Supervisory Control of Fair Discrete-Event Systems: A Canonical Temporal Logic Foundation

Kiam Tian Seow, *Senior Member, IEEE*

***Abstract*—This paper studies the linear-time temporal logic (LTL) control of a class of fair discrete-event systems (DES's). It is motivated by the curious extent in which the use of LTL can be strengthened and differentiated in control theory development. Over a fair DES model, a marker-progressive supervisory control problem is formulated in LTL. The problem formulation admits a more flexible specification of multiple markers to distinguish different DES tasks, and seeks to find a supervisor – a passive control function by convention for specified temporal safety – such that a fair DES under its control is guaranteed to make constant progress to these markers. The problem is studied in terms of DES marker-controllability – a new controllability concept formulation of temporal safety for constant marker progress. This new formulation sheds light on how event fairness in DES's coachieves such marker progress with supervision that exists. It is shown that a solution supervisor may be found by canonical LTL verification. Three examples are provided for illustration.**

***Index Terms*—Discrete-event systems, event fairness, linear-time temporal logic, supervisory control.**

I. Introduction

The rapid evolution of technology in artificial intelligence (AI), robotics, and the Internet has brought about a disruptive trend towards building a cyber-physical world of innovative service and engineered biological system applications. These applications are embodied in ubiquitous electronic devices and robots – including autonomous cars and drones – offering capabilities that go beyond industrial production to directly impact human welfare and well-being in everyday life and work. Arguably limited only by human imagination, these applications can be modeled as discrete-event systems (DES's) and controlled to behave as specified at some level of design abstraction by a systems and control design approach. A seminal theory [1], [2] of supervisory control was founded in the 1980's and has been extended to-date in various ways in the control literature to support this approach. That these applications are amenable to DES modeling is because a DES is a model generating discrete structures of system states and possible state transitions; and these transitions represent abrupt occurrences of discrete qualitative changes termed events that are human-defined labels at the heart of the behavioral design matter for an application, such as 'cup grasped' and 'bowl filled' for a home service robot [3], [4]. Supervisory control theory seeks to understand and control DES's in discrete mathematics, since behavioral dynamics of the DES kind is non-continuous in time and so generally not amenable to system modeling and control in continuous or discrete time differential-equations.

In this paper, the linear-time temporal logic (LTL) of Manna and Pnueli [5] is adopted for the study of supervisory control [6], [7], [8] of fair DES's. This LTL is a branch of symbolic logic in discrete mathematics developed in computer science. A fair DES contains a subset of fair events that directs the system evolution. By a fair event, its transition occurs at infinitely many states of a DES evolution, in which the event is either defined at infinitely many states, or permanently defined from some state onwards. Towards augmenting AI and making LTL control verification and synthesis an integral part of system application design, this paper takes a renewed interest to investigate the extent in which the use of LTL [5], [9] can be strengthened and differentiated in control theory development. This interest is driven in part by the unrealized potential of available industrial-strength software tools for application DES modeling and transparent LTL control synthesis.

There are two time views depicting the generation or evolution of discrete structures by a DES: linear and branching. In the temporal logic literature [10], the semantics of formulas constructed in the syntax of a temporal logic language is determined by evaluating or interpreting the formulas (as true or false) over discrete structures in either time view. LTL is developed with the linear-time view. By 'linear time', a discrete structure is in the form of a sequence of states, over which an LTL formula is interpreted. Developed with the branching-time view is Computational Tree Logic (CTL). By 'branching time', a discrete structure is in the form of a tree structure of alternative state sequences, over which a CTL formula is interpreted. While acknowledging the usefulness of CTL, one reason for adopting LTL over CTL in this paper is the generally greater ease of use and intuitiveness of LTL [11]. Another is the neat classification of LTL formulas hierarchically into syntactic canonical classes [9], backed by their (assertional) proof rules and verification methods for a fair transition model. This LTL classification over a fair model forms a well-organized basis for control theory development.

The canonical forms for the various LTL classes have a set of restricted future modalities, namely operators *always* $\Box$ and *eventually* $\Diamond$, applied to past formulas [9]. Two of the classes, namely safety [12] at the base level and response [13] – a kind of progress at a level higher up in the classification hierarchy, are relevant to this paper; their canonical forms have operator $\Box$ and combined operator $\Box\Diamond$ applied to a past formula, respectively. LTL formulas in the safety and response classes assert, respectively, that 'nothing bad' will ever occur, and 'something good' such as accomplishment of tasks will occur 'regularly' in the sense of infinite oftenity. Together, they cover

K.T. Seow is with the Robot Intelligence Technology Laboratory, School of Electrical Engineering, KAIST, Daejeon 305-701, South Korea. E-mail: ktseow@rit.kaist.ac.kr

a useful range of control specifications about finishing tasks regularly without compromising safety.

In the same spirit as the founding, now standard nonblocking control theory [1], [2] and its rich extensions (see, e.g., in [6], [7], [8]) – all in formal languages and finite (state) automata under the *event space* formulation, this paper extends existing LTL research on DES controllability and control synthesis of temporal safety [14], [15], [16] under the alternate *state space* formulation. Also actively investigated in the Petri nets paradigm (see, e.g., in [17], [18], [19]), the state space formulation originates in a predicate and predicate transformer context [20], [21], [7, Ch. 7] that has been shown to be generalizable [14], [15], [16] to the richer LTL context subsumed in this paper, which presents a more complete LTL control theory with a new notion called marker-progressive supervisory control. This notion as defined produces subclasses of marker-progressive control that correspond to subclasses of control for standard nonblocking [1], [2] and multitasking [8].

Standard nonblocking control theory [1] is for supervising a DES designed to handle one set of marker states that represents but does not distinguish different DES tasks. In addition, this marker state set is termed global, in that each member marker state is defined only at where all the component processes, that a DES is often modularly composed of, complete one of their own process tasks. The control existence of a global marker state set in DES's is mandatory in (nontrivial) standard nonblocking control synthesis [2], and this might be too conservative. Multitasking control theory [8], a useful nonblocking generalization, is for supervising a DES extended to handle multiple sets of marker states, such that every set modeling and distinguishing the completion of a different task can be independently entered via an arbitrary DES state and subsequent transitions admissible under control. But, as with nonblocking, multitasking control theory implicitly assumes that the DES will proactively traverse states not blocked by control to regularly reach a state of every specified marker state set. This means that a kind of fair event subset somehow invariably exists post-synthesis, for the DES to be able to realize such assumed proactivity under a control solution synthesized independently of system fairness.

In contrast, incorporating event fairness into DES's at the outset, this paper relaxes the implicit proactivity assumption and investigates, in LTL [5], a marker-progressive supervisory control problem in the spirit of *verification and synthesis of control dynamic invariants by state feedback*, and supports the resolution of the problem in terms of a new *system* concept formulation called marker-controllability for specified temporal safety over the resultant class of fair DES's. In the formulation, the concept of dynamic invariant is naturally defined by a formula of an unchanging past concomitant with temporal safety. Importantly, mapped onto the state space in an LTL paradigm, this control problem has temporal past extending and DES event fairness refining, respectively, the marker and reachability conditions of standard nonblocking and multitasking control. It seeks to find a supervisor controlling an invariant for a given fair DES, to controllably meet a given temporal-safety specification, such that the DES not only can but also will accomplish – infinitely often – all tasks specified by a set of marker conditions expressed in past formulas, of which marker state sets are a special case; such supervision thereby ensures constant, guaranteed progress to multiple markers under specified temporal safety. Event fairness in DES's is shown to play a cooperative role induced in such controllability ensuring marker progress, shedding light on how the role plays out in coachieving this kind of progress with supervision that exists. By the LTL classification [9], marker-progressive control for fair DES's is shown to be (solvable as) a canonical safety-response verification problem.

The rest of this paper is organized as follows. Section II lays an LTL background for DES research on state feedback control. The main contributions of this paper are presented in Sections III to IV, and include the following:

1) The definition of two new concepts, namely condition invariance and marker-liveness under conditional invariance, adding to the set of basic system and control concepts defined in LTL over the notion of an invariant (Section III).
2) The main results of solution existence and problem classification for the marker-progressive control problem based on a fair DES model definition. The results are based on concepts presented in Section III, the relevant canonical LTL classes of temporal safety and response, and the aggregated concepts and sets defined (Section IV). The aggregated concepts include the overarching system concept of marker-controllability for temporal safety.

Section V discusses the control results of this paper along with three illustrative examples. Section VI discusses related work. Section VII concludes this paper.

II. DES & Linear-Time Temporal Logic

A. DES Model

The discrete-event system (DES) to be controlled – called the plant in control theory – is modeled by a basic transition system G, defined as follows:

$$G \overset{\text{def}}{=} (\Pi, Q, \Sigma, \delta, \theta). \tag{1}$$

Π denotes the finite state variable set which is typed; the type of each state variable $v \in \Pi$ indicates the domain $Range(v)$ over which the variable ranges. Q denotes the state set, defined by the cross product of the ranges of the variables in Π, i.e., $Q \overset{\text{def}}{=} \bigotimes_{v \in \Pi} Range(v)$, such that every state $q \in Q$ assigns domain values to all state variables in Π, and is unique, i.e., every state-assignment is different. Σ denotes the finite event set partitioned into two sets, namely the set of controllable events Σ_c and the set of uncontrollable events Σ_u, and this is denoted by $\Sigma = \Sigma_c \dot{\cup} \Sigma_u$.[1] $\delta : \Sigma \times Q \rightarrow Q$ is a (deterministic) state transition function that is partial, in that for each $q \in Q$, $\delta(\sigma, q)$ is defined for a subset of Σ in general. θ is the initial condition – a boolean valued formula that characterizes the set of initial states $Q_0 \subseteq Q$ of G, such that $q \in Q_0$ provided (the value assignment by) $q \in Q$ satisfies (i.e., 'makes true') θ.

[1] Given sets E, E_1, E_2, E is a partition of E_1, E_2, denoted by $E = E_1 \dot{\cup} E_2$, if $E = E_1 \cup E_2$ and $E_1 \cap E_2 = \varnothing$.

In DES model G, it is assumed that $Q_0 \neq \varnothing$, $\Sigma \neq \varnothing$ due to nontrivial system modeling.

In the case that DES model G is finite state, it may be represented by an edge-labeled directed graph. In this graph, a node denotes a DES state; a σ-labeled edge, directing a node denoting a state q to a node denoting a state q', denotes the transition of event σ from q to q', as defined by $\delta(\sigma, q) = q'$. A node with an entering arrow denotes a state in initial state set Q_0.

B. *Temporal Logic Syntax*

LTL [5] is a language of predicate logic that is augmented with a temporal operator set to facilitate reasoning over sequences of states, with predicate logic, which subsumes propositional logic, for reasoning over individual states. There are two subsets of temporal operators, namely past and future, for abstracting arbitrary sequences of states in a logical passage of time, in the temporal past and future as LTL formulas expressed over predicates of state information. Predicates of arity $m \geqslant 0$ are written in symbolic form $F(x_1, \cdots, x_m)$, where every argument $x_i \in \Pi$ $(1 \leqslant i \leqslant m)$ of predicate F is non-propositional, and propositional variables are viewed as 0-ary predicates. To be defined later, future operators include *always* $\Box$, *eventually* $\Diamond$, *next* $\bigcirc$, *until* $\mathcal{U}$, and *unless* $\mathcal{W}$. Past operators include *has-always-been* $\boxminus$, *once* $\diamondminus$, *previously* $\ominus$ and its weak version $\circleddash$, *since* $\mathcal{S}$, and *back-to* $\mathcal{B}$.

Formally, LTL formulas are constructed using formula formation rules over a finite set $\mathcal{P}$ of atomic propositions, the Boolean connectives *and*, *not* denoted by a dot $\cdot$ and an overhead bar $^-$, respectively, the quantifier '*there exists*' denoted by $\exists$, and temporal operators. The atomic propositions in $\mathcal{P}$ are expressed by predicates, quantified or otherwise, in terms of state variables in Π of DES G (over their domains) and the system and control logics which will be defined later. Let $\mathcal{T}_1$ and $\mathcal{T}_2$ denote an arbitrary unary and an arbitrary binary temporal operator, respectively. Then the formula formation rules are as follows:

1) Every atomic proposition of $\mathcal{P}$ is a formula.
2) If ω, ω_1 and ω_2 are formulas, so are $\overline{\omega}$, $\omega_1 \cdot \omega_2$, $\mathcal{T}_1(\omega)$ and $\omega_1 \mathcal{T}_2\ \omega_2$.

