\Delta\Pi \geq 0$.

Since

$$\bar{\mu}^* = [\underbrace{\mu_1^* \quad \cdots \quad \mu_\ell^*}_{\geq 0} \mid \underbrace{\mu_{\ell+1}^* \quad \cdots \quad \mu_n^*}_{<0}]^{\mathrm{T}}$$

and $[I - \tilde{\Pi}]^{-1} \geq 0$ elementwise, and $\tilde{\bar{\mu}} - \bar{\mu}^* = [I - \tilde{\Pi}]^{-1}[\tilde{\Pi} - \Pi^*]\mu^*$, it follows that

$$\tilde{\bar{\mu}} - \bar{\mu}^* = \underbrace{[I - \tilde{\Pi}]^{-1}}_{\geq 0} \left(\underbrace{\sum_{j=1}^{\ell} \mathrm{Col}_j \cdot \mu_j^*}_{\leq 0} + \underbrace{\sum_{j=\ell+1}^{n} \mathrm{Col}_j \cdot \mu_j^*}_{\leq 0} \right) \leq 0$$

where Col_j indicates jth column of the matrix $[\tilde{\Pi} - \Pi^*]$.

Therefore, $\tilde{\bar{\mu}} \leq \bar{\mu}^*$ for any arbitrary choice of $0 \leq \tilde{\Pi} \leq \Pi^0$. □

Proposition 4: *The control policy induced by the optimal Π^*-matrix in Proposition 3 is unique in the sense that the controlled language is most permissive (i.e. least restrictive) among all controller(s) having the best performance.*

Proof: Disabling controllable event(s) leading to a state q_j with performance measure $\mu_j^* = 0$ does not alter the performance vector $\bar{\mu}^*$. The optimal control does not disable any controllable event leading to the states with zero performance. This implies that, among all controllers with the identical performance $\bar{\mu}^*$, the control policy induced by the Π^*-matrix is most permissive. □

Propositions 3 and 4 suffice to conclude that the Π^*-matrix yields the most permissive controller with the best performance $\bar{\mu}^*$. The optimal control policy (without event disabling cost) can be realized as:

- all controllable events leading to the states q_j, for which $\mu_j^* < 0$, are disabled;
- all controllable events leading to the states q_j, for which $\mu_j^* \geq 0$, are enabled.

3.1. *Optimal policy construction without event disabling cost*

A procedure is proposed for construction of the optimal control policy that maximizes the performance of the controlled language of DFSA (without event disabling cost), starting from any initial state $q \in Q$. Let G be a DFSA plant model without any constraint (i.e., operational specifications) and have the state transition cost matrix of the unsupervised plant as: $\Pi^0 \equiv \Pi^{\mathrm{plant}} \in \Re^{n \times n}$ and the characteristic vector as: $\bar{\chi} \in \Re^n$. Then, the performance vector at $k = 0$ is given as: $\bar{\mu}^0 = [\mu_1^0 \; \mu_2^0 \; \cdots \; \mu_n^0]^{\mathrm{T}} = (I - \Pi^0)^{-1}\bar{\chi}$, where the jth element μ_j^0 of the vector $\bar{\mu}^0$ is the performance of the language, with state q_j as the initial state. Then, $\mu_j^0 < 0$ implies that, if the state q_j is reached, then the plant will yield bad performance thereafter. Intuitively, the control system should attempt to prevent the automaton from reaching q_j by disabling all controllable events that lead to this state. Therefore, the optimal control algorithm starts with disabling all controllable events that lead to every state q_j for which $\mu_j^0 < 0$. This is equivalent to reducing all elements of the corresponding columns of the Π^0-matrix by disabling those controllable events. In the next iteration, i.e. $k = 1$, the updated cost matrix Π^1 is obtained as: $\Pi^1 = \Pi^0 - \Delta^0$ where $\Delta^0 \geq 0$ (the inequality being implied elementwise) is composed of event costs corresponding to all controllable events that have been disabled. Using Proposition 2, $\bar{\mu}^0 \leq \bar{\mu}^1 \equiv [I - \Pi^1]^{-1}\bar{\chi}$. Although all controllable events leading to every state corresponding to a negative element of μ^1 are disabled, some of the controllable events that were disabled at $k = 0$ may now lead to states corresponding to positive elements of $\bar{\mu}^1$. Performance could be further enhanced by re-enabling these controllable events. For $k \geq 1$, $\Pi^{k+1} = \Pi^k + \Delta^k$ where $\Delta^k \geq 0$ is composed of the state transition costs of all re-enabled controllable events at k.

If $\bar{\mu}^0 \geq 0$, i.e. there is no state q_j such that $\mu_j^0 < 0$, then the plant performance cannot be improved by event disabling and the null controller S^0 (i.e. no disabled event) is the optimal controller for the given plant. Therefore, the cases are considered where $\mu_j^0 < 0$ for some state q_j.

Starting with $k = 0$ and $\Pi^0 \equiv \Pi^{\mathrm{plant}}$, the control policy is constructed by the following two-step procedure.

Step 1: For every state q_j for which $\mu_j^0 < 0$, disable controllable events leading to q_j. Now, $\Pi^1 = \Pi^0 - \Delta^0$, where $\Delta^0 \geq 0$ is composed of event costs corresponding to all controllable events, leading to q_j for which $\mu_j^0 < 0$, which have been disabled at $k = 0$.

Step 2: For $k \geq 1$, if $\mu_j^k \geq 0$, re-enable all controllable events leading to q_j, which were disabled in Step 1. The cost matrix is updated as: $\Pi^{k+1} = \Pi^k + \Delta^k$ for $k \geq 1$, where $\Delta^k \geq 0$ is composed of event costs corresponding to all currently re-enabled controllable events. The iteration is terminated if no controllable event leading to q_j remains disabled for which $\mu_j^k > 0$. At this stage, the optimal performance $\bar{\mu}^* \equiv [I - \Pi^*]^{-1}\bar{\chi}$.

Proposition 5: *The number of iterations needed to arrive at the optimal control law without event disabling cost does not exceed the number, n, of states of the DFSA.*

Proof: Following Proposition 2, the sequence of performance vectors $\{\Pi^k\}$ in successive iterations of the two-step procedure is monotonically increasing. The first iteration at $k = 0$ disables controllable events following Step 1 of the two-step procedure in § 3.1. During each subsequent iteration in Step 2, the controllable events leading to at least one state are re-enabled. When Step 2 is terminated, there remains at least one negative element, $\mu_j^k < 0$ by 2. Therefore, as the number of iterations in Step 2 is at most $n - 1$, the total number of iterations to complete the two-step procedure does not exceed n. □

Since each iteration in the synthesis of the optimal control requires a single Gaussian elimination of n unknowns from n linear algebraic equations, computational complexity of the control algorithm is polynomial in n.

4. Optimal control with event disabling cost

This section presents the optimal control policy with (state-based) event disabling cost by including the cost of all (controllable) events, disabled by the supervisor, in the performance cost; the disabling cost is incurred each time the event is disabled at the state. As the cost of disabled event(s) approaches zero, the optimal control policy with event disabling cost converges to the optimal control policy without event disabling cost, described in § 3.

Definition 8: Let the cost of disabling a (controllable) event σ_j that causes transition from q_i be denoted as c_{ij} where $c_{ij} \in [0, 1]$. The $(n \times m)$ disabling cost matrix is denoted as $C = [c_{ij}]$.

Since the (controllable) supervisor never disables any uncontrollable event, the entries c_{ij} for uncontrollable events have no importance. For implementation, they can be set to an arbitrarily large positive $M < \infty$.

Definition 9: The action of disabling a (controllable) event σ_j at state q_i by a supervisor S is defined as

$$d_{ij}^S = \begin{cases} 1 & \text{if } \sigma_j \text{ is disabled at state } q_i \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

The $(n \times m)$ action matrix of disabling controllable events by a supervisor S is denoted as: $D^S = [d_{ij}^S]$.

Definition 10: The event disabling cost characteristic of a supervisor S that selectively disables controllable events σ_j at state q_i is defined as:

$$\gamma_i^S = \sum_{j:d_{ij}^S=1} c_{ij}\,\tilde{\pi}_{ij} \tag{15}$$

The disabling cost characteristic is proportional to event cost of the controllable event disabled by the supervisor S.

The $(n \times 1)$ disabling cost characteristic vector of a supervisor S is denoted as: $\bar{\gamma}^S \equiv [\gamma_1^S \ \ \gamma_2^S \ \ \cdots \ \ \gamma_n^S]^{\mathrm{T}}$.

