Relative Observability and Coobservability of Timed Discrete-Event Systems

Kai Cai, *Member, IEEE*, Renyuan Zhang, *Member, IEEE*, and W. M. Wonham, *Life Fellow, IEEE*

***Abstract*—We study supervisory control of timed discrete-event systems (TDES) under partial observation, and propose new observability concepts effective for supervisor synthesis. First, we consider monolithic/centralized supervisory control, and introduce *timed relative observability* and *timed relative weak observability*. The former concept extends our previous work to the timed case, while the latter exploits choices of forcible events to preempt the clock event *tick*. We prove that timed relative (respectively, weak) observability is stronger than timed (respectively, weak) observability, weaker than normality, and closed under set union; hence there exists the supremal relatively (respectively, weakly) observable sublanguage of a given language. We move on to study decentralized supervisory control of TDES, and propose *timed relative coobservability* and *timed relative weak coobservability* as extensions of their centralized counterparts. It is shown that timed relative (respectively, weak) coobservability is stronger than timed (respectively, weak) coobservability, weaker than conormality, and closed under set union; therefore the supremal relatively (respectively, weakly) coobservable sublanguage of a given language exists. Finally, algorithms are designed to compute the supremal relatively (weakly) (co)observable and controllable sublanguages, which are demonstrated with a Guideway example.**

***Index Terms*—Automata, decentralized supervisory control, partial observation, supervisory control, timed discrete-event systems, timed relative (weak) coobservability, timed relative (weak) observability.**

I. Introduction

We study supervisory control of timed discrete-event systems (TDES) under partial observation, and propose new observability concepts effective for supervisor synthesis. Many time-critical applications can be modeled as TDES, such as communication channels, sensor networks, logistic management and scheduling [1]. The correctness and optimality of TDES depend not only on the system's logical behavior, but also on the times at which various events occur. In practice it may well be the case that the occurrence of some events cannot be observed because of a lack of sensors (possibly due to cost). Therefore it is important to develop supervisory control of timed DES based only on partial event observation.

Partially-observed supervisory control of untimed DES in the Ramadge-Wonham (RW) framework [2]–[4] has been actively studied (e.g., [5]–[9]); *observability* and *normality* are two familiar concepts [5], [6]. Observability is necessary for the existence of a partially-observed supervisor, but it is not preserved under set union, and consequently the supremal observable sublanguage of a given language need not exist in general. Normality is closed under union, but may result in overly conservative controlled behavior inasmuch as unobservable events are not allowed to be disabled. In [10] observability was extended to supervisory control of TDES in the Brandin-Wonham (BW) framework [11], [4, Chapter 9]. Like its untimed counterpart, timed observability is not preserved under set union. In [12] a concept called *weak observability* was proposed for a distinct class of timed supervisors. In particular, the observability requirement for the special clock event $tick$ is relaxed by exploiting choices of *forcible events* (formal definitions are given below). Weak observability, however, is again not closed under set union.

We introduced *relative observability* in [13] for untimed DES, which is proved to be stronger than observability, weaker than normality, and preserved under set union; hence there exists the supremal relatively observable sublanguage of a given language. In this paper and its conference precursor [14], we extend relative observability to supervisory control of TDES in the BW framework. Specifically, we propose *timed relative observability* and *timed relative weak observability*, extending respectively [10] and [12]. First, we introduce timed relative observability, and prove that it is stronger than timed observability [10], weaker than normality, and closed under set union. Second, we introduce timed relative weak observability, and show that it is stronger than weak observability [12], weaker than normality, and closed under set union. We design an algorithm for computing the supremal relatively weakly observable sublanguage. The concepts proposed and relations proved, together with those of [10] and [12], are summarized on the left of Fig. 1.

Timed relative (weak) observability is formulated in a centralized setup where a monolithic supervisor partially observes and controls the TDES plant as a whole. We move on to study a *decentralized* setup, where multiple decentralized supervisors operate jointly, each of which observes and controls only part of the TDES plant. Decentralized supervisory control is an effective means of managing computational complexity for large-scale systems, and has been extensively investigated for untimed DES in the RW framework (e.g., [6], [15]–[17]). The

Manuscript received June 8, 2015; revised October 5, 2015; accepted December 18, 2015. Date of publication December 29, 2015; date of current version October 25, 2016. This work was supported in part by JSPS KAKENHI Grant no. JP16K18122 and Program to Disseminate Tenure Tracking System, MEXT, Japan; the Fundamental Research Funds for the Central Universities, China, Grant 3102014JCQ01069, and the National Nature Science Foundation of China, Grant 61403308; the Natural Sciences and Engineering Research Council, Canada, Grant no. 7399. Recommended by Associate Editor S. Takai.

K. Cai is with the Urban Research Plaza, Osaka City University, Osaka 558 8585, Japan (e-mail: kai.cai@eng.osaka-cu.ac.jp).

R. Zhang is with the Department of Traffic and Control Engineering, Northwestern Polytechnical University, Xi'an 710072, China (e-mail: ryzhang@nwpu.edu.cn).