Over arbitrary formulas ω_1, ω_2, ω, the following abbreviations (*always-equals* $\equiv$) are used, about which related connectives *or* $+$, *implies* $\rightarrow$ and *equals* $=$, and the related quantifier '*for all*' $\forall$ are, respectively, defined: $(\omega_1 + \omega_2) \equiv \overline{(\overline{\omega_1} \cdot \overline{\omega_2})}$, $(\omega_1 \rightarrow \omega_2) \equiv (\overline{\omega_1}{+}\omega_2)$, $(\omega_1 = \omega_2) \equiv (\omega_1 \rightarrow \omega_2){\cdot}(\omega_2 \rightarrow \omega_1)$, and $(\forall x)\omega \equiv \overline{(\exists x)\overline{\omega}}$, where $x \in \Pi$ is an argument of some predicate contained in ω. The LTL language also includes *validity* $true$ and *inconsistency* $false$ - propositional constants which are defined, respectively, by the following abbreviations over an arbitrary formula ω: $true \equiv \overline{\omega}{+}\omega$ and $false \equiv \overline{\omega}{\cdot}\omega$.

Aggregation connectives $\sum$, $\prod$ denote the *or*-ing (or logical sum) and *and*-ing (or logical product) of a number of formulas, respectively.

A past formula is one that contains no future operators; a future formula is one that contains no past operators; and a state formula is one that contains no future or past operators.

C. *Temporal Logic Semantics & DES Model Behavior*

A string is a sequence of events that can be finite or infinite (in length). An arbitrary string over the event set Σ of DES G can be generally viewed as a map $e : \{1, \cdots, k, \cdots, \cdots\} \rightarrow \Sigma$, such that $e \overset{\text{def}}{=} e(1)e(2) \cdots e(k) \cdots$, where $e(k) \in \Sigma$. Then e is an event string generated by DES G, provided there exists a 'labeling' of the string by states $I : \{0, \cdots, k, \cdots, \cdots\} \rightarrow Q$ under G's state transition function δ, such that $I \overset{\text{def}}{=} I(0) - I(1) - \cdots - I(k) \cdots$, where $I(k) = q_k \in Q$ for which

1) $I(0) = q_0 \in Q_0$ (an initial state), and
2) $I(k) = \delta(e(k), I(k-1))$, where $k \geqslant 1$.

Such a labeling I (that exists) is an arbitrary state trajectory or interpretation of G. With $k \geqslant 0$, the k-prefix of I is $q_0 - q_1 - \cdots - q_k$, and denoted by $I_{(k)}$. A state $q \in Q$ is said to be terminal (in G) if $(\forall \sigma \in \Sigma)(\delta(\sigma, q)$ is not defined). An interpretation I is finite and said to be terminating if it ends in a state q_k that is terminal, i.e., $I = I_{(k)}$; otherwise, it is infinite and said to be non-terminating, i.e., $I = I_{(\infty)}$. The string labeled by prefix $I_{(k)}$ $(k \geqslant 0)$ is called a prefix string. Note that $I_{(0)} = I(0)$. Two state trajectories of DES G, or, respectively, their k-prefixes, are defined to be the same (or equal) if the two have the same sequence of states and label the same string.

The LTL formulas expressed over (set $\mathcal{P}$ of) DES model G are interpreted over models of the form (I, π), where $\pi : \{0, \cdots, k, \cdots, \cdots\} \times \mathcal{P} \rightarrow \{true, false\}$ is a binary function that evaluates an atomic proposition p_a in state $I(k)$ (or $q_k \in Q$) as follows:

$$\pi(k, p_a) = \begin{cases} true, & \text{if } p_a \text{ is } true \text{ in } q_k \in Q \\ false, & \text{otherwise.} \end{cases}$$

The satisfaction relation $\left(\models^{I^{(k)}} \omega\right) \in \{true, false\}$ (read: 'I at its state q_k satisfies ω', or simply 'I satisfies ω' if $k = 0$, since $I^{(0)} \overset{\text{def}}{=} I$) defines the semantics of an arbitrary LTL formula ω at state q_k $(k \geqslant 0)$ along an arbitrary interpretation I of G. Rewriting in terms of this relation, it follows that, for an atomic proposition $p_a \in \mathcal{P}$,

0) $\models^{I^{(k)}} p_a$ iff $\pi(k, p_a) = true$.

It should be clear that, over $I^{(k)}$ and in state q_k, the respective evaluations of an atomic proposition, and more inductively of a state formula p_s, are the same, i.e., $\models^{I^{(k)}} p_s$ iff $\models^{q_k} p_s$, where $(\models^{q_k} p_s) \in \{true, false\}$ (read: 'q_k satisfies p_s') defines the semantics of formula p_s in state q_k.

In addition to the standard rules for Boolean connectives, LTL uses satisfaction relation rules for temporal operators to inductively evaluate the satisfaction of an arbitrary $I^{(k)}$ $(k \geqslant 0)$ over an LTL formula. Below, the rules defined for the basis sets $\{\Box, \bigcirc, \mathcal{U}\}$, $\{\boxminus, \ominus, \mathcal{S}\}$ of future and past operators are presented. The rule for operator $\bigcirc$ requires the following event-transition logic to account for a trajectory I that is finite.

Definition 1 (The σ-Transition Logic): Given $\sigma \in \Sigma$, for an arbitrary state trajectory I of DES G, $I = q_0 - q_1 - \cdots - q_k \cdots$, the function $\tau : \sigma \rightarrow (I \rightarrow \{true, false\})$ is a system σ-transition logic, defined at $q_k \in Q$ such that

$$\models^{I^{(k)}} \tau_\sigma \text{ iff } (\exists I_{(k+1)})\ q_{k+1} = \delta(\sigma, q_k).$$

Now, given formulas ω, ω_1, ω_2:

1) $\models^{I^{(k)}} \Box\omega$ iff for all $j \geqslant k$, $\models^{I^{(j)}} \omega$.
2) $\models^{I^{(k)}} \bigcirc\omega$ iff $\models^{I^{(k)}} \tau \;\rightarrow \models^{I^{(k+1)}} \omega$, where $\tau \equiv \sum_{\sigma\in\Sigma} \tau_\sigma$.
3) $\models^{I^{(k)}} \omega_1 \mathcal{U} \omega_2$ iff there is a j $(j \geqslant k)$ such that $\models^{I^{(j)}} \omega_2$ and for all i $(k \leqslant i < j)$, $\models^{I^{(i)}} \omega_1$.
4) $\models^{I^{(k)}} \boxminus\omega$ iff for all j $(0 \leqslant j \leqslant k)$, $\models^{I^{(j)}} \omega$.
5) $\models^{I^{(k)}} \ominus\omega$ iff $k > 0$ and $\models^{I^{(k-1)}} \omega$.
6) $\models^{I^{(k)}} \omega_1 \mathcal{S} \omega_2$ iff there is a j $(0 \leqslant j \leqslant k)$ such that $\models^{I^{(j)}} \omega_2$ and for all i $(j < i \leqslant k)$, $\models^{I^{(i)}} \omega_1$.

The other temporal operators may be defined by the following abbreviations: 7) $\Diamond\omega \equiv \overline{\Box(\overline{\omega})} \equiv true\,\mathcal{U}\omega$, 8) $\omega_1 \mathcal{W} \omega_2 \equiv \Box\omega_1 + \omega_1 \mathcal{U} \omega_2$, 9) $\diamondminus\omega \equiv \overline{\boxminus(\overline{\omega})} \equiv true\,\mathcal{S}\omega$, 10) $\circledtilde\omega \equiv \overline{\ominus(\overline{\omega})}$, and 11) $\omega_1 \mathcal{B} \omega_2 \equiv \boxminus\omega_1 + \omega_1 \mathcal{S} \omega_2$.

The model operational premise is this: From every non-terminal state that DES G is in, one event will occur and transition the DES into another state.

Interpretations that restrict to the actual behavior of DES G are termed legal. Let $\mathcal{I}(G)$ be the set of legal interpretations defined over DES G. Then, since only actual DES behavior is of interest, the following notion of G-validity of a formula ω, denoted by $G \models \omega$, is fundamental:

$$G \models \omega \text{ iff } (\forall I \in \mathcal{I}(G)) \models^{I} \omega.$$

In LTL semantics, for an arbitrary set $\mathcal{I}(G)$, $\omega_1 \equiv \omega_2$ denotes $G \models \Box(\omega_1 = \omega_2)$; in addition, let $\omega_1 \approx \omega_2$ denote $G \models (\omega_1 = \omega_2)$, where the connective $\approx$ is said to be the anchored version of $\equiv$. An LTL formula ω is said to be satisfiable if $\omega \not\equiv false$, i.e., $(\exists I \in \mathcal{I}(G))(\exists k \geqslant 0) \models^{I^{(k)}} \omega$.

D. State Feedback Supervisory Control

A supervisor for DES G specifies whether controllable events are to be enabled or disabled at state $q_k \in Q$ of an arbitrary input state history $I_{(k)}$, where $I \in \mathcal{I}(G)$.

Definition 2 (The σ-Definition Logic): Given $\sigma \in \Sigma$, for an arbitrary state $q \in Q$ of DES G, the function $\xi : \sigma \rightarrow (q \rightarrow \{true, false\})$ is a system σ-definition logic, defined such that

$$\models^{q} \xi_\sigma \text{ iff } (\exists q' \in Q) q' = \delta(\sigma, q).$$

Formally then, for every $I \in \mathcal{I}(G)$, a supervisor is a function $f : \Sigma \rightarrow (I \rightarrow \{true, false\})$, defined at $q_k \in Q$ with the supervisor Σ_u-completeness constraint

$$(\forall \sigma \in \Sigma_u) \models^{I^{(k)}} (f_\sigma = true),$$

such that $f_\sigma = true$ and $f_\sigma = false$ enables and disables event $\sigma \in \Sigma$ at current state $q_k \in Q$ of history $I_{(k)}$, respectively, if σ is defined at the state (i.e., $\models^{q_k} \xi_\sigma$); otherwise, $f_\sigma \in \{true, false\}$ does not enable σ. Only an enabled event at the DES current state can occur; but the supervisor is by convention not the cause of its occurrence, hence supervision is termed *passive*. Set up in the closed-loop system depicted in Fig. 1, the supervisor is said to issue a new control pattern $\{f_\sigma \mid \sigma \in \Sigma\}$ for enabling and disabling events, in response to new state values (i.e., state information) fed back by a discrete state change triggered by an enabled event occurrence in the DES. It is hence termed a state feedback supervisor. The choice of event for execution among those enabled at a current DES state is deemed to be made by some underlying event-selection mechanism of the DES. This mechanism is generally unmodeled in the DES control literature.

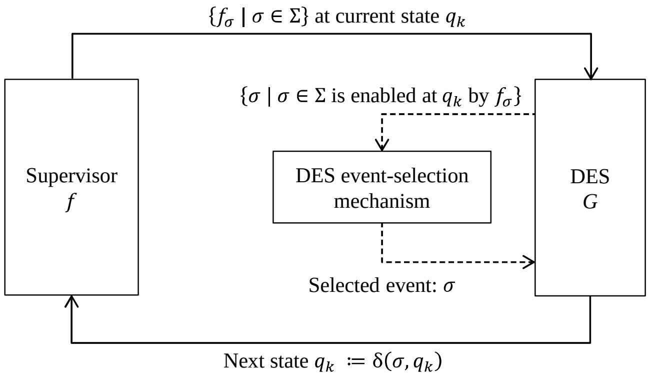

$\{f_\sigma \mid \sigma \in \Sigma\}$ at current state q_k
$\{\sigma \mid \sigma \in \Sigma$ is enabled at q_k by $f_\sigma\}$
Supervisor
f
DES event-selection mechanism
DES
G
Selected event: σ
Next state $q_k := \delta(\sigma, q_k)$

Fig. 1. The state feedback supervisory control loop, with DES event-selection mechanism explicitly shown.

By imposing f on DES G in the state feedback loop, the resulting controlled model, denoted by G^f, is of the same type (1) but with state uniqueness relaxed, and is defined as follows:

1) $\{I'_{(0)} \mid I' \in \mathcal{I}\left(G^f\right)\} = \{I_{(0)} \mid I \in \mathcal{I}\left(G\right)\}$, and
2) $(\forall I \in \mathcal{I}\left(G\right))(\forall k \geqslant 0)(\forall \sigma \in \Sigma)\Big(\models^{I^{(k)}} f_\sigma \cdot \xi_\sigma$ iff $(\exists I' \in \mathcal{I}(G^f))\; I'_{(k+1)} = I_{(k)} - \delta(\sigma, q_k)\Big)$.