Definition 11: The modified characteristic of a state $q_i \in Q$ is defined as

$$\chi_i^S \equiv \chi_i - \gamma_i^S. \tag{16}$$

The $(n \times 1)$ modified characteristic vector under a supervisor S is defined as

$$\bar{\chi}^S \equiv \bar{\chi} - \bar{\gamma}^S \tag{17}$$

where $\bar{\chi}^S \equiv [\chi_i^S \ \ \chi_i^S \ \ \cdots \ \ \chi_n^S]^{\mathrm{T}}$.

Definition 12: The disabling cost measure vector under a supervisor S is defined as

$$\bar{\theta}^S \equiv [I - \Pi^S]^{-1}\bar{\gamma}^S \tag{18}$$

with θ_i^S being the ith element of $\bar{\theta}^S$, which is the disabling cost incurred by the state.

Definition 13: The performance measure vector of a supervisor S is defined as

$$\bar{\eta}^S \equiv [I - \Pi^S]^{-1}\bar{\chi}^S \tag{19}$$

with η_i^S being the ith element of $\bar{\eta}^S$.

The performance index vector $\bar{\eta}^S$of a supervisor S can be interpreted as the difference between the measure vector $\bar{\mu}^S$ of the supervised language $L(S/G)$ of the DFSA G and the respective disabling cost measure vector $\bar{\theta}^S$. That is

$$\bar{\eta}^S = \bar{\mu}^S - \bar{\theta}^S. \tag{20}$$

Following the approach taken for optimal control without event disabling cost in § 3, let $\mathcal{S} \equiv \{S^0, S^1, \ldots, S^N\}$ be the finite set of supervisory control policies that can be realized as regular languages. For a supervisor $S^k \in \mathcal{S}$, the control policy selectively disables certain controllable events. Consequently, the corresponding elements of the $\tilde{\Pi}$-matrix become zero and those of the event disabling characteristic vector $\bar{\gamma}^S$ are entered in the modified characteristic vector $\bar{\chi}^S$ as seen in Definition 11; therefore, $L(S^k/G) \subseteq L(G)\ \forall S^k \in \mathcal{S}$. Denoting $\Pi^k \equiv \Pi(S^k)$, $\quad k \in \{1, 2, \ldots, N\}$, the performance measure vector (with event disabling cost) of the supervised plant $L(S^k/G)$ is expressed as

$$\bar{\eta}^k \equiv [I - \Pi^k]^{-1}(\bar{\chi} - \bar{\gamma}^k) \tag{21}$$

where $\bar{\eta}^k \equiv \bar{\eta}^{s^k}$, $\bar{\gamma}^k \equiv \bar{\gamma}^{s^k}$, and jth element of the vector $\bar{\eta}^k$ is denoted as η_j^k. The null supervisor S^0 (i.e. no disabled event) has zero disabling cost, i.e. $\bar{\gamma}^0 = 0$ and consequently $\bar{\eta}^0 = \bar{\mu}^0$. The construction of optimal policy is extended to include the event disabling cost.

4.1. *Optimal policy construction with event disabling cost*

This subsection formulates an optimal control policy with event disabling cost, which maximizes all elements of the performance vector $\bar{\eta}^S$ of the supervised language of a DFSA G with event cost matrix $\tilde{\Pi} \in \Re^{n \times m}$; state transition cost matrix $\Pi \in \Re^{n \times n}$; characteristic vector $\bar{\chi} \in \Re^n$; and the disabling cost matrix $C \in \Re^{n \times m}$. For the unsupervised plant, the initial conditions of the optimal synthesis procedure are set as: $\Pi^0 \equiv \Pi^{\text{plant}}$; $\bar{\chi}^0 = \bar{\chi}$; $\bar{\gamma}^0 = 0$; $D^0 = \mathbf{0}$ (no event disabled so far). For optimal control without event disabling cost in § 3.1, all controllable events that lead to states q_ℓ, for which $\mu_\ell^0 < 0$ are first disabled. Subsequently, for $k \geq 1$, all previously disabled controllable events leading to q_j are re-enabled if $\mu_j^k \geq 0$. In contrast, for optimal control with event disabling cost, the first action is to disable all controllable events σ_j leading to states q_ℓ for which $\eta_\ell^0 < -c_{ij}$ with $\delta(q_i, \sigma_j) = q_\ell$. Subsequently, for $k \geq 1$, these disabled events are re-enabled if $\eta_\ell^k \geq -c_{ij}$. The rationale for this procedure is that disabling of controllable events leading to states with small negative performance may not be advantageous because of incurring additional event disabling cost.

The control policy with event disabling cost is constructed by the following two-step procedure.

Step 1: Starting at $k = 0$ and $\Pi^0 \equiv \Pi^{\text{plant}}$, disable all controllable events σ_j, leading to each state q_ℓ if the inequality: $\eta_\ell^0 < -c_{ij}$ with $\delta(q_i, \sigma_j) = q_\ell$ is satisfied. The algorithm for dealing with this inequality is delineated below:

- If the inequality is not satisfied for any single case, stop the iterative procedure. No event disabling can improve the plant performance beyond that of the open loop plant, i.e., the null supervisor S^0 achieves optimal control.
- If the inequality is satisfied for at least one case, disable the qualified event(s) and update the state transition cost matrix to $\Pi^1 \leq \Pi^0$ (elementwise); the disabling matrix to D^1 for generating the cost characteristic function $\bar{\gamma}^1$; and the modified characteristic vector $\bar{\chi}^1 \equiv \bar{\chi} - \bar{\gamma}^1$. Go to Step 2.

Step 2: This step starts at $k = 1$ and the performance measure vector for $k \geq 1$ is

$$\bar{\eta}^k \equiv [I - \Pi^k]^{-1} \bar{\chi}^k = [I - \Pi^k]^{-1} (\bar{\chi} - \bar{\gamma}^k)$$

The algorithm at Step 2 re-enables all previously (i.e. at $k \geq 0$) disabled controllable events σ_j that lead to states q_ℓ if the inequality $\eta_\ell^k \geq -c_{ij}$ with $\delta(q_i, \sigma_j) = q_\ell$ is satisfied. The algorithm for dealing with this inequality is as follows:

- If the inequality is not satisfied for any single case, an optimal control is achieved and the iterative procedure is complete. No further event re-enabling can improve the controlled plant performance beyond that of the current supervisor that is the optimal controller.
- If the inequality is satisfied for at least one case, re-enable all qualified events and update the state transition cost matrix to $\Pi^{k+1} \geq \Pi^k$ (elementwise); the disabling matrix to D^k; the cost characteristic function to $\bar{\gamma}^{k+1}$; and the modified characteristic vector $\bar{\chi}^{k+1} \equiv \bar{\chi} - \bar{\gamma}^{k+1}$. Update $k \leftarrow (k+1)$ and repeat Step 2 until the inequality $\eta_\ell^k \geq -c_{ij}$ with $\delta(q_i, \sigma_j) = q_\ell$ is not satisfied for all j and ℓ. Then, the current supervisor is optimal in terms of the performance measure in Definition 13.

The above procedure for optimal control with event disabling cost is an extension of that without event disabling cost described in § 3.1. For zero event disabling cost, the two procedures become identical. Following the rationale of Proposition 5, the computational complexity of the control synthesis with disabling cost is also polynomial in n.

The underlying theory of unconstrained optimal control with event disabling cost is presented as two additional propositions, which simultaneously maximize all elements of the performance vector $\bar{\eta}$.