W. M. Wonham is with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M55364, Canada (e-mail: wonham@ece.utoronto.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TAC.2015.2513370

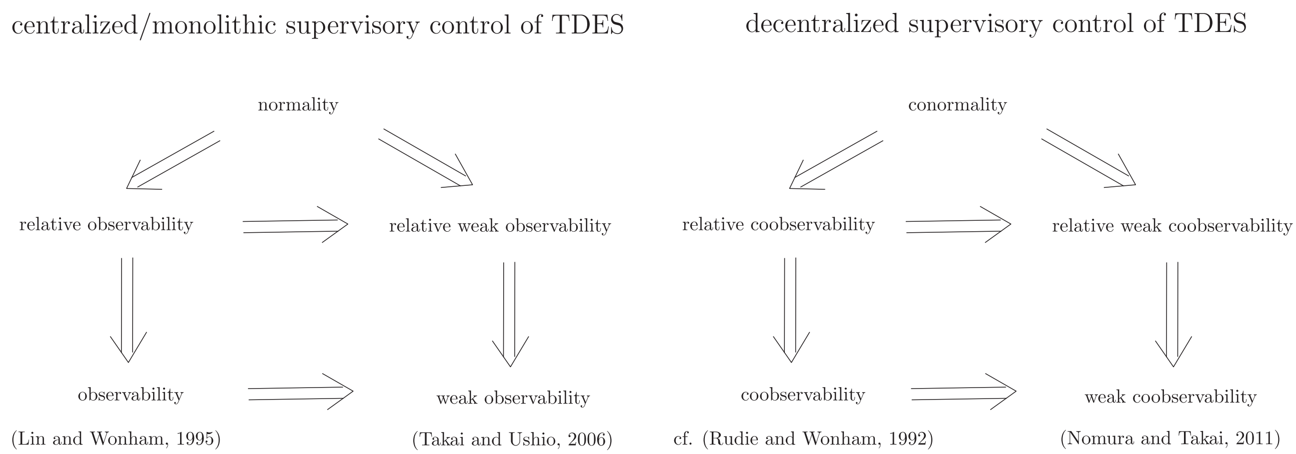


Fig. 1. Observability concepts and their relations in centralized/monolithic and decentralized supervisory control of TDES under partial observation.

fundamental concepts are *coobservability* [6], [15] (and its variations [16], [17]) and conormality [15]. Coobservability specifies the (AND/OR) rule of integrating local control decisions, and is necessary for the existence of decentralized supervisors. Just like observability and normality in the centralized setup, coobservability is not closed under set union while conormality may result in overly conservative controlled behavior. By a method similar to that of [10], coobservability may be extended to decentralized control of TDES in the BW framework. Moreover in [18] the authors studied weak coobservability conditions, but these are again not closed under set union.

Recently, we introduced *relative coobservability* [19] in decentralized supervisory control of untimed DES. Relative coobservability is shown to be stronger than (any variations of) coobservability, weaker than conormality, and preserved under set union; hence, there exists the supremal relatively coobservable sublanguage of a given language. In the second part of this paper, we extend relative coobservability to decentralized control of TDES in the BW framework. First, we introduce timed relative coobservability, and prove that it is stronger than timed coobservability (cf. [15]), weaker than conormality, and closed under set union. Second, we propose timed relative weak coobservability, and show that it is stronger than weak coobservability [18], weaker than conormality, and closed under set union. The concepts proposed and relations proved, together with those of [18], are summarized on the right of Fig. 1.

Finally, we design algorithms for computing the supremal relatively (weakly) (co)observable (and controllable, $L_m(\mathbf{G})$-closed) sublanguage of a given language. The algorithms and the proposed concepts are demonstrated with a Guideway example of partially-observed centralized/decentralized supervisory control.

We note that, for a given supervisor synthesis problem, even if the supremal relatively (weakly) (co)observable sublanguage of a given language is empty, there may still exist a nonempty (weakly) (co)observable sublanguage. The latter is however difficult to compute for non-prefix-closed languages. See [9] for recent work on this problem in the untimed centralized setting. In the decentralized setting, the existence of a nonempty solution is generally undecidable [20].

We also note that many timed DES models and approaches have been studied in the literature, including Brave and Heymann's "clock automata" [21], Ostroff's "timed transition models" [22], Brandin and Wonham's TDES [11], and Cofer and Garg's model based on "timed Petri nets" [23]. We adopt Brandin and Wonham's TDES as the framework of developing new observability concepts mainly for technical convenience in extending our own previous work as well as for easy comparison with relevant results in the literature. As demonstrated in [11], [4, Ch. 9], the BW framework captures a variety of timing issues useful in real-time discrete-event control problems including communication delays and operational hard deadlines.

The rest of the paper is organized as follows. In Section II we review the basics of the BW framework of timed supervisory control. Section III introduces timed relative observability and investigates its properties. Section IV proposes timed relative weak observability and studies its properties; an algorithm is designed to compute the supremal relatively weakly observable sublanguage. Section V introduces timed relative coobservability and timed relative weak coobservability; their properties are studied. An algorithm is developed to compute the supremal relatively (weakly) (co)observable, controllable, and $L_m(\mathbf{G})$-closed sublanguage. Section VI presents a Guideway example for demonstration of the proposed concepts and algorithms. Finally in Section VII we state our conclusions.

