Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Decentralized Supervisory Control of Discrete-Event Systems in Canonical Temporal Logic

KIAM TIAN SEOW, (Senior Member, IEEE)
The Robot Intelligence Technology Laboratory, School of Electrical Engineering, KAIST, Daejeon 305-701, South Korea (e-mail: kiamtian@singnet.com.sg; ktseow@ieee.org)

ABSTRACT

The existence theory of decentralized control is given a new development in canonical linear-time temporal logic (canonical LTL) for a refined class of discrete-event systems (DES's). A discussion at length in the introduction motivates this theoretical exposition. In extending an existing monolithic control foundation for a specification of canonical LTL safety, the main thrust is a new LTL characterization of the system concept of co-observability in its general form, called universal co-observability. A control-theoretic study of universal co-observability along with its key boundary cases, all specializable to counterparts of existing language-based versions, demonstrates the merits of transparency and structural generality that the characterization embodies; importantly, this study reveals a more complete understanding, namely, decentralized control under co-observability is specification-modular control under local partial observation, exercised without non-trivial coordination.

INDEX TERMS Fair discrete-event systems, linear-time temporal logic, supervisory control architecture.

I. INTRODUCTION

CONTROL architecture is an organizational structure for control of a system by generally not one monolithic, but many smaller entities. Research in control architectures for discrete-event systems (DES's) [1] is motivated by their strategic role in smart complexity management, which is about not only computationally efficient control synthesis that has been the research's primary focus, but also modularly cleaner final solution for supervisory control that has perhaps received less attention. As a founding premise, DES modeling permits defining cyber and physical qualitative changes of the abrupt and non-differentiable type present in modern application systems of all kinds as discrete events – the model transition labels at the heart of the system behavioral design matter. Supervisory control theory of DES's – architecturally enhanced – can aptly cover the control of such, often complex, application systems. These systems are continually borne out of human imagination, in constant contemplation of the evolving technology in artificial intelligence, robotics, and the Internet that has, at the outset, motivated the 1980's founding of the DES control field [2], [3] and its continual development [4].

The need for smart complexity management arises from the fact that nonblocking control in its simplest existential setting [2], [3], perhaps fundamentally the most important problem that founded the DES control field [1], [5], is already NP-hard for a modular DES [6] – a DES of subsystems, and its basic solution is not only monolithic but also prescribed in a rudimentary framework, of formal languages and automata [1], [5], that somewhat lacks a transparent structure. This monolithic solution [1], [5] is about control or supervision to keep a DES in temporal safety implicit in a given language specification, without blocking the DES from reaching a state of the DES marker state set targeted within the specification to signal some task completion as well, but in general not guaranteeing that the DES will enter the marker state set with the specification-intended liveness or eventuality. In the formal languages and automata framework, temporal safety is prescribed by prefix-closure of the given specification for the DES. The standard setup for implementing a control solution is a closed loop interconnecting control and system via feedback. In managing complexity targeting synthesis efficiency, the use of control architecture is extensively studied in the literature to date, extending control to multiple closed loops in a modular, decentralized, or hierarchical setup [1]. However, the rudimentary basis has rendered concept formulations often arduous to tease out, inevitably making the resulting research results and control solutions less appealing to application developers in general. This is one important issue if supervisory control theory is to become widely used in real-world applications.

This paper studies a new system-theoretic development of a control architecture for DES's. The philosophy adopted is that, in order to more effectively accentuate the role of a control architecture in smart complexity management, it is imperative to formulate and investigate transparent characterizations of system concepts underlying the existence of control in the architecture, which is the focus of this paper. Specifically, this paper investigates decentralized control – the architectural type central in an increasingly cyber-physicalized real world of innovative applications that include connected smart cars and robots, and home and office automation, where limited actuator and sensor capabilities have to be distributed among local controls. Set in the same syntax-based framework of canonical linear-time temporal logic (canonical LTL) [7], [8], the investigation extends the supervisory control of a class of fair DES's as developed in two recent papers [9], [10]. Therein, a fundamental state feedback control problem is defined for a fair DES model, and studied in terms of its solution existence conditions [9] and a method for its solution control synthesis [10]. Called marker-progressive supervisory control, the problem is about ensuring constant marker progress – a form of liveness signaling a certain modality or regularity of DES task completion – under specified temporal safety. In canonical LTL [7], [8], the part on constant marker progress is specified by the infinite oftenity of every past formula of some system marker set $\mathcal{M}$, while the part on temporal safety is specified by the invariance of some 'non-redundant' past formula P[1]; the resultant logical product is the specification input of the problem, which is denoted by the pair $(P, \mathcal{M})$. Two unique findings distinguish this LTL control problem formulation. One, as explained in [9], this problem subsumes control nonblockingness first studied in [2], [3] and its multitasking generalization [11], both as important special cases but refined in LTL for fair DES's admitting event fairness – the model feature missing in the conventional supervisory control foundation [1], [5], [11] that guaranteeing constant marker-progress is found to require in general. Two, as pointed out in [10], as long as it satisfies specification pair $(P, \mathcal{M})$, a control solution to this problem can, in its generality, be finite [2], infinite [12], [13], or both, unifying both types of controlled behavior as the natural outcome of supervisory control. In extending the marker-progressive control problem to decentralized control, this paper is motivated by the potential of canonical LTL in continued theory refinement and transparency enhancement towards smart complexity management.

In general, decentralized control of a DES is about one or more supervisory controls working concurrently, with each responding to and making decisions to act on state transitions of events from the subset under their jurisdiction; these event subsets of the DES are generally different but not necessarily pairwise disjoint. Broadly speaking, research in this area has evolved on two goals:

1) To meet a global specification on a monolithic DES.
2) To meet different local specifications (concurrently) on, in practice, generally different subsystems of a modular DES.

The generic setups of decentralized control architecture for the first and second goal listed above are depicted in Figs. 1 and 2, respectively, albeit through the lenses of LTL marker-progressive control on event enablement logic.

To elaborate, the former goal entails some type of co-observability structure that a DES must have in relation to a global specification, by which a number of local controls exist that can each decide and act concurrently on a controllable event subset to meet the specification. Referring to Fig. 1, each local control f^i at site i $(1 \leqslant i \leqslant n_o)$ is on a subset Σ_c^i of the system controllable event set Σ_c, and over a submodel $G[\Sigma_o^i]$. Ideally a size-reduced system model, the submodel $G[\Sigma_o^i]$ contains only events of subset Σ_o^i, locally observable via the observation channel $O_{\Sigma_o^i}$ for DES G modeled on event set Σ. Originated in [14] and first generalized in [15], the system concept of co-observability and its variants have been developed [14]–[21]; the way decisions of local controls (on an event) are combined and acted upon in a decentralized control architecture is in accordance to its characteristic fusion rule F – a static logic function or its equivalent – for which the formulation of the original or a variant of the co-observability concept is based. Being static means F is memoryless, i.e., independent of past system evolution; it is thus distributable over different fusion nodes each hosting the actuator of a controllable event.

The latter goal entails synthesis of (high-level, logical product) coordination between individual subsystem controls concurrently supervising a modular DES, such that every individual control that meets a local specification when standalone is refined to forestall potential conflicts when in concurrency with other controls – a conflict being different controls acting in contradiction on the same system controllable event that results in blocking [1], [5], or progress termination [9], [10] in its generality as considered in this paper.

The role of (nonblocking or, more generally under DES event fairness, marker-progressive) coordination is thus to further restrict so as to retain the meeting of local specification in every subsystem running in concurrency. It is well known that this coordination is done by exercising disablement control as needed on each system controllable event otherwise product-enabled by local controls, to preempt any potential conflict ahead arising from some shared controllable event between different local specifications (to be met as some logical product) on the system. The control literature has focused mostly on obtaining decentralized con-

[1] Formally called the kernel of some invariant [9], both the terms *invariant* and its *kernel* will be reviewed together later.

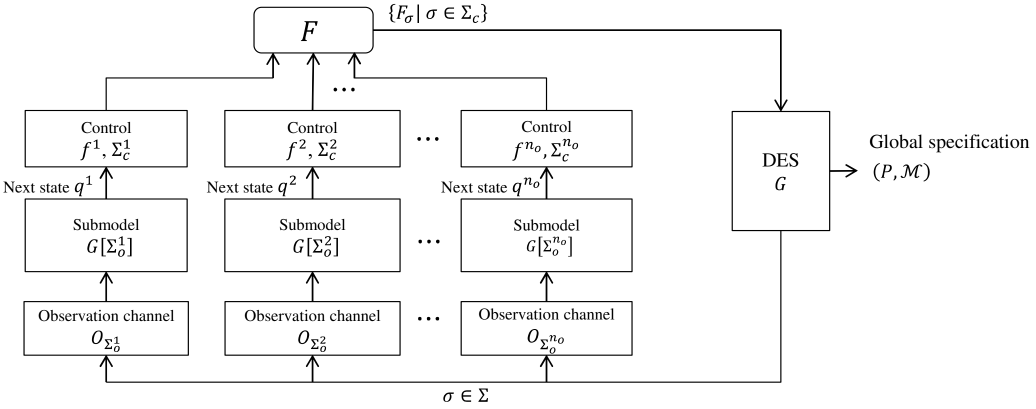

F
$\{F_\sigma | \sigma \in \Sigma_c\}$
Control f^1, Σ_c^1
Control f^2, Σ_c^2
Control $f^{n_o}, \Sigma_c^{n_o}$
Next state q^1
Next state q^2
Next state q^{n_o}
Submodel $G[\Sigma_o^1]$
Submodel $G[\Sigma_o^2]$
Submodel $G[\Sigma_o^{n_o}]$
Observation channel $O_{\Sigma_o^1}$
Observation channel $O_{\Sigma_o^2}$
Observation channel $O_{\Sigma_o^{n_o}}$
DES G
Global specification $(P, \mathcal{M})$
$\sigma \in \Sigma$

FIGURE 1. The generic decentralized control architecture setup for meeting a global specification on a monolithic DES.

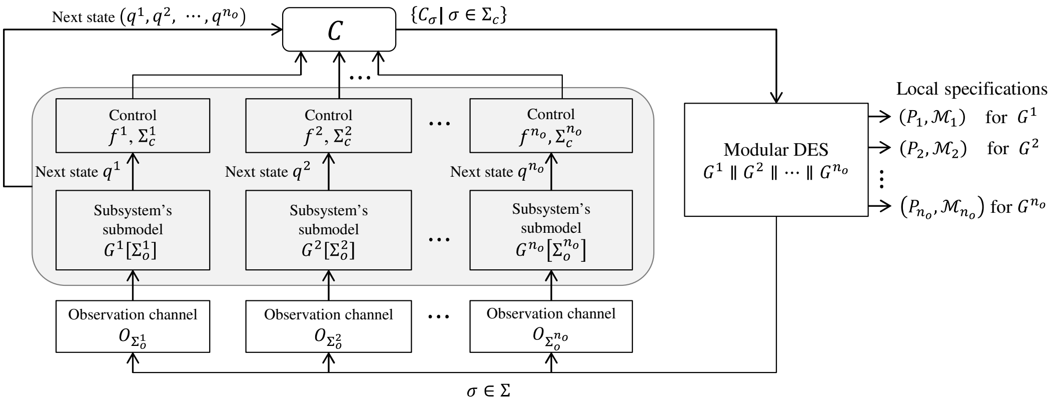

Next state $(q^1, q^2, \cdots, q^{n_o})$
C
$\{C_\sigma | \sigma \in \Sigma_c\}$
Control f^1, Σ_c^1
Control f^2, Σ_c^2
Control $f^{n_o}, \Sigma_c^{n_o}$
Next state q^1
Next state q^2
Next state q^{n_o}
Subsystem's submodel $G^1[\Sigma_o^1]$
Subsystem's submodel $G^2[\Sigma_o^2]$
Subsystem's submodel $G^{n_o}[\Sigma_o^{n_o}]$
Observation channel $O_{\Sigma_o^1}$
Observation channel $O_{\Sigma_o^2}$
Observation channel $O_{\Sigma_o^{n_o}}$
Modular DES $G^1 \parallel G^2 \parallel \cdots \parallel G^{n_o}$
Local specifications
$(P_1, \mathcal{M}_1)$ for G^1
$(P_2, \mathcal{M}_2)$ for G^2
$(P_{n_o}, \mathcal{M}_{n_o})$ for G^{n_o}
$\sigma \in \Sigma$

FIGURE 2. The generic decentralized control architecture setup for meeting different local specifications on different subsystems of a modular DES.

trol admitting nonblocking solution optimality [2], [3], i.e., with equivalence to monolithic, optimal nonblocking control of a specification 'globalizing'[2] every local specification in a logical product. And, in targeting computationally simpler control synthesis, the key idea has been to do away with having to build the whole DES from its original subsystems. To this end are two ways.

Firstly, sufficient conditions of abstraction, structure, and consistency for nonconflictingness, in the sense of nonblockingness of a control solution in which coordination is trivial (i.e., not needed) if the solution then exists, are introduced and studied in the literature, with nonblocking solution optimality preserved (e.g., as in [22]–[27]), compromised (e.g., in favor of enhancing synthesis efficiency as in [28]), or compromisable (e.g., in favor of privacy between subsystems in concurrent synthesis of their controls as in [29]). This line of work may be regarded as originating and extending from the research work [30] that treats the DES as monolithic in theory and disregards the issue of nonblockingness.

However, conflicts do often exist necessitating non-trivial coordination as a means to nonblocking solution optimality. Thus, secondly, a lot more research has focused on complexity mitigation of NP-hard synthesis [6] in both control and coordination for nonblocking solution optimality, to which different sufficient conditions have been studied for the synthesis efficiency they bring to obtaining coordinated, decentralized nonblocking control [25], [31] and a nonblocking generalization to a decentralized multitasking version [32].

Referring to Fig. 2, in either way, these sufficient conditions lead to synthesis of individual local control f^i at site i $(1 \leqslant i \leqslant n_o)$ that is on a subset of the controllable event set Σ_c^i of subsystem G^i, in turn a subset of the system controllable event set Σ_c, and over a submodel $G^i[\Sigma_o^i]$ for specification pair $(P_i, \mathcal{M}_i)$. Ideally a size-reduced subsystem model, the submodel $G^i[\Sigma_o^i]$ contains only events of subset Σ_o^i, locally observable via the observation channel $O_{\Sigma_o^i}$ for subsystem G^i in modular DES $G^1 \parallel G^2 \parallel \cdots \parallel G^{n_o}$ modeled on event set Σ, where operator $\parallel$ denotes synchronous interactions on state transitions between subsystems. And in the latter way, the sufficient conditions lead additionally to more efficient synthesis of coordination C – a dynamic logic function. Coordination C is dynamic because, for nonblockingness [33] or multitasking nonblockingness [11], whether C enables a shared controllable event product-enabled by individual controls is dependent on the state reached over the

[2] In formal languages, 'globalizing' is done by inverse projection [1], [5] with respect to the system closed-language.

synchronization $\|$ of the individually controlled submodels. Note, however, that under conditions in which coordination C is trivial, C is reduced to a logical product fusion rule; this is basic, and is just one of the possible fusion rules for function F of the other architecture depicted in Fig. 1.