From a theoretical viewpoint following [7], a standard criterion imposed on the 'control technology' for supervisor f is that its '*control should at most restrict uncontrolled behavior, never enlarge it*'. To formalize this criterion, let $\mathcal{I}^{\#}(G) = \{I_{(k)} \mid I \in \mathcal{I}(G),\ \text{finite } k \geqslant 0,\ \text{and } I_{(k)} \notin \mathcal{I}(G)\}$, called the legally prefix-admissible set. Now, let $\mathcal{I}^{\circledast}(G) = \mathcal{I}(G) \mathbin{\dot\cup} \mathcal{I}^{\#}(G)$. Then formally, this criterion states logically that $I \in \mathcal{I}(G^f) \rightarrow I \in \mathcal{I}^{\circledast}(G)$. If this criterion is obeyed and $\mathcal{I}(G^f) \neq \varnothing$, then the supervisor f is said to be proper. In other words, for an arbitrary proper f, $\varnothing \subset \mathcal{I}(G^f) \subseteq \mathcal{I}^{\circledast}(G)$.

E. Fundamental Problem of Supervisory Control

Consider the specification pair $(P, \mathcal{M})$ for DES G, where P is an arbitrary past formula over G, and $\mathcal{M} = \{M_1, M_2, \cdots, M_m\}$, where each $M_i \in \mathcal{M}$ $(1 \leqslant i \leqslant m)$ is an arbitrary past formula over G specifying a system marker condition. $\mathcal{M}$ is called the system marker set; and the marker conditions are meant to represent and distinguish the completion of different tasks or jobs by the processes of DES G. Given this pair $(P, \mathcal{M})$, the basic problem of supervisory control is defined as finding a proper state feedback supervisor f for G that can confine the DES to state trajectories in a subset of $\mathcal{I}^{\circledast}(G)$, along which all marker conditions in $\mathcal{M}$ are met infinitely often under the invariance of P, i.e., $\Box P$, an LTL formula in canonical temporal-safety form [5], [9]. Temporal safety includes criteria such as deadlock avoidance, mutual exclusion, etc. That each marker condition $M_i \in \mathcal{M}$ is to be met infinitely often in G is specified by $\Box\Diamond M_i$, an LTL formula in canonical temporal-response form [5], [9].

By the problem description above, the pair $(P, \mathcal{M})$ denotes

$$\Box\left(P \cdot \prod_{i=1}^{m} \Diamond M_i\right)$$

– the control specification to realize. The problem described is called the marker-progressive supervisory control problem in this paper, and is in essence about marker progressiveness under supervised temporal safety. This problem is formalized in Section IV, and the conditions under which it is solvable are investigated therein.

F. DES Model Behavioral Logics

The characterizations of behavioral transition logics and operations are presented. Except for the conditioned transition logic and related operators, much of the supporting material herein originates in [14], [15], [16].

Proposition 1: Consider an arbitrary state trajectory I of DES G, $I = q_0 - q_1 - \cdots - q_k \cdots$. Then, for an arbitrary $\sigma \in \Sigma$:

P1.1) $\models^{I^{(k)}} (\tau_\sigma \rightarrow \xi_\sigma)$. P1.2) $\models^{I^{(k)}} \left(\tau_\sigma \rightarrow \prod_{\sigma' \in \Sigma\setminus\{\sigma\}} \overline{\tau_{\sigma'}}\right)$.

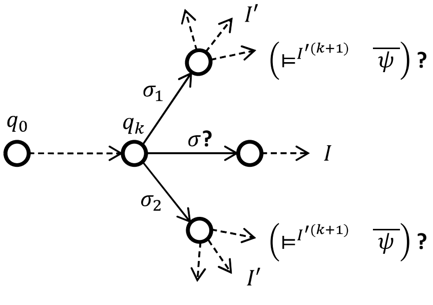
The converse of P1.2 is also true if state q_k is not terminal.

Behavioral operators over arbitrary LTL formulas follow next.

Definition 3 (Dynamic σ-Transition Operators): Given an arbitrary LTL formula φ over DES G and $\sigma \in \Sigma$, the system dynamic event-operators $\ominus_\sigma$, $\bigcirc_\sigma$ over an arbitrary state trajectory I of G, $I = q_0 - q_1 - \cdots - q_k \cdots$, are defined as follows:

D3.1) $\models^{I^{(k)}} \ominus_\sigma(\varphi) = \ominus(\tau_\sigma \cdot \varphi)$.
D3.2) $\models^{I^{(k)}} \bigcirc_\sigma(\varphi) = (\tau_\sigma \rightarrow \bigcirc\varphi)$.

Definition 4 (The Conditioned σ-Transition Logic): Given an arbitrary LTL formula ψ over DES G and $\sigma \in \Sigma$, for an arbitrary state trajectory I of G, $I = q_0 - q_1 - \cdots - q_k \cdots$, and an arbitrary $I' \in \mathcal{I}(G)$, $I' = I_{(k)} - q'_{k+1} \cdots$ (if it exists), the function $\tau_x : (\sigma, \psi) \rightarrow (I \rightarrow \{true, false\})$ is a system ψ-conditioned transition logic, defined at $q_k \in Q$ such that $\models^{I^{(k)}} \left(\tau_{x|\sigma}(\psi) = \tau_\sigma \cdot (\forall I', I'_{(k+1)} \neq I_{(k+1)}) \models^{I'^{(k)}} \bigcirc\overline{\psi}\right)$.



I'
$\left(\models^{I'^{(k+1)}} \quad \overline{\psi}\right)$?
σ_1
q_0
q_k
σ?
I
σ_2
$\left(\models^{I'^{(k+1)}} \quad \overline{\psi}\right)$?
I'

Fig. 2. An illustration of conditioned σ-transition logic $\tau_{x|\sigma}(\psi)$. Each dotted arrow depicts an arbitrary sequence of state transitions (of events). A fanout of three dotted arrows from each of two of the states is meant to depict an arbitrary number of legal state trajectories that exist, each denoted by I' that extends $I_{(k)}$ by branching off I from state q_k via either event σ_1 or σ_2.

For illustration, it is assumed, as depicted in Fig. 2, that other than the event defined at state q_k along an arbitrary state trajectory I of DES G, only two other events σ_1, σ_2 are defined at q_k. Then intuitively, along I, $\tau_{x|\sigma}(\psi)$ is true at state q_k if: (i) τ_σ is true at q_k along I, and (ii) ψ is false at the next state after q_k along every other trajectory I' that is legal, shares the same k-prefix as I, and branches off I from q_k into the next state via (the transition of) some event of Σ, which in Fig. 2 is either event σ_1 or σ_2; $\tau_{x|\sigma}(\psi)$ is false otherwise. By Condition (ii) above, it may be said that a true $\tau_{x|\sigma}(\psi)$ at a state q_k along I means that the σ-transition has no (legal) (Σ, ψ)-peers at q_k.

Proposition 2: Consider an arbitrary LTL formula ψ over DES G and an arbitrary state trajectory I of G, $I = q_0 - q_1 - \cdots - q_k \cdots$. Then, for an arbitrary $\sigma \in \Sigma$,

$$\models^{I^{(k)}} \left(\tau_{x|\sigma}(\psi) \rightarrow \prod_{\sigma' \in \Sigma\setminus\{\sigma\}} \overline{\tau_{x|\sigma'}(\psi)}\right).$$

Definition 5 (Dynamic Conditioned σ-Transition Operators): Given arbitrary LTL formulas ψ, φ over DES G and $\sigma \in \Sigma$, the system dynamic conditioned event-operators $\ominus_{x|\sigma}(.,.)$, $\bigcirc_{x|\sigma}(.,.)$, over an arbitrary state trajectory I of G, $I = q_0 - q_1 - \cdots - q_k \cdots$, are defined as follows:

D5.1) $\models^{I^{(k)}} \ominus_{x|\sigma}(\psi, \varphi) = \ominus\left(\tau_{x|\sigma}(\psi) \cdot \varphi\right)$.
D5.2) $\models^{I^{(k)}} \bigcirc_{x|\sigma}(\psi, \varphi) = \left(\tau_{x|\sigma}(\psi) \rightarrow \bigcirc\varphi\right)$.

Definition 6 (The Event-Transitions): Consider the event set $\Sigma = \Sigma_c \dot\cup \Sigma_u$ of DES G. The respective system controllable, uncontrollable, and conditioned event-transitions τ_c, τ_u, $\tau_x(.)$ are characterized as follows:

D6.1) $\tau_c \equiv \sum_{\sigma \in \Sigma_c} \tau_\sigma$. D6.2) $\tau_u \equiv \sum_{\sigma \in \Sigma_u} \tau_\sigma$.
D6.3) $\tau_x(.) \equiv \sum_{\sigma \in \Sigma} \tau_{x|\sigma}(.)$.

Proposition 3: Consider an arbitrary state trajectory I of DES G, $I = q_0 - q_1 - \cdots - q_k \cdots$, where q_k is not terminal. Then, $\models^{I^{(k)}} (\tau_u = \overline{\tau_c})$.

Definition 7 (Dynamic Event-Transition Operators): Consider the event set $\Sigma = \Sigma_c \dot\cup \Sigma_u$ of DES G. The system dynamic event-transition operators $\ominus_u$, $\bigcirc_u$, $\ominus_x(.,.)$, $\bigcirc_x(.,.)$ are characterized as follows:

D7.1) $\ominus_u \equiv \sum_{\sigma \in \Sigma_u} \ominus_\sigma$; $\bigcirc_u \equiv \prod_{\sigma \in \Sigma_u} \bigcirc_\sigma$.
D7.2) $\ominus_x(.,.) \equiv \sum_{\sigma \in \Sigma} \ominus_{x|\sigma}(.,.)$; $\bigcirc_x(.,.) \equiv \prod_{\sigma \in \Sigma} \bigcirc_{x|\sigma}(.,.)$.

Proposition 4: Consider arbitrary LTL formulas ψ, φ over DES G, and an arbitrary state trajectory I of G, $I = q_0 - q_1 - \cdots - q_k \cdots$. Then:

P4.1) $\models^{I^{(k)}} \ominus_u(\varphi) = \ominus(\tau_u \cdot \varphi)$.
P4.2) $\models^{I^{(k)}} \bigcirc_u(\varphi) = (\tau_u \rightarrow \bigcirc\varphi)$.
P4.3) $\models^{I^{(k)}} \ominus_x(\psi, \varphi) = \ominus\left(\tau_x(\psi) \cdot \varphi\right)$.
P4.4) $\models^{I^{(k)}} \bigcirc_x(\psi, \varphi) = (\tau_x(\psi) \rightarrow \bigcirc\varphi)$.

Proposition 5 (Operator Duality): Consider arbitrary LTL formulas ψ, φ, ϕ over DES G. Then:

P5.1) $G \models \Box\left(\ominus\psi \rightarrow \varphi\right) = \Box\left(\psi \rightarrow \bigcirc\varphi\right)$.
P5.2) $G \models \Box\left(\ominus_u(\psi) \rightarrow \varphi\right) = \Box\left(\psi \rightarrow \bigcirc_u(\varphi)\right)$.
P5.3) $G \models \Box\left(\ominus_x\left(\psi, \phi\right) \rightarrow \varphi\right) = \Box\left(\phi \rightarrow \bigcirc_x(\psi, \varphi)\right)$.

III. DES & Control Concepts Over An Invariant

The concept of (dynamic) invariant and its kernel is fundamental in the logic framework of supervisory control.

Definition 8 (The Invariant & Its Kernel): Consider an arbitrary LTL formula φ over DES G. Then φ is said to be an invariant if $\varphi \equiv \boxminus\psi$ for some past formula ψ. If this ψ

has no operator $\boxminus$ in its outermost scope, it is said to be the kernel of φ.

An invariant φ and its kernel ψ over DES G are said to be initially satisfied if $G \models \psi$. An invariant may be 'upper-bounded' by another of given interest, as defined next.