For easy reference, we list the main symbols used in the paper.

Symbols	Meanings
$E_K(s)$	Set of events eligible following string s in language $\overline{K}$
sup$\mathcal{O}(K,C)$	Supremal relatively observable sublanguage of K with ambient language C
sup$\mathcal{WO}(K,C)$	Supremal relatively weakly observable sublanguage of K with ambient language C
K^{O}_{sup}	Supremal relatively observable, controllable and $L_m(\mathbf{G})$-closed sublanguage of K with ambient language K
K^{WO}_{sup}	Supremal relatively weakly observable, controllable and $L_m(\mathbf{G})$-closed sublanguage of K with ambient language K
K^{CO}_{sup}	Supremal relatively coobservable, controllable and $L_m(\mathbf{G})$-closed sublanguage of K with ambient language K
K^{WCO}_{sup}	Supremal relatively weakly coobservable, controllable and $L_m(\mathbf{G})$-closed sublanguage of K with ambient language K

II. Preliminaries on Brandin-Wonham TDES Framework

This section reviews the TDES model proposed by Brandin and Wonham [11], [4, Ch. 9]. First consider the untimed DES model

$$\mathbf{G}_{\rm act} = (A, \Sigma_{\rm act}, \delta_{\rm act}, a_0, A_m). \tag{1}$$

Here, A is the finite set of *activities*, $\Sigma_{\rm act}$ the finite set of *events*, $\delta_{\rm act} : A \times \Sigma_{\rm act} \to A$ the (partial) *activity transition function*, $a_0 \in A$ the *initial activity*, and $A_m \subseteq A$ the set of *marker activities*. Let $\mathbb{N}$ denote the natural numbers $\{0, 1, 2, \ldots\}$. We introduce *time* into $\mathbf{G}_{\rm act}$ by assigning to each event $\sigma \in \Sigma_{\rm act}$ a *lower time bound* $l_\sigma \in \mathbb{N}$ and an *upper time bound* $u_\sigma \in \mathbb{N} \cup \{\infty\}$, such that $l_\sigma \le u_\sigma$; typically, l_σ represents a delay in communication or in control enforcement, while u_σ is often a hard deadline imposed by legal specification or physical necessity. With these assigned time bounds, the event set $\Sigma_{\rm act}$ is partitioned into two subsets: $\Sigma_{\rm act} = \Sigma_{\rm spe} \dot\cup \Sigma_{\rm rem}$ ($\dot\cup$ denotes *disjoint union*) with $\Sigma_{\rm spe} := \{\sigma \in \Sigma_{\rm act} | u_\sigma \in \mathbb{N}\}$ and $\Sigma_{\rm rem} := \{\sigma \in \Sigma_{\rm act} | u_\sigma = \infty\}$; here "spe" denotes "prospective," i.e., σ will occur within some prospective time (with a finite upper bound), while "rem" denotes "remote," i.e., σ will occur at some *indefinite* time (with no upper bound), or possibly will never occur at all.

A distinguished event, written $tick$, is introduced which represents "tick of the global clock." Then a TDES model

$$\mathbf{G} := (Q, \Sigma, \delta, q_0, Q_m) \tag{2}$$

is constructed from $\mathbf{G}_{\rm act}$ [11], [4, Chapter 9] with Q the finite set of *states*, $\Sigma := \Sigma_{\rm act} \dot\cup \{tick\}$ the finite set of events, $\delta : Q \times \Sigma \to Q$ is the (partial) *state transition function*, q_0 the *initial state*, and Q_m the set of *marker states*.

Let Σ^* be the set of all finite strings of elements in $\Sigma = \Sigma_{\rm act} \dot\cup \{tick\}$, including the empty string ϵ. We introduce the languages generated by TDES $\mathbf{G}$ in (2). The transition function δ is extended to $\delta : Q \times \Sigma^* \to Q$ in the usual way. The *closed behavior* of $\mathbf{G}$ is the language $L(\mathbf{G}) := \{s \in \Sigma^* | \delta(q_0, s)!\}$, and the *marked behavior* is $L_m(\mathbf{G}) := \{s \in L(\mathbf{G}) | \delta(q_0, s) \in Q_m\}$. Let $K \subseteq \Sigma^*$ be a language; its *prefix closure* is $\overline{K} := \{s \in \Sigma^* | (\exists t \in \Sigma^*) st \in K\}$. We say that K is $L_m(\mathbf{G})$-*closed* if

$$\overline{K} \cap L_m(\mathbf{G}) = K. \tag{3}$$

TDES $\mathbf{G}$ is *nonblocking* if $\overline{L_m(\mathbf{G})} = L(\mathbf{G})$.