Proposition 6: *For all supervisors S^k in the iterative procedure, $\bar{\eta}^{k+1} \geq \bar{\eta}^k$ elementwise.*

Proof: Given $\bar{\chi}^k \equiv \bar{\chi} - \bar{\gamma}^k$ and $\bar{\eta}^k \equiv [I - \Pi^k]^{-1} \bar{\chi}^k$, let us denote the change in event disabling characteristic vector as

$$\bar{\omega}^k \equiv \bar{\gamma}^{k+1} - \bar{\gamma}^k = \bar{\chi}^k - \bar{\chi}^{k+1}.$$

Note that, elementwise

$$\bar{\omega}^k \begin{cases} > 0 & \text{for event disabling} \\ \leq 0 & \text{for event re-enabling} \end{cases}$$

The performance increment at iteration k is given by:

$$\begin{aligned}
\bar{\eta}^{k+1} - \bar{\eta}^k &= \left[I - \Pi^{k+1}\right]^{-1} \bar{\chi}^{k+1} - \left[I - \Pi^k\right]^{-1} \bar{\chi}^k \\
&= \left[I - \Pi^{k+1}\right]^{-1} \left[\bar{\chi}^k - \bar{\omega}^k\right] - \left[I - \Pi^k\right]^{-1} \bar{\chi}^k \\
&= \left(\left[I - \Pi^{k+1}\right]^{-1} - \left[I - \Pi^k\right]^{-1}\right) \bar{\chi}^k \\
&\quad - \left[I - \Pi^{k+1}\right]^{-1} \bar{\omega}^k \\
&= \left[I - \Pi^{k+1}\right]^{-1} \left[\Pi^{k+1} - \Pi^k\right] \left[I - \Pi^k\right]^{-1} \bar{\chi}^k \\
&\quad - \left[I - \Pi^{k+1}\right]^{-1} \bar{\omega}^k \\
&= -\left\{\left[I - \Pi^{k+1}\right]^{-1} \left[\Pi^k - \Pi^{k+1}\right] \bar{\eta}^k \right. \\
&\quad \left. + \left[I - \Pi^{k+1}\right]^{-1} \bar{\omega}^k \right\}.
\end{aligned}$$

At $k = 0$, the state transition cost matrix changes from Π^0 to Π^1 as a result of disabling selected controllable events leading to states with sufficiently negative performance. Let the ith column of a matrix A be denoted as $(A)_i$; ijth element of a matrix A be denoted as $(A)_{ij}$, and the ith element of a vector v be denoted as $(v)_i$; and let ℓ and j satisfy the following conditions: $\delta(q_\ell, \sigma_j) = q_p$ and $d_{\ell j}^{S^k} \neq d_{\ell j}^{S^{k+1}}$.

Then, $\Pi^1 \le \Pi^0$; $\omega_\ell^0 = \sum_j c_{\ell j}\{\tilde{\Pi}^0 - \tilde{\Pi}^1\}_{\ell j}$, and

$$(\bar{\eta}^1 - \bar{\eta}^0)_i = -\Big([I-\Pi^1]^{-1}[\Pi^0-\Pi^1]\bar{\eta}^0 - [I-\Pi^1]^{-1}\bar{\omega}^0\Big)_i$$
$$= -\sum_\ell \Big([I-\Pi^1]^{-1}\Big)_{i\ell}\left(\sum_p\left(\sum_j(\tilde{\pi}_{\ell j}\eta_p^0 + c_{\ell j}\tilde{\pi}_{\ell j})\right)\right)$$
$$= -\sum_\ell \Big([I-\Pi^1]^{-1}\Big)_{i\ell}\left(\sum_p\left(\sum_j\tilde{\pi}_{\ell j}(\eta_p^0 + c_{\ell j})\right)\right).$$

Since $[I-\Pi^1]^{-1} \ge 0$ elementwise and event disabling requires $(\eta_p^0 + c_{\ell j}) < 0$ for all admissible ℓ, j and p (see Step 1 of the control policy with event disabling cost in §4.1), it follows from the above equation that $\bar{\eta}^1 - \bar{\eta}^0 \ge 0$ elementwise.

Next, iterations $k \ge 1$ are considered, for which some of the events disabled at $k=0$ are (possibly) re-enabled

$$\omega_\ell^k = -\sum_j c_{\ell j}\Big(\tilde{\Pi}^{k+1} - \tilde{\Pi}^k\Big)_{\ell j}$$

$$\Big(\bar{\eta}^{k+1} - \bar{\eta}^k\Big)_i = \Big(\big[I-\Pi^{k+1}\big]^{-1}\big[\Pi^{k+1}-\Pi^k\big]\bar{\eta}^k - \big[I-\Pi^{k+1}\big]^{-1}\bar{\omega}^k\Big)_i$$
$$= \sum_\ell\Big(\big[I-\Pi^{k+1}\big]^{-1}\Big)_{i\ell} \times\left(\sum_p\left(\sum_j\tilde{\pi}_{\ell j}(\eta_p^k + c_{\ell j})\right)\right).$$

Since $[I-\Pi^k]^{-1} \ge 0$ elementwise and event re-enabling requires $(\eta_p^k + c_{\ell j}) \ge 0$ for all admissible ℓ, j and p (see Step 1 of the control policy with event disabling cost in §4.1), it follows from the above equations that $\bar{\eta}^{k+1} - \bar{\eta}^k \ge 0$ for $k \ge 0$. □

Proposition 7: *The supervisor generated upon completion of the algorithm in* §4.1 *is optimal in terms of the performance in Definition* 13.

Proof: Based on the algorithm in §4.1, let the supervisor S^* be synthesized by disabling and re-enabling certain controllable events at selected states. It is to be shown that S^* is optimal in the following sense. The performance $\bar{\eta}$ of any (controllable) supervisor $S \in \mathcal{S}$ is not superior to the performance $\bar{\eta}^*$ of S^*, i.e. $\bar{\eta}^* \ge \bar{\eta}$, $\forall S \in \mathcal{S}$.

Let an arbitrary supervisor $S \in \mathcal{S}$ disable ceratin controllable events σ_j at selected states q_ℓ, which are not disabled by S^*. Then, by Step 1 of the control policy with event disabling cost in §4.1, it follows that $(\eta_p^* + c_{\ell j}) \ge 0$ with $\delta(q_\ell, \sigma_j) = q_p$. Let the same supervisor S enable some other controllable events σ_k at selected states q_ℓ, leading to state q_r, which are disabled by S^*, i.e. $(\eta_r^* + c_{\ell k}) < 0$ with $\delta(q_\ell, \sigma_k) = q_r$. In non-trivial cases, ℓ, j and k satisfy the conditions: $d_{\ell j}^{S^*} \ne d_{\ell j}^S$ and $d_{\ell k}^{S^*} \ne d_{\ell k}^S$.

Denoting the differences between event disabling characteristic vectors and the state transition cost matrices of S^* and S, respectively, as

$$\bar{\omega} \equiv \bar{\gamma}^S - \bar{\gamma}^{S^*} = \bar{\chi}^{S^*} - \bar{\chi}^S$$
$$\Delta \equiv \Pi^{S^*} - \Pi^S$$

the difference between corresponding performance vectors is obtained as

$$\bar{\eta}^* - \bar{\eta} = \Big[I-\Pi^{S^*}\Big]^{-1}\bar{\chi}^{S^*} - \big[I-\Pi^S\big]^{-1}\bar{\chi}^S$$
$$= \Big[I-\Pi^{S^*}\Big]^{-1}\bar{\chi}^{S^*} - \big[I-\Pi^S\big]^{-1}\Big[\bar{\chi}^{S^*} - \bar{\omega}\Big]$$
$$= \left(\Big[I-\Pi^{S^*}\Big]^{-1} - \big[I-\Pi^S\big]^{-1}\right)\bar{\chi}^{S^*} + \big[I-\Pi^S\big]^{-1}\bar{\omega}$$
$$= \big[I-\Pi^S\big]^{-1}\Big[\Pi^{S^*} - \Pi^S\Big]\Big[I-\Pi^{S^*}\Big]^{-1}\bar{\chi}^{S^*} + \big[I-\Pi^S\big]^{-1}\bar{\omega}$$
$$= \big[I-\Pi^S\big]^{-1}\Big[\Pi^{S^*} - \Pi^S\Big]\bar{\eta}^{S^*} + \big[I-\Pi^S\big]^{-1}\bar{\omega}$$
$$= \big[I-\Pi^S\big]^{-1}\Big(\Delta\,\bar{\eta}^{S^*} + \bar{\omega}\Big).$$