Definition 9 (P-History Boundedness): Given the kernel P of some invariant over DES G, an arbitrary invariant φ over G is said to be P-history bounded (with respect to G) if

$$G \models \Box\,(\varphi \rightarrow \boxminus P).$$

Henceforth, in the specification pair $(P, \mathcal{M})$ first introduced in Section II-E, the past formula P is assumed, for succinctness with no loss of generality[2], to be the kernel of some invariant over DES G. Then if the invariance (specifiable with $\Box$) of P is to be met by supervisory control, an invariant φ needs to be found that is P-history bounded, i.e., φ is not weaker than $\boxminus P$. To satisfy or meet $\Box P$ as part of solving the marker-progressive control problem, the invariant φ also needs to be a satisfiable formula whose truth a supervisor can feasibly maintain in DES G. The maintenance is done by the supervisor performing next-state control of the kernel of φ. To also meet $\Box\left(\prod_{i=1}^{m}\Diamond M_i\right)$, the whole control problem is studied in the next section in terms of several basic system and control concepts defined over an invariant that are presented in the following. The first two are the LTL concepts of control invariance and Σ_u-invariance that originate in [14], and are herein reformulated over the more refined Definition 8 of an LTL invariant first introduced in [16].

Definition 10 (Control Invariance): An arbitrary invariant φ over DES G is said to be control invariant (with respect to G) if, for some state feedback supervisor f,

$$G \models \Box\left(\varphi \cdot \sum_{\sigma\in\Sigma}(f_\sigma \cdot \tau_\sigma) \rightarrow \bigcirc\varphi\right).$$

Definition 11 (Σ_u-Invariance): An arbitrary invariant φ over DES G is said to be Σ_u-invariant (with respect to G) if

$$G \models \Box\,(\ominus_u(\varphi) \rightarrow \varphi).$$

Proposition 6: An arbitrary invariant φ over DES G is Σ_u-invariant if and only if it is control invariant (under Σ_u-completeness of supervisor f).

Proof: Based on the more refined Definition 8 of an invariant, proof is similar to that of [14, Proposition 3.13]. ■

Note that a supervisor f for DES G, by definition, is Σ_u-complete. Therefore, for a control invariant φ over DES G,

$$(\exists f)\ G \models \Box\left(\varphi \cdot \sum_{\sigma\in\Sigma}(f_\sigma \cdot \tau_\sigma) \rightarrow \bigcirc\varphi\right)$$

may be logically rewritten as follows:

$$(\exists f)(\forall \sigma \in \Sigma_c)\ G \models \Box\,(\varphi \cdot f_\sigma \rightarrow \bigcirc_\sigma(\psi)),$$

where ψ is the kernel of φ. It follows that the ψ-locally optimal or most permissive supervisor f that exists for G is such that

$$(\forall \sigma \in \Sigma_c)\ G \models \Box\,(\varphi \rightarrow (f_\sigma = \bigcirc_\sigma(\psi))),$$

which may be abbreviated in the algebraic form:

$$(\forall \sigma \in \Sigma_c) \quad f_\sigma = \bigcirc_\sigma(\psi) \quad \text{[rel to } (\varphi, G)],$$

where '[rel to (φ, G)]' reads 'relative to φ over G' and may be omitted when the context is understood. The supervisor f is said to be *static* if ψ is or abbreviates to a state formula with respect to DES G; otherwise, f is *dynamic*.

Remark 1: Since the state transition function δ of DES model G is deterministic, a realization of f_σ, an arbitrary σ-component of supervisor f for $\sigma \in \Sigma_c$, may be computed by LTL reasoning over model G's possible transitions axiomatized as transition relations [5], [14], to remove the operator $\bigcirc$ and obtain some past formula ψ_σ, such that

$$\bigcirc_\sigma(\psi) \equiv (\tau_\sigma \rightarrow \psi_\sigma).$$

Recall from Definition 1 that the σ-transition logic τ_σ is defined at a state q along an arbitrary state trajectory I of DES G. In interpreting this logic when the DES is at state q where several other events may also be defined, a $true$ assertion of τ_σ is taken to mean that the event σ is exclusively selected for execution by the event-selection mechanism of the DES. This selection is, however, generally not known *a priori*. The logic τ_σ is therefore set to $true$, so that

$$(\forall \sigma \in \Sigma_c) \quad f_\sigma = \psi_\sigma \quad \text{[rel to } (\varphi, G)].$$

In this form, an event-disabling by f_σ need not first detect the imminence of event σ. ■

Definition 12 ($(\mathcal{M}, \varphi)$-Uncertain State): Consider an arbitrary invariant φ over DES G with system marker set $\mathcal{M} = \{M_1, M_2, \cdots, M_m\}$. Then a state $q_k \in Q$ along an arbitrary $I \in \mathcal{I}(G)$, $I = q_0 - q_1 - \cdots - q_k \cdots$, is said to be $(\mathcal{M}, \varphi)$-uncertain if $\models^{I^{(k)}} \tau_x(\varphi) \cdot \sum_{i=1}^{m}\overline{M_i}$.

With $\tau_x(\varphi) \equiv \sum_{\sigma\in\Sigma}\tau_{x|\sigma}(\varphi)$ by D6.3, and by Proposition 2, an $(\mathcal{M}, \varphi)$-uncertain state along a legal state trajectory I of DES G has a true $\tau_{x|\sigma}(\varphi)$ for some $\sigma \in \Sigma$, whose exclusive σ-transition is therefore (Σ, ψ)-peerless at that state. An $(\mathcal{M}, \varphi)$-uncertain state along I is also where not all marker conditions are met.

Definition 13 ($(\mathcal{M}, \varphi)$-Condition Invariance): An arbitrary invariant φ over DES G with system marker set $\mathcal{M} = \{M_1, M_2, \cdots, M_m\}$ is said to be $(\mathcal{M}, \varphi)$-condition invariant (with respect to G) if

$$G \models \Box\left(\ominus_x\left(\varphi, \varphi \cdot \sum_{i=1}^{m}\overline{M_i}\right) \rightarrow \varphi\right).$$

Consider an invariant φ over DES G that is $(\mathcal{M}, \varphi)$-condition invariant. Then since

$$\ominus_x\left(\varphi, \varphi \cdot \sum_{i=1}^{m}\overline{M_i}\right) \equiv \ominus\left(\varphi \cdot \tau_x(\varphi) \cdot \sum_{i=1}^{m}\overline{M_i}\right)$$

by P4.3, it intuitively means that φ remains true at an arbitrary state q_{k+1} along an arbitrary legal state trajectory I of DES G, if it is true at the previous state q_k that is $(\mathcal{M}, \varphi)$-uncertain. In essence, this means that if φ is true at some arbitrary state q_k reached along I and not all marker conditions in $\mathcal{M}$ are true there, a next state exists into which the DES can transition to maintain the φ-information (i.e., the truth of φ), albeit along some legal state trajectory of DES G evolved onto that shares the same k-prefix, but not necessarily the same $(k+1)$-prefix as I unless the transition at q_k along I has no (Σ, ψ)-peers.

[2] Because $G \models \Box P = \Box(\boxminus P)$.

In other words, whenever φ is true, then unless all marker conditions in $\mathcal{M}$ are simultaneously met, the DES can evolve to maintain the φ-information.

Remark 2 (Dual-Operator Definitions): By Proposition 5, the dual-operator versions of Definitions 10 (control invariance), 11 (Σ_u-invariance), and 13 [$(\mathcal{M}, \varphi)$-condition invariance] are obtained. ■

Definition 14 ($\mathcal{M}$-Liveness under Conditional Invariance): An arbitrary invariant φ over DES G with system marker set $\mathcal{M} = \{M_1, M_2, \cdots, M_m\}$ is said to be $\mathcal{M}$-alive under conditional invariance (with respect to G) if

$$G \models \Box\varphi \to \Box\left(\prod_{i=1}^{m} \Diamond M_i\right).$$

Intuitively, along an arbitrary legal state trajectory of DES G for an invariant φ over G that is $\mathcal{M}$-alive under conditional invariance, every marker condition in $\mathcal{M}$ is true infinitely often if φ is always true.

Note that, if ψ is the kernel of invariant φ, the G-validity conditions in Definitions 13 and 14 may be replaced by the following, respectively:

$$G \models \Box\left(\ominus_x\left(\psi, \varphi \cdot \sum_{i=1}^{m} \overline{M_i}\right) \to \psi\right), \text{ and}$$

$$G \models \Box\psi \to \Box\left(\prod_{i=1}^{m} \Diamond M_i\right).$$

The basic concepts in this section are defined for DES model G with an arbitrarily ascertained legal set $\mathcal{I}(G)$ (representing the actual behavior of G). As inferred from the adopted framework of Manna and Pnueli on canonical LTL and fair transition systems [5], [9], one cannot generally talk about making marker progress without the system actual behavior exhibiting some kind of fairness, suggesting a deeper model characterization to refine the set $\mathcal{I}(G)$. The next section presents one such DES model refinement adapted from [5] that is quite general for control-theoretic investigation using the concepts defined herein, and over which the marker-progressive supervisory control problem is studied.

IV. Marker-Progressive Supervisory Control

A. Fair DES Model

Let $\Sigma_{\mathcal{F}} = \Sigma_{\mathcal{C}} \cup \Sigma_{\mathcal{J}}$ denote the set of fair events, where $\Sigma_{\mathcal{C}}$ denotes the strongly fair set of compassionate events, and $\Sigma_{\mathcal{J}}$ denotes the weakly fair set of just events. Without loss of generality, assume $\Sigma_{\mathcal{C}} \cap \Sigma_{\mathcal{J}} = \varnothing$.

Definition 15 (The Fair DES Model): The DES model G (1) is said to be fair [5, p. 256] (with respect to $\Sigma_{\mathcal{F}} \subseteq \Sigma_u$), where $\Sigma_{\mathcal{F}} = \Sigma_{\mathcal{C}} \dot\cup \Sigma_{\mathcal{J}}$ such that, for every state trajectory I of G, $I \in \mathcal{I}(G)$ iff I satisfies the event-fairness formulas:

1) $(\forall \sigma \in \Sigma_{\mathcal{C}}) \models^I \Box\Diamond\xi_\sigma \to \Box\Diamond\tau_\sigma$. (Strong fairness)
2) $(\forall \sigma \in \Sigma_{\mathcal{J}}) \models^I \Diamond\Box\xi_\sigma \to \Box\Diamond\tau_\sigma$. (Weak fairness)

Intuitively, an arbitrary $\sigma \in \Sigma_{\mathcal{C}}$ that is defined at infinitely many states must occur next at infinitely many states; and an arbitrary $\sigma \in \Sigma_{\mathcal{J}}$ that is defined henceforth from a certain state must occur next at infinitely many states. The event-fairness formulas constitute the legal conditions that characterize the interpretation set $\mathcal{I}(G)$ of the fair DES model G.

By the standard criterion imposed on supervisory control technology discussed in Section II-D, a (non-terminating) state trajectory of DES G that is not legal in the absence of control does not become legal under control.[3] In view of this, fair events are set as uncontrollable; otherwise, being able to 'disable' a fair event could contradict the criterion, as it might then become possible for DES G under control f to be kept along a state trajectory I of G that is not legal, i.e., $I \notin \mathcal{I}(G)$ but $I \in \mathcal{I}(G^f)$, in that, over G, the antecedent condition of the event's fairness formula is true, but not the consequent condition.

Henceforth in this paper, unless otherwise stated, G refers to the fair DES model of Definition 15.

With $\Sigma_{\mathcal{F}} \subseteq \Sigma_u$ (as specified in Definition 15), a new auxiliary DES concept over an invariant, logically weaker than Σ_u-invariance and called $\Sigma_{\mathcal{F}}$-invariance, is introduced. Substitute Σ_u with $\Sigma_{\mathcal{F}}$ and 'u' with '$\mathcal{F}$' in D6.2, D7.1, and the definitions of $\tau_{\mathcal{F}}$, $\ominus_{\mathcal{F}}$, $\bigcirc_{\mathcal{F}}$ are obtained for characterizing the concept, around which the $\mathcal{F}$-substituted versions of P4.1, P4.2, and P5.2 apply.

Definition 16 ($\Sigma_{\mathcal{F}}$-Invariance): An arbitrary invariant φ over DES G is said to be $\Sigma_{\mathcal{F}}$-invariant (with respect to G) if $G \models \Box\left(\ominus_{\mathcal{F}}(\varphi) \to \varphi\right)$.

Intuitively, for an invariant φ over DES G that is $\Sigma_{\mathcal{F}}$-invariant, the DES does not 'slip out of φ' on a fair event along an arbitrary legal state trajectory, just as it does not on an uncontrollable event if φ is Σ_u-invariant. In other words, to maintain the φ-information, the DES can branch off a state trajectory that is not legal in G onto one that is.

B. Problem Formulation, Statement & Solvability

The LTL control problem described in Section II-E may now be formulated with a class of supervisors, defined as follows.