Finally, noteworthy are two fundamental special cases for the latter goal, where a modular DES of one subsystem, in essence monolithic, is considered for nonblocking control. One case, called modular control – without [34] and with coordination [35] for nonblockingness – is perhaps the earliest DES control architecture first studied in [34]; it is about decentralized control of two or more global specifications on (therefore the full event set of) the same DES; the research efforts [34], [35] may be regarded as extended to decentralized nonblocking control [26], [36]–[38], modular control with nonblockingness guaranteed without coordination by construction [39], and modular multitasking-nonblocking control [40], of two or more abstracted or local specifications on (therefore essentially different event subsets of) the same modular DES. The other case is about decentralized control [25], [41] of one local specification on (therefore an event subset of) a DES, where coordination is therefore absent.

Against this research background, this paper contributes new transparent characterizations and existence results in decentralized control, in the syntax-based control framework [9], [10] of canonical LTL [7], [8] that extends nonblockingness [2], [3] to marker-progressiveness of control [9], [10], necessarily refined to fair DES's. Limited to conventional nonblockingness of control of the marker state set type for DES's modeled with no regard for event fairness, the current state-of-the-art research on supervisory control in fact does not furnish a logically complete or unified understanding that also entails connecting system fairness and control under readable specification for its marker-progress guarantee; the line of research [9], [10] this paper extends postulates that it is apparently due to not grounding supervisory control research on a more adequate and transparently descriptive supporting formalism that can always match a correct logical control design against a system's actual behavioral dynamics. Towards system control theory refinement and better intuitive natural language understanding, canonical LTL provides this supporting formalism.

In systematically extending the control theory initiated in [9], [10] for specification pair $(P, \mathcal{M})$ to decentralized control in the same canonical LTL framework, the original essence of an observability concept [1, Ch. 6] – the monolithic or centralized special case of and predecessor to co-observability, is first characterized and studied, followed by the co-observability concept that extends, with some natural specification modularity, the decentralized control architecture of Fig. 1. The characterizations are for the *invariance of the past formula* P, categorically an LTL formula of canonical temporal safety. On co-observability, as this paper will present, the new formulation is an LTL characterization that logically refines and structurally extends the concept from the essence of its founding formulation [14], [15] to one modularizing *the unchanging past of* P concomitant with temporal safety, such that the control fusion rule is generalized to some combinatorial function of logical products and summations. The resultant concept is herein called universal co-observability.

This paper will also define and explain some significant special cases derivable from LTL universal co-observability, including its logically strongest product-boundary case, its logically weakest summation-boundary case, and its monolithic case. The first two cases structurally specialize to the respective LTL counterparts of the founding co-observability formulation [15] and its dual [16], [17], while the last is an LTL counterpart of the founding formulation [42] of observability. Then, informally speaking, the overarching main result of this paper has it that, co-observability, whenever implied, is by which feasible[3] decentralized control of a DES exists in general for specification pair $(P, \mathcal{M})$, on the provision that the invariance of the past formula P, naturally modularized, is also marker-controllable or $\mathcal{M}$-controllable [9].

Of perhaps more research and practical interest is the logically strongest product-boundary case of universal co-observability. This case provides the necessity that equates the control architecture of Fig. 1 with that of Fig. 2, which in turn accentuates the former including arbitrarily distributing the system marker set $\mathcal{M}$ among the local sites, under a refinement of the latter architecture to treating the DES as observedly monolithic[4], and with coordination reducible to a fusion rule (of the logical product type), meaning coordination is not needed to ensure marker-progress or $\mathcal{M}$-progress. In manifesting the overlap between the two decentralized control architectures, this important special case adds to and advances, from the canonical LTL perspective, the current state-of-the-art research on nonblocking modular and decentralized control under logical product *without* coordination. This current state-of-the-art includes some condition-dependent synthesis [43], [44] of a given marker-language specification for a modular sublanguage that is co-observable [15] and controllable [2]. This modular sublanguage has any required coordination latently built-in, using suitably enlarged local observable event sets to also model communication receipt of additional events that are locally unobservable but observable elsewhere and thus communicable, that the built-in coordination entails.

For observedly monolithic DES's beyond the product-boundary case, universal co-observability extends the modularity in decentralized control without non-trivial coordination, from a logical product to an arbitrary logical combination that the term 'universal' refers to.

[3] By definition in this paper, supervisory control is a function that always permits uncontrollable events to occur. This control is *feasible* [1] provided its action on every controllable event can always be determined with certainty with partial state feedback from locally observed event evolution of the DES.

[4] Being observedly monolithic means, for each site i, $G^i[\Sigma_o^i] = G[\Sigma_o^i]$, which subsumes the case that $G^i = G$.

On a different logic front, the use of epistemic logic [45]–[47] and one temporal logic version [48] of it (albeit for control specification in formal language, not temporal logic) in related work supports a more human-ascribing reasoning perspective, with anthropomorphic knowledge characterization on the control decision-making side. The research efforts [46], [48] give epistemic interpretations to co-observability in decentralized nonblocking control architectures by [14], [15], [17]. An earlier initiated effort [45] cum a subsequently improved version [47] give the same to co-observability in their local non-binary control and so-called conditioning extensions by [19], and conceivably, according to [47], to that in the more general so-called inferencing extension by [20], called N-inference observability – a variant of co-observability characterized with index $N \geqslant 0$. This general N-inferencing extension [20] is shown in [49] to be realizable in some arborescent architecture. It is also shown, in [50], to be organizable in the so-called multi-decision extension by [21], as a ‘blending’ of $(N + 1)$ control architectures at zero-level inferencing or, equivalently, with no conditioning [17], by which, in principle then, the use of local binary controls suffices, with each local control site that is submodel-unique[5] exercising at least one such local control, and at least one site exercising the maximum of $(N + 1)$ local controls. Notably, the need to extend local controls from binary to non-binary arises only with the optional use of conditioning [19] – equivalently established as unity-level inferencing in [20], or more generally, non-zero level inferencing [20], [45] as an alternative means of local decision synthesis and subsequent fusion admitted by their co-observability formulations. The multi-decision extension can in fact generalize the inferencing extension to the more modularized form of inference-based multi-decisioning [21, Sec. V-B], and is shown to be the most general among comparable language-based architectures studied to date.

In contrast, towards exposition of system structure amenable to decentralized control, the use of canonical LTL in this paper supports a different perspective, of a more human natural language-paraphrasable reasoning, with transparent model characterization on the (conventional) system dynamics side, by which the original and key variants of the co-observability concept [14], [15] in the cited literature above re-emerge in the unified LTL formulation of universal co-observability, for marker-progressive control in the well-motivated refined setting of fair DES’s as reviewed earlier on. It turns out that universal co-observability can in fact be explained as one that structurally specializes to an LTL counterpart of the language-based multi-decision variant [21] of the co-observability concept. Not unexpectedly then, as with language-based multi-decision co-observability [21], the LTL formulation of universal co-observability for decentralized control is invariably linked to the simpler baseline of local binary control decisions, only clearer with each of event enablement or disablement by determination or appropriate default dynamically set that this paper will detail, that are combinable by fusion in combinatorial logic for overall control action. Importantly, the existence theory of decentralized control in canonical LTL can be and is developed in a logic fashion that is clean and independent of any formulation supporting decision-making by inferencing. This is a clear merit, as any alternative LTL formulation that directly involves inferencing would require mirroring the language-based iterative formulation [20] also used in the multi-decision version [21], which is unintuitive and complex. As this paper will make self-evident, the LTL system characterization of universal co-observability, decoupled from inferencing right at the definitional outset, makes the decentralized control theory presented more clearly understandable, potentially paving new informed ways for transparent, decentralized control synthesis in future work. That said, in drawing the parallels with language-based efforts, prospective formal development of decentralized control synthesis methods in LTL might benefit from prying for pertinent insights from those research efforts [19]–[21], [45] on or related to inference observability, and their prior special-case and subsequent developments.

The rest of this paper is organized as follows. Section II presents the relevant technical background and preliminaries. Section III reviews the basic, LTL marker-progressive control theory for fair DES’s [9]. Extending this theory, the main contributions are then presented and discussed in detail in Sections IV and V, and include the following: 1) In LTL syntax, the DES concept characterizations of observability and universal co-observability for a canonical LTL safety formula, and 2) the main results of solution existence for marker-progressive control, namely an extension to monolithic or centralized partial observation, and a successive extension to decentralized control. Section VI concludes this paper.

II. BACKGROUND AND PRELIMINARIES

A. DES MODELING AND LTL

1) DES Model and State Trajectories

The DES model G is a basic transition system that, formally, is a 5-tuple $(\Pi, Q, \Sigma, \delta, \theta)$. Π denotes the finite state variable set that is typed, with the type of an arbitrary state variable $x \in \Pi$ defining the domain $Range(x)$ over which x ranges. Q denotes the state set defined by $Q \overset{\text{def}}{=} \bigotimes_{x \in \Pi} Range(x)$ – the cross product of the ranges of the variables in Π, such that every state $q \in Q$ is unique in terms of domain value-assignmentof the state variables of Π. Σ denotes the finite event set partitioned into the set of controllable events Σ_c and the set of uncontrollable events Σ_u, i.e., $\Sigma = \Sigma_c \cup \Sigma_u$ and $\Sigma_u = \Sigma \backslash \Sigma_c$. $\delta : \Sigma \times Q \to Q$ is a (deterministic) partial state-transition function. θ is the system initial condition – a Boolean valued formula that represents the initial state set $Q_0 \subseteq Q$, according to which $q \in Q_0$ provided $q \in Q$

[5] That the control sites are submodel-unique means that, at different sites x, y $(x \neq y)$ among them, the two respective submodels $G[\Sigma_o^x]$, $G[\Sigma_o^y]$ as depicted in Fig. 1 are then not the same, in the sense that the respective languages locally observed of the DES G at the sites x, y are not equal.

satisfies θ (by its value assignment). Assuming non-trivial DES modeling, $Q_0 \neq \emptyset$ and $\Sigma \neq \emptyset$.

A state trajectory of a DES is a sequence of DES states in temporal concurrence with a string – a sequence of DES events that can be finite or infinite (in length). Formally, an arbitrary string over the event set Σ of DES G is a map $e : \{1, \ldots, k, \ldots, \ldots\} \rightarrow \Sigma$, such that $e \overset{\text{def}}{=} e(1)e(2)\cdots e(k)\cdots$, where $e(k) \in \Sigma$. Then a string e can be generated by DES G provided there exists a state trajectory, formally a map $I : \{0, \ldots, k, \ldots, \ldots\} \rightarrow Q$, that is said to be labeling the string under G's state transition function δ, such that $I \overset{\text{def}}{=} I(0) - I(1) - \cdots - I(k)\cdots$, where $I(k) \in Q$, with 1) $I(0) \in Q_0$ (an initial state), and 2) $I(k) = \delta(e(k), I(k-1))$, for $k \geqslant 1$.

Let $I(k) = q_k \in Q$ ($k \geqslant 0$). Then the k-prefix of I, denoted by $I_{(k)}$, is $q_0 - q_1 - \cdots - q_k$. A state $q \in Q$ is said to be terminal (in G) if $(\forall \sigma \in \Sigma)(\delta(\sigma, q)$ is not defined). A state trajectory 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 $I_{(0)} = I(0) = q_0 \in Q_0$; it labels the empty string ε that is formally defined later. Two state trajectories of DES G, or, respectively, their k-prefixes, are defined to be equal, i.e., the same, if the two have the same sequence of states and label the same string.

Note that, since the founding work of supervisory control [2], [3], the implied premise for DES G has always been that, in *runtime*, one event will occur and transition the DES from a non-terminal state reached into another state.

2) LTL – Syntax and Semantics

LTL [8] is a language of predicate logic augmented with a temporal operator set for writing high-level control specifications for DES G. An arbitrary LTL formula ω can be constructed inductively, as combinations of Boolean connectives *not* ($\overline{}$) and *and* ($\cdot$), quantifier '*there exists*' ($\exists$), and temporal operators, namely future operators *next* ($\bigcirc$) and *until* ($\mathcal{U}$), and past operators *previously* ($\ominus$) and *since* ($\mathcal{S}$), all of which are fundamental, using the following syntax-based grammar:

$$\omega ::= true \mid p_a \mid \overline{\omega} \mid \omega_1 \cdot \omega_2 \mid \mathcal{T}_1(\omega) \mid \omega_1 \mathcal{T}_2\, \omega_2 \mid (\exists x)\, \omega.$$

Now, $true$ denotes *validity*, a propositional constant defined in an abbreviation below; $p_a \in \mathcal{AP}$, where $\mathcal{AP}$ denotes the finite set of atomic propositions expressed by predicates in terms of, over their domains, state variables in Π of DES G and auxiliary variables (defined for system and control); unary operator $\mathcal{T}_1 \in \{\bigcirc, \ominus\}$; binary operator $\mathcal{T}_2 \in \{\mathcal{U}, \mathcal{S}\}$; and x is a state or auxiliary variable in ω.

Given LTL formulas ω_1, ω_2, ω, the following abbreviations stated with *always-equals* ($\equiv$) are used, about which derived connectives *or* ($+$), *implies* ($\rightarrow$) and *equals* ($=$), derived quantifier '*for all*' ($\forall$), and propositional constants *validity* ($true$) and *inconsistency* ($false$) are, respectively, defined: $(\omega_1 + \omega_2) \equiv \overline{(\overline{\omega_1} \cdot \overline{\omega_2})}$, $(\omega_1 \rightarrow \omega_2) \equiv (\overline{\omega_1} + \omega_2)$, $(\omega_1 = \omega_2) \equiv (\omega_1 \rightarrow \omega_2) \cdot (\omega_2 \rightarrow \omega_1)$, $(\forall x)\omega \equiv \overline{(\exists x)\overline{\omega}}$, where x is a variable in ω, $true \equiv \overline{\omega} + \omega$, and $false \equiv \overline{true}$.

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

A past formula has no future operators; a future formula has no past operators; and a non-temporal or state formula has no future or past operators.

To evaluate the truth semantics of an arbitrary LTL formula ω at index $k \geqslant 0$ of an arbitrary state trajectory I of DES G entails the satisfaction relation $\left(\models^{I^{(k)}} \omega\right) \in \{true, false\}$ (read: 'I at index k satisfies ω', or simply 'I satisfies ω' if $k = 0$, since $I^{(0)} \overset{\text{def}}{=} I$). If ω is a state formula, then

$$\models^{I^{(k)}} \omega \text{ iff } \models^{I(k)} \omega \text{ (read: `state } I(k) \text{ satisfies } \omega\text{')}.$$

Where convenient, write 'I at index k' as (I, k) when it is not superscripted to the satisfaction relation symbol $\models$.