Since the matrix $[I-\Pi^S]^{-1} \ge 0$, it suffices to show that $(\bar{\eta}^{S^*} + \bar{\omega}) \ge 0$ to prove $(\bar{\eta}^{S^*} - \bar{\eta}^S) \ge 0$ elementwise. The proof will make use of the fact that $\pi_{i\ell} = \sum_u \tilde{\pi}_{iu}$, where $\sigma_u \in \Sigma$: $\delta(\sigma_u, q_i) = q_\ell$ for all pairs of states q_i and q_ℓ (see Definitions 5 and 7) under any given supervisor S. Then, the matrix Δ can be partitioned as two sets of columns such that, for $\ell \in \{1, 2, \ldots, n\}$, (non-zero) elements in each row of these column sets are obtained as

$$\Delta_{\ell p} = \sum_{j:d_{\ell j}^S=1}\tilde{\pi}_{\ell j} \quad \text{and} \quad \Delta_{\ell r} = -\sum_{k:d_{\ell k}^{S^*}=1}\tilde{\pi}_{\ell k}$$

where the subscript p depends on both ℓ and j, and the subscript r depends on both ℓ and k. The product of the matrix $\Delta \in \Re^{n\times n}$ and the performance vector $\bar{\eta}^* \in \Re^n$ is derived as

$$(\Delta\,\bar{\eta}^*)_\ell \equiv \sum_p \Delta_{\ell p}\eta_p^* + \sum_r \Delta_{\ell r}\eta_r^*$$
$$= \sum_p\left(\sum_j\tilde{\pi}_{\ell j}\,\eta_p^*\right) - \sum_r\left(\sum_k\tilde{\pi}_{\ell k}\eta_r^*\right).$$

Similarly, the change in the event disabling characteristic vector is expressed as

$$\begin{aligned}(\bar{\omega})_\ell &= \sum_i \tilde{\pi}_{\ell i} c_{\ell i}\left(d^S_{\ell i} - d^{S^*}_{\ell i}\right)\\ &= \sum_p \left(\sum_j \tilde{\pi}_{\ell j} c_{\ell j}\right) - \sum_r \left(\sum_k \tilde{\pi}_{\ell k}\, c_{\ell k}\right)\end{aligned}$$

Summing the above two expressions yields the ℓth element of the vector $(\Delta\bar{\eta}^{S^*} + \bar{\omega})$ as

$$\begin{aligned}(\Delta\bar{\eta}^* + \bar{\omega})_\ell &= \sum_p \left(\sum_j \tilde{\pi}_{\ell j}\, \eta^*_p\right) - \sum_r \left(\sum_k \tilde{\pi}_{\ell k}\, \eta^{S^*}_r\right)\\ &\quad + \sum_p \left(\sum_j \tilde{\pi}_{\ell j} c_{\ell j}\right) - \sum_r \left(\sum_k \tilde{\pi}_{\ell \tilde{j}} c_{\ell \tilde{j}}\right)\\ &= \sum_p \left(\sum_j \tilde{\pi}_{\ell j}\, (\eta^*_p + c_{\ell j})\right)\\ &\quad - \sum_r \left(\sum_k \tilde{\pi}_{\ell k}\, (\eta^*_r + c_{\ell k})\right) \geq 0\end{aligned}$$

because $(\eta^*_p + c_{\ell j}) \geq 0$ and $(\eta^*_r + c_{\ell k}) < 0$ as stated at the beginning of the proof. □

5. Examples of discrete event optimal supervisory control

This section presents two examples to demonstrate different applications of discrete-event optimal supervisors. The first example addresses health monitoring of a twin-engine unmanned aircraft that is used for surveillance and data collection. The second example presents controlled interactions of a multiprocessor message decoding system.

5.1. *Optimal supervisory control of a twin-engine unmanned aircraft*

The control objective is to enhance engine safety operation. Engine health and operating conditions, which are monitored in real time based on avionic sensor information, are classified into three mutually exclusive and exhaustive categories: (i) *good*; (ii) *unhealthy* (*but operable*); and (iii) *inoperable*. Upon occurrence of any observed abnormality, the supervisor decides to continue or abort the mission.

The deterministic finite state automaton model of the unsupervised plant (i.e. engine operation) has 13 states, of which three are marked (i.e., accepted) states, and nine events, of which four are controllable. The dump state is not included as it is not of interest in the supervisory control synthesis (Ramadge and Wonham 1987, Fu *et al*. 2004). All events are assumed to be observable. The states and events of the plant model are listed in tables 1 and 2, respectively. As indicated in table 1, the marked states are: 11, 12 and 13, of which the states 11 and 13 are bad marked states, and the state 12 is a good marked state.

The state transition function δ (see the beginning of §2), the entries $\tilde{\pi}_{ij}$ (see Definition 4) of the event cost matrix $\tilde{\Pi}$, and the entries c_{ij} (see Definition 8) of the event disabling cost matrix C are entered simultaneously in relevant cells of Table 3. The dump state and any transitions to the dumped state are not shown in Table 3. The empty cells in Table 3 imply that the state transition function δ is undefined for the respective state and event. In each non-empty cell in Table 3, the

State	Description
1	Safe in base
2	Mission executing—two good engines
3	One engine unhealthy during mission execution
4	Mission executing—one good and one unhealthy engine
5	Both engines unhealthy during mission execution
6	One engine good and one engine inoperable
7	Mission execution with two unhealthy engines
8	Mission execution with only one good engine
9	One engine unhealthy and one engine inoperable
10	Mission execution with only one unhealthy engine
11	Mission aborted/not completed (bad marked state)
12	Mission successful (good marked state)
13	Aircraft destroyed (bad marked state)

Table 1. Plant automaton states of the aircraft engine system.

Event	Event description	Controllable events
s	Start and take-off	√
b	A good engine becoming unhealthy	
t	An unhealthy engine becoming inoperable	
v	A good engine becoming inoperable	
k	Keep engine(s) running	√
a	Mission abortion	√
f	Mission completion	
d	Destroyed aircraft	
l	Landing	√

Table 2. Plant event alphabet of the aircraft engine system.

	s	*b*	*t*	*v*	*k*	*a*	*f*	*d*	*l*
1	(2) 0.500 0.000					(1) 0.020 0.005			
2		(3) 0.050 N/A		(6) 0.010 N/A			(12) 0.800 N/A	(3) 0.100 N/A	
3					(4) 0.450 0.050	(11) 0.450 0.005			
4		(5) 0.120 N/A	(6) 0.160 N/A	(9) 0.100 N/A			(12) 0.500 N/A	(13) 0.120 N/A	
5					(7) 0.450 0.080	(11) 0.450 0.002			
6					(8) 0.450 0.010	(11) 0.450 0.004			
7			(9) 0.250 N/A				(12) 0.500 N/A	(13) 0.200 N/A	
8		(9) 0.200 N/A		(13) 0.010 N/A			(12) 0.300 N/A	(13) 0.400 N/A	
9					(10) 0.450 0.35	(11) 0.450 0.002			
10			(13) 0.350 N/A				(12) 0.200 N/A	(13) 0.400 N/A	
11									(1) 0.95 0.000
12									(1) 0.95 0.000
13									

Characteristic vector $\bar{\chi} = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ -0.05\ 0.25\ -1.0]^{\mathrm{T}}$ (see Definition 4).

Table 3. State transition δ event cost $\tilde{\Pi}$ and disabling cost C matrices of the aircraft engine system.

positive integer in the first entry signifies the destination state of the transition; the non-negative fraction in the second entry is the state-based event cost $\tilde{\pi}_{ij}$; and the non-negative fraction in the third entry is the state-based event disabling cost c_{ij} of the four controllable events (i.e. events s, k, a and ℓ); event disabling cost is not applicable to the remaining five uncontrollable events (i.e. events b, t, v, f and d) and the corresponding entries are marked as 'N/A'. (Note that the event cost $\tilde{\pi}_{ij}$ and event disabling cost c_{ij} of a given event could be different at different states.)

The values of $\tilde{\pi}_{ij}$ were selected by extensive simulation experiments on gas turbine engine models and were also based on experience of gas turbine engine operation and maintenance. The state-based event cost $\tilde{\pi}_{ij}$ such that each row sum of the event cost matrix $\tilde{\Pi}$ is strictly less than one as given in Definition 5 and explained in detail in a previous publication (Surana and Ray 2004). The event disabling cost c_{ij} for controllable events indicates the difficulty of disabling from the respective states and the values were chosen based on operational experience. The elements of the characteristic vector (see Definition 4) are chosen as non-negative weights based on the perception of each marked state's role on the gas turbine system performance. In this simulation example, the characteristic value of the good marked state 12 is taken to be 0.25 and those of the bad marked states 11 and 13 are taken to be -0.05 and

− 1.0, respectively, to quantify their respective importance; each of the remaining non-marked states is assigned zero characteristic value as seen at the bottom of table 3. The information provided in table 3 is sufficient to generate the state transition cost matrix Π (see Definition 7).