Definition 17 ($(P, \mathcal{M})$-Supervisor): Consider the kernel P of an arbitrary invariant over DES G with system marker set $\mathcal{M} = \{M_1, M_2, \cdots, M_m\}$. Then a state feedback supervisor f for G is said to be P-regulating and $\mathcal{M}$-progressive if, respectively, $G^f \models \Box P$ and $G^f \models \Box\left(\prod_{i=1}^{m} \Diamond M_i\right)$. A $(P, \mathcal{M})$-supervisor f is a state feedback supervisor that is P-regulating and $\mathcal{M}$-progressive.

Given the specification pair $(P, \mathcal{M})$ over DES G, formally, the marker-progressive supervisory control problem (MP-SCP) is stated as follows:

MP-SCP: Find a proper $(P, \mathcal{M})$-supervisor f for fair DES G.

The conditions under which the MP-SCP is solvable, i.e., a general solution to the problem exists, are established by the following result.

Theorem 1: Consider an arbitrary invariant φ with ψ as its kernel, over fair DES G with system marker set $\mathcal{M}$. Then there exists a proper $(P, \mathcal{M})$-supervisor f for G, such that

$$(\forall \sigma \in \Sigma_c) \quad f_\sigma = \bigcirc_\sigma(\psi) \quad [\text{rel to } (\varphi, G)]$$

iff φ is: 1) initially satisfied, 2) Σ_u-invariant, 3) $(\mathcal{M}, \varphi)$-condition invariant, 4) $\mathcal{M}$-alive under conditional invariance, and 5) P-history bounded.

[3] Note that a state trajectory *of* DES G as formalized in Section II-C is either legal or is not; and any terminating state trajectory of DES G satisfies every event-fairness formula of the DES and hence is legal.

Proof: Consider an arbitrary invariant φ with ψ as its kernel, over fair DES G with system marker set $\mathcal{M}$.

(If) That φ is initially satisfied and Σ_u-invariant implies φ is initially satisfied and control invariant with a (Σ_u-complete) supervisor f, such that $(\forall\sigma \in \Sigma_c) \quad f_\sigma = \bigcirc_\sigma(\psi)$ [rel to (φ, G)]. It follows that $G^f \models \Box\varphi$. That φ is P-history bounded implies $G^f \models \Box P$.

Next, because φ is initially satisfied and $(\mathcal{M}, \varphi)$-condition invariant, $\mathcal{I}(G^f) \neq \varnothing$; and because φ is Σ_u-invariant and hence $\Sigma_{\mathcal{F}}$-invariant, every $I \in \mathcal{I}(G^f)$ satisfies all the legal (or fairness) conditions of, and over, DES G. By definition, the supervisor f that exists is proper. That φ is initially satisfied and $(\mathcal{M}, \varphi)$-condition invariant also implies every $I \in \mathcal{I}(G^f)$ that is terminating due to supervisor f satisfies $\Box\left(\prod_{i=1}^{m}\Diamond M_i\right)$. And by f being proper and that φ is $\mathcal{M}$-alive under conditional invariance, every other $I \in \mathcal{I}(G^f)$ is in $\mathcal{I}(G)$ and thus satisfies $\Box\left(\prod_{i=1}^{m}\Diamond M_i\right)$. Therefore, $G^f \models \Box\left(\prod_{i=1}^{m}\Diamond M_i\right)$.

Together, by definition, a proper $(P, \mathcal{M})$-supervisor f for G exists, such that $(\forall\sigma \in \Sigma_c) \quad f_\sigma = \bigcirc_\sigma(\psi)$ [rel to (φ, G)].

(Only if) Suppose there is a proper $(P, \mathcal{M})$-supervisor f for DES G, such that $(\forall\sigma \in \Sigma_c) \quad f_\sigma = \bigcirc_\sigma(\psi)$ [rel to (φ, G)]. That f is Σ_u-complete and in the given algebraic form implies φ is control invariant, and hence Σ_u-invariant.

That f is proper implies $\mathcal{I}(G^f) \neq \varnothing$; that it is P-regulating implies $G^f \models \Box P$, together implying $G \models P$, i.e., $\boxminus P$ is initially satisfied (in G). Now, assume φ is not initially satisfied. Then control by the given f on G need not guarantee the invariance of P, contradicting the fact that $G^f \models \Box P$. Therefore, φ is initially satisfied, and has to be P-history bounded for f on G to obtain $G^f \models \Box P$, with $G^f \models \Box\varphi$.

Next, that f is ψ-locally optimal and proper implies $\mathcal{I}(G^f)$ is the largest legal set of state trajectories satisfying $\Box\varphi$ without violating any of the legal (or fairness) conditions of, and over, DES G. That f is $\mathcal{M}$-progressive implies $G^f \models \Box\left(\prod_{i=1}^{m}\Diamond M_i\right)$. It thus follows by this nonempty set $\mathcal{I}(G^f)$ that:

1) Since every $I \in \mathcal{I}(G^f)$ that is terminating due to supervisor f satisfies $\Box\left(\prod_{i=1}^{m}\Diamond M_i\right)$, φ is $(\mathcal{M}, \varphi)$-condition invariant.
2) Since every other $I \in \mathcal{I}(G^f)$ satisfies $\Box\left(\prod_{i=1}^{m}\Diamond M_i\right)$ and is in $\mathcal{I}(G)$, φ is $\mathcal{M}$-alive under conditional invariance.

■

For an arbitrary LTL formula φ over DES G and an arbitrary state trajectory I of G, define operator $\widetilde{\ominus}_u$ such that $\widetilde{\ominus}_u(\varphi) \equiv \widetilde{\ominus}(\tau_u \cdot \varphi)$. It can be shown that

$$\models^{I^{(k)}} \widetilde{\ominus}_u(\varphi) \text{ iff } (k = 0) \text{ or } \models^{I^{(k)}} \ominus_u(\varphi).$$

Then $G \models (\varphi \cdot \Box(\ominus_u(\varphi) \to \varphi) = \Box(\widetilde{\ominus}_u(\varphi) \to \varphi))$. It follows by Theorem 1 that the MP-SCP is solvable iff there exists an invariant φ such that:

INV1) $G \models \Box(\widetilde{\ominus}_u(\varphi) \to \varphi)$, INV2) $G \models \Box(\varphi \to \boxminus P)$,

INV3) $G \models \Box\left(\ominus_x\left(\varphi, \varphi \cdot \sum_{i=1}^{m}\overline{M_i}\right) \to \varphi\right)$, and

INV4) $G \models \Box\varphi \to \Box\left(\prod_{i=1}^{m}\Diamond M_i\right)$.

Lemma 1 ([5, p. 290]): The class of LTL response formulas is closed under Boolean *and* $\cdot$, in that, for arbitrary past formulas ϕ_1, ϕ_2 over DES G,

$$\Box\Diamond\phi_1 \cdot \Box\Diamond\phi_2 \approx \Box\Diamond\left(\phi_2 \cdot \ominus(\overline{\phi_2}\mathcal{S}\phi_1)\right),$$

i.e., $\Box\Diamond\phi_1 \cdot \Box\Diamond\phi_2$ is an LTL response formula (under $\approx$).

Lemma 2: Given arbitrary past formulas φ, M_1, M_2, $\cdots, M_m$ over DES G,

$$\Box\varphi \to \Box\left(\prod_{i=1}^{m}\Diamond M_i\right) \approx \prod_{i=1}^{m}\Box\Diamond\left(\boxminus\varphi \to M_i\right),$$

and is an LTL response formula (under $\approx$).

Proof: By applying temporal logic rules [5]:

$$\begin{aligned}
\Box\varphi \to \Box\Diamond M_i \quad &\approx \overline{\Box\overline{(\overline{\varphi})}} + \Box\Diamond M_i \\
&\approx \Diamond(\overline{\varphi}) + \Box\Diamond M_i \\
&\approx \Box\Diamond(\diamondminus\overline{\varphi}) + \Box\Diamond M_i \\
&\approx \Box\left(\boxminus\Diamond(\diamondminus\overline{\varphi}) + \boxminus\Diamond M_i\right) \\
&\approx \Box\left(\Diamond(\diamondminus\overline{\varphi}) + \Diamond M_i\right) \\
&\approx \Box\Diamond\left(\diamondminus\overline{\varphi} + M_i\right) \\
&\approx \Box\Diamond\left(\boxminus\varphi \to M_i\right).
\end{aligned}$$

Extending $\Box\Diamond M_i$ to $\Box\left(\prod_{i=1}^{m}\Diamond M_i\right)$:

$$\begin{aligned}
\Box\varphi \to \Box\left(\prod_{i=1}^{m}\Diamond M_i\right) \quad &\approx \Box\varphi \to \prod_{i=1}^{m}\Box\Diamond M_i \\
&\approx \prod_{i=1}^{m}\left(\Box\varphi \to \Box\Diamond M_i\right) \\
&\approx \prod_{i=1}^{m}\Box\Diamond\left(\boxminus\varphi \to M_i\right).
\end{aligned}$$

$\Box\Diamond\left(\boxminus\varphi \to M_i\right)$ is a response formula (in canonical form). By Lemma 1,

$$\prod_{i=1}^{m}\Box\Diamond\left(\boxminus\varphi \to M_i\right)$$

is an LTL response formula. Hence,

$$\Box\varphi \to \Box\left(\prod_{i=1}^{m}\Diamond M_i\right)$$

is also an LTL response formula. ■

Theorem 2: The MP-SCP is a canonical safety-response verification problem.

Proof: The expressions in INV1 – INV3 are LTL safety formulas (in canonical form). Since φ, M_i $(1 \leqslant i \leqslant m)$ in INV4 are past formulas, by Lemma 2, the expression in INV4 is an LTL response formula that can be written as a product of m response formulas (in canonical form). Hence the result. ■

C. Controllability of Temporal Safety for Marker Progress

The LTL controllability concept of temporal safety originating in [14] is extended to admit constant progression to markers.

Definition 18 (Controllability): Consider the kernel P of an arbitrary invariant over DES G. Then $\Box P$ is said to be controllable (with respect to G) if $\boxminus P$ is: CT1) initially satisfied, and CT2) Σ_u-invariant.

Definition 19 ($\mathcal{M}$-Directingness): Consider the kernel P of an arbitrary invariant over DES G with system marker set $\mathcal{M}$. Then $\Box P$ is said to be $\mathcal{M}$-directing (with respect to G) if $\boxminus P$ is: MD1) initially satisfied, MD2) $(\mathcal{M}, \boxminus P)$-condition invariant, and MD3) $\mathcal{M}$-alive under conditional invariance.

Condition MD1, equivalently of past formula P being G-valid, is necessary for an arbitrary legal state trajectory I of DES G to satisfy $\boxminus P$ at a state q_k of I, or equivalently, for a k-prefix of I to satisfy $\Box P$. Condition MD2 ensures that every k-prefix, of an arbitrary legal state trajectory of G satisfying $\boxminus P$ at state q_k, can always be extended to a legal state trajectory I that either satisfies $\Box P$, or $\boxminus P$ at some non-terminal state q_j $(j \geqslant k)$ along I, at where all marker conditions in $\mathcal{M}$ are simultaneously satisfied. Condition MD3 ensures that every legal state trajectory of G that satisfies $\Box P$ also satisfies every marker condition in $\mathcal{M}$ infinitely often. Taken together, $\Box P$ is $\mathcal{M}$-directing if every legal state trajectory of DES G has some k-prefix $(0 \leqslant k \leqslant \infty)$ satisfying $\Box P$, and every such prefix can be extended to or is a legal state trajectory of G or its j-prefix $(j \geqslant k)$, satisfying $\Box P$ and infinitely often, every marker condition in $\mathcal{M}$.

Remark 3: Returning to Theorem 1 which characterizes the existence of a proper, ψ-locally optimal solution supervisor for the MP-SCP, Conditions 1, 3, and 4 therein define the $\mathcal{M}$-directingness while Conditions 1 and 2 define the controllability, both of $\Box\psi$ with respect to fair DES model G. As the only liveness requirement, Condition 4 depends on how the set $\mathcal{I}(G)$ is restricted by the fair events of the DES (see Definition 15) for its G-validity. It follows that achieving marker progress under supervised temporal safety depends on DES event fairness in general. With reference to the closed-loop system setup of Fig. 1, one may then interpret a solution supervisor (that exists) as inducing a subset of the given fair events of the DES to render the underlying event-selection mechanism marker-directable under the controllable $\Box\psi$. The supervisor maintains the invariance of ψ, within which the event-selection mechanism is directed by a fair event subset to drive the DES to meeting, infinitely often, every marker condition in $\mathcal{M}$. ■

Definition 20 ($\mathcal{M}$-Controllability): Consider the kernel P of an arbitrary invariant over DES G with system marker set $\mathcal{M}$. Then $\Box P$ is said to be $\mathcal{M}$-controllable (with respect to G) if $\Box P$ is controllable and $\mathcal{M}$-directing.