To use TDES $\mathbf{G}$ in (2) for supervisory control, it is necessary to specify certain transitions that can be controlled by an external supervisor. First, as in the untimed theory [4], we need a subset of events that may be *disabled*. Since disabling an event usually requires preventing that event indefinitely from occurring, only remote events belong to this category. Thus, let a new subset $\Sigma_{\rm hib} \subseteq \Sigma_{\rm rem}$ denote the *prohibitible* events; the supervisor is allowed to disable any prohibitible event. Next, and specific to TDES, we bring in another category of events which can *preempt* event $tick$. Note that $tick$ may not be disabled, inasmuch as no control technology can stop the global clock indefinitely. On this basis let a new subset $\Sigma_{\rm for} \subseteq \Sigma_{\rm act}$ denote the *forcible* events; a forcible event is one that preempts event $tick$: if, at a state q of $\mathbf{G}$, $tick$ is defined and so are one or more forcible events, then $tick$ can be effectively erased from the current list of defined events (contrast with indefinite erasure). There is no particular relation postulated *a priori* between $\Sigma_{\rm for}$ and any of $\Sigma_{\rm hib}$, $\Sigma_{\rm rem}$ or $\Sigma_{\rm spe}$; in particular, a remote event may be both forcible and prohibitible. It is now convenient to define the *controllable* event set $\Sigma_c := \Sigma_{\rm hib} \dot\cup \{tick\}$. Here designating both $\Sigma_{\rm hib}$ and $tick$ controllable is to simplify terminology. We emphasize that events in $\Sigma_{\rm hib}$ can be disabled indefinitely, while $tick$ may be preempted only by events in $\Sigma_{\rm for}$. The *uncontrollable* event set Σ_u is $\Sigma_u := \Sigma \setminus \Sigma_c = \Sigma_{\rm spe} \dot\cup (\Sigma_{\rm rem} \setminus \Sigma_{\rm hib})$.

We introduce the notion of controllability in TDES as follows. Let $K \subseteq L(\mathbf{G})$ and $s \in \overline{K}$; define the *eligible event subset*

$$E_K(s) := \{\sigma \in \Sigma \mid s\sigma \in \overline{K}\}. \tag{4}$$

We say that K is *controllable* with respect to $\mathbf{G}$ in (2) if, for all $s \in \overline{K}$

$$E_K(s) \supseteq \begin{cases} E_{L(\mathbf{G})}(s) \cap (\Sigma_u \dot\cup \{tick\}) & \text{if } E_K(s) \cap \Sigma_{\rm for} = \emptyset \\ E_{L(\mathbf{G})}(s) \cap \Sigma_u & \text{if } E_K(s) \cap \Sigma_{\rm for} \neq \emptyset. \end{cases} \tag{5}$$

Thus, K controllable means that an event σ is eligible to occur in $\overline{K}$ if: (i) σ is currently eligible in $L(\mathbf{G})$ and (ii) either σ is uncontrollable or $\sigma = tick$ when there is no forcible event currently eligible in $\overline{K}$. Controllability plays the central role in the TDES supervisory control framework for the case of full-event observation.

III. Partially-Observed Supervisory Control of TDES by Relative Observability

Supervisory control of TDES under partial-event observation was studied in [10], where the concepts of timed observability and normality were introduced. This work is first reviewed. Then we introduce timed relative observability, which is stronger than timed observability, weaker than normality, and closed under set union.

A. Observability of TDES

Let $\Sigma_o \subseteq \Sigma$ be a subset of *observable* events. Define the *natural projection* $P : \Sigma^* \to \Sigma_o^*$ according to

$$\begin{aligned} P(\epsilon) &= \epsilon, \ \ \epsilon \text{ is the empty string} \\ P(\sigma) &= \begin{cases} \epsilon, & \text{if } \sigma \notin \Sigma_o \\ \sigma, & \text{if } \sigma \in \Sigma_o \end{cases} \\ P(s\sigma) &= P(s)P(\sigma), \ \ s \in \Sigma^*, \sigma \in \Sigma. \end{aligned} \tag{6}$$

As usual, P is extended to $P : Pwr(\Sigma^*) \to Pwr(\Sigma_o^*)$, where $Pwr(\cdot)$ denotes *powerset*. Write $P^{-1} : Pwr(\Sigma_o^*) \to Pwr(\Sigma^*)$ for the *inverse-image function* of P.

A *supervisor* V under partial observation is any map $V : P(L(\mathbf{G})) \to Pwr(\Sigma)$. Denote by $V/\mathbf{G}$ the closed-loop system where $\mathbf{G}$ is under the supervision of V; then the closed language $L(V/\mathbf{G}) \subseteq L(\mathbf{G})$ is defined inductively according to:

(i) $\epsilon \in L(V/\mathbf{G})$;
(ii) $s \in L(V/\mathbf{G}), \sigma \in V(Ps), s\sigma \in L(\mathbf{G}) \Rightarrow s\sigma \in L(V/\mathbf{G})$;
(iii) no other strings belong to $L(V/\mathbf{G})$.