Based on the data given in tables 1–3, two optimal control policies—Case 1 without event disabling cost and the Case 2 with event disabling cost—have been synthesized following the respective two-step procedures in §§ 3 and 4. The results of optimal supervisor syntheses without and with event disabling cost are presented in tables 4 and 5 supported by respective finite state machine diagrams in figures 1 and 2. For Case 1, the event disabling cost matrix C (i.e. the relevant elements in table 3) are set to zero for synthesis of the optimal control without event disabling cost. In contrast, for Case 2, all elements the event disabling cost matrix C in table 3 are used for synthesis of the optimal control with event disabling cost. At successive iterations, table 4 lists the performance vectors in Case 1: $\bar{\mu}^0$ for the unsupervised (i.e. open loop) plant, $\bar{\mu}^1$ in iteration 1, and $\bar{\mu}^2$ in iteration 2 when the synthesis is completed because of no sign change between elements of $\bar{\mu}^1$ and $\bar{\mu}^2$. Table 4 shows that $\bar{\mu}^2 \geq \bar{\mu}^1 \geq \bar{\mu}^0$ elementwise. This is due to disabling the controllable event k leading to states 7, 8 and 10 as indicated by the dashed arcs in the state transition diagram of figure 1. Consequently, the states 7, 8 and 10 become isolated as there are no other events leading to these states. Starting with the initial state 1, indicated by an external arrow in figure 1, the optimal performance is 0.0850 that is the first element μ_1^2 of the performance vector $\bar{\mu}^2$ as seen in the top right-hand corner in table 4.

Iteration 0	Iteration 1	Iteration 2
0.0823	0.0840	0.0850
0.1613	0.1646	0.1665
0.0062	0.0134	0.0366
−0.0145	0.0500	0.0506
−0.0367	0.0134	0.0138
−0.1541	0.0134	0.0138
−0.1097	−0.0317	−0.0312
−0.3706	−0.3084	−0.3080
−0.2953	0.0134	0.0138
−0.6844	−0.6840	−0.6839
0.0282	0.0298	0.0307
0.3282	0.3298	0.3307
−1.0000	−1.0000	−1.0000

Table 4. Supervised engine performance without event disabling cost.

Iteration 0	Iteration 1	Iteration 2
0.0823	0.0839	0.0841
0.1613	0.1645	0.1649
0.0062	0.0134	0.0188
−0.0145	0.0117	0.0118
−0.0367	−0.0356	−0.0354
−0.1541	0.0034	0.0035
−0.1097	−0.1088	−0.1086
−0.3706	−0.3700	−0.3699
−0.2953	−0.2944	−0.2943
−0.6844	−0.6841	−0.6840
0.0282	0.0297	0.0299
0.3282	0.3297	0.3299
−1.0000	−1.0000	−1.0000

Table 5. Supervised engine performance with event disabling cost.

The results are different for Case 2 because the event disabling cost is taken into account in optimal supervisor synthesis as seen in table 5 and figure 2; in this case, only the state 8 is isolated due to disabling of the controllable event k at the state 6. At successive iterations, table 5 lists the performance vectors for this Case 2 where $\bar{\eta}^0 = \bar{\mu}^0$ for the unsupervised (i.e. open loop) plant; $\bar{\eta}^1$ in iteration 1, and $\bar{\eta}^2$ in iteration 2 when the synthesis is completed because of no sign change between elements of $\bar{\eta}^1$ and $\bar{\eta}^2$. (Note that, in general, the number of iterations needed for supervisor synthesis without and with event disabling cost may not be the same.) Table 5 shows that $\bar{\eta}^2 \geq \bar{\eta}^1 \geq \bar{\eta}^0$ elementwise. This is due to disabling of the controllable event k leading to the state 8 as indicated by the dashed arcs in the state transition diagram of figure 2. Consequently, the state 8 (shown in a dotted circle in figure 2) becomes isolated as there are no other events leading to this state. Starting with the initial state 1, indicated by an arrow in figure 2, the optimal performance is 0.0841 that is the first element $\bar{\eta}_1^2$ of the performance vector $\bar{\eta}^2$ as seen in the top right-hand corner in table 5. Clearly, the supervisor in Case 2 is suboptimal relative to the optimal supervisor in Case 1. That is, the supervised plant performance with event disabling cost cannot exceed the performance for which the event disabling cost is not taken into account.

5.2. *Optimal supervisory control of a three-processor message decoding system*

This section presents the design of a discrete-event (controllable) supervisor for a multiprocessor message decoding system as described below.

Figure 3 depicts the arrangement of the message decoding system, where each of the three processors, p_1, p_2 and p_3, receives encoded message to be decoded. The processor p_3 normally receives the most important

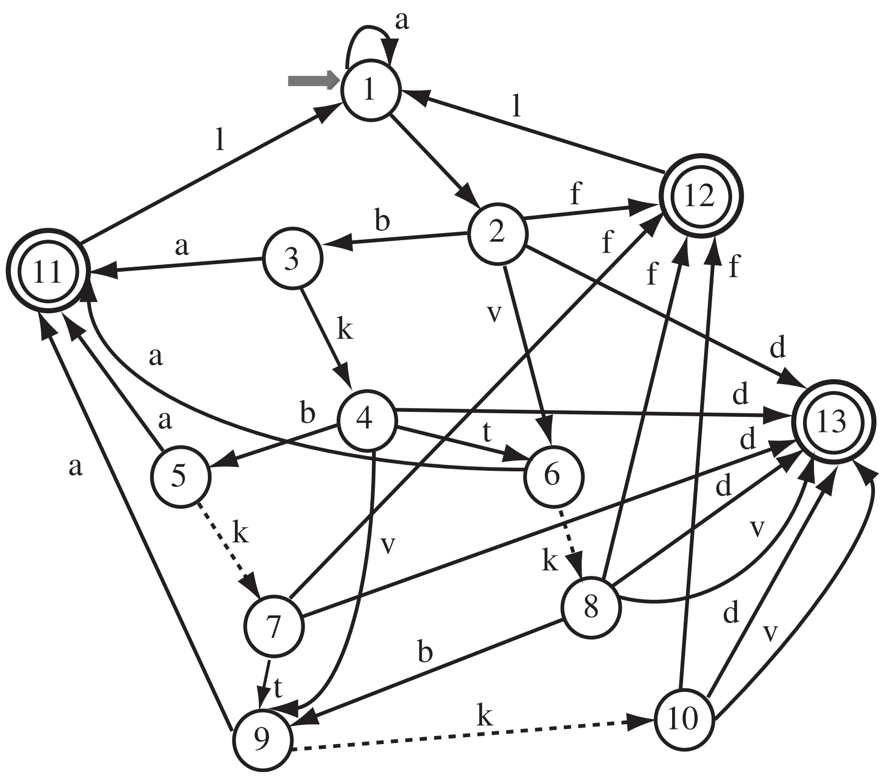

Figure 1. Optimal supervision of engines without disabling cost.

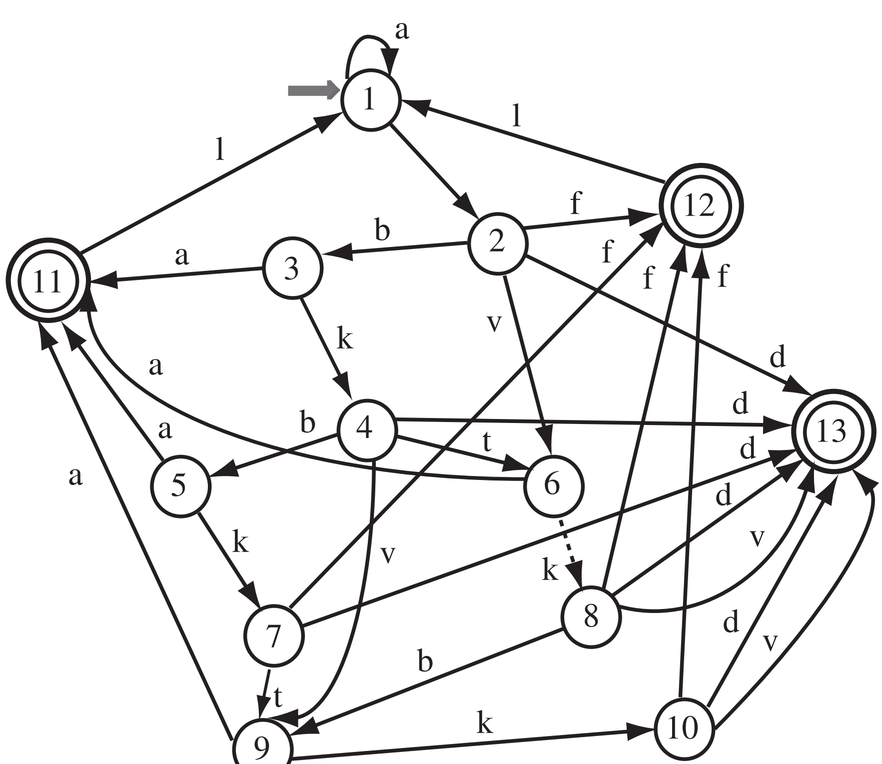

Figure 2. Optimal supervision of engines with disabling cost.