With respect to the specification pair $(P, \mathcal{M})$, Definition 20 of $\mathcal{M}$-controllability reduces to Definition 18 of controllability if $\mathcal{M}$ is empty or is the set $\{M_1, M_2, \cdots, M_m\}$ of trivial marker conditions, i.e., $M_i \equiv P$ or $M_i \equiv true$ $(1 \leqslant i \leqslant m)$.

Theorem 3: Consider the kernel P of an arbitrary invariant over fair DES G with system marker set $\mathcal{M}$. Then there exists a proper $(P, \mathcal{M})$-supervisor f for G, such that

$$(\forall \sigma \in \Sigma_c) \quad f_\sigma = \bigcirc_\sigma(P) \quad \text{[rel to } (\boxminus P, G)]$$

iff $\Box P$ is $\mathcal{M}$-controllable.

Proof: Let $\varphi \equiv \boxminus P$ in Theorem 1. The result follows by Definitions 18 to 20. ■

For the specification pair $(P, \mathcal{M})$ input to the MP-SCP, the set of all $\mathcal{M}$-controllable invariance formulas whose invariants are not weaker than $\boxminus P$ is introduced:

$$\mathcal{C}(P, \mathcal{M}) = \left\{ \Box\psi \;\middle|\; \begin{array}{l} \Box\psi \text{ is } \mathcal{M}\text{-controllable, where} \\ \psi \text{ is the kernel of an invariant} \\ \text{that is } P\text{-history bounded} \end{array} \right\}.$$

Proposition 7: Consider the kernel P of an arbitrary invariant over fair DES G with system marker set $\mathcal{M}$, and assume $\mathcal{C}(P, \mathcal{M}) \neq \varnothing$. Then $\mathcal{C}(P, \mathcal{M})$ is closed under arbitrary *or*-ings. Specifically, $\mathcal{C}(P, \mathcal{M})$ contains a (unique) supremal element (which is hereby denoted by $\sup \mathcal{C}(P, \mathcal{M})$).

Proof: On the assumption that $\mathcal{C}(P, \mathcal{M}) \neq \varnothing$, let $\Box\psi_i \in \mathcal{C}(P, \mathcal{M})$ for all i in some index set N, and let $\varphi \equiv \sum_{i \in N} \boxminus\psi_i$. It follows that φ is an invariant because, equivalently, $\varphi \equiv \boxminus\psi$, where $\psi \equiv \sum_{i \in N} \boxminus\psi_i$ is therefore the kernel of φ. By temporal logic reasoning, it can be shown that φ is P-history bounded and satisfies the conditions of $\mathcal{M}$-controllability. Therefore, $\Box\psi \in \mathcal{C}(P, \mathcal{M})$. Because $\Box\psi \approx \Box\left(\sum_{i \in N} \boxminus\psi_i\right) \approx \sum_{i \in N} \Box\psi_i$, the arbitrary *or*-ing of $\mathcal{M}$-controllable $\Box\psi_i$'s over N is $\mathcal{M}$-controllable $\Box\psi$. It follows that, over the whole set $\mathcal{C}(P, \mathcal{M})$, the supremal $\mathcal{M}$-controllable element is $\sup \mathcal{C}(P, \mathcal{M}) \approx \Box\left(\sum_{\Box\psi_i \,\in\, \mathcal{C}(P,\mathcal{M})} \boxminus\psi_i\right)$. ■

In logic terms, $\sup \mathcal{C}(P, \mathcal{M}) \approx false$ provided $\mathcal{C}(P, \mathcal{M}) = \varnothing$. Therefore, in general, $\sup \mathcal{C}(P, \mathcal{M}) \in \mathcal{C}(P, \mathcal{M}) \cup \{false\}$. Provided $\mathcal{C}(P, \mathcal{M}) \neq \varnothing$, $\sup \mathcal{C}(P, \mathcal{M})$ is the weakest $\mathcal{M}$-controllable formula that is not weaker than $\Box P$, and it is then called the supremal $\mathcal{M}$-controllable subformula of $\Box P$.

Theorem 4: Consider the kernel P of an arbitrary invariant over fair DES G with system marker set $\mathcal{M}$, and assume $\mathcal{C}(P, \mathcal{M}) \neq \varnothing$. Let $\sup \mathcal{C}(P, \mathcal{M}) \approx \Box\psi$, where ψ is the kernel of some invariant. Then there exists a proper $(P, \mathcal{M})$-supervisor f for G, such that

$$(\forall \sigma \in \Sigma_c) \quad f_\sigma = \bigcirc_\sigma(\psi) \quad \text{[rel to } (\boxminus\psi, G)].$$

Proof: $\Box\psi$ is $\mathcal{M}$-controllable. The result follows by Theorem 3. ■

The solution supervisor characterized in Theorems 3 and 4 is said to be globally optimal for the MP-SCP (if it exists). By this global optimality which is with respect to $(P, \mathcal{M})$, the solution supervisor under Σ_u-completeness is maximally permissive in maintaining the invariance of P while ensuring the constant, guaranteed progress to multiple markers in $\mathcal{M}$.

V. Discussion

A. Example 1: Nonblocking Control of Fair DES's

A general $(P, \mathcal{M})$-supervisor that exists by Theorem 1 is nonblocking [1] for a fair DES G, if the special case of

$$\mathcal{M} = \left\{ \sum_{q \in Q_m} p_q \right\} \tag{2}$$

is considered for the MP-SCP, where p_q is the proposition characterizing the unique state $q \in Q$ of DES G (1) and is given by

$$p_q \equiv \prod_{v_i \in \Pi} (v_i = a_i) \text{ for some } a_i \in Range(v_i)$$

– a state formula which is a predicate such that $\models^q p_q$ and $(\forall q' \in Q \backslash \{q\})\ \models^{q'} \overline{p_q}$, and $\varnothing \subset Q_m \subseteq Q$, where Q_m is a state subset designated as the set of global marker states in DES G that is explained in the introduction. This set Q_m is a key feature in the standard DES model used in standard nonblocking control theory [1], [2], [6], [7]; and the standard DES model is a 5-tuple automaton $(Q, \Sigma, \delta, q_0, Q_m)$ which, in essence, is model (1) with Π abstracted out, Q_0 conventionalized to containing one initial state q_0, and Q_m included.

Note, however, that for this special case of $\mathcal{M}$ (2) reducing the MP-SCP to nonblocking control, the globally optimal $(P, \mathcal{M})$-supervisor that exists by Theorem 4 is in general more restricting on the DES than the globally optimal supervisor that exists by standard nonblocking control theory [1], [7] for some equivalent specification in formal language, i.e., a set of strings (over the event set Σ). The following example illustrates this fact.

For this example, consider a simple standard DES model shown in Fig. 3, with its only global marker state q_0 denoted by a darkened node. For nonblocking control, set $\mathcal{M} = \{p_{q_0}\}$, and treat the model as a fair DES (over which Theorem 4 can then apply). Suppose $\Sigma_u = \{\sigma_2\}$, $\Sigma_{\mathcal{F}} = \varnothing$, and for the specification pair $(P, \mathcal{M})$, $P \equiv true$. By inspection of the DES model in Fig. 3 against the specification pair, it is easy to see that $\sup \mathcal{C}(P, \mathcal{M}) \approx \Box\overline{p_{q_1}}$. Thus the resultant globally optimal $(P, \mathcal{M})$-supervisor disables (controllable) event σ_1 at state q_0, and enables every other event defined at each state the DES can reach under control. But by standard nonblocking control synthesis [1], [7], a language deemed equivalent to the specification pair $(true, \{p_{q_0}\})$ is the (largest) set of strings that start and end in state q_0 under the DES model's transition function; it is thus representable by the same automaton as the DES in Fig. 3, and the resultant globally optimal nonblocking supervisor trivially obtained enables every event defined at each DES state, all of which the DES can reach under control. Clearly then, the former solution supervisor is more restricting on the DES than the latter.

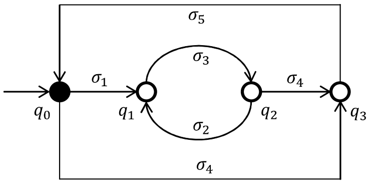


Fig. 3. A simple standard DES model for Example 1. It is an automaton with $Q_m = \{q_0\}$.

In fact, in not explicitly modeling and accounting for system event fairness, a globally optimal supervisor that exists for a DES under standard nonblocking control theory is the most 'optimistic' with regard to global marker state reachability. This is in the sense that, as long as specified safety is never violated, the supervisor permits the DES to enter any state from where it *can* logically transition to regularly reach a state of the global marker state set, but *need not* in runtime without the assumed DES proactivity discussed in the introduction. To elaborate, consider the example above, where the DES under globally optimal, standard nonblocking supervision is permitted to enter state q_1 and traverse in the loop formed by two transitions, namely $\delta(\sigma_3, q_1) = q_2$ and $\delta(\sigma_2, q_2) = q_1$ (see Fig. 3). Without the assumed DES proactivity, the DES need not, in runtime, regularly transition out of this supervisor-permitted loop to reach the global marker state q_0.

B. Example 2: Role of Event Fairness in $\mathcal{M}$-Controllability

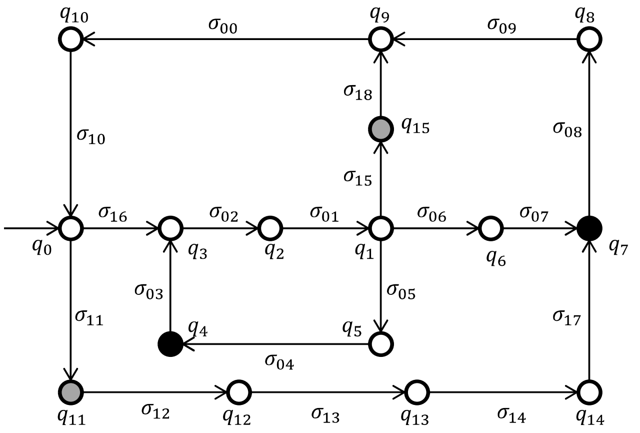


Fig. 4. The DES G for Example 2. The marker states are denoted by darkened nodes. The forbidden states specified by P are denoted by nodes shaded in grey.

This example illustrates the cooperative role of fair events in $\mathcal{M}$-controllability, using a DES G with system marker set $\mathcal{M} = \{p_{q_4}, p_{q_7}\}$, as depicted in Fig. 4. Each marker condition of the given set $\mathcal{M}$ corresponds to a marker state.

Suppose $\Sigma_c = \{\sigma_{03}, \sigma_{11}, \sigma_{12}, \sigma_{15}\}$, $\Sigma_{\mathcal{C}} = \{\sigma_{05}, \sigma_{06}\}$, $\Sigma_{\mathcal{J}} \subseteq \Sigma_u \backslash \Sigma_{\mathcal{C}}$,[4] and for the specification pair $(P, \mathcal{M})$, $P \equiv \overline{p_{q_{11}} + p_{q_{15}}}$. In this example, $\Box P$ is $\mathcal{M}$-controllable: No terminal state results under control that disables events σ_{11}, σ_{15} only, and the strong fairness in uncontrollable events σ_{05}, σ_{06} ensures that marker states q_4, q_7 are visited infinitely often, assuring $\mathcal{M}$-directingness under the controllable $\Box P$.

C. Example 3: Solution With No Global DES Marker State

Although the nonblocking case of $\mathcal{M}$ (2) for a DES can be addressed by standard control theory, it is by the LTL counterpart theory of marker-progressive control in this paper that a globally optimal nonblocking control solution that exists can guarantee constant progress to a global marker state. Furthermore, unlike standard nonblocking control synthesis [1], [2], [7], marker-progressive control synthesis in LTL can also be transparently applied to a modular DES without a

[4] Note that, for this example DES G (see Fig. 4), $\Sigma_{\mathcal{J}}$ can be arbitrarily fixed, since, once $\Sigma_{\mathcal{C}}$ is fixed, $\mathcal{I}(G)$ is the same regardless of whether the other uncontrollable events are just or not.

global marker state, or without one surviving under control as the following example shows.