In general, evaluating the satisfaction relation proceeds inductively based on standard rules for Boolean connectives and quantifiers, and the satisfaction relation rules for temporal operators. Presented below are the rules for operators $\bigcirc$, $\mathcal{U}$, $\ominus$, $\mathcal{S}$ that are fundamental to LTL, and also the rules for future operator *always* ($\Box$) and past operator *has-always-been* ($\boxminus$) that are syntax-definable using operators $\mathcal{U}$ and $\mathcal{S}$, respectively, but presented to aid in better understanding their important role in control theory development. The rule for operator $\bigcirc$ entails the following event-transition logic to account for a state trajectory I that is finite.

Definition 1 (The σ-Transition Logic): [9, Def. 1] Given $\sigma \in \Sigma$, for an arbitrary state trajectory I of DES G, $I = I(0) - I(1) - \cdots - I(k)\cdots$, the function $\tau : \sigma \rightarrow (I \rightarrow \{true, false\})$ is a system σ-transition logic, defined at index k such that

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

Given LTL formulas ω, ω_1, ω_2, the satisfaction relations are defined [8] as follows:

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

All other temporal operators may be syntax-defined or derived in terms of the fundamental ones via abbreviations as presented in [9], [10] and [8]. Here, only the derived operators of interest, namely *eventually* ($\Diamond$) and *weak previously* ($\circledtilde$), are presented: 7) $\Diamond\omega \equiv \overline{\Box(\overline{\omega})} \equiv true\mathcal{U}\omega$, and 8) $\circledtilde\omega \equiv \overline{\ominus(\overline{\omega})}$. To aid explanation, Abbreviation (8) may be given by the following satisfaction relation:

$$\models^{I^{(k)}} \circledtilde\omega \text{ iff } k = 0 \text{ or } \models^{I^{(k)}} \ominus\omega.$$

Lastly, LTL formulas may be expandable using rules [8, p. 206, p. 219] that recursively define basic operators $\Box$, $\mathcal{U}$, $\boxminus$, $\mathcal{S}$, and their operator derivations. Here, only expansion rules defining the basic operators are given: 9) $\Box\omega \equiv \omega \cdot \bigcirc\Box\omega$, 10) $\omega_1\mathcal{U}\omega_2 \equiv \omega_2 + \omega_1 \cdot \tau \cdot \bigcirc\,(\omega_1\mathcal{U}\omega_2)$, 11) $\boxminus\omega \equiv \omega \cdot \circleddash\boxminus\omega$, and 12) $\omega_1\mathcal{S}\omega_2 \equiv \omega_2 + \omega_1 \cdot \circleddash\,(\omega_1\mathcal{S}\omega_2)$.

State trajectories that confine to actual, possible runtime behavior of DES G are termed legal. Let $\mathcal{I}(G)$ be the legal state trajectory set of DES G. Since only legal DES behavior is of interest, the notion of G-validity of LTL formula ω, denoted by $G \models \omega$, is fundamental. It is defined as follows:

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

Under G-validity, for LTL formulas ω_1, ω_2, $\omega_1 \equiv \omega_2$ denotes $G \models \Box(\omega_1 = \omega_2)$; and for the connective *always-implies* ($\Rightarrow$), $\omega_1 \Rightarrow \omega_2$ denotes $G \models \Box(\omega_1 \rightarrow \omega_2)$. Thus, $(\omega_1 \equiv \omega_2) \equiv (\omega_1 \Rightarrow \omega_2) \cdot (\omega_2 \Rightarrow \omega_1)$. Finally, an LTL formula ω is said to be satisfiable if $\omega \not\equiv false$, i.e., $(\exists I \in \mathcal{I}(G))(\exists k \geqslant 0) \models^{I(k)} \omega$.

B. PROPER STATE FEEDBACK CONTROL

Based on an input state history $I_{(k)}$ – a prefix of an arbitrary state trajectory $I \in \mathcal{I}(G)$, a supervisor or control for DES G specifies whether controllable events are to be enabled or disabled at an arbitrary state $I(k) \in Q$.

Definition 2 (The σ-Definition Logic): [9, Def. 2] 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 $\models^{q} \xi_\sigma$ iff $(\exists q' \in Q)q' = \delta(\sigma, q)$.

Formally then, given an arbitrary $I \in \mathcal{I}(G)$, a supervisor is a control function $f : \Sigma \rightarrow (I \rightarrow \{true, false\})$, defined at index k subject to the Σ_u-completeness constraint that $(\forall \sigma \in \Sigma_u) \models^{I^{(k)}} (f_\sigma = true)$, such that, respectively, $f_\sigma = true$ and $f_\sigma = false$ enables and disables event $\sigma \in \Sigma$ at state $I(k) \in Q$ (of history $I_{(k)}$), if $\models^{I(k)} \xi_\sigma$, i.e., σ is defined at the state $I(k)$. Thus, control f leaves events in the uncontrollable event set Σ_u enabled, and can only disable events in the controllable event set Σ_c, where $\Sigma_c = \Sigma \backslash \Sigma_u$ as defined. An event occurs provided it is not only enabled by control f but also 'activated' by the DES; the role of f is thus only passive [1]. Interconnected with the DES in a closed loop, the control f decides as defined, in response to new value-assignments of state variables fed back by a discrete state change triggered by an enabled event occurrence in the DES. It is thus called state feedback control.

Now, let $\mathcal{I}^{\#}(G) = \{I_{(k)} \mid I \in \mathcal{I}(G)$, finite $k \geqslant 0$, and $I_{(k)} \notin \mathcal{I}(G)\}$, called the legally prefix-admissible set of DES model G, and let $\mathcal{I}^{\circledast}(G) = \mathcal{I}(G) \cup \mathcal{I}^{\#}(G)$, noting that $\mathcal{I}(G) = \mathcal{I}^{\circledast}(G) \backslash \mathcal{I}^{\#}(G)$.

In what follows, in supervising DES G with control f, the resultant closed-loop controlled model, denoted by G^f, is also a DES model but with an arbitrary state of G^f denoting a nonempty and possibly non-unique subset of the state set Q of G. It is described as follows:

1) $\{I_{(0)} \mid I \in \mathcal{I}(G)\} \subseteq \mathcal{I}^{\circledast}(G^f)$, and
2) $(\forall I \in \mathcal{I}(G))\,(\forall k \geqslant 0)(\forall \sigma \in \Sigma)$ if $I_{(k)} \in \mathcal{I}^{\circledast}(G^f)$ and $\models^{I^{(k)}} f_\sigma \cdot \xi_\sigma$, then $I_{(k)} - \delta(\sigma, I(k)) \in \mathcal{I}^{\circledast}(G^f)$.

Additionally, following [1], a standard criterion [1] for supervisory control is adopted, namely, $I \in \mathcal{I}(G^f) \rightarrow I \in \mathcal{I}^{\circledast}(G)$, meaning that the control f should at most be restrictive on DES G. In what follows, an arbitrary control f is said to be proper if $\mathcal{I}(G^f) \neq \varnothing$ and $\mathcal{I}(G^f) \subseteq \mathcal{I}^{\circledast}(G)$.

III. REVIEW OF CONTROL UNDER FULL OBSERVATION

This review section summarizes the basic LTL control theory [9], and only includes the notation and technical formalities required in this paper to extend the theory to partial observation.

Fair DES Model

The DES model G considered is a fair transition system for which the LTL framework [7], [8] is developed. Called a fair model, DES G contains a system subset $\Sigma_{\mathcal{F}}$ of fair events that directs the free system evolution. A fair event is one whose transition occurs at infinitely many states of a DES state trajectory, in which the event is either defined at infinitely many states, in which case the event is called *compassionate* or *strongly fair*, or permanently defined from some state onwards, in which case it is called *just* or *weakly fair*. Events designated as fair are additionally set as uncontrollable for their unperturbable role in DES G, i.e., $\Sigma_{\mathcal{F}} \subseteq \Sigma_u$. Importantly, this setting sufficiently ensures that an arbitrary control f with a nonempty set $\mathcal{I}(G^f)$ is proper, as explained in [9]. These fair events impose the legal conditions that characterize the state trajectory set $\mathcal{I}(G)$ of the fair DES model G.

Control Specification in Canonical LTL

LTL formulas are classified by a complete hierarchy of syntactic canonical classes, with the canonical forms for the classes defined by the restricted future modalities *always* ($\Box$) and *eventually* ($\Diamond$), and their different combinations, applied to past formulas [7]. Of specification interest in supervisory control are two of the classes, namely safety at the base level and response – a kind of progress at a level higher up in the classification hierarchy; their canonical forms have operator $\Box$ and combined operator $\Box\Diamond$ applied to a past formula, respectively. Together, these two classes are useful for expressing a range of control specifications about finishing tasks regularly in the sense of infinite oftenity, without compromising temporal safety. In the interest of control theory development, an LTL formula φ is called an invariant if $\varphi \equiv \boxminus\psi$, where ψ is some past formula; and this ψ is called the kernel of φ if it has no operator $\boxminus$ in its outermost scope. Then, the said control specification of interest, denoted by specification pair $(P, \mathcal{M})$ over DES G, is

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

where P is the kernel of some arbitrary invariant, and $\mathcal{M} = \{M_1, M_2, \dots, M_m\}$ is the system marker set, where each $M_i \in \mathcal{M}$ $(1 \leqslant i \leqslant m)$ is an arbitrary past formula specifying a system marker condition. In their respective forms [7], [8], $\Box P$ is called a canonical LTL safety formula while $\Box\Diamond M_i$ is called a canonical LTL response formula.

The Basic LTL Supervisory Control Problem

In what follows, the basic problem of interest, called the marker-progressive supervisory control problem (MP-SCP), is reviewed. A state feedback $(P, \mathcal{M})$-supervisor f for fair DES G is defined such that $G^f \models \Box\left(P \cdot \prod_{i=1}^{m} \Diamond M_i\right)$. Then the problem is formally stated as follows:

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

Consider an arbitrary kernel ψ of some invariant over DES G. For the system conditions that constitute the LTL concept of $\mathcal{M}$-controllability of $\Box\psi$, refer to [9] for details. Suffice for understanding at this juncture, it is shown in [9] that, provided $\boxminus\psi \Rightarrow \boxminus P$, this $\mathcal{M}$-controllability, which comprises of controllability that is a counterpart of the language version [2], and some $\mathcal{M}$-directingness for which fair events play a role in general, is necessary and sufficient for proper control of temporal safety ensuring constant marker progress that is the objective of the MP-SCP. The invariant-exact solvability condition for the MP-SCP then follows; this condition is $\mathcal{M}$-controllability of $\Box\psi$ with $\boxminus\psi \equiv \boxminus P$, under which a solution that fully realizes specification pair $(P, \mathcal{M})$ exists, as established by the following result that this paper extends. First define $\bigcirc_\sigma(.) \equiv (\tau_\sigma \rightarrow \bigcirc(.))$. Then, for some ψ-locally optimal control f that exists by the control invariance of $\boxminus\psi$, rewrite the Σ_c-part, i.e.,

$$(\forall\sigma \in \Sigma_c)\; G \models \Box\left(\boxminus\psi \rightarrow (f_\sigma = \bigcirc_\sigma(\psi))\right)$$

in the following algebraic form:

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

where '[rel to $(\boxminus\psi, G)$]' reads 'relative to $\boxminus\psi$ over G'. The result may then be stated as follows.

Theorem 1: [9, Thm. 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

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

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

In the context of decentralized control as overviewed in the introduction, the MP-SCP is clearly about centralized or monolithic control under full observation.

IV. CONTROL UNDER PARTIAL OBSERVATION

In this section, the supervisory control of DES G on event set Σ is extended to partial observation. Let the sets $\Sigma_o \subseteq \Sigma$, $\Sigma_{uo} = \Sigma \backslash \Sigma_o$ denote the sets of observable and unobservable events, respectively. This means that, via feedback through a Σ_{uo}-lossy observation channel in closed loop with the DES, an implementation of the supervisor can observe strings of only events in Σ_o.

A. THE PARTIAL OBSERVATION CONTROL PROBLEM

The theory extension to partial observation control in LTL is a new development that requires the following additional theoretical support:

The operator str

Define the empty string ε such that for an arbitrary $q \in Q$ of DES G, $\delta(\varepsilon, q) = q$, and for an arbitrary string s, $s = \varepsilon s = s\varepsilon$. The operator str is then defined inductively as follows:

For an arbitrary $I \in \mathcal{I}(G)$,

$$str\,[I(0)] = \varepsilon.$$

And for $k \geqslant 0$, $\sigma \in \Sigma$ and $I(k+1) = \delta(\sigma, I(k))$,

$$str\left[I_{(k+1)}\right] = str\left[I_{(k)}\right]\sigma.$$

Intuitively, str 'stringifies' the prefix of a state trajectory I, extending in the limit $\lim_{j \to \infty} I_{(j)}$ for a non-terminating I.

The string-based projection O_{Σ_o}

The string-based projection O_{Σ_o} modeling the observation channel of Σ_o may then be defined inductively as follows:

$$O_{\Sigma_o}(\varepsilon) = \varepsilon.$$

$$O_{\Sigma_o}(\sigma) = \begin{cases} \sigma & \text{, if } \sigma \in \Sigma_o \\ \varepsilon & \text{, otherwise.} \end{cases}$$

Given $I \in \mathcal{I}(G)$, for $k \geqslant 0$, $\sigma \in \Sigma$ and $str\left[I_{(k+1)}\right] = s\sigma$,

$$O_{\Sigma_o}(s\sigma) = O_{\Sigma_o}(s)O_{\Sigma_o}(\sigma).$$

Based on str, O_{Σ_o}, the following logic of duplication is defined.

Definition 3 (The Duplication Logic): Given arbitrary LTL formulas ϕ_1, ϕ_2 over DES G, for an arbitrary $I \in \mathcal{I}(G)$ at index k, and an arbitrary $I' \in \mathcal{I}(G)$ at index j with

$$O_{\Sigma_o}\left(str\left[I_{(k)}\right]\right) = O_{\Sigma_o}\left(str\left[I'_{(j)}\right]\right),$$

the function $D : (\Sigma_o, \phi_1, \phi_2) \rightarrow (I \rightarrow \{true, false\})$ is a system (ϕ_1, ϕ_2)-duplication logic (with respect to Σ_o), defined at $I(k) \in Q$ such that $\models^{I^{(k)}} \Big(D_{\Sigma_o}(\phi_1, \phi_2) =$

$$\forall\,(I', j)\,(I'(0) = I(0)) \models^{I'^{(j)}} \phi_1 \rightarrow \phi_2\Big).$$

By this logic, given a state trajectory $I \in \mathcal{I}(G)$ at index k, every j-prefix of an arbitrary state trajectory $I' \in \mathcal{I}(G)$ that shares the same initial state as I and looks like, string-wise under an observation channel, the k-prefix of I, has I' at index j 'duplicating' (the satisfaction of) the same formula $\phi_1 \rightarrow \psi_2$ along I at index k. As this paper will show, this logic is fundamental in defining concepts related to observation.