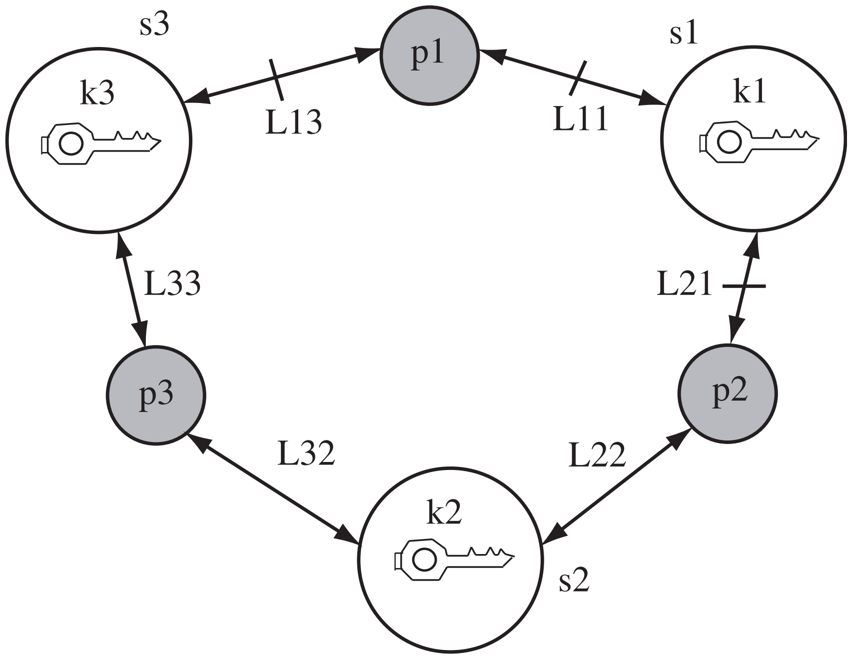

Figure 3. Multiprocessor interactions layout.

messages, and p_1 receives the least important messages. There is a server between each pair of processors—s_1 between p_1 and p_2; s_2 between p_2 and p_3; and s_3 between p_3 and p_1. Each server is connected to each of its two adjacent processors by a link—the server s_j is connected to the adjacent processors p_i and p_k through the links L_{ij} and L_{kj}, respectively. Out of these six links, each of the three links, L_{11}, L_{12} and L_{21}, is equipped with a switch to disable the respective connection whenever it is necessary; each of the remaining three links, L_{22}, L_{32} and L_{33}, always remain connected. Each server s_i is equipped with a decoding key k_i that, at any given time, can only be accessed by only one of the two processors, adjacent to the server, through the link connecting the processor and the server. In order to decode the message, the processor holds the information on both keys of the servers next to it, one at a time. After decoding, the processor simultaneously releases both keys so that other processors may get hold of them.

The unsupervised plant model of the decoding system is depicted as a finite state machine in figure 4, where state 1 is the initial state; the states 26 and 27, shaded in red, are bad marked states representing deadlocks; the states, 11, 12, 13, 14, 15 and 16, shaded in grey, are good marked states; and the remaining unshaded states are unmarked. The event p_{ij} indicates that processor p_i has accessed the key k_j; and the event f_i indicates that the processor p_i has finished decoding and (simultaneously) released both keys in its possession upon completion of decoding. The events f_i are uncontrollable because, after the decoding is initiated, there is no control on when a processor finishes decoding. Table 6 lists the event cost matrix $\tilde{\Pi}$ and the characteristic vector $\bar{\chi}$. In the right-hand column of table 6, positive values are assigned to the χ values of the states 8 to 16 that represent successful decoding of each processor. The χ values of the deadlock states 26 and 27 (where each processor holds exactly one key and hence no processor releases its key) are assigned the negative value of -1. The remaining states are non-marked and are assigned zero χ values. Table 7 lists the event disabling cost matrix C where the entries c_{ij} are non-negative fractions in the first three columns corresponding to the controllable events, p_{11}, p_{13} and p_{21}. These disabling costs are assigned in accordance with the characteristic values of the respective marked states. The entries, marked as 'N/A' correspond to either uncontrollable events or the states at which the plant model does not generate a controllable event.

Based on the optimal control policies described in § 3 and § 4, two supervisors have been synthesized without and with event disabling cost, as shown in figures 5 and 6, respectively. A comparison of these two diagrams reveals that different controllable events are disabled at different states, as indicated by arcs with dashed lines. Specifically, controllable events, causing state transitions $1 \rightarrow 2$, $3 \rightarrow 22$, $4 \rightarrow 19$, $5 \rightarrow 22$, $6 \rightarrow 17$, $7 \rightarrow 18$, $14 \rightarrow 16$, $20 \rightarrow 27$, $23 \rightarrow 26$ and $25 \rightarrow 27$, are disabled under the control policy without event disabling cost, as seen in figure 5. Similarly, controllable events, causing state transitions $1 \rightarrow 2$, $1 \rightarrow 3$, $3 \rightarrow 22$, $4 \rightarrow 19$, $5 \rightarrow 22$, $6 \rightarrow 17$, $7 \rightarrow 20$, $20 \rightarrow 27$, $23 \rightarrow 26$

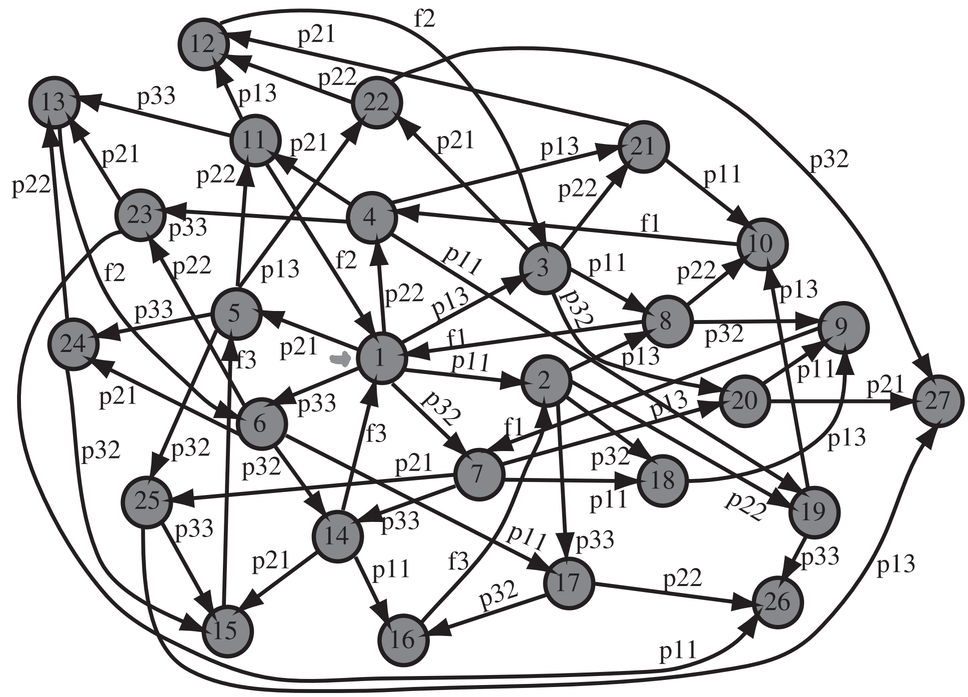

Figure 4. DFSA model of unsupervised processor interactions.

	$p11$ (1)	$p13$ (2)	$p21$ (3)	$p22$ (4)	$p32$ (5)	$p33$ (6)	$f1$ (7)	$f2$ (8)	$f3$ (9)	X
1	0.15	0.11	0.15	0.15	0.15	0.3				0
2		0.1		0.25	0.25	0.3				0
3	0.2		0.2	0.2	0.2					0
4	0.25	0.1	0.25			0.3				0
5		0.1		0.25	0.25	0.3				0
6	0.2		0.2	0.2	0.2					0
7	0.25	0.1	0.25			0.3				0
8				0.35	0.35		0.2			0.01
9							0.5			0.01
10							0.5			0.01
11		0.1				0.3		0.3		0.02
12								0.5		0.02
13								0.5		0.02
14	0.3		0.3						0.3	0.04
15									0.4	0.04
16									0.4	0.04
17				0.4	0.4					0
18		0.1				0.3				0
19		0.1				0.3				0
20	0.4		0.4							0
21	0.4		0.4							0
22				0.4	0.4					0
23	0.4		0.4							0
24				0.4	0.4					0
25		0.1				0.3				0
26										−1
27										−1

Table 6. Event cost matrix $\tilde{\Pi}$ and characteristic vector $\tilde{\chi}$ of the multiprocessor decoding system.