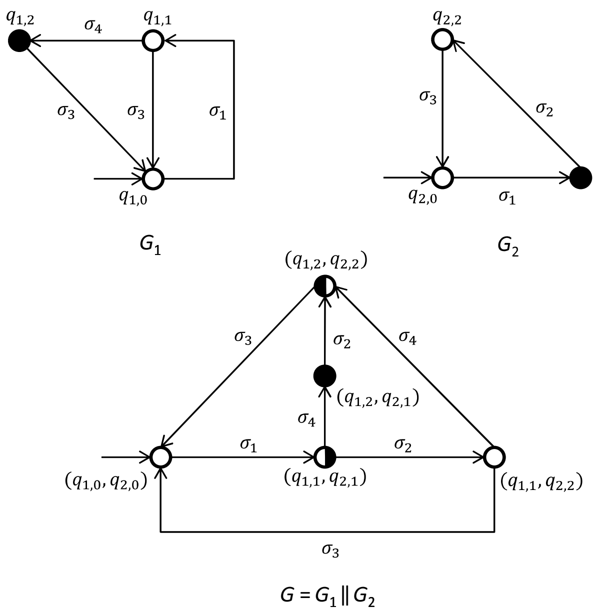


Fig. 5. The DES $G = G_1 \parallel G_2$ for Example 3, where $G = (\Pi, Q, \Sigma, \delta, p_{q_0})$, $G_i = (\Pi_i, Q_i, \Sigma_i, \delta_i, p_{q_{i,0}})$ $(1 \leqslant i \leqslant 2)$. Herein, $\parallel$ is a modified version of the synchronous operator [6], [7], by which G is constructed such that $\Pi = \Pi_1 \dot{\cup} \Pi_2$, $Q = Q_1 \times Q_2$, $\Sigma = \Sigma_1 \cup \Sigma_2$, $p_{q_0} \equiv p_{q_{1,0}} \cdot p_{q_{2,0}}$ with $q_0 = (q_{1,0}, q_{2,0})$, and $\delta(\sigma, (q_1, q_2))$ is: 1) $(\delta_1(\sigma, q_1), \delta_2(\sigma, q_2))$ if $\sigma \in \Sigma_1 \cap \Sigma_2$ and both $\delta_1(\sigma, q_1)$ and $\delta_2(\sigma, q_2)$ are defined, 2) $(\delta_1(\sigma, q_1), q_2)$ if $\delta_1(\sigma, q_1)$ is defined and $\sigma \notin \Sigma_2$, 3) $(q_1, \delta_2(\sigma, q_2))$ if $\delta_2(\sigma, q_2)$ is defined and $\sigma \notin \Sigma_1$, and 4) undefined, otherwise. A darkened node denotes a marker state that in G_i is identified by the corresponding marker condition of the specified set $\mathcal{M}$, and in G by the product of its component marker states. For modular G, a node that is half-darkened denotes a state containing the marker state of one G_i.

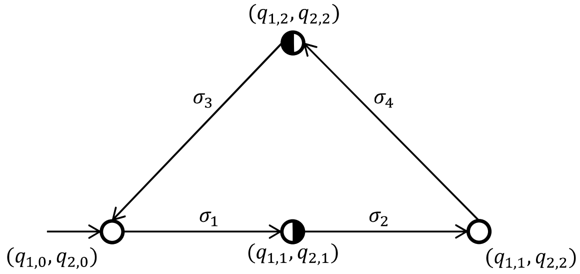


Fig. 6. Transition model equivalent of the globally optimal $(P, \mathcal{M})$-supervisor that exists for Example 3. It is an automaton with no marker state.

For this example, refer to a modular DES G with system marker set $\mathcal{M} = \{p_{q_{1,2}}, p_{q_{2,1}}\}$, as depicted and described in Fig. 5. Then suppose $\Sigma_c = \{\sigma_3, \sigma_4\}$, $\Sigma_{\mathcal{C}} = \varnothing$, $\Sigma_{\mathcal{J}} = \varnothing$ (as arbitrarily fixed), and for the specification pair $(P, \mathcal{M})$, $P \equiv \big(\overline{p_{q_{1,2}}}\ \mathcal{S}\ p_{q_{1,1}} \rightarrow \overline{p_{q_{2,0}}}\big) \cdot \big(\overline{p_{q_{2,2}}}\ \mathcal{S}\ p_{q_{2,1}} \rightarrow \overline{p_{q_{1,2}}}\big)$. The first product component of the temporal-safety part $\Box P$ may be paraphrased as follows: 'Whenever Process G_1 has not proceeded to its state $q_{1,2}$ since entering its state $q_{1,1}$, Process G_2 must not enter its state $q_{2,0}$.' The second component may be paraphrased similarly. It can be shown that this temporal-safety part may be equivalently rewritten with $P \equiv \big(\circleddash p_{q_{1,1}} \rightarrow \overline{p_{q_{2,0}}}\big) \cdot \big(\circleddash p_{q_{2,1}} \rightarrow \overline{p_{q_{1,2}}}\big)$, and $\Box P$ is $\mathcal{M}$-controllable: $\mathcal{M}$-directingness is trivially assured under the controllable $\Box P$.

By Theorem 3, it follows that a proper $(P, \mathcal{M})$-supervisor that is globally optimal exists. However, though correct, this solution supervisor is deemed blocking with respect to the global DES marker state set $\{(q_{1,2}, q_{2,1})\}$, as may be observed from the transition model equivalent of the solution shown in Fig. 6, which has no marker state. As a result, applying standard nonblocking control synthesis [1], [7] will yield the unwarranted outcome that rejects the transition model equivalent, and in fact returns no nontrivial solution. To see this, note that standard control synthesis entails imposing a finite automaton equivalent of the given specification pair $(P, \mathcal{M})$ on the standard DES model $(Q, \Sigma, \delta, (q_{1,0}, q_{2,0}), \{(q_{1,2}, q_{2,1})\})$. This is to select their common formal marker sublanguage, with a marker sublanguage of an automaton being a subset of finite strings that start from the initial state and end in a marker state under the automaton's transition function. Thus this selection actually corresponds to enforcing the refined system marker set $\mathcal{M}' = \{p_{q_{1,2}} \cdot p_{q_{2,1}}\}$ under the given $\Box P$, and it returns an empty marker language.

Beyond the specifications for the three examples in this section, the use of past formulas enables an even more flexible specification of system marker conditions, in particular among the component processes of a modular DES. This built-in flexibility, coupled with the introduction of fair events in DES's, distinguishes LTL marker-progressive control from formal language multitasking control [8] and the nonblocking special case [1]. This is besides their treating different feedback spaces as fundamental – with the former in Π-state space and the latter in Σ-event space.

VI. Related Work

In the DES field of supervisory control, initial efforts propose LTL for specifying and verifying safety and liveness properties (see, e.g., in [22], [23], [24]). However, since the LTL control theory of temporal safety for DES's subsequently reported in [14], [15], [16], there has been relatively less research using temporal logic for *control-theoretic* development of logical DES's, other than the following subsequent major efforts. The first [25] is the control theory in CTL* for nonterminating DES's, to handle a class of CTL* control specifications for safety, reachability, liveness, and stability. The full branching time logic of CTL* is a hybrid of CTL and LTL. The second effort [26] reformulates the controllability results, among others, of supervisory control for model checking in an epistemic temporal logic – a logic based on CTL* that is augmented with one version of additional operators for reasoning about knowledge and belief. The third effort [27] is an algorithm for computing static state feedback supervisors based on specifications in a fragment of CTL that supports writing requirements of continual reachability of multiple marker state sets for multitasking [8]. However, although these research efforts have their own merits, the use of past formulas fundamental to this paper is not considered, and no temporal logic control research prior to this paper has studied the role of DES event fairness in dynamic state feedback control. Besides, the CTL* concept of controllability defined in the first effort [25] does not have the familiar system-theoretic treatment reminiscent of standard control theory [1], [2]. The control specifications considered in the second effort [26] are in a formal language, not a temporal logic formula, and so no direct leverage can be made of the specification merits of natural language expressiveness and readability that temporal logic can offer to the system designer. Finally, while the LTL combined operator $\Box\Diamond$ can specify different sets of marker states, asserting that each set *must* regularly be entered in a DES modeled by a transition system under event fairness, the corresponding CTL combined operator used in the third effort [27] is only meant to assert that each marker state set specified *can* logically be entered regularly in a DES represented by a transition system with no modeled event fairness.

For a setting where controlled and monitored actions are considered in an environment modeled by a labeled transition system (LTS), a related but different type of control problem is studied [28]. This problem is about finding an LTS-based controller that 'legally synchronizes' with the environment (without deadlocking), so as to satisfy a fluent LTL formula of the form $C \cdot (A \rightarrow B)$, where A, B, C are event-predicated formulas, with A, B modeling the liveness, in the sense of infinite oftenity, of the environment assumptions and system goals, respectively, and with C specifying a temporal safety. The fluent LTL used is a variant of LTL developed for specifying state-based temporal properties about event occurrences [29]. In [30], the problem is further investigated where success and failure of each controlled action in a given subset are modeled, over which a specialized notion of strong fairness is defined that a success-recurrent solution controller is based on. Besides the settings for the problem that include treating the event space as fundamental as in standard control theory [1], also different from this paper is the fact that neither the concept of controllability embracing liveness nor the optimality of control is formulated and studied in these research efforts [28], [30].

Other related research efforts in less comparable settings use LTL primarily as a specification language, with controller synthesis that is not syntax-based and uses ω-automata – automata that generate (or accept) ω-languages which are sets of infinite strings, and are of either the Rabin or Büchi type, with each type referring to a different (string) acceptance condition used. The research efforts therefore require the first steps of translating specified LTL formulas into ω-automata. One effort [31], [32] presents an evaluation semantics of LTL formulas for Petri net (PN) models and deterministic PN control synthesis of DES's; this synthesis entails a composition between the Büchi automaton translated from an LTL formula and a PN DES model. Another effort [33] uses LTL for designing *hierarchically-organized* controllers for *concurrent* DES's [34], while yet another [35] uses LTL extended with some quality operators that are 'normalized real-number evaluable' for designing *directed* controllers [36], [37] of a specified minimal quality. The latter two efforts focus on developing Rabin [33] or Büchi [35] game-based control synthesis methods.

In yet another effort [38], a translator of propositional LTL, i.e., LTL restricted to its propositional fragment only, is specially developed for standard nonblocking control synthesis in finite automata. Over the standard, finite state DES model augmented with propositional state variables, the translator tailors and limits a useful class of state-based, response LTL formulas to selecting, as control specifications, formal marker sublanguages of the DES whose corresponding state trajectories satisfy the LTL formulas, and which the translator outputs as finite trim automata – a trim automaton being one generating strings that can be extended to end in a marker state under its transition function. The research [38] also reviews a number of early research efforts on control of DES's using temporal logic that are not covered in this paper.

In a different language setting, the problem of controlling DES's generating ω-languages to meet progress or liveness specifications expressed also in ω-languages has been investigated in [39], [40]. Among the earliest research efforts on liveness in supervisory control, these studies entail a safety bound of ω-closure on specifications relative to (the non-terminating behavior of) a given DES, as originally introduced in [39]. In the automata-theoretic case of ω-languages technically related to LTL, a supervisor that exists can be constructed [40] to 'enforce' liveness specifications in a 'live' DES model, based only on its making control decisions over finite prefix strings. This essence of control is analogously manifested in the LTL approach of this paper as controlling temporal safety for marker progress in a fair DES model, where, relative to the respective DES models, $\mathcal{M}$-directingness is analogous to ω-closure.

On fairness, a conceptually related but technically different treatment in a formal ω-languages and finite automata framework for modular DES's is presented in [41]. This is a rare paper in the DES control literature, besides an early study of some bounded versions of fairness for DES's in Petri nets [42], both motivated by an awareness of the significance of fairness in controlled systems. In that paper [41], a bounded version of strong fairness is defined instead for an ω-language, which is such that for every event in each infinite string, the number of events between successive occurrences of the event cannot exceed a bound specified *a priori*. That paper, however, is on control synthesis to realize the defined bounded fairness in modular DES's, and not on that to realize constant marker progress in fair DES's under specified temporal safety that this paper is about.

VII. Conclusion

In LTL, the existence and synthesis results of state feedback for marker-progressive control of fair DES's, namely Theorems 1 to 4, are developed. Concluding, the regular temporal-safety execution of multiple DES tasks can be coachieved by supervision and event fairness of the DES under its control.