Finally, the observed DES submodel of DES G is a basic transition system $G[\Sigma_o] \overset{\text{def}}{=} (\Pi, Q_o, \Sigma_o, \delta_o, \theta_o)$, where $Q_o \subseteq 2^Q$, and Π is as defined on Q. This submodel can be defined by adapting the generic construction procedure for a system observer [5, Ch. 2, Sec. 2.5.2], also slightly generalized to handle the initial state set characterized by the initial condition θ of DES G. Introduced here for completeness' sake, this submodel is explicitly required only later in a further extension to decentralized control.

Now, under partial observation, a DES supervisor f needs to be not only proper; it has to be feasible, as defined below.

Definition 4 (Feasibility of Supervisor): A state feedback supervisor f for DES G is said to be feasible if

FM1) f is proper, and
FM2) $(\forall \sigma \in \Sigma_c)\ G \models \Box\,(\tau_\sigma \rightarrow (f_\sigma \rightarrow D_{\Sigma_o}(\tau_\sigma, f_\sigma)))$.

To help explain supervisor feasibility and other concepts more clearly, the following lemma is provided.

Lemma 1: Consider arbitrary LTL formulas $\psi_1, \psi_2, \phi_1, \phi_2$ over DES G. Then,

$$G \models \Box(\psi_1 \rightarrow (\psi_2 \rightarrow D_{\Sigma_o}(\phi_1, \phi_2))) \quad \text{iff } G \models \Box\left(\phi_1 \rightarrow \left(\overline{\phi_2} \rightarrow D_{\Sigma_o}\left(\psi_1, \overline{\psi_2}\right)\right)\right).$$

Proof: Follows by Definition 3 of D_{Σ_o} and LTL reasoning at the level of satisfaction relation for G-validity. ■

By Lemma 1 (with $\psi_1 \equiv \tau_\sigma, \psi_2 \equiv f_\sigma, \phi_1 \equiv \tau_\sigma, \phi_2 \equiv f_\sigma$), Condition FM2 may be equivalently rewritten as

FM2•E) $(\forall \sigma \in \Sigma_c)\ G \models \Box\left(\tau_\sigma \rightarrow \left(\overline{f_\sigma} \rightarrow D_{\Sigma_o}(\tau_\sigma, \overline{f_\sigma})\right)\right)$.

Thus, a feasible supervisor is a proper one for DES G, whose control actions (on a controllable event) are the same for different, control-admissible DES state trajectory histories that are on the same event string observation giving partial state feedback. Being the same, such actions on each controllable event can, in effect, be determined with certainty in one control action by the supervisor, via an implementation that responds to partial state feedback upon each observable event occurrence in the DES.

Of interest then is the marker-progressive supervisory control and observation problem (MP-SCOP) that extends the MP-SCP. It may be stated as follows:

MP-SCOP: Find a feasible $(P, \mathcal{M})$-supervisor f for fair DES G (under partial observation).

In what follows, a new observability concept formulation of temporal safety is first presented and studied. The invariant-exact solvability conditions for the MP-SCOP are then shown to be $\mathcal{M}$-controllability and observability, under which a solution that fully realizes specification pair $(P, \mathcal{M})$ exists, as stated in Theorem 2 – a main result of this paper.

B. LTL CHARACTERIZATIONS FOR OBSERVABILITY

The concept of observation consistency for DES G is first formulated and discussed.

Definition 5 (Observation Consistency): Consider an arbitrary invariant φ, with ψ as its kernel, over DES G (thus, $\varphi \equiv \boxminus\psi$). Then invariant φ is said to be Σ_o-observation consistent (with respect to G) if

$(\forall \sigma \in \Sigma)\ G \models \Box\,(\boxminus\psi \cdot \tau_\sigma \rightarrow (\bigcirc\psi \rightarrow D_{\Sigma_o}(\boxminus\psi, \bigcirc_\sigma(\psi))))$.

By Lemma 1 (with $\psi_1 \equiv \boxminus\psi{\cdot}\tau_\sigma, \psi_2 \equiv \bigcirc\psi, \phi_1 \equiv \boxminus\psi{\cdot}\tau_\sigma$, $\phi_2 \equiv \bigcirc\psi$), the condition of Definition 5 is equivalent to

$(\forall \sigma \in \Sigma)\ G \models \Box\left(\boxminus\psi \cdot \tau_\sigma \rightarrow \left(\bigcirc\overline{\psi} \rightarrow D_{\Sigma_o}\left(\boxminus\psi, \bigcirc_\sigma\left(\overline{\psi}\right)\right)\right)\right)$.

Thus, this concept means that, in the presence of uncertainty due to partial observation – as to which state the DES may be in while a given invariant is true, there is certainty as to whether or not the invariant can continue to be true at the next state upon an event transition.

In the remark below, a semantics interpretation of the concept of Definition 5 exposes its origins in the founding language concept of observability [42].

Remark 1 (Observation Consistency): In semantically interpreting Definition 5, invariant φ, with ψ as its kernel, is said to be Σ_o-observation consistent (with respect to G) if, for all $(I, k), (I', j)$, where $I, I' \in \mathcal{I}(G)$ and $I(0) = I'(0)$ such that $\models^{I^{(k)}} \boxminus\psi$ and $\models^{I'^{(j)}} \boxminus\psi$, and for all $\sigma \in \Sigma$, if $O_{\Sigma_o}\left(str\left[I_{(k)}\right]\right) = O_{\Sigma_o}\left(str\left[I'_{(j)}\right]\right)$,
$\models^{I^{(k)}} \tau_\sigma \cdot \bigcirc\boxminus\psi$, and $\models^{I'^{(j)}} \tau_\sigma$,
then $\models^{I'^{(j)}} \tau_\sigma \cdot \bigcirc\boxminus\psi$.

Assuming reader familiarity, one gets the (prefix) closed language version of observability formulated in the rudimentary form stated in [51, Def. 7], by logically replacing, in the interpretation above, the semantics-based components with their language-based component counterparts. ■

An invariant φ and its kernel ψ over DES G are said to be initially satisfied if $G \models \psi$. The concept definition for observability follows.

Definition 6 (Observability): Consider the kernel P of an arbitrary invariant over DES G. Then $\Box P$ is said to be observable (with respect to G) if invariant $\boxminus P$ is initially satisfied and Σ_o-observation consistent.

Unlike language observability [51, Def. 7] which is a characterization counterpart of LTL observation consistency for an invariant $\boxminus P$ that admits kernel $P \equiv false$ as (trivially) observation consistent, the LTL version of observability in Definition 6 is for $\Box P$, and necessarily entails initial P-satisfaction to be a sensible standalone concept.

C. PROBLEM SOLVABILITY: MAIN RESULT

The main result on the necessary and sufficient conditions for an ideal solution to the MP-SCOP is presented after the following lemma supporting its proof.

Lemma 2: Consider the kernel P of an arbitrary invariant over DES G. Then, with f as state feedback supervisor,
$\exists f (\forall \sigma \in \Sigma_c)\ G \models \Box\,(\tau_\sigma \rightarrow (f_\sigma \rightarrow D_{\Sigma_o}(\tau_\sigma, f_\sigma)))$ iff
$\exists f (\forall \sigma \in \Sigma_c)\ G \models \Box\,(\boxminus P \cdot \tau_\sigma \rightarrow (f_\sigma \rightarrow D_{\Sigma_o}(\boxminus P \cdot \tau_\sigma, f_\sigma)))$.

Proof: By specializing Lemma 3 (presented ahead in Section V-C) to centralized control, and applying Lemma 1 accordingly. ■

Theorem 2: Consider the kernel P of an arbitrary invariant over fair DES G with system marker set $\mathcal{M}$. Then there exists a feasible $(P, \mathcal{M})$-supervisor f for G, such that

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

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

Proof: Given the kernel P of an arbitrary invariant over fair DES G with system marker set $\mathcal{M}$, it follows that:

$\square P$ is $\mathcal{M}$-controllable and observable

iff there exists a proper $(P, \mathcal{M})$-supervisor f for G, such that $(\forall \sigma \in \Sigma_c)\ f_\sigma = \bigcirc_\sigma(P)$ [rel to $(\boxminus P, G)$], and $(\forall \sigma \in \Sigma)$ $G \models P \cdot \square\,(\boxminus P \cdot \tau_\sigma \rightarrow (\bigcirc P \rightarrow D_{\Sigma_o}(\boxminus P, \bigcirc_\sigma(P))))$
[by Theorem 1 and Definition 6]

iff there exists a proper $(P, \mathcal{M})$-supervisor f for G, such that $(\forall \sigma \in \Sigma_c)\ f_\sigma = \bigcirc_\sigma(P)$ [rel to $(\boxminus P, G)$], and $(\forall \sigma \in \Sigma_c)$ $G \models \square\,(\boxminus P \cdot \tau_\sigma \rightarrow (\bigcirc_\sigma(P) \rightarrow D_{\Sigma_o}(\boxminus P, \bigcirc_\sigma(P))))$, and $(\forall \sigma \in \Sigma_u)\ G \models \square\,(\boxminus P \cdot \tau_\sigma \rightarrow$
$(\bigcirc P \rightarrow D_{\Sigma_o}(\boxminus P, \bigcirc_\sigma(P))))$
[by LTL reasoning and the fact that $\Sigma = \Sigma_c \cup \Sigma_u$ and $\Sigma_u = \Sigma \backslash \Sigma_c$]

iff there exists a proper $(P, \mathcal{M})$-supervisor f for G, such that $(\forall \sigma \in \Sigma_c)\ f_\sigma = \bigcirc_\sigma(P)$ [rel to $(\boxminus P, G)$], and $(\forall \sigma \in \Sigma_c)$ $G \models \square\,(\boxminus P \cdot \tau_\sigma \rightarrow (\bigcirc_\sigma(P) \rightarrow D_{\Sigma_o}(\boxminus P, \bigcirc_\sigma(P))))$
[$\because$ $(\forall \sigma \in \Sigma_u)\ D_{\Sigma_o}(\boxminus P, \bigcirc_\sigma(P)) \equiv true,$ provided $\boxminus P$ is Σ_u-invariant, i.e., $(\forall \sigma \in \Sigma_u)\ \boxminus P \Rightarrow \bigcirc_\sigma(P)$, which it is since $\square P$ is controllable, i.e., $\boxminus P$ is initially satisfied and Σ_u-invariant [9]]

iff there exists a proper $(P, \mathcal{M})$-supervisor f for G, such that $(\forall \sigma \in \Sigma_c)\ f_\sigma = \bigcirc_\sigma(P)$ [rel to $(\boxminus P, G)$], and $(\forall \sigma \in \Sigma_c)$ $G \models \square\,(\boxminus P \cdot \tau_\sigma \rightarrow (f_\sigma \rightarrow D_{\Sigma_o}(\boxminus P \cdot \tau_\sigma, f_\sigma)))$
[by LTL reasoning, and logic substitution of $f_\sigma = \bigcirc_\sigma(P)$ [rel to $(\boxminus P, G)$] for all $\sigma \in \Sigma_c$]

iff there exists a proper $(P, \mathcal{M})$-supervisor f for G, such that $(\forall \sigma \in \Sigma_c)\ f_\sigma = \bigcirc_\sigma(P)$ [rel to $(\boxminus P, G)$], and $(\forall \sigma \in \Sigma_c)$ $G \models \square\,(\tau_\sigma \rightarrow (f_\sigma \rightarrow D_{\Sigma_o}(\tau_\sigma, f_\sigma)))$
[by Lemma 2]

iff there exists a feasible $(P, \mathcal{M})$-supervisor f for G, such that $(\forall \sigma \in \Sigma_c)\ f_\sigma = \bigcirc_\sigma(P)$ [rel to $(\boxminus P, G)$]
[by Definition 4].

Hence the theorem. ∎

Remark 2: As used in the proof of Theorem 2,
$\boxminus P$ is Σ_u-invariant iff
$$(\forall \sigma \in \Sigma_u)\ D_{\Sigma_o}(\boxminus P, \bigcirc_\sigma(P)) \equiv true.$$
Because of this, there is an 'overlap' between Σ_u-invariance and Σ_o-observation consistency, which are the key constituent condition for controllability [9] and observability, respectively. Thus, the observation consistency condition in observability for supervisory control that exists (according to Theorem 2) may be restated by only that part of the condition for controllable events, with the G-valid part for uncontrollable events dropped as it is implied and hence covered by Σ_u-invariance. ∎

V. CONTROL UNDER DECENTRALIZATION

In this section, the supervisory control of partially observed DES G on event set Σ is extended to decentralized control, as introduced and depicted in Fig. 1.

Let $\mathcal{N} = \{1, 2, \ldots, n_o\}$ be the index set of control sites for DES G on event set Σ. Each local control site $i \in \mathcal{N}$ is associated with local observable event set $\Sigma_o^i \subseteq \Sigma$, and local controllable event set $\Sigma_c^i \subseteq \Sigma$ such that the system controllable event set $\Sigma_c = \bigcup_{i=1}^{n_o} \Sigma_c^i$, and the system uncontrollable event set $\Sigma_u = \Sigma \backslash \Sigma_c$. For an arbitrary event $\sigma \in \Sigma_c$, let $\mathcal{N}_\sigma = \{i \in \mathcal{N} \mid \sigma \in \Sigma_c^i\}$ be the index subset of control sites that event σ is under the jurisdiction of, for which, equivalently then to the system controllable event set distribution, $\mathcal{N}_\sigma \neq \varnothing$; and let an arbitrary site $i \in \mathcal{N}$ have some $\sigma \in \Sigma_c$ under its jurisdiction, i.e., $\Sigma_c^i \neq \varnothing$. Then, that $\mathcal{N}_\sigma \neq \varnothing$ for an arbitrary event $\sigma \in \Sigma_c$ and $\Sigma_c^i \neq \varnothing$ for an arbitrary site $i \in \mathcal{N}$ is the same as $\mathcal{N} = \bigcup_{\sigma \in \Sigma_c} \mathcal{N}_\sigma$.