The marked language $L_m(V/\mathbf{G})$ of $V/\mathbf{G}$ is defined by

$$L_m(V/\mathbf{G}) := L(V/\mathbf{G}) \cap L_m(\mathbf{G}).$$

A supervisor V is *nonblocking* if $\overline{L_m(V/\mathbf{G})} = L(V/\mathbf{G})$, and *admissible* if for each $s \in L(V/\mathbf{G})$:

(i) $\Sigma_u \subseteq V(Ps)$;
(ii) $(E_{L(\mathbf{G})}(s) \cap V(Ps) \cap \Sigma_{\text{for}} = \emptyset \ \& \ tick \in E_{L(\mathbf{G})}(s)) \Rightarrow tick \in V(Ps)$.

Let $K \subseteq L_m(\mathbf{G})$, and recall $\Sigma_c = \Sigma_{\text{hib}} \dot\cup \{tick\}$. We say that K is *observable* (with respect to $\mathbf{G}$ and P) [10] if for every pair of strings $s, s' \in \Sigma^*$ with $Ps = Ps'$, there holds

$$(\forall \sigma \in \Sigma_c)\ s\sigma \in \overline{K}, s' \in \overline{K}, s'\sigma \in L(\mathbf{G}) \Rightarrow s'\sigma \in \overline{K}. \quad (7)$$

In the definition, the event $tick$ is allowed to be unobservable, i.e., $P(tick) = \epsilon$. Note, however, that owing to the role of $tick$ in the TDES $\mathbf{G}$, $tick$ being unobservable may render the observability condition difficult to be satisfied for $K \subseteq L_m(\mathbf{G})$. The following is the main result of [10].

Theorem 1: Let $K \subseteq L_m(\mathbf{G})$ be a nonempty language. There exists a nonblocking, admissible supervisor V such that $L_m(V/\mathbf{G}) = K$ if and only if:

(i) K is observable [as in (7)];
(ii) K is controllable [as in (5)];
(iii) K is $L_m(\mathbf{G})$-closed [as in (3)].

While controllability and $L_m(\mathbf{G})$-closedness are properties closed under set union, observability is not; consequently the supremal sublanguage that satisfies the above three conditions (or the optimal supervisor) need not exist in general. This problem motivates us to propose the concept of relative observability below, which in fact is closed under set union.

B. Relative Observability of TDES

Fix a sublanguage $C \subseteq L_m(\mathbf{G})$. We introduce *relative observability* which sets C to be the *ambient language* in which observability is tested.

Definition 1: Let $K \subseteq C \subseteq L_m(\mathbf{G})$. We say that K is *relatively observable* with respect to C, $\mathbf{G}$, and P, or simply C*-observable*, if for every pair of strings $s, s' \in \Sigma^*$ with $Ps = Ps'$, there holds

$$(\forall \sigma \in \Sigma_c)\ s\sigma \in \overline{K}, s' \in \overline{C}, s'\sigma \in L(\mathbf{G}) \Rightarrow s'\sigma \in \overline{K} \quad (8)$$

where $\Sigma_c = \Sigma_{\text{hib}} \dot\cup \{tick\}$.

Relative observability was first proposed in [13] for untimed DES. Here, for TDES, we extend the concept by accounting for the event $tick$ which may be preempted only by a forcible event, in contrast with direct disablement of prohibitible events.

Let $C_1 \subseteq C_2 \subseteq L_m(\mathbf{G})$ be two ambient languages. By Definition 1 it is easily verified that C_2-observability implies C_1-observability. In other words, relative observability is weaker for smaller ambient language. In the special case where the ambient $C = K$, Definition 1 becomes (standard) timed observability [10] for the given K. This immediately implies the following.

Proposition 1: If $K \subseteq C$ is C-observable, then K is also observable.

The reverse statement need not be true (refer to [13] for a counterexample). Timed observability is not closed under set union: even if two sublanguages $K_1, K_2 \subseteq L_m(\mathbf{G})$ are observable, their union $K_1 \cup K_2$ need not be. This is because for timed observability of each K_i, $i = 1, 2$, one checks lookalike string pairs only in $\overline{K_i}$, ignoring all candidates permitted by the other language. By contrast, timed relative observability exploits a fixed ambient $C \subseteq L_m(\mathbf{G})$: for $K_1, K_2 \subseteq C$, no matter which K_i one checks for timed relative observability, all lookalike string pairs in $\overline{C}$ must be considered. It is indeed this more stringent requirement that renders timed relative observability algebraically well-behaved: an arbitrary union of relatively observable languages is again relatively observable.

Proposition 2: Let $K_i \subseteq C$, $i \in I$ (some index set), be C-observable. Then $K = \bigcup\{K_i \mid i \in I\}$ is also C-observable.