Research studies in terms of system and control concepts in different mathematical formalisms are standard problems for the control theorist. For the practitioner, the important problem is concrete control synthesis. In future work, this paper provides an LTL control foundation for research on two approaches to accomplish the control synthesis task for fair DES's in a transparent, syntax-based or symbolic fashion, by which available industrial-strength software tools for theorem proving and model checking may be adapted, enhanced, and used. One approach to the control synthesis task is by canonical LTL verification [12], [13] based on Theorem 2 for finding a general control solution (characterized) in Theorem 1. In the case [10], [43] of a decidable specification pair $(P, \mathcal{M})$ over a finite state DES, the other approach is by fixpoint computation of $\sup \mathcal{C}(P, \mathcal{M})$ that uses the synthesis method in [16] as a foundation for finding the globally optimal solution in Theorem 3, or more generally in Theorem 4.

Finally, recent developments [44], [45] have formalized the mathematical links between some supervisory control problems from the DES control engineering field and some reactive synthesis problems from the computer science field, resulting in reductions between the problems that allow synthesis algorithms from one field to be applied to the other. In the same vein, it may be of interest to investigate the possible links between the LTL problem of supervisory control studied in this paper and an LTL problem of reactive synthesis [46], by which some problem reduction of the former may be found to which the solution algorithm for the latter can be applied, complementing the two approaches suggested above for future work.

References

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

[2] W. M. Wonham and P. J. Ramadge, "On the supremal controllable sublanguage of a given language," *SIAM Journal of Control and Optimization*, vol. 25, no. 3, pp. 637–659, May 1987.

[3] D.-H. Kim, G.-M. Park, Y.-H. Yoo, S.-J. Ryu, I.-B. Jeong, and J.-H. Kim, "Realization of task intelligence for service robots in an unstructured environment," *Annual Reviews in Control*, vol. 44, pp. 9–18, October 2017.

[4] U.-H. Kim and J.-H. Kim, "A stabilized feedback episodic memory (SF-EM) and home service provision framework for robot and IoT collaboration," *IEEE Transactions on Cybernetics*, vol. 50, no. 5, pp. 2110–2123, May 2020.

[5] Z. Manna and A. Pnueli, *The Temporal Logic of Reactive and Concurrent Systems: Specification*. Springer-Verlag New York, Inc, 1992.

[6] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*, 2nd ed. Springer, 2008.

[7] W. M. Wonham and K. Cai, *Supervisory Control of Discrete-Event Systems*, A. Isidori, J. H. van Schuppen, E. D. Sontag, and M. Krstic, Eds. Springer, Cham, Switzerland, 2019.

[8] M. H. de Queiroz, J. E. R. Cury, and W. M. Wonham, "Multitasking supervisory control of discrete-event systems," *Discrete Event Dynamic Systems : Theory and Applications*, vol. 15, no. 4, pp. 375–395, December 2005.

[9] Z. Manna and A. Pnueli, "Completing the temporal picture," *Theoretical Computer Science*, vol. 83, no. 1, pp. 97–130, 1991.

[10] N. Piterman and A. Pnueli, "Temporal logic and fair discrete systems," in *Handbook of Model Checking*, E. M. Clarke, T. A. Henzinger, H. Veith, and R. Bloem, Eds. Springer, Cham, Switzerland, 2018, pp. 27–73.

[11] M. Y. Vardi, "Branching vs. linear time: Final showdown," in *Lecture Notes in Computer Science: Tools and Algorithms for the Construction and Analysis of Systems - TACAS 2001, Vol. 2031*, M. Tiziana and Y. Wang, Eds. Springer-Verlag Berlin, Heidelberg, 2001, pp. 1–22. [Online]. Available: https://link.springer.com/content/pdf/10.1007/3-540-45319-9_1.pdf

[12] Z. Manna and A. Pnueli, *Temporal Verification of Reactive Systems: Safety*. Springer-Verlag New York, Inc, 1995.

[13] ——, "Temporal verification of reactive systems: Progress," Draft, 1996. [Online]. Available: http://theory.stanford.edu/~zm/tvors3.html (Accessed October 2018).

[14] K. T. Seow and R. Devanathan, "A temporal logic approach to discrete event control for the safety canonical class," *Systems and Control Letters*, vol. 28, no. 4, pp. 205–217, August 1996.

[15] K. T. Seow, "Existence characterizations of temporal-safety supervisors," *IEEE Transactions on Automatic Control*, vol. 47, no. 10, pp. 1779–1783, October 2002.

[16] ——, "Syntax-based synthesis for temporal-safety supervision," *Automatica*, vol. 41, no. 11, pp. 1965–1972, November 2005.

[17] Y. F. Chen, Z. W. Li, K. Barkaoui, and M. Uzam, "New Petri net structure and its application to optimal supervisory control: Interval inhibitor arcs," *IEEE Transactions on Systems, Man and Cybernetics: Systems*, vol. 44, no. 10, pp. 1384–1400, September 2014.

[18] J. H. Ye, Z. W. Li, and A. Giua, "Decentralized supervision of Petri nets with a coordinator," *IEEE Transactions on Systems, Man and Cybernetics: Systems*, vol. 45, no. 6, pp. 955–966, June 2015.

[19] Y. F. Chen, Z. W. Li, K. Barkaoui, N. Q. Wu, and M. C. Zhou, "Compact supervisory control of discrete event systems by Petri nets with data inhibitor arcs," *IEEE Transactions on Systems, Man and Cybernetics: Systems*, vol. 47, no. 2, pp. 364–379, February 2017.

[20] P. J. Ramadge and W. M. Wonham, "Modular feedback logic for discrete event systems," *SIAM Journal of Control and Optimization*, vol. 25, no. 5, pp. 1202–1218, September 1987.

[21] R. Kumar, V. K. Garg, and S. I. Marcus, "Predicate and predicate transformers for supervisory control of discrete event dynamical systems," *IEEE Transactions on Automatic Control*, vol. 38, no. 2, pp. 232–247, February 1993.

[22] J. G. Thistle and W. M. Wonham, "Control problems in a temporal logic framework," *International Journal of Control*, vol. 44, no. 4, pp. 943–976, 1986.

[23] J.-Y. Lin and D. Ionescu, "Verifying a class of nondeterministic discrete event systems in a generalized temporal logic," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 22, no. 6, pp. 1461–1469, 1992.

[24] ——, "A reachability synthesis procedure for discrete eventvsystems in a temporal logic framework," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 24, no. 9, pp. 1397–1406, September 1994.

[25] S. Jiang and R. Kumar, "Supervisory control of discrete event systems with CTL* temporal logic specifications," *SIAM Journal of Control and Optimization*, vol. 44, no. 6, pp. 2079–2103, November 2006.

[26] G. Aucher, "Supervisory control theory in epistemic temporal logic," in *Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014)*, Paris, France, May 2014, pp. 333–340.

[27] B. C. Rawlings, S. Lafortune, and B. E. Ydstie, "Supervisory control of labeled transition systems subject to multiple reachability requirements via symbolic model checking," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 2, pp. 644–652, March 2020.

[28] N. D'Ippolito, V. A. Braberman, N. Piterman, and S. Uchitel, "Synthesis of live behaviour models," in *Proceedings of the 18th ACM SIGSOFT International Symposium on Foundations of Software Engineering (FSE)*, Sante Fe, New Mexico, USA, November 2010, pp. 77–86.

[29] D. Giannakopoulou and J. Magee, "Fluent model checking for event-based systems," *ACM Software Engineering Notes*, vol. 28, no. 5, pp. 257–266, September 2003.

[30] N. D'Ippolito, V. A. Braberman, N. Piterman, and S. Uchitel, "Synthesis of live behaviour models for fallible domains," in *Proceedings of the 33rd International Conference on Software Engineering (ICSE)*, Waikiki, Honolulu , HI, USA, May 2011, pp. 211–220.

[31] B. Lacerda, "Supervision of discrete event systems based on temporal logic specifications," Instituto Superior Técnico, Universidade Tecnica de Lisboa, Lisbon, Portugal, Doctor of Philosophy (Ph.D) Thesis, September 2013.

[32] B. Lacerda and P. U. Lima, "On the notion of uncontrollable marking in supervisory control of Petri nets," *IEEE Transactions on Automatic Control*, vol. 59, no. 11, pp. 53–61, November 2014.

[33] A. Sakakibara and T. Ushio, "Hierarchical control of concurrent discrete event systems with linear temporal logic specifications," *IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences*, vol. E101-A, no. 2, pp. 313–321, February 2018.

[34] Y. M. Willner and M. Heymann, "Supervisory control of concurrent discrete-event systems," *International Journal of Control*, vol. 54, no. 5, pp. 1143–1169, 1991.

[35] A. Sakakibara and T. Ushio, "Directed control of discrete event systems with LTL[$\mathcal{F}$] specifications," in *Proceedings of the 57th IEEE IEEE International Conference on Decision and Control*, Miami Beach, FL, USA, December 2018, pp. 3962–3967.

[36] J. Huang and R. Kumar, "An optimal directed control framework for discrete event systems," *IEEE Transactions on Systems, Man, and*

Cybernetics - Part A: Systems and Humans, vol. 37, no. 5, pp. 780–791, September 2007.

[37] ——, “Optimal nonblocking directed control of discrete event systems,” *IEEE Transactions on Automatic Control*, vol. 53, no. 7, pp. 1592–1603, August 2008.

[38] K. T. Seow, “Integrating temporal logic as a state-based specification language for discrete-event control design in finite automata,” *IEEE Transactions on Automation Science and Engineering*, vol. 4, no. 3, pp. 451–464, 2007.

[39] P. J. Ramadge, “Some tractable supervisory control problems for discrete-event systems modeled by Büchi automata,” *IEEE Transactions on Automatic Control*, vol. 34, no. 1, pp. 10–19, January 1989.

[40] J. G. Thistle and W. M. Wonham, “Supervision of infinite behaviour of discrete event systems,” *SIAM Journal of Control and Optimization*, vol. 32, no. 4, pp. 1098–1113, July 1994.

[41] P. Gohari and W. M. Wonham, “Efficient implementation of fairness in discrete-event systems using queues,” *IEEE Transactions on Automatic Control*, vol. 50, no. 11, pp. 1845–1849, November 2005.

[42] T. Murata and M. Silva, “Petri-net-based fairness concepts for discrete event systems,” in *Progress in Systems and Control Theory: Realization and Modeling in System Theory - MTNS 1989, Vol. 3*, M. A. Kaashoek, J. H. van Schuppen, and A. C. M. Ran, Eds. Birkhäuser, Boston, USA, 1990, pp. 549–557.

[43] I. Hodkinson, F. Wolter, and M. Zakharyaschev, “Decidable fragments of first-order temporal logics,” *Annals of Pure and Applied Logic*, vol. 106, no. 1, pp. 85–134, 2000, [Online]. Available: https://annals-of-pure-and-applied-logic/vol/106/issue/1.

[44] R. Ehlers, S. Lafortune, S. Tripakis, and M. Y. Vardi, “Supervisory control and reactive synthesis: a comparative introduction,” *Discrete Event Dynamic Systems : Theory and Applications*, vol. 27, no. 2, pp. 209–260, June 2017.

[45] A.-K. Schmuck, T. Moor, and R. Majumdar, “On the relation between reactive synthesis and supervisory control of non-terminating processes,” *Discrete Event Dynamic Systems : Theory and Applications*, vol. 30, no. 1, pp. 81–124, March 2020.

[46] R. Majumdar, N. Piterman, and A.-K. Schmuck, “Environmentally-friendly GR(1) synthesis,” in *Lecture Notes in Computer Science: Tools and Algorithms for the Construction and Analysis of Systems - TACAS 2019, Vol. 11428*, T. Vojnar and L. Zhang, Eds. Springer, Cham, Switzerland, 2019, pp. 229–246.

Kiam Tian Seow (SM’10) received the B.Eng. degree (Hons.) in electrical engineering from the National University of Singapore, Singapore, in 1990, and the M.Eng. and Ph.D. degrees in electrical and computer engineering from Nanyang Technological University (NTU), Singapore, in 1993 and 1998, respectively.

Since 2014, he has been a Visiting Professor with the Robot Intelligence Technology Laboratory (http://rit.kaist.ac.kr/home/Members), KAIST, Daejeon, South Korea. From 2014 to 2016, he was an Adjunct Associate Professor with the School of Computer Science and Engineering, NTU, where he was a full-time Faculty Member from 2003 to 2014. He has held visiting research appointments with the Systems Control Group, University of Toronto, Toronto, ON, Canada, in 1997; the School of Electrical Engineering, KAIST, in 2002; the Nippon Telegraph and Telephone Corporation (NTT) Communication Science Laboratories, Kyoto, Japan, in 2003; and the Institute of Information Science, Academia Sinica, Taipei, Taiwan, in 2005. His current research interests are in modeling, control design, and applications of discrete-event and agent systems.