A. THE DECENTRALIZED CONTROL PROBLEM

Extend accordingly, the foregoing single-site notation for centralized control under partial observation to multi-site for decentralized control. Then, for an arbitrary $I \in \mathcal{I}(G)$ at index k, such that $I^i \in \mathcal{I}(G[\Sigma_o^i])$ at index k_i is defined with
$$I(0) \in I^i(0) \text{ and } str\left[I^i_{(k_i)}\right] = O_{\Sigma_o^i}\left(str\left[I_{(k)}\right]\right),$$
let the $\mathcal{N}$-decentralized state feedback supervisor $f : \Sigma \rightarrow (I \rightarrow \{true, false\})$ be defined at index k as
$$\models^{I^{(k)}} f_\sigma = \begin{cases} F_\sigma\left(\left\{\models^{I^{i(k_i)}} f_\sigma^i\right\}_{i \in \mathcal{N}}\right), & \text{if } \sigma \in \Sigma_c \\ true, & \text{if } \sigma \in \Sigma_u, \end{cases}$$
where $F_\sigma(f_\sigma^1, \ldots, f_\sigma^{n_o})$ is a logic function implementing some general fusion rule F over the control decision f^i of every site $i \in \mathcal{N} \supseteq \mathcal{N}_\sigma$ on controllable event σ, such that
$$\models^{I^{i(k_i)}} f_\sigma^i \in \{\text{det}, \text{dft}\},$$
where dft $\in \{true, false\}$ is a default control logic exercisable at site $i \in \mathcal{N}$ whenever there is no certainty of control under Σ_o^i-observation, to 'pass the buck' [15] on event σ under the fusion rule F, and det $\in \{true, false\}$ is the determinable control logic, such that for $\models^{I^{(k)}} \tau_\sigma \rightarrow (f_\sigma^i = \text{det})$,
$$\models^{I^{i(k_i)}} f_\sigma^i = \text{det}$$
if it is the case that

- $i \in \mathcal{N}_\sigma$, and
- for all $I' \in \mathcal{I}(G)$ at index j with
$$O_{\Sigma_o^i}\left(str\left[I_{(k)}\right]\right) = O_{\Sigma_o^i}\left(str\left[I'_{(j)}\right]\right),$$
$$\models^{I'^{(j)}} \tau_\sigma \rightarrow \left(f_\sigma^i = \text{det}\right);$$

otherwise,
$$\models^{I^{i(k_i)}} f_\sigma^i = \text{dft}.$$

Note that an event $\sigma \in \Sigma_c$ is not under the jurisdiction of site $i \in \mathcal{N}$ if $i \in \mathcal{N} \backslash \mathcal{N}_\sigma$, and f_σ^i in this case is called a *phantom* control, which is one that always passes the buck on σ.

Two fundamental special cases of the general fusion rule F are of interest: For an arbitrary $\sigma \in \Sigma_c$, F is said to be a product fusion rule if

$$\models^{I^{(k)}} F_\sigma = \left(\prod_{i \in \mathcal{N}} \models^{I^{i(k_i)}} f_\sigma^i \right), \tag{1}$$

and a summation fusion rule if

$$\models^{I^{(k)}} F_\sigma = \left(\sum_{i \in \mathcal{N}} \models^{I^{i(k_i)}} f_\sigma^i \right). \tag{2}$$

The default setting, i.e., the assignment of $true$ or $false$ to dft as the means to pass the buck in event control, is required provided $|\mathcal{N}| \geqslant 2$ (i.e., $n_o \geqslant 2$). For product fusion, the default setting has to be that dft $\equiv true$, and the fusion rule is said to be default permissive; for summation fusion, the default setting has to be that dft $\equiv false$, and the fusion rule is said to be default anti-permissive. Unlike these two rules where the default setting is static, the setting for the general fusion rule may not be.

Now, consider the general fusion rule F of the controllable event set Σ_c; every component F_σ, $\sigma \in \Sigma_c$, is then expressible as a combinatorial logic function in the same sum-of-product (SoP) form, as follows. Let Λ be the (nonempty) index set for the constituent SoP terms of $F_\sigma(f_\sigma^1, \ldots, f_\sigma^{n_o})$ in minimized form. Then, for an arbitrary $\sigma \in \Sigma_c$ and $j \in \Lambda$, let $\mathcal{N}^j \subseteq \mathcal{N}$ denote the (nonempty) subset of indices of all the local control sites (that appear) in the j-th SoP term $f_\sigma^{\bullet j}$ of F_σ, with $\bigcup_{j \in \Lambda} \mathcal{N}^j = \mathcal{N}$; and $\mathcal{N}_\sigma^j \subseteq \mathcal{N}^j$ denote the index subset, used in $f_\sigma^{\bullet j}$, identifying the control sites that the event σ is under the jurisdiction of, i.e., $\mathcal{N}_\sigma^j = \mathcal{N}^j \cap \mathcal{N}_\sigma$. Then, for an arbitrary $\sigma \in \Sigma_c$, F is said to be the general or universal fusion rule, if

$$\models^{I^{(k)}} F_\sigma = \left(\sum_{j \in \Lambda} f_\sigma^{\bullet j} \right), \tag{3}$$

where, for an arbitrary $j \in \Lambda$,

$$\models^{I^{(k)}} f_\sigma^{\bullet j} = \left(\prod_{i \in \mathcal{N}^j} \models^{I^{i(k_i)}} f_\sigma^i \right).$$

For this fusion rule, the default setting across SoP terms, required provided $|\mathcal{N}| \geqslant 2$, is in general dynamic. Let dft $\equiv$ dft$^j \in \{true, false\}$ be a default control logic exercisable at site $i \in \mathcal{N}^j \supseteq \mathcal{N}_\sigma^j$ for the j-th SoP term $f_\sigma^{\bullet j}$ of F_σ. Then the setting has to be that, with respect to each SoP term $f_\sigma^{\bullet j}$, $j \in \Lambda$, along an arbitrary $I \in \mathcal{I}(G)$ at index k:

$$\text{If } |\Lambda| = 1 \text{ or } |\mathcal{N}^j| > 1 \text{ and } (\exists i \in \mathcal{N}^j) \models^{I^{i(k_i)}} f_\sigma^i = \text{det}, \\ \text{then dft}^j \equiv true, \text{ else dft}^j \equiv false. \tag{4}$$

Under this default setting, equivalently for an arbitrary $j \in \Lambda$ of fusion rule (3),

$$\models^{I^{(k)}} f_\sigma^{\bullet j} = \begin{cases} \left(\prod_{i \in \mathcal{N}_\sigma^j} \models^{I^{i(k_i)}} f_\sigma^i \right) & \text{, if } \mathcal{N}_\sigma^j \neq \varnothing \\ false & \text{, otherwise.} \end{cases}$$

In essence, this syntactically removes every default-assigned phantom control f_σ^i, i.e., $i \in \mathcal{N} \backslash \mathcal{N}_\sigma$, simplifying F_σ. The general fusion rule F is said to be default open.

Following, the feasibility of monolithic supervisor, as in Definition 4, can be generalized to that of $\mathcal{N}$-decentralized control under the formulated universal fusion rule, as in the following. Define a logic variable $\lambda^\bullet$ that indicates whether or not an $\mathcal{N}$-decentralized control f is under product fusion rule:

$$\lambda^\bullet \equiv true \text{ iff } |\Lambda| = 1. \tag{5}$$

Characteristically then, the $\mathcal{N}$-decentralized supervisor f defined under the default open, universal fusion rule (3) is said to be feasible, if

UDF1) f is proper, and

UDF2) $(\forall \sigma \in \Sigma_c)$

$$G \models \Box \Big(\tau_\sigma \rightarrow (\exists j \in \Lambda)(\exists i \in \mathcal{N}_\sigma^j) \\ \big(f_\sigma \rightarrow D_{\Sigma_o^i}(\overline{\lambda^\bullet} \cdot \tau_\sigma, f_\sigma^{\bullet j} \cdot f_\sigma^i)\big) \cdot \\ \Big(\overline{f_\sigma} \rightarrow D_{\Sigma_o^i} \left(\lambda^\bullet \cdot \tau_\sigma, \overline{f_\sigma^i} \right) \Big) \Big).$$

Note that $\lambda^\bullet \equiv true$ means $|\Lambda| = 1$ and $(\forall \sigma \in \Sigma_c)$ $f_\sigma \equiv f_\sigma^{\bullet j}$ for $j \in \Lambda$, with $\mathcal{N}^j = \mathcal{N}$. Intuitively then, the defined supervisor f is said to be feasible if it is proper and, for an arbitrary $\sigma \in \Sigma_c$, there always exists a site $i \in \mathcal{N}_\sigma^j$ for some $j \in \Lambda$, such that, if the fused control action f_σ is $true$, i.e., enable [respectively, $false$, i.e., disable], it is, under an appropriate default setting for SoP term $f_\sigma^{\bullet j}$ of f_σ, effectively the result of a local control decision f_σ^i; under the fusion rule that is not product [respectively, product], this result is obtainable at site i by determination, and (impliedly) by default setting or determination if the fused control action f_σ is otherwise.

In the special case of feasibility of supervisor under the default anti-permissive, summation fusion rule (2), where $|\Lambda| = |\mathcal{N}|$ and for all $j \in \Lambda$, $|\mathcal{N}^j| = 1$, it follows that in the non-trivial summation case of $|\Lambda| > 1$, i.e., $\lambda^\bullet \equiv false$, Condition UDF2 reduces to the following condition:

SDF2) $(\forall \sigma \in \Sigma_c)\ G \models \Box \left(\tau_\sigma \rightarrow (\exists i \in \mathcal{N}_\sigma) \right.$
$\left. \left(f_\sigma \rightarrow D_{\Sigma_o^i} \left(\tau_\sigma, f_\sigma^i \right) \right) \right)$.

In the special case of feasibility of supervisor under the default permissive, product fusion rule (1), i.e., $\lambda^\bullet \equiv true$, where $|\Lambda| = 1$, Condition UDF2 reduces to the following condition:

PDF2) $(\forall \sigma \in \Sigma_c)\ G \models \Box \Big(\tau_\sigma \rightarrow (\exists i \in \mathcal{N}_\sigma)$
$\Big(\overline{f_\sigma} \rightarrow D_{\Sigma_o^i} \left(\tau_\sigma, \overline{f_\sigma^i} \right) \Big) \Big)$.

Consider the former special case (2). Relaxing to full observation at every site (i.e., for all $i \in \mathcal{N}$, $\Sigma_o^i = \Sigma$), this feasible $\mathcal{N}$-decentralized supervisor f reduces to a proper supervisor that is said to be summation $\mathcal{N}$-modularized. In this case, Condition SDF2 reduces to the following condition:

SDF2-M) $(\forall \sigma \in \Sigma_c)\, G \models \Box\left(\tau_\sigma \rightarrow (\exists i \in \mathcal{N}_\sigma)\left(f_\sigma \rightarrow f_\sigma^i\right)\right).$

And relaxing further to full control jurisdiction at every site (i.e., for all $i \in \mathcal{N}$, $\Sigma_c^i = \Sigma_c$), it follows that $\mathcal{N}_\sigma = \mathcal{N}$, and f thus reduces further to the case of a proper, summation $\mathcal{N}$-modular supervisor, which is the one with trivially valid Condition SDF2-M.

Consider the latter special case (1). Relaxing to full observation at every site (i.e., for all $i \in \mathcal{N}$, $\Sigma_o^i = \Sigma$), this feasible $\mathcal{N}$-decentralized supervisor f reduces to a proper supervisor that is said to be product $\mathcal{N}$-modularized. In this case, Condition PDF2 reduces to the following condition:

PDF2-M) $(\forall \sigma \in \Sigma_c)\, G \models \Box\left(\tau_\sigma \rightarrow (\exists i \in \mathcal{N}_\sigma)\left(\overline{f_\sigma} \rightarrow \overline{f_\sigma^i}\right)\right).$

And relaxing further to full control jurisdiction at every site (i.e., for all $i \in \mathcal{N}$, $\Sigma_c^i = \Sigma_c$), it follows that $\mathcal{N}_\sigma = \mathcal{N}$, and f thus reduces further to the conventional case of a proper, product $\mathcal{N}$-modular supervisor, which is the one with trivially valid Condition PDF2-M, and is thus not only the earliest but also practically the simplest DES control architecture first studied in [34].

By the cases above, clearly, feasible $\mathcal{N}$-decentralized control, as formulated, is general. Of interest then is the marker-progressive decentralized supervisory control problem (MP-DSCP) that extends the MP-SCOP. It may be stated as follows:

MP-DSCP: Find a feasible $\mathcal{N}$-decentralized $(P, \mathcal{M})$-supervisor f for fair DES G (under local partial observation).

In what follows, a new concept formulation of general co-observability of temporal safety, called universal co-observability, is studied as a culmination of two special cases, with one at each boundary end of its logical spectrum. The invariant-exact solvability conditions for the MP-DSCP are then shown to be $\mathcal{M}$-controllability and universal observability, under which a solution that fully realizes specification pair $(P, \mathcal{M})$ exists, as stated in Theorem 3 – the overarching main result of this paper. Two modular optimality principles for the case of full observation, both specializations of the two boundary cases of universal co-observability, are also stated; these principles subsume temporal-safety specification modularity in either the strongest product or weakest summation form for the set $\mathcal{N}$. One may then readily write down, for different control architectural settings, the different control existence results, refining or specializing Theorem 3 to the successively specialized definitions of $\mathcal{N}$-decentralized supervisor feasibility given earlier, with each matching the $\mathcal{M}$-controllability and a specialized condition of universal co-observability for $\Box P$. The relations among these results are depicted in Fig. 3.

B. LTL CHARACTERIZATIONS FOR CO-OBSERVABILITY

Co-observability of the logical product and summation types for DES G are first formulated and discussed.

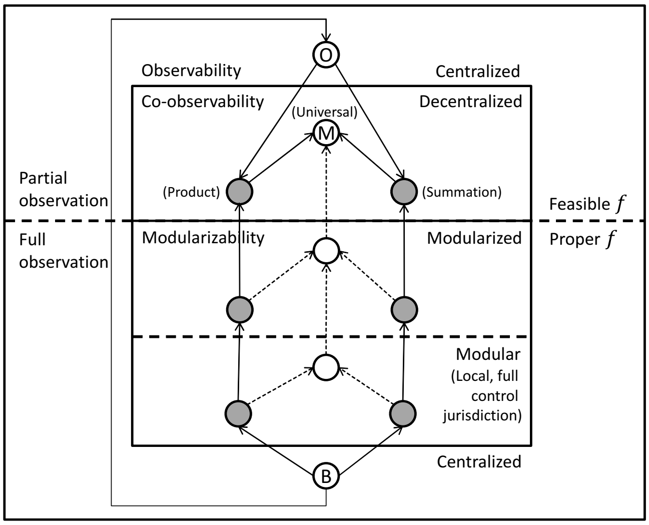


FIGURE 3. A results-relation map: Each node depicts a different control existence result for a horizontal architecture (centralized, modular, modularized, or decentralized) under a partitioning of the system event set for control and observation. The directed edge is read as 'is a special case of'. The nodes Ⓑ, Ⓞ, Ⓜ depict the driver Theorems 1 [9, Thm. 3], 2, and 3, respectively. Other than $\mathcal{M}$-controllability, the additional concepts for the results depicted by white, unlettered nodes are not explicitly named in this paper; each such result can also be readily specialized from that depicted by its node's successor, and stated as would be for the results depicted by its node's grey predecessors, except with the adjective 'product' or 'summation' replaced with 'universal'.