A proof is in [13] (identical to the untimed case). Whether or not $K \subseteq C$ is C-observable, write

$$\mathcal{O}(K, C) := \{K' \subseteq K \mid K' \text{ is } C\text{-observable}\} \quad (9)$$

for the family of C-observable sublanguages of K. Note that the empty language $\emptyset$ is trivially C-observable, thus a member of $\mathcal{O}(K, C)$. By Proposition 2, moreover, $\mathcal{O}(K, C)$ has a unique supremal element $\sup \mathcal{O}(K, C)$ given by

$$\sup \mathcal{O}(K, C) := \bigcup\{K' \mid K' \in \mathcal{O}(K, C)\}. \quad (10)$$

This is the supremal C-observable sublanguage of K. An algorithm that computes $\sup \mathcal{O}(K, C)$ was presented in [13]. Note that

$$\sup \mathcal{O}(K, C) \subseteq \sup \mathcal{O}(K, K) \text{ for } K \subseteq C \subseteq L_m(\mathbf{G}). \quad (11)$$

Now we show that relative observability is weaker than *normality* of TDES ([10]), a property that is also preserved by set union. A sublanguage $K \subseteq C$ is $(L(\mathbf{G}), P)$*-normal* if

$$\overline{K} = P^{-1}P\overline{K} \cap L(\mathbf{G}). \quad (12)$$

This implies that no string in $\overline{K}$ may exit $\overline{K}$ via an unobservable transition. Thus normality excludes, when control is present, the disablement of unobservable, prohibitible events, or the preemption of $tick$ in case $tick$ is unobservable. By contrast, timed relative observability does not impose this restriction, i.e., one may exercise disablement/preemption over unobservable events.

Proposition 3: If $K \subseteq C$ is $(L(\mathbf{G}), P)$-normal, then K is C-observable.

A proof is in [13].

Finally we turn to control. Let $K \subseteq L_m(\mathbf{G})$ be a nonempty specification language, and let the ambient language $C = K$ [because of (11)]. Since K-observability, controllability, and $L_m(\mathbf{G})$-closedness are all closed under set union, there exists a unique supremal sublanguage of K that satisfies these three properties. Denote this supremal sublanguage by K^O_{sup}; according to Proposition 1, K^O_{sup} is observable, controllable, and $L_m(\mathbf{G})$-closed. Therefore, by Theorem 1, there exists a nonblocking, admissible supervisor V such that $L_m(V/\mathbf{G}) = K^O_{\text{sup}}$. In Section V-C, we present an algorithm to compute K^O_{sup}.

IV. Partially-Observed Supervisory Control of TDES under Relative Weak Observability

A distinct type of supervisory control for TDES was proposed in [12], and a *weak observability* condition derived for the case of partial observation. This work is first reviewed. Then we introduce timed relative weak observability, which is stronger than weak observability but closed under set union. Computation of the supremal relatively weakly observable sublanguage of a given language will be discussed.

A. Weak Observability of TDES

Again let $\Sigma_o \subseteq \Sigma$ be a subset of observable events, and $P : \Sigma^* \to \Sigma_o^*$ be the natural projection. A *supervisor* V under partial observation is any map $V : P(L(\mathbf{G})) \to Pwr(\Sigma_{\text{act}}) \times Pwr(\Sigma_{\text{for}})$ such that for each $t \in P(L(\mathbf{G}))$, $V(t) = (V_1(t), V_2(t))$ satisfies the following two conditions:

(i) $\Sigma_u \subseteq V_1(t)$;
(ii) $V_2(t) \subseteq V_1(t) \cap \Sigma_{\text{for}}$.

Here, $V_1(t)$ is the set of events in Σ_{act} to be enabled, which must always include the uncontrollable subset Σ_u; $V_2(t)$ is the set of events in Σ_{for} which are candidates for forcing, and which must be enabled by V_1. The closed language $L(V/\mathbf{G})$ of the closed-loop system $V/\mathbf{G}$ is defined inductively according to

(i) $\epsilon \in L(V/\mathbf{G})$;
(ii) $s \in L(V/\mathbf{G}), \sigma \in \Sigma_{\text{act}} \cap V_1(Ps), s\sigma \in L(\mathbf{G}) \Rightarrow s\sigma \in L(V/\mathbf{G})$;
(iii) $s \in L(V/\mathbf{G}), E_{L(\mathbf{G})}(s) \cap V_2(Ps) = \emptyset, s.tick \in L(\mathbf{G}) \Rightarrow s.tick \in L(V/\mathbf{G})$;
(iv) no other strings belong to $L(V/\mathbf{G})$.

The marked language $L_m(V/\mathbf{G})$ of $V/\mathbf{G}$ is given by

$$L_m(V/\mathbf{G}) := L(V/\mathbf{G}) \cap L_m(\mathbf{G}).$$

Let $K \subseteq L_m(\mathbf{G})$. We say that K is *weakly observable* (with respect to $\mathbf{G}$ and P) [12] if the following two conditions hold:

(1) K is observable with respect to Σ_{hib}, i.e., for every pair of strings $s, s' \in \Sigma^*$ with $Ps = Ps'$ there holds

$$(\forall\, \sigma \in \Sigma_{\text{hib}})\; s\sigma \in \overline{K}, s' \in \overline{K}, s'\sigma \in L(\mathbf{G}) \Rightarrow s'\sigma \in \overline{K}.$$

(2) For each $t \in P(\overline{K})$, there exists a subset

$$F(t) \subseteq \left(\bigcup_{s \in \overline{K} \cap P^{-1}(t)} E_K(s) \right) \cap \Sigma_{\text{for}}$$

such that for each $s \in \overline{K} \cap P^{-1}(t)$ with $tick \in E_{L(\mathbf{G})}(s)$ there holds

$$tick \in E_K(s) \Leftrightarrow E_{L(\mathbf{G})}(s) \cap F(t) = \emptyset. \tag{14}$$

Weak observability is identical to observability with respect to Σ_{hib}, but exploits choices of forcible events to address preemption of the event $tick$. It was shown [12] that if K is observable and controllable, then it is weakly observable. The following is the main result of [12].