	$p11$ (1)	$p13$ (2)	$p21$ (3)	$p22$ (4)	$p32$ (5)	$p33$ (6)	$f1$ (7)	$f2$ (8)	$f3$ (9)
1	0.1	0.06	0.0333	N/A	N/A	N/A	N/A	N/A	N/A
2	N/A	0.06	N/A	N/A	N/A	N/A	N/A	N/A	N/A
3	0.075	N/A	0.025	N/A	N/A	N/A	N/A	N/A	N/A
4	0.06	0.06	0.02	N/A	N/A	N/A	N/A	N/A	N/A
5	N/A	0.06	N/A	N/A	N/A	N/A	N/A	N/A	N/A
6	0.075	N/A	0.02	N/A	N/A	N/A	N/A	N/A	N/A
7	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
8	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
9	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
10	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
11	N/A	0.06	N/A	N/A	N/A	N/A	N/A	N/A	N/A
12	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
13	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
14	0.05	N/A	0.01667	N/A	N/A	N/A	N/A	N/A	N/A
15	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
16	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
17	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
18	N/A	0.06	N/A	N/A	N/A	N/A	N/A	N/A	N/A
19	N/A	0.06	N/A	N/A	N/A	N/A	N/A	N/A	N/A
20	0.0375	N/A	0.0125	N/A	N/A	N/A	N/A	N/A	N/A
21	0.0375	N/A	0.0125	N/A	N/A	N/A	N/A	N/A	N/A
22	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
23	0.0375	N/A	0.0125	N/A	N/A	N/A	N/A	N/A	N/A
24	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
25	N/A	0.06	N/A	N/A	N/A	N/A	N/A	N/A	N/A
26	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
27	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A

Table 7. Event disabling cost matrix C of the multiprocessor decoding system.

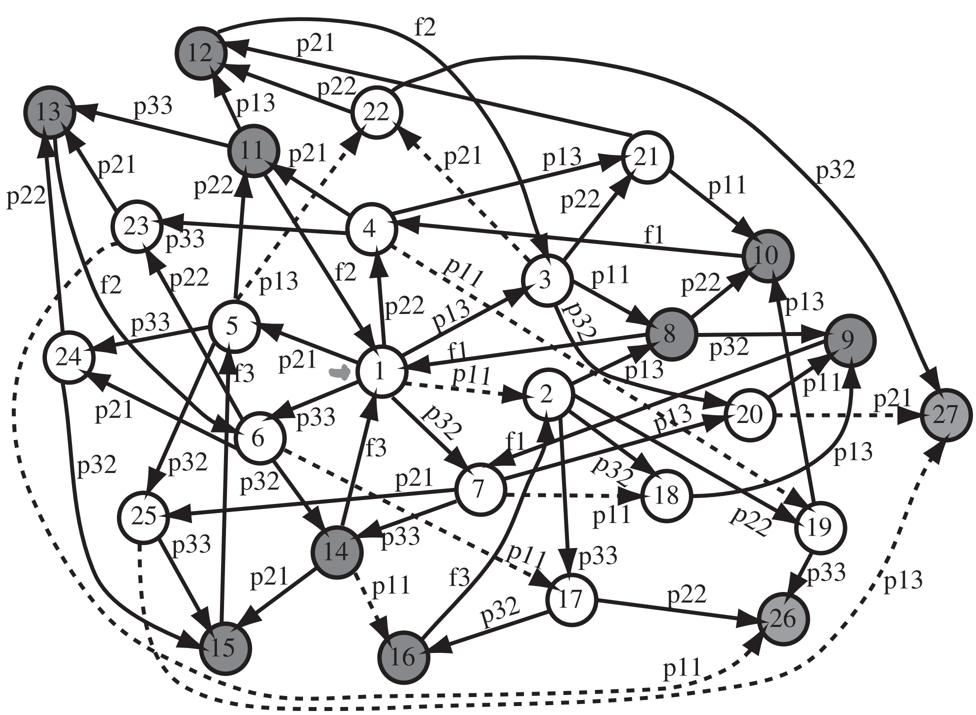

Figure 5. Optimal supervision of processors without disabling cost.

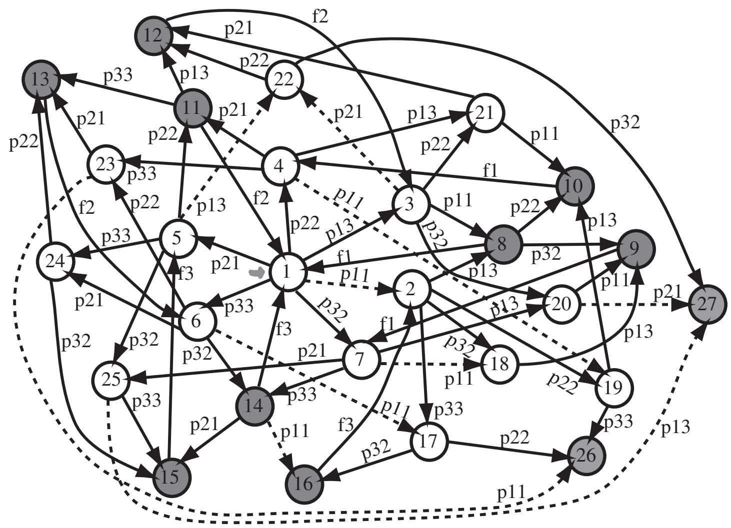

Figure 6. Optimal supervision of processors with disabling cost.

and 25 → 27, are disabled under the the control policy with event disabling cost, as seen in figure 6. The rationale for the difference in event disabling in these two cases is that extremal point(s) on the optimization surface shift due to introduction of non-zero disabling costs in the supervisor synthesis algorithms as described in §§ 3.1 and 4.1.

Tables 8 and 9 show the corresponding performance vectors for optimal control without and with event disabling at each iteration; the optimization is terminated after iterations 2 and 4, respectively, when optimality is reached and the performance cannot be improved any further by event disabling or re-enabling. The performance is non-decreasing at every state from one iteration to the next as seen in tables 8 and 9. Since signs of the performance vector elements have changed for several states at iteration 1 in table 8, the procedure is continued to iteration 2 where it is terminated because of no further sign change. Similarly, in table 9, sign change in the performance vector element(s) continues until iteration 3 and the procedure is terminated at iteration 4 because of no further sign change. It is also seen that, starting from the same condition at iteration 0 for the unsupervised plant, the performance of the supervisor with event disabling cost is always inferior to that without event disabling cost because of the additional penalty. For example, with starting state 1, the performance of the supervised plant without event disabling cost is 0.0161 and that with event disabling cost is −0.0216 while the performance of the unsupervised plant is −0.1646.

6. Summary and conclusions

This paper presents the theory, formulation, and validation of optimal supervisory control policies for dynamical systems, modelled as deterministic finite state automata (DFSA), which may have already been subjected to constraints such as control specifications. The synthesis procedure for optimal control without and with event disabling cost relies on a signed real measure

States	Iteration 0	Iteration 1	Iteration 2
1	−0.1646	0.059	0.0161
2	−0.2141	−0.2030	−0.1991
3	−0.1970	0	0.0104
4	−0.2277	0	0.0145
5	−0.0902	0.0198	0.0223
6	−0.1788	0.0112	0.0207
7	−0.0765	0.0168	0.0224
8	−0.0692	0.0211	0.0267
9	−0.0283	0.0184	0.0212
10	−0.1039	0.0100	0.0172
11	−0.0581	0.0295	0.0365
12	−0.0785	0.0200	0.0252
13	−0.0694	0.0256	0.0303
14	−0.0219	0.0561	0.0595
15	0.0039	0.0479	0.0489
16	−0.0456	−0.0412	−0.0396
17	−0.4183	−0.4165	−0.4159
18	−0.0165	−0.0124	−0.0098
19	−0.3104	−0.3000	−0.2983
20	−0.4113	0	0.0085
21	−0.0729	0	0.0170
22	−0.4134	−0.3920	−0.3899
23	−0.4278	0	0.0121
24	−0.0262	0.0294	0.0317
25	−0.0988	0.0144	0.0147
26	−1.0000	−1.0000	−1.0000
27	−1.0000	−1.0000	−1.0000

Table 8. Supervised processor performance without event disabling cost.