Definition 7 (Product Co-observability): Consider an arbitrary kernel P of some invariant over DES G. At local control site $i \in \mathcal{N} = \{1, 2, \ldots, n_o\}$, let local P_i be the kernel of some invariant over G. Then $\Box P$ is said to be product co-observable with respect to G and $(P_i, \Sigma_o^i, \Sigma_c^i)$ for all $i \in \mathcal{N}$, if

KO1) $G \models P \cdot \Box\left(\boxminus P = \prod_{i=1}^{n_o} \boxminus P_i\right)$, and

KO2) $(\forall \sigma \in \Sigma_c)$

$$G \models \Box\,(\boxminus P \cdot \tau_\sigma \rightarrow (\exists i \in \mathcal{N}_\sigma)$$
$$(\bigcirc \overline{P} \rightarrow D_{\Sigma_o^i}\left(\boxminus P, \bigcirc_\sigma\left(\overline{P_i}\right)\right)))).$$

Definition 7 carries with it a neat interpretation of product co-observability: Condition KO1 asserts that the unchanging past of initially satisfied kernel P is to be exactly realized by the unchanging past of local kernels P_i's in a logical product. Condition KO2 asserts that if kernel P has been true (i.e., $\boxminus P$ is maintained) and the next state transition is of a controllable event that falsifies P, then the event is of some control site i at which the falsity of its local invariant $\boxminus P_i$ next upon the event transition, and thus resultantly of $\boxminus P$, is of definite certainty under Σ_o^i-observation.

Remark 3: Let $\widehat{\Sigma}_c^i = \Sigma_c^i \backslash \bigcup_{\text{all } j} \Sigma_c^j$, where $j \in \{1, 2, \ldots, n_o\} \backslash \{i\}$ (the subset of Σ_c^i, in which every event cannot be found in the local controllable event set at any other site), and $\check{\Sigma}_c = \Sigma_c \backslash \bigcup_{i=1}^{n_o} \widehat{\Sigma}_c^i$ (the subset of events controllable at more than one site). Note, then, that Condition KO2 of Definition 7 can be decomposed equivalently into Conditions KO2a and KO2b, stated as follows:

KO2a) $(\forall i)\left(\forall \sigma \in \hat{\Sigma}_c^i\right)$
$$G \models \Box\left(\boxminus P \cdot \tau_\sigma \rightarrow \left(\bigcirc \overline{P} \rightarrow D_{\Sigma_o^i}\left(\boxminus P, \bigcirc_\sigma\left(\overline{P_i}\right)\right)\right)\right), \text{ and}$$

KO2b) $\left(\forall \sigma \in \check{\Sigma}_c\right)$
$$G \models \Box\left(\boxminus P \cdot \tau_\sigma \rightarrow (\exists i \in \mathcal{N}_\sigma)\left(\bigcirc \overline{P} \rightarrow D_{\Sigma_o^i}\left(\boxminus P, \bigcirc_\sigma\left(\overline{P_i}\right)\right)\right)\right).$$

This aspect emulates formulating the original language version of co-observability [15]. The second condition for co-observability of the summation and universal type, defined in succession next, can also be decomposed similarly. ■

Definition 8 (Summation Co-observability): Consider an arbitrary kernel P of some invariant over DES G. At local control site $i \in \mathcal{N} = \{1, 2, \ldots, n_o\}$, let local P_i be the kernel of some invariant over G. Then $\Box P$ is said to be summation co-observable with respect to G and $(P_i, \Sigma_o^i, \Sigma_c^i)$ for all $i \in \mathcal{N}$, if

KO+1) $$G \models P \cdot \Box\left(\boxminus P = \boxminus\left(\sum_{i=1}^{n_o} \boxminus P_i\right)\right),$$

KO+2) $(\forall \sigma \in \Sigma_c)$
$$G \models \Box\left(\boxminus P \cdot \tau_\sigma \rightarrow (\exists i \in \mathcal{N}_\sigma)\left(\bigcirc P \rightarrow D_{\Sigma_o^i}\left(\boxminus P, \bigcirc_\sigma\left(\boxminus P_i\right)\right)\right)\right).$$

Definition 8 carries with it a neat interpretation of summation co-observability in similar paraphrasing style as that of Definition 7.

Remark 4: Noted are two special cases for each of Definition 7 (product co-observability) and Definition 8 (summation co-observability), with one special case being the same:

1) Setting $P_i \equiv P$ for all $i \in \mathcal{N} = \{1, 2, \ldots, n_o\}$ reduces co-observability of the product and summation types to special cases, the conditions of which are, respectively, deduced as the following:

   KO•S) $(\forall \sigma \in \Sigma_c)$
   $$G \models P \cdot \Box\left(\boxminus P \cdot \tau_\sigma \rightarrow (\exists i \in \mathcal{N}_\sigma)\left(\bigcirc \overline{P} \rightarrow D_{\Sigma_o^i}\left(\boxminus P, \bigcirc_\sigma\left(\overline{P}\right)\right)\right)\right).$$

   KO+S) $(\forall \sigma \in \Sigma_c)$
   $$G \models P \cdot \Box\left(\boxminus P \cdot \tau_\sigma \rightarrow (\exists i \in \mathcal{N}_\sigma)\left(\bigcirc P \rightarrow D_{\Sigma_o^i}\left(\boxminus P, \bigcirc_\sigma(P)\right)\right)\right).$$

   The former and latter special cases are counterparts of two original language versions, respectively, the founding version [14], [15] referred to in [17] as conjunctive and permissive (C&P) co-observability, and disjunctive and anti-permissive (D&A) co-observability [17]. The 'permissiveness' in being C&P is brought about by default setting of event-enablement, and the 'anti-permissiveness' in being D&A by default setting of event-disablement. Be it of the product or summation type, co-observability in LTL is logically stronger but structurally more general than their special case, and may also be referred to as C&P or D&A, respectively.

2) Setting $n_o = 1$, i.e., considering the centralized case of one control site, reduces each definition, by LTL reasoning and Lemma 1 (with $\psi_1 \equiv \boxminus P \cdot \tau_\sigma$, $\psi_2 \equiv \bigcirc P$, $\phi_1 \equiv \boxminus P \cdot \tau_\sigma$, $\phi_2 \equiv \bigcirc P$), to the same characterization that in essence is observability for supervisory control that exists; in this context, the characterization may be treated as an LTL counterpart of the original language version [42] of observability. ■

Modular optimality principles, that clearly are co-observability specializations under full observation, are now stated and discussed.

Definition 9 (Product $\mathcal{N}$-Modular Optimality Principle): Consider an arbitrary kernel P of some invariant over DES G. At an arbitrary local control site $i \in \mathcal{N} = \{1, 2, \ldots, n_o\}$ with local observable event set $\Sigma_o^i = \Sigma$ (i.e., with full observation at site i), let local P_i be the kernel of some invariant over G. Then $\Box P$ is said to be product-modularizable with respect to G and $(P_i, \Sigma, \Sigma_c^i)$ for all $i \in \mathcal{N}$, if

KO1) $$G \models P \cdot \Box\left(\boxminus P = \prod_{i=1}^{n_o} \boxminus P_i\right), \text{ and}$$

KO2-M) $(\forall \sigma \in \Sigma_c)$
$$G \models \Box\left(\boxminus P \cdot \tau_\sigma \cdot \bigcirc \overline{P} \rightarrow (\exists i \in \mathcal{N}_\sigma) \bigcirc_\sigma\left(\overline{P_i}\right)\right).$$

And $\Box P$ is said to be product modular with respect to G and P_i for all $i \in \mathcal{N}$, if Condition KO1 is true.

Definition 10 (Summation $\mathcal{N}$-Modular Optimality Principle): Consider an arbitrary kernel P of some invariant over DES G. At an arbitrary local control site $i \in \mathcal{N} = \{1, 2, \ldots, n_o\}$ with local observable event set $\Sigma_o^i = \Sigma$ (i.e., with full observation at site i), let local P_i be the kernel of some invariant over G. Then $\Box P$ is said to be summation-modularizable with respect to G and $(P_i, \Sigma, \Sigma_c^i)$ for all $i \in \mathcal{N}$, if

KO+1) $$G \models P \cdot \Box\left(\boxminus P = \boxminus\left(\sum_{i=1}^{n_o} \boxminus P_i\right)\right), \text{ and}$$

KO+2-M) $(\forall \sigma \in \Sigma_c)$
$$G \models \Box\left(\boxminus P \cdot \tau_\sigma \cdot \bigcirc P \rightarrow (\exists i \in \mathcal{N}_\sigma) \bigcirc_\sigma\left(\boxminus P_i\right)\right).$$

And $\Box P$ is said to be summation modular with respect to G and P_i for all $i \in \mathcal{N}$, if Condition KO+1 is true.

Most modular control research, such as those cited in the introduction, focuses on the type product, conjunction, or intersection, with the latter two terms often used interchangeably in the cited language-based research; the research furnishes a modular control solution by a conjunction of individually controllable, marker-language specification components. In contrast, by some specialization depicted in Fig. 3 of Theorem 3 informally described earlier, the modular optimality principle, be it product, summation or universal (unstated), suggests, if satisfied, that in exercising modularizing or modular control, there is no need for the component LTL safety specifications to be individually marker-controllable, as long as the overall combined specification is.

Finally, towards formalizing Theorem 3, the required concept of universal co-observability may be defined and explained as a combinatorial culmination of co-observability of the product and summation types.

Definition 11 (Universal Co-observability): Consider an arbitrary kernel P of some invariant over DES G. At local control site $i \in \mathcal{N} = \{1, 2, \ldots, n_o\}$, let local P_i be the kernel of some invariant over G. Define U^Λ as the logical sum-of-product (SoP) fusion function of local invariants $\boxminus P_1, \ldots, \boxminus P_{n_o}$ combined in (*and* $\cdot$, *or* $+$) only[6], given by

$$U^\Lambda(\boxminus P_1, \ldots, \boxminus P_{n_o}) \equiv \boxminus \left(\sum_{j \in \Lambda} \boxminus P^j \right)$$

in minimized invariant form, where, recalling that $\mathcal{N} = \bigcup_{j \in \Lambda} \mathcal{N}^j$, each fusion-rule associated j-th SoP term $\boxminus P^j$ is defined with $P^j \equiv \prod_{i \in \mathcal{N}^j} P_i$. Then, using logic variable $\lambda^\bullet$ (5), $\Box P$ is said to be Λ-universal co-observable (or U^Λ-co-observable) with respect to G and $(P_i, \Sigma_o^i, \Sigma_c^i)$ for all $i \in \mathcal{N}$, if

UKO1) $G \models P \cdot \Box \left(\boxminus P = U^\Lambda(\boxminus P_1, \ldots, \boxminus P_{n_o}) \right)$, and

UKO2) $(\forall \sigma \in \Sigma_c)$

$$\begin{aligned} G \models \Box \big(\boxminus P \cdot \tau_\sigma \to (\exists j \in \Lambda)(\exists i \in \mathcal{N}_\sigma^j) \\ \left(\bigcirc P \to D_{\Sigma_o^i} \left(\overline{\lambda^\bullet} \cdot \boxminus P, \bigcirc_\sigma \left(\boxminus P^j \right) \right) \right) \\ \cdot \left(\bigcirc \overline{P} \to D_{\Sigma_o^i} \left(\lambda^\bullet \cdot \boxminus P, \bigcirc_\sigma \left(\overline{P_i} \right) \right) \right) \big). \end{aligned}$$

Definition 11 carries with it an interpretation of universal co-observability in similar paraphrasing style as Definition 7's; in particular for Condition UKO2 that has two parts, the part where $\lambda^\bullet \equiv true$ has the same paraphrase as Definition 7's.

Remark 5: With respect to the decentralized setting of $(P_i, \Sigma_o^i, \Sigma_c^i)$ for an arbitrary control site $i \in \mathcal{N}$, two boundary instances, of the local invariant fusion function U^Λ defined in Definition 11 of universal co-observability, are given by $U^\Lambda(.) \equiv \prod_{i \in \mathcal{N}} \boxminus P_i$, where $|\Lambda| = 1$, and

$$U^\Lambda(.) \equiv \sum_{i \in \mathcal{N}} \boxminus P_i, \text{ where } |\Lambda| = |\mathcal{N}|.$$

They are the $\min$ (or strongest) and the $\max$ (or weakest) invariant fusion functions, respectively, by which Definition 7 of product co-observability and Definition 8 of summation co-observability are, respectively, defined, that together culminate in Definition 11, with the former two at each boundary end of the concept's combinatorial spectrum of given local invariants.

C. PROBLEM SOLVABILITY: MAIN RESULT

As before, the main result on the necessary and sufficient conditions for an ideal solution to the MP-DSCP is presented after the following lemma supporting its proof.

Lemma 3: Consider an arbitrary kernel P of some invariant over DES G, and a constant $\lambda^\odot \in \{true, false\}$. Then, with f as $\mathcal{N}$-decentralized state feedback supervisor under the default open, universal fusion rule (3),

$$\begin{aligned} \exists f (\forall \sigma \in \Sigma_c)\, G \models \Box \Big(\tau_\sigma \to (\exists j \in \Lambda)(\exists i \in \mathcal{N}_\sigma^j) \\ (f_\sigma \to D_{\Sigma_o^i}(\overline{\lambda^\odot} \cdot \tau_\sigma, f_\sigma^{\bullet j} \cdot f_\sigma^i)) \cdot \\ \left(\overline{f_\sigma} \to D_{\Sigma_o^i} \left(\lambda^\odot \cdot \tau_\sigma, \overline{f_\sigma^i} \right) \right) \Big) \end{aligned}$$

iff

$$\begin{aligned} \exists f (\forall \sigma \in \Sigma_c)\, G \models \Box \Big(\boxminus P \cdot \tau_\sigma \to (\exists j \in \Lambda)(\exists i \in \mathcal{N}_\sigma^j) \\ (f_\sigma \to D_{\Sigma_o^i}(\overline{\lambda^\odot} \cdot \boxminus P \cdot \tau_\sigma, f_\sigma^{\bullet j} \cdot f_\sigma^i)) \cdot \\ \left(\overline{f_\sigma} \to D_{\Sigma_o^i} \left(\lambda^\odot \cdot \boxminus P \cdot \tau_\sigma, \overline{f_\sigma^i} \right) \right) \Big). \end{aligned}$$

Proof: Consider the given settings for DES G.