Theorem 2: Let $K \subseteq L_m(\mathbf{G})$ be a nonempty language. There exists a nonblocking supervisor V such that $L_m(V/\mathbf{G}) = K$ if and only if:

(i) K is weakly observable [as in (13)];
(ii) K is controllable [as in (5)];
(iii) K is $L_m(\mathbf{G})$-closed [as in (3)].

Like timed observability, weak observability is not closed under set union; consequently the supremal sublanguage that satisfies the above three conditions (or the optimal supervisor) need not exist in general. This motivates us to propose relative weak observability below, which in fact is closed under set union.

Remark 1: The implementation of the supervisor $V = (V_1, V_2)$ in Theorem 2 is as follows. After a string $s \in L(\mathbf{G})$ such that $s.tick \in L(\mathbf{G})$, V observes the string $t = Ps \in P(L(\mathbf{G}))$. Then V enables all events in $V_1(t)$, and forces all events in $V_2(t) = F(t)$. If one or more events in $F(t)$ is eligible after s, then $tick$ is preempted; if no event in $F(t)$ is eligible after s, then $tick$ is enabled. In comparison, the implementation of the supervisor V in Theorem 1 is simpler inasmuch as no explicit $F(t)$ is needed for $tick$ preemption; indeed, V directly decides to enable or disable $tick$, and controllability ensures the availability of forcible events for the disabling/preempting action.

B. Relative Weak Observability of TDES

Fixing a sublanguage $C \subseteq L_m(\mathbf{G})$, we introduce *timed relative weak observability* which sets C to be the ambient language (as is done in Definition 1 for relative observability). The key idea here is to distinguish different "control patterns" for $tick$ preemption in each set of lookalike strings; we do so by imposing on each such set a special *equivalence relation*. The equivalence classes of this equivalence relation have mutually disjoint subsets of forcible events, so that in each equivalence class $tick$ preemption may be carried out independently.

Let $P : \Sigma^* \to \Sigma_o^*$ and $s \in L(\mathbf{G})$. Write $[s] := \{s' \in L(\mathbf{G}) | Ps' = Ps\}$ for the set of lookalike strings to s in $L(\mathbf{G})$. Define a binary relation $\equiv$ on $[s]$ as follows: for all $s, s' \in [s]$, $s \equiv s'$ if either (i) $E_{L(\mathbf{G})}(s) \cap E_{L(\mathbf{G})}(s') \cap \Sigma_{\text{for}} \neq \emptyset$ or (ii) there exist $s_1, \ldots, s_k \in [s]$, $k \geq 1$, such that

$$\begin{aligned} &E_{L(\mathbf{G})}(s) \cap E_{L(\mathbf{G})}(s_1) \cap \Sigma_{\text{for}} \neq \emptyset \\ &\qquad \vdots \\ &E_{L(\mathbf{G})}(s_k) \cap E_{L(\mathbf{G})}(s') \cap \Sigma_{\text{for}} \neq \emptyset. \end{aligned} \tag{14}$$

In words, two strings $s, s' \in [s]$ satisfy $s \equiv s'$ if either (i) they are followed by some common forcible events that are eligible in $L(\mathbf{G})$, or (ii) there is a finite chain of strings in $[s]$ that "connects" s to s' through some common forcible events that are eligible in $L(\mathbf{G})$. This implies that for $s, s' \in [s]$, $s \equiv s'$ is false if and only if for every $s'' \in [s]$ with $s'' \equiv s'$ there holds $E_{L(\mathbf{G})}(s) \cap E_{L(\mathbf{G})}(s'') \cap \Sigma_{\text{for}} = \emptyset$. It is easily verified that $\equiv$ is reflexive, symmetric, and transitive, and thus an equivalence relation on $[s]$.

Definition 2: Let $K \subseteq C \subseteq L_m(\mathbf{G})$. We say that K is *relatively weakly observable* with respect to C, $\mathbf{G}$, and P, or simply *weakly C-observable*, if the following two conditions hold:

(1) K is C-observable with respect to $\Sigma_{\rm hib}$, i.e., for every pair of strings $s, s' \in \Sigma^*$ with $Ps = Ps'$ there holds

$$(\forall\, \sigma \in \Sigma_{\rm hib})\ s\sigma \in \overline{K}, s' \in \overline{C}, s'\sigma \in L(\mathbf{G}) \Rightarrow s'\sigma \in \overline{K}.$$

(2) For every pair of strings $s, s' \in \Sigma^*$ with $Ps = Ps'$, there holds

$$s.tick \in \overline{K}, s' \in \overline{C}, s'.tick \in L(\mathbf{G}), s \equiv s' \Rightarrow s'.tick \in \overline{K}.$$

The first condition above is the relative observability of K with respect to $\Sigma_{\rm hib}$. The second condition deals with the event $tick$: two lookalike strings $s, s' \in \overline{C}$ which satisfy $s \equiv s'$ are required to have identical one-step continuations of $tick$, if allowed in $L(\mathbf{G})$, with respect to membership in $\overline{K}$. This is weaker than relative observability with respect to $tick$, inasmuch as the requirement is imposed only on lookalike strings satisfying $s \equiv s'$. Therefore, the following result is immediate.