States	Iteration 0	Iteration 1	Iteration 2	Iteration 3	Iteration 4
1	−0.1646	−0.0364	−0.0228	−0.0228	−0.0216
2	−0.2141	−0.2110	−0.2032	−0.2015	−0.2012
3	−0.1970	−0.0194	−0.0045	−0.0011	−0.0005
4	−0.2277	−0.0270	−0.0134	−0.0133	−0.0061
5	−0.0902	0.0034	0.0081	0.0081	0.0082
6	−0.1788	−0.0184	−0.0050	−0.0049	−0.0049
7	−0.0765	−0.0057	0.0031	0.0032	0.0033
8	−0.0692	0.0040	0.0106	0.0107	0.0122
9	−0.0283	0.0072	0.0115	0.0116	0.0117
10	−0.1039	−0.0035	0.0033	0.0033	0.0069
11	−0.0581	0.0063	0.0202	0.0204	0.0208
12	−0.0785	0.0103	0.0177	0.0194	0.0198
13	−0.0694	0.0108	0.0175	0.0175	0.0176
14	−0.0219	0.0282	0.0337	0.0340	0.0343
15	0.0039	0.0414	0.0432	0.0433	0.0433
16	−0.0456	−0.0444	−0.0413	−0.0406	−0.0405
17	−0.4183	−0.4178	−0.4165	−0.4162	−0.4162
18	−0.0165	−0.0126	−0.0112	−0.0110	−0.0110
19	−0.3104	−0.3060	−0.3060	−0.2997	−0.2993
20	−0.4113	−0.0021	−0.0004	−0.0004	−0.0003
21	−0.0729	−0.0200	−0.0079	0.0091	0.0107
22	−0.4134	−0.3959	−0.3929	−0.3922	−0.3921
23	−0.4278	−0.0200	−0.0080	−0.0080	−0.0080
24	−0.0262	0.0209	0.0243	0.0243	0.0243
25	−0.0988	0.0064	0.0070	0.0070	0.0070
26	−1.0000	−1.0000	−1.0000	−1.0000	−1.0000
27	−1.0000	−1.0000	−1.0000	−1.0000	−1.0000

Table 9. Supervised processor performance with event disabling cost.

of regular languages (Ray and Phoha 2003, Surana and Ray 2004) to construct the performance index. The language measure is based on a specified state transition cost matrix and a characteristic vector.

The state-based optimal control policy without event disabling cost maximizes the language measure vector $\bar{\mu}$ to avoid termination on bad marked states by selectively disabling controllable events that may lead to bad marked states and simultaneously ensuring that the remaining controllable events are kept enabled. The goal is to maximize the measure of the controlled plant language without any further constraints. The control policy induced by the updated state transition cost matrix yields maximal performance and is unique in the sense that the controlled language is most permissive (i.e. least restrictive) among all controller(s) having the optimal performance.

The performance measure vector $\bar{\eta}$, for optimal control with disabling cost, is obtained as the language measure vector of the supervised plant minus the disabling cost characteristic vector. The optimal control policy maximizes the performance vector elementwise to avoid termination on bad marked states by selectively disabling controllable events with reasonable disabling costs, and simultaneously ensuring that the remaining controllable events are kept enabled. As the cost of event disabling approaches zero, the optimal control policy with event disabling cost converges to that without event disabling cost.

Derivation of the optimal supervisory control policies requires at most n iterations, where n is the number of states of the DFSA model and each iteration is required to solve a set of n simultaneous linear algebraic equations having complexity of $O(n^3)$ (Ray and Phoha 2003, Surana and Ray 2004). As such computational complexity of the control synthesis procedure is polynomial in the number of DFSA model states. Two examples have been presented to demonstrate applications of discrete-event optimal supervisors. The first example addresses health monitoring of a twin-engine unmanned aircraft that is used for surveillance and data collection. The second example presents controlled interactions of a multiprocessor message decoding system.

The novel concept of language-based control synthesis, presented in this paper, allows quantification of plant performance instead of evaluating its qualitative performance (e.g. permissiveness), which is the current state of the art for discrete event supervisory control (Cassandras and Lafortune 1999). The following conclusion is drawn in view of using the language measure for construction of the performance index for deriving an optimal control policy. Like any other optimization procedure, it is possible to choose different performance indices to arrive at different optimal policies for discrete event supervisory control. Nevertheless, usage of the language measure provides a systematic procedure for precise comparative evaluation of different supervisors so that the optimal control policy(ies) can be unambiguously identified.

There are several issues that need to be addressed for implementation of the theory of discrete-event supervisory control in an operating plant. For example, the events must be generated in real time, based on physical measurements, to provide the supervisor with the current information on the plant; this is beyond what is done off-line for construction of the DFSA plant model and control synthesis. Similarly, the event disabling/enabling decisions of the supervisor must be translated in real time as appropriate actions to control the plant.

Future research for advancement of the theory of optimal supervisory control for discrete event systems include the following areas:

- Robustness of the control policy relative to unstructured and structured uncertainties in the plant model including variations in the language measure parameters (Fu *et al.* 2003 a).
- Control synthesis under partial observation to accommodate loss of observability at the supervisory level possibly due to faults in sensors and/or communication links (Chattopadhyay and Ray 2004).
- Construction of grammar-based measures, instead of memory-less state-based measures (Ray and Phoha 2003, Surana and Ray 2004), for non-regular languages when details of transitions in plant dynamics cannot be captured by finitely many states (Chattopadhyay *et al.* 2004).

Acknowledgements

This work has been supported in part by Army Research Office (ARO) under Grant No. DAAD19-01-1-0646; and NASA Glenn Research Center under Grant No. NNC04GA49G.

References

Brave, Y., and Heyman, M., 1993, On optimal attraction of discrete-event processes. *Information Science*, **67**, 245–276.

Cassandras, C. G., and Lafortune, S., 1999, *Introduction to Discrete Event Systems* (Boston, MA: Kluwer Academic).

Chattopadhyay, I., and Ray, A., 2004, A language measure for partially observed discrete-event supervisory control systems. In *Proceedings of 43rd IEEE Conference on Decision and Control*, Atlantis, Paradise Island, Bahamas, December 2004 (in press).

Chattopadhyay, I., Ray, A., and Wang, X., 2004, A complex measure of non-regular languages for discrete-event supervisory control. In *Proceedings of American Control Conference*, Boston, MA, USA, June–July, pp. 5120–5125.

Fu, J., Lagoa, C., and Ray, A., 2003 a, Robust optimal control of regular languages. In *Proceedings of 42nd IEEE Conference on Decision and Control*, Maui, Hawaii, USA, December, pp. 3209–3214.

Fu, J., Ray, A., and Lagoa, C. M., 2003 b, Optimal control of regular languages with event disabling cost. In *Proceedings of American Control Conference*, Denver, Colorado, June, pp. 1691–1695.

Fu, J., Ray, A., and Lagoa, C. M., 2004, Unconstrained optimal control of regular languages. *Automatica*, **40**, 639–646.

Kumar, R., and Garg, V., 1995, Optimal supervisory control of discrete event dynamical systems. *SIAM Journal of Control and Optimization*, **33**, 419–439.

Naylor, A. W., and Sell, G. R., 1982, *Linear Operator Theory in Engineering and Science* (New York, NY: Springer-Verlag).

Passino, K., and Antsaklis, P., 1989, On the optimal control of discrete event systems. In *Proceedings of 28th IEEE Decision and Control Conference*, Tampa, FL, USA, December.

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., and Phoha, S., 2003, Signed real measure of regular languages for discrete-event automata. *International Journal of Control*, **76**, 1800–1808.

Sengupta, R., and Lafortune, S., 1998, An optimal control theory for discrete event systems. *SIAM Journal of Control and Optimization*, **36**, 488–541.

Surana, A., and Ray, A., Signed real measure of regular languages. *Demonstratio Mathematica*, **37**, 485–503.

Wang, X., and Ray, A., 2004, A language measure for performance evaluation of discrete-event supervisory control systems. *Applied Mathematical Modelling*, **28**, 817–833.