(If) By Definition 3 of $D_{\Sigma_o^i}$ – the duplication logic (with respect to Σ_o^i), an equivalent fact refining f that is $\mathcal{N}$-decentralized under the default open, universal fusion rule, and therefore an implication, is that

$$\begin{aligned} &\exists f \, (\forall \sigma \in \Sigma_c) \\ &G \models \Box \Big(\boxminus P \cdot \tau_\sigma \to (\exists j \in \Lambda)(\exists i \in \mathcal{N}_\sigma^j) \\ &\quad \left(f_\sigma \to D_{\Sigma_o^i} \left(\overline{\lambda^\odot} \cdot \boxminus P \cdot \tau_\sigma, f_\sigma^{\bullet j} \cdot f_\sigma^i \right) \right) \\ &\quad \cdot \left(f_\sigma \to D_{\Sigma_o^i} \left(\overline{\lambda^\odot} \cdot \overline{\boxminus P} \cdot \tau_\sigma, f_\sigma^{\bullet j} \cdot f_\sigma^i \right) \right) \\ &\quad \cdot \left(\overline{f_\sigma} \to D_{\Sigma_o^i} \left(\lambda^\odot \cdot \boxminus P \cdot \tau_\sigma, \overline{f_\sigma^i} \right) \right) \\ &\quad \cdot \left(\overline{f_\sigma} \to D_{\Sigma_o^i} \left(\lambda^\odot \cdot \overline{\boxminus P} \cdot \tau_\sigma, \overline{f_\sigma^i} \right) \right) \Big). \end{aligned}$$

By LTL reasoning, this reduces equivalently to the fact that

$$\begin{aligned} &\exists f \, (\forall \sigma \in \Sigma_c) \\ &G \models \Box \Big(\boxminus P \cdot \tau_\sigma \to (\exists j \in \Lambda)(\exists i \in \mathcal{N}_\sigma^j) \\ &\quad \left(f_\sigma \to D_{\Sigma_o^i} \left(\overline{\lambda^\odot} \cdot \tau_\sigma, f_\sigma^{\bullet j} \cdot f_\sigma^i \right) \right) \\ &\quad \cdot \left(\overline{f_\sigma} \to D_{\Sigma_o^i} \left(\lambda^\odot \cdot \tau_\sigma, \overline{f_\sigma^i} \right) \right) \Big). \end{aligned}$$

An equivalent fact refining f further is that

$$\begin{aligned} &\exists f \, (\forall \sigma \in \Sigma_c) \\ &G \models \Box \Big(\boxminus P \cdot \tau_\sigma \to (\exists j \in \Lambda)(\exists i \in \mathcal{N}_\sigma^j) \\ &\quad \left(f_\sigma \to D_{\Sigma_o^i} \left(\overline{\lambda^\odot} \cdot \tau_\sigma, f_\sigma^{\bullet j} \cdot f_\sigma^i \right) \right) \\ &\quad \cdot \left(\overline{f_\sigma} \to D_{\Sigma_o^i} \left(\lambda^\odot \cdot \tau_\sigma, \overline{f_\sigma^i} \right) \right) \Big) \\ &\cdot \Box \Big(\overline{\boxminus P} \cdot \tau_\sigma \to (\exists j \in \Lambda)(\exists i \in \mathcal{N}_\sigma^j) \\ &\quad \left(f_\sigma \to D_{\Sigma_o^i} \left(\overline{\lambda^\odot} \cdot \tau_\sigma, f_\sigma^{\bullet j} \cdot f_\sigma^i \right) \right) \\ &\quad \cdot \left(\overline{f_\sigma} \to D_{\Sigma_o^i} \left(\lambda^\odot \cdot \tau_\sigma, \overline{f_\sigma^i} \right) \right) \Big). \end{aligned}$$

By LTL reasoning, this in turn is the equivalent fact that

$$\begin{aligned} \exists f (\forall \sigma \in \Sigma_c)\, G \models \Box \Big(\tau_\sigma \to (\exists j \in \Lambda)(\exists i \in \mathcal{N}_\sigma^j) \\ (f_\sigma \to D_{\Sigma_o^i}(\overline{\lambda^\odot} \cdot \tau_\sigma, f_\sigma^{\bullet j} \cdot f_\sigma^i)) \cdot \\ \left(\overline{f_\sigma} \to D_{\Sigma_o^i} \left(\lambda^\odot \cdot \tau_\sigma, \overline{f_\sigma^i} \right) \right) \Big). \end{aligned}$$

(Only if) Immediate by LTL reasoning.

Hence the lemma. ■

Theorem 3: Consider the kernel P of an arbitrary invariant over fair DES G with system marker set $\mathcal{M}$. At local control site $i \in \mathcal{N} = \{1, 2, \ldots, n_o\}$, let local P_i be the kernel of some invariant over G. Then there exists a feasible $\mathcal{N}$-decentralized $(P, \mathcal{M})$-supervisor f under the default open, universal fusion rule (3) for G, such that

$$\begin{aligned} &(\forall i \in \mathcal{N}) \\ &\quad (\forall \sigma \in \Sigma_c^i) \quad f_\sigma^i = \bigcirc_\sigma(\boxminus P_i) \qquad [\text{rel to } (\boxminus P, G)] \\ &\text{and} \qquad \boxminus P \Rightarrow U^\Lambda(\boxminus P_1, \ldots, \boxminus P_{n_o}) \text{ (with respect to } G), \end{aligned}$$

iff $\Box P$ is $\mathcal{M}$-controllable with respect to G, and U^Λ-co-observable with respect to G and $(P_i, \Sigma_o^i, \Sigma_c^i)$ for all $i \in \mathcal{N}$.

Proof: Consider the kernel P of an arbitrary invariant over fair DES G with system marker set $\mathcal{M}$. With

$$U^\Lambda(\boxminus P_1, \ldots, \boxminus P_{n_o}) \equiv \boxminus \left(\sum_{j \in \Lambda} \boxminus P^j \right),$$

[6] I.e., the connective *not* $^-$ is not used in the logic combination of the invariants, but is permitted within their kernels.

where kernel P_i of some invariant over G is at local site $i \in \mathcal{N} = \{1, 2, \ldots, n_o\} = \bigcup_{j\in\Lambda} \mathcal{N}^j$, $P^j \equiv \prod_{i\in\mathcal{N}^j} P_i$, and $\mathcal{N}^j_\sigma = \mathcal{N}^j \cap \mathcal{N}_\sigma$, it follows that:

$\Box P$ is $\mathcal{M}$-controllable and U^Λ-co-observable

iff there exists a proper $(P, \mathcal{M})$-supervisor f for G, such that

$(\forall \sigma \in \Sigma_c)\ f_\sigma = \bigcirc_\sigma(P)$ [rel to $(\boxminus P, G)$], and

$$\begin{aligned}(\forall \sigma \in \Sigma_c)\ G \models \Box\, \big(\boxminus P \cdot \tau_\sigma \to {} & (\exists j \in \Lambda)(\exists i \in \mathcal{N}^j_\sigma) \\ & \left(\bigcirc P \to D_{\Sigma^i_o}\left(\overline{\lambda^\bullet} \cdot \boxminus P, \bigcirc_\sigma\left(\boxminus P^j\right)\right)\right) \\ & \cdot \left(\bigcirc \overline{P} \to D_{\Sigma^i_o}\left(\lambda^\bullet \cdot \boxminus P, \bigcirc_\sigma\left(\overline{P_i}\right)\right)\right)\big),\end{aligned}$$

and $\boxminus P \equiv U^\Lambda(\boxminus P_1, \ldots, \boxminus P_{n_o})$ (with respect to G)

[by Theorem 1 and Definition 11]

iff there exists a proper $(P, \mathcal{M})$-supervisor f for G, such that

$$\begin{aligned} &(\forall \sigma \in \Sigma_c) && f_\sigma = \left(\sum_{j\in\Lambda} f^{\bullet j}_\sigma\right) && \text{[rel to } (\boxminus P, G)], \\ &\text{where} && f^{\bullet j}_\sigma \equiv \left(\prod_{i\in\mathcal{N}^j} f^i_\sigma\right) && \text{for an arbitrary } j \in \Lambda, \\ &\text{with} && f^i_\sigma = \bigcirc_\sigma(\boxminus P_i) && \text{[rel to } (\boxminus P, G)],\end{aligned}$$

$$\begin{aligned}\text{and } G \models \Box \Big(\boxminus P \cdot \tau_\sigma \to {} & (\exists j \in \Lambda)(\exists i \in \mathcal{N}^j_\sigma) \\ & \left(f_\sigma \to D_{\Sigma^i_o}\left(\overline{\lambda^\bullet} \cdot \boxminus P \cdot \tau_\sigma, f^{\bullet j}_\sigma \cdot f^i_\sigma\right)\right) \\ & \cdot \Big(\overline{f_\sigma} \to D_{\Sigma^i_o}\Big(\lambda^\bullet \cdot \boxminus P \cdot \tau_\sigma, \overline{f^i_\sigma}\Big)\Big)\Big),\end{aligned}$$

and $\boxminus P \Rightarrow U^\Lambda(\boxminus P_1, \ldots, \boxminus P_{n_o})$ (with respect to G)

[By definition of $(P, \mathcal{M})$-supervisor, LTL reasoning, and logic substitutions for f_σ and, in essence, f^i_σ defined for each $i \in \mathcal{N}$, for all $\sigma \in \Sigma_c$]

iff there exists a proper $\mathcal{N}$-decentralized $(P, \mathcal{M})$-supervisor f under the default open, universal fusion rule for G, such that $(\forall i \in \mathcal{N})$

$(\forall \sigma \in \Sigma^i_c)\quad f^i_\sigma = \bigcirc_\sigma(\boxminus P_i)\quad$ [rel to $(\boxminus P, G)$],

$$\begin{aligned}\text{and } G \models \Box \Big(\boxminus P \cdot \tau_\sigma \to {} & (\exists j \in \Lambda)(\exists i \in \mathcal{N}^j_\sigma) \\ & \left(f_\sigma \to D_{\Sigma^i_o}\left(\overline{\lambda^\bullet} \cdot \boxminus P \cdot \tau_\sigma, f^{\bullet j}_\sigma \cdot f^i_\sigma\right)\right) \\ & \cdot \Big(\overline{f_\sigma} \to D_{\Sigma^i_o}\Big(\lambda^\bullet \cdot \boxminus P \cdot \tau_\sigma, \overline{f^i_\sigma}\Big)\Big)\Big),\end{aligned}$$

and $\boxminus P \Rightarrow U^\Lambda(\boxminus P_1, \ldots, \boxminus P_{n_o})$ (with respect to G)

[By supervisor $\mathcal{N}$-decentralization according to the default open, universal fusion rule (3)]

iff there exists a proper $\mathcal{N}$-decentralized $(P, \mathcal{M})$-supervisor f under the default open, universal fusion rule for G, such that $(\forall i \in \mathcal{N})$

$(\forall \sigma \in \Sigma^i_c)\quad f^i_\sigma = \bigcirc_\sigma(\boxminus P_i)\quad$ [rel to $(\boxminus P, G)$],

$$\begin{aligned}\text{and } G \models \Box \Big(\tau_\sigma \to {} & (\exists j \in \Lambda)(\exists i \in \mathcal{N}^j_\sigma) \\ & \left(f_\sigma \to D_{\Sigma^i_o}\left(\overline{\lambda^\bullet} \cdot \tau_\sigma, f^{\bullet j}_\sigma \cdot f^i_\sigma\right)\right) \\ & \cdot \Big(\overline{f_\sigma} \to D_{\Sigma^i_o}\Big(\lambda^\bullet \cdot \tau_\sigma, \overline{f^i_\sigma}\Big)\Big)\Big),\end{aligned}$$

and $\boxminus P \Rightarrow U^\Lambda(\boxminus P_1, \ldots, \boxminus P_{n_o})$ (with respect to G)

[By Lemma 3 with $\lambda^\odot \equiv \lambda^\bullet$]

iff there exists a feasible $\mathcal{N}$-decentralized $(P, \mathcal{M})$-supervisor f under the default open, universal fusion rule for G, such that $(\forall i \in \mathcal{N})$

$(\forall \sigma \in \Sigma^i_c)\quad f^i_\sigma = \bigcirc_\sigma(\boxminus P_i)\quad$ [rel to $(\boxminus P, G)$],

and $\boxminus P \Rightarrow U^\Lambda(\boxminus P_1, \ldots, \boxminus P_{n_o})$ (with respect to G)

[By supervisor feasibility under the default open, universal fusion rule (3)].

Hence the theorem. ■

By the condition $\boxminus P \Rightarrow U^\Lambda(\boxminus P_1, \ldots, \boxminus P_{n_o})$, a feasible $\mathcal{N}$-decentralized $(P, \mathcal{M})$-supervisor f that exists according to Theorem 3 is said to be optimal.

D. THE GENERAL LTL CO-OBSERVABILITY PICTURE

LTL universal co-observability may be equivalently redefined by replacing Condition UKO2 in the original Definition 11 with Condition UKO2E given below – essentially obtainable by setting $\lambda^\bullet \equiv false$ in the former condition, and applying the dynamic default setting obtained by removing the or-term $|\Lambda| = 1$ in the original setting (4).

UKO2E) $(\forall \sigma \in \Sigma_c)$

$$\begin{aligned} G \models \Box\, \big(\boxminus P \cdot \tau_\sigma \to {} & (\exists j \in \Lambda)(\exists i \in \mathcal{N}^j_\sigma) \\ & \left(\bigcirc P \to D_{\Sigma^i_o}\left(\boxminus P, \bigcirc_\sigma\left(\boxminus P^j\right)\right)\right)\big).\end{aligned}$$

By the foregoing exposition, Theorem 3 shows LTL universal co-observability as orchestrating a general logic combination blending together a convolution of multiple control architectures, with each delineable as a counterpart of the language-formulated type (C&P + D&A) [17] or, equivalently, zero-level inferencing [20], but is under a non-static default setting unlike (C&P + D&A)-co-observability [17]. Each is also under a partition of the controllable event set into permissive and anti-permissive sets that is not known *a priori*, unlike that for (C&P + D&A)-co-observability [17] and implicitly like each delineated one for N-inference observability [20].

Examining further, under LTL universal co-observability, local sites having the same locally observed submodel[7] or, equivalently, with the same local observation channel, may be aggregated into one, effectively reducing the total number of sites which then all become submodel-unique (see Footnote 5). It follows that each aggregated site handles more than one component invariant of the given invariant, thus rendering its local control multi-decisioning. In what then follows, the equivalent multi-decision version, of universal (or U^Λ-) co-observability given by Definition 11, may be obtained by re-indexing, according to site aggregations, the control sites and the component invariants constituting the given invariant,

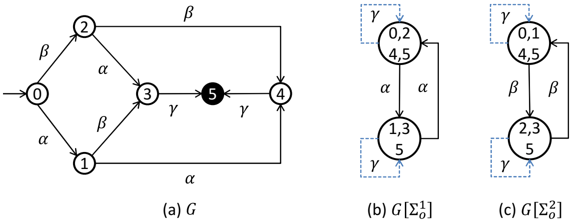
[7] That the submodel is the same is in the sense of equality of formal languages locally observed of the DES at these sites. And note that, at any two different sites x, y $(x \neq y)$, if every event in $\Sigma^x_o \cup \Sigma^y_o$ is defined at some state along some state-trajectory of DES model G, then their respective submodels $G[\Sigma^x_o]$, $G[\Sigma^y_o]$ as depicted in Fig. 1 are the same provided they have the same local observable event set, i.e., $\Sigma^x_o = \Sigma^y_o$.

and re-expressing the fusion rule in terms of local control decisions, each correspondingly index-redefined to associate with its submodel-unique control site and component invariant. Specifically, introduce two new nonempty auxiliary sets: 1) The index set $\mathcal{PF}$ that collects every index z such that (i, z) identifies control $f^{(i,z)}$ and correspondingly P_i^z – the z-indexed kernel of an invariant at control site $i \in \mathcal{N}$, and 2) the set $\mathcal{P}_i$, of indexed kernels of invariants at site $i \in \mathcal{N}$. Using the new sets, equivalently redefine Definition 11, now with respect to DES G and $(\mathcal{P}_i, \Sigma_o^i, \Sigma_c^i)$ for all $i \in \mathcal{N}$, and Condition UDF2 of control feasibility. Then a translation of Theorem 3 to a multi-decision version easily follows, mapping the architectural blending noted above into a general picture of $|\Lambda|$ architectures, with each under product fusion and all running in parallel under their overall fusion summation, over $|\mathcal{N}|$ control sites. This picture standardizes the original multi-decision control architecture depicted in [21, Fig. 1]. And in the special case with every component invariant logically subsumed by the given invariant, multi-decision co-observability structurally specializes to an LTL counterpart of the original concept [21] that is the most general language-based version to date. This concept subsumes N-inference observability [20] that is realizable by a blending of $(N+1)$ (C&P + D&A) control architectures [50], as reviewed by their equivalent type of zero-level inferencing in the introduction. In turn, the latter subsumes all other comparable versions known to date [15], [17]–[19], all in all underlining the concept formulation generality of LTL universal co-observability.