Proposition 4: If $K \subseteq C$ is C-observable, then K is also weakly C-observable.

As a corollary of Propositions 3 and 4, relative weak observability is weaker than normality. Next, we show that relative weak observability is stronger than weak observability.

Proposition 5: If $K \subseteq C$ is weakly C-observable and controllable, then K is also weakly observable.

Proof: First, since K is weakly C-observable, it is C-observable with respect to $\Sigma_{\rm hib}$; and by Proposition 1, K is observable with respect to $\Sigma_{\rm hib}$. Thus the first condition of weak observability is satisfied.

Now let $t \in P(\overline{K})$, and

$$F(t) = \bigcup \{E_K(s) \cap \Sigma_{\rm for} | s \in \overline{C} \cap P^{-1}(t), \\ (\exists s' \in [s])(s' \equiv s \,\&\, s'.tick \notin \overline{K})\}.$$

Moveover let $s_1 \in \overline{K} \cap P^{-1}(t)$ with $s_1.tick \in L(\mathbf{G})$. Then $s_1 \in \overline{C} \cap P^{-1}(t)$. Suppose that $s_1.tick \in \overline{K}$; it follows from K being weakly C-observable that for every $s_1' \in [s_1]$ with $s_1 \equiv s_1'$, $s_1' \in \overline{C}$, and $s_1'.tick \in L(\mathbf{G})$, there holds $s_1'.tick \in \overline{K}$. This implies that $E_{L(\mathbf{G})}(s_1) \cap F(t) = \emptyset$ owing to the definition of the equivalence relation $\equiv$.

Conversely, suppose that $s_1.tick \notin \overline{K}$. By controllability of K we have $E_K(s_1) \cap \Sigma_{\rm for} \neq \emptyset$. Let $\sigma \in E_K(s_1) \cap \Sigma_{\rm for}$; then $\sigma \in E_{L(\mathbf{G})}(s_1)$ and also $\sigma \in F(t)$. Hence $E_{L(\mathbf{G})}(s_1) \cap F(t) \neq \emptyset$. We have thus proved

$$tick \in E_K(s_1) \Leftrightarrow E_{L(\mathbf{G})}(s_1) \cap F(t) = \emptyset.$$

Therefore, the second condition of weak observability is satisfied. □

The reverse statement of Proposition 5 need not be true. An example is provided in Fig. 2, which displays a weakly observable language that is not relatively weakly observable because of violation of the second condition of Definition 2. (Since the first condition of Definition 2 is identical to that of relative observability, an example of violating the first condition may be found in [13].)

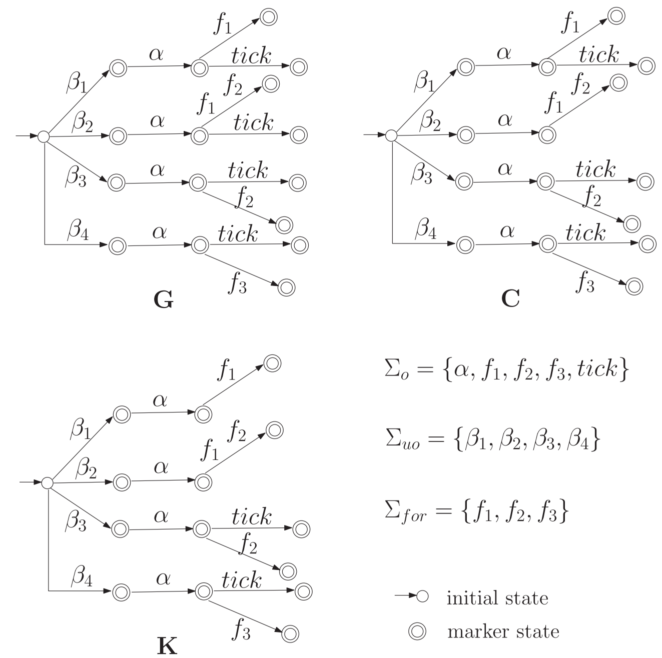


Fig. 2. $L_m(\mathbf{K})$ is weakly observable but not relatively weakly observable. The first condition of both definitions holds. For the second condition of weak observability, let $F(\alpha) = \{f_1\}$; then (13) holds, and hence $L_m(\mathbf{K})$ is weakly observable. On the other hand, let $s_1 = \beta_1\alpha$ and $s_3 = \beta_3\alpha$; then $Ps_1 = Ps_3$ and $s_1 \equiv s_3$. The equivalence $s_1 \equiv s_3$ holds because there is $s_2 = \beta_2\alpha \in [s_1]$ such that $E_{L(\mathbf{G})}(s_1) \cap E_{L(\mathbf{G})}(s_2) \cap \Sigma