E. PRODUCT BOUNDARY CASE: A TOY EXAMPLE

Consider a toy example adapted from a seminal paper [15] on co-observability, with the key components as follows: 1) DES G, with $\Sigma = \Sigma_c = \{\alpha, \beta, \gamma\}$, $Q_0 = \{0\}$ (initial state set), and $Q = \{0, \ldots, 5\}$ such that proposition $p_x \in \Pi$ is $true$ at state $x \in Q$ and $false$ elsewhere; and it is graphically depicted in Fig. 4(a) with an x-numbered node for state x, an event-labeled edge for a transition of the event from one state to another as edge-directed, and a node with an entering arrow for an initial state; 2) Specification pair $(P, \mathcal{M})$, where $P \equiv (\ominus(p_1 + p_2) \rightarrow \overline{p_4})$ and $\mathcal{M} = \{p_5\}$, depicted with the node for state $5 \in Q$ darkened; 3) $\mathcal{N} = \{1, 2\}$, i.e., two sites indexed 1 and 2 with $(P_1, \Sigma_o^1, \Sigma_c^1)$ and $(P_2, \Sigma_o^2, \Sigma_c^2)$, respectively, where $P_1 \equiv (\ominus p_1 \rightarrow \overline{p_4})$, $\Sigma_o^1 = \{\alpha\}$, $\Sigma_c^1 = \{\alpha, \gamma\}$, and $P_2 \equiv (\ominus p_2 \rightarrow \overline{p_4})$, $\Sigma_o^2 = \{\beta\}$, $\Sigma_c^2 = \{\beta, \gamma\}$.

$\Box P$ can be easily shown to be $\mathcal{M}$-controllable. Since $P \equiv P_1 \cdot P_2$ and P is initially satisfied, Condition KO1 of Definition 7 of product co-observability is true; so is Condition KO2 due to the following G-validities, verified against the models (a) to (c) in Fig. 4 by inspection: 1) $\boxminus P \cdot \tau_{\gamma} \Rightarrow \bigcirc P$ (at γ-transitions to state 5), 2) $\boxminus P \cdot \tau_{\alpha} \cdot \bigcirc \overline{P} \Rightarrow D_{\Sigma_o^1}\left(\boxminus P, \bigcirc_{\alpha}\left(\overline{P_1}\right)\right)$ (at α-transition to state 4), 3) $\boxminus P \cdot \tau_{\beta} \cdot \bigcirc \overline{P} \Rightarrow D_{\Sigma_o^2}\left(\boxminus P, \bigcirc_{\beta}\left(\overline{P_2}\right)\right)$ (at β-transition to state 4). Thus, by the result specializing Theorem 3 to product co-observability, a feasible and optimal $\mathcal{N}$-decentralized $(P, \mathcal{M})$-supervisor f under the default permissive, product fusion rule (1) for DES G exists, such that $(\forall i \in \mathcal{N})(\forall \sigma \in \Sigma_c^i)\ f_\sigma^i = \bigcirc_\sigma(P_i)$ [rel to $(\boxminus P, G)$].



(a) G

(b) $G[\Sigma_o^1]$

(c) $G[\Sigma_o^2]$

FIGURE 4. For a toy example: (a) DES model G, (b) submodel $G[\Sigma_o^1]$, (c) submodel $G[\Sigma_o^2]$. The submodels are self-looped augmented with their site's unobservable, controllable event γ.

VI. CONCLUSION

A new LTL characterization of the co-observability concept in its general form, called universal co-observability, is studied for marker-progressive decentralized control of a class of fair DES's under temporal safety. This LTL concept development is unifying in scope, with demonstrated transparency and structural generality underpinning a more coherent understanding of co-observability and specification modularity of canonical LTL safety for decentralized supervisory control.

A rich formalism, canonical LTL provides neither a decidable nor an efficient basis for synthesis in general. Thus, future work includes studying versions of the MP-DSCP that are decidable, and solvable by efficient computer-aided synthesis leveraging the uniformity of LTL syntax-based reasoning.

Future work also includes studying a new but complementary LTL variant of universal co-observability. This is to be an extended counterpart of an existing language-cum-automaton-based formulation of a co-observability variant, studied in [52] for a so-called state-estimator-intersection(-fusion)-based architecture; generalizing that initiated in [53], this variant is proved [52] to be generally incomparable with those purely language-based formulations to date that universal co-observability studied herein is an extended counterpart of.

REFERENCES

[1] 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.

[2] 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.

[3] 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.

[4] W. J. Fokkink and M. Goorden, "Offline supervisory control synthesis: Taxonomy and recent developments," *Discrete Event Dynamic Systems: Theory and Applications*, vol. 34, no. 4, pp. 605–657, December 2024.

[5] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*, 3rd ed. Springer Cham, 2021.

[6] P. Gohari-Moghadam and W. M. Wonham, "On the complexity of supervisory control design in the rw framework," *IEEE Transactions on Systems,*

Man and Cybernetics – Part B : Cybernetics, vol. 30, no. 5, pp. 643–652, October 2000.

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

[8] ——, *The Temporal Logic of Reactive and Concurrent Systems: Specification*. Springer-Verlag New York, Inc, 1992.

[9] K. T. Seow, "Supervisory control of fair discrete-event systems: A canonical temporal logic foundation," *IEEE Transactions on Automatic Control*, vol. 66, no. 11, pp. 5269–5282, November 2021.

[10] ——, "Supremal marker-controllable subformula of a given canonical temporal-safety formula," *IEEE Access*, vol. 10, pp. 66 300–66 320, June 2022.

[11] 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.

[12] 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.

[13] 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.

[14] R. Cieslak, C. Desclaux, F. A. S., and P. Varaiya, "Supervisory control of discrete event processes with partial observations," *IEEE Transactions on Automatic Control*, vol. 33, no. 3, pp. 249–260, March 1988.

[15] K. Rudie and W. M. Wonham, "Think globally, act locally : Decentralized supervisory control," *IEEE Transactions on Automatic Control*, vol. 37, no. 11, pp. 1692–1708, November 1992.

[16] J. H. Prosser, M. Kam, and H. G. Kwatny, "Decision fusion and supervisor synthesis in decentralized discrete-event systems," in *Proceedings of the American Control Conference*, Albuquerque, New Mexico, USA, June 1997, pp. 2251–2255.

[17] T.-S. Yoo and S. Lafortune, "A general architecture for decentralized supervisory control of discrete-event systems," *Discrete Event Dynamic Systems: Theory and Applications*, vol. 12, no. 3, pp. 335–377, July 2002.

[18] S. Takai, R. Kumar, and T. Ushio, "Characterization of co-observable languages and formulas for their super/sublanguages," *IEEE Transactions on Automatic Control*, vol. 50, no. 4, pp. 434–447, April 2005.

[19] T.-S. Yoo and S. Lafortune, "Decentralized supervisory control with conditional decisions: Supervisor existence," *IEEE Transactions on Automatic Control*, vol. 49, no. 11, pp. 1886–1904, November 2004.

[20] R. Kumar and S. Takai, "Inference-based ambiguity management in decentralized decision-making: Decentralized control of discrete event systems," *IEEE Transactions on Automatic Control*, vol. 52, no. 10, pp. 1783–1794, October 2007.

[21] H. Chakib and A. Khoumsi, "Multi-decision supervisory control: Parallel decentralized architectures cooperating for controlling discrete event systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 11, pp. 2608–2622, November 2011.

[22] S.-H. Lee and K. C. Wong, "Structural decentralized control of concurrent discrete-event systems," *European Journal of Control*, vol. 8, no. 5, pp. 477–491, 2002.

[23] M. H. de Queiroz and J. E. R. Cury, "Modular control of composed systems," in *Proceedings of the American Control Conference*, Chicago, IL, USA, 2000, pp. 4051–4055.

[24] ——, "Modular supervisory control of large scale discrete event systems," in *Discrete Event Systems*, ser. The Springer International Series in Engineering and Computer Science, R. K. Boel and G. Stremersch, Eds. Springer US, 2000, vol. 569, pp. 103–110.

[25] L. Feng and W. M. Wonham, "Supervisory control architecture for discrete-event systems," *IEEE Transactions on Automatic Control*, vol. 53, no. 6, pp. 1449–1461, July 2008.

[26] S. Jiang, R. Kumar, S. Takai, and W. Qiu, "Decentralized control of discrete-event systems with multiple local specifications," *IEEE Transactions on Automation Science and Engineering*, vol. 7, no. 3, pp. 512–522, July 2010.

[27] J. Komenda and T. Masopust, "Supervisory control of modular discrete-event systems under partial observation: Normality," *IEEE Transactions on Automatic Control*, vol. 69, no. 6, pp. 3796–3807, June 2024.

[28] K. Schmidt, H. Marchand, and B. Gaudin, "Modular and decentralized supervisory control of concurrent discrete event systems using reduced system models," in *Proceedings of the 8th International Workshop on Discrete-Event Systems*, Ann Arbor, MI, USA, July 2006, pp. 149–154.

[29] A. M. Mainhardt and A.-K. Schmuck, "Assume-guarantee synthesis of decentralised supervisory control," in *Proceedings of the 16th International Workshop on Discrete Event Systems (WODES'22)*, Prague, Czech Republic, September 2022, pp. 165–172.

[30] F. Lin and W. M. Wonham, "Decentralized supervisory control of discrete event systems," *Information Sciences*, vol. 44, pp. 199–224, 1988.

[31] K. Schmidt and C. Breindl, "Maximally permissive hierarchical control of decentralized discrete event systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 4, pp. 723–737, April 2011.

[32] K. Schmidt, M. H. de Queiroz, and J. E. R. Cury, "Hierarchical and decentralized multitasking control of discrete event systems," in *Proceedings of the 46th IEEE International Conference on Decision and Control*, New Orleans, LA, U.S.A, December 2007, pp. 5936–5941.

[33] P. J. Ramadge and W. M. Wonham, "The control of discrete event systems," *Proceedings of the IEEE*, vol. 77, no. 1, pp. 81–98, January 1989.

[34] W. M. Wonham and P. J. Ramadge, "Modular supervisory control of discrete-event systems," *Mathematics of Control, Signals and Systems*, vol. 1, no. 1, pp. 13–30, January 1988.

[35] K. C. Wong and W. M. Wonham, "Modular control and coordination of discrete-event systems," *Discrete Event Dynamic Systems : Theory and Applications*, vol. 8, no. 3, pp. 247–297, October 1998.

[36] F. Lin and W. M. Wonham, "Decentralized control and coordination of discrete-event systems with partial observation," *IEEE Transactions on Automatic Control*, vol. 35, no. 12, pp. 1330–1337, December 1990.

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

[38] S. Jiang and R. Kumar, "Decentralized control of discrete event systems with specializations to local control and concurrent systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 30, no. 5, pp. 653–660, October 2000.

[39] R. C. Hill and D. M. Tilbury, "Incremental hierarchical construction of modular supervisors for discrete-event systems," *International Journal of Control*, vol. 81, no. 9, pp. 1364–1381, September 2008.

[40] M. H. de Queiroz and J. E. R. Cury, "Modular multitasking supervisory control of composite discrete-event systems," in *IFAC Proceedings Volumes (The 16th IFAC World Congress)*, vol. 38, no. 1, Prague, Czech Republic, July 2005, pp. 91–96.

[41] K. Schmidt and C. Breindl, "On maximal permissiveness of hierarchical and modular supervisory control approaches for discrete event systems," in *Proceedings of the 9th International Workshop on Discrete Event Systems*, Goteborg, Sweden, May 2008, pp. 462–467.

[42] F. Lin and W. M. Wonham, "On observability of discrete event systems," *Information Sciences*, vol. 44, pp. 173–198, 1988.

[43] J. Komenda and T. Masopust, "A bridge between decentralized and coordination control," in *Proceedings of the 51st Annual Allerton Conference on Communication, Control, and Computing*, Monticello, Illinois, USA, October 2013, pp. 966–972.

[44] ——, "Computation of controllable and coobservable sublanguages in decentralized supervisory control via communication," *Discrete Event Dynamic Systems: Theory and Applications*, vol. 27, no. 4, pp. 585–608, December 2017.

[45] S. L. Ricker and K. Rudie, "Knowledge is a terrible thing to waste: Using inference in discrete-event control problems," *IEEE Transactions on Automatic Control*, vol. 52, no. 3, pp. 428–441, 2007.

[46] K. Ritsuka and K. Rudie, "Epistemic interpretations of decentralized discrete-event system problems," *Discrete Event Dynamic Systems: Theory and Applications*, vol. 32, no. 3, pp. 359–398, September 2022.

[47] ——, "Do what you know: Coupling knowledge with action in discrete-event systems," *Discrete Event Dynamic Systems: Theory and Applications*, vol. 33, no. 3, pp. 257–277, September 2023.

[48] 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.

[49] A. Khoumsi and H. Chakib, "Decentralized supervisory control of discrete event systems: An arborescent architecture to realize inference-based control," *IEEE Transactions on Automatic Control*, vol. 63, no. 12, pp. 4278–4285, December 2018.

[50] ——, "Decentralized supervisory control of discrete event systems: Using multi-decision control as an alternative to inference-based control," in *Proceedings of the 13th International Conference on Informatics in Control, Automation and Robotics - Volume 1: ICINCO*, Lisbon, Portugal, July 2016, pp. 213–220.

[51] F. Lin, "Opacity of discrete event systems and its applications," *Automatica*, vol. 47, no. 3, pp. 496–503, March 2011.

[52] A. Hayano and S. Takai, "A general architecture for intersection-based decentralized supervisory control of discrete event systems," *IEEE Transactions on Automatic Control*, vol. 69, no. 1, pp. 674–680, November 2016.

[53] X. Yin and S. Lafortune, "Decentralized supervisory control with intersection-based architecture," *IEEE Transactions on Automatic Control*, vol. 61, no. 11, pp. 3644–3650, November 2016.

KIAM TIAN SEOW (Senior Member, IEEE) 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 holding the title of Visiting Professor with the Robot Intelligence Technology Laboratory, 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 include modeling, control design, and applications of discrete-event and agent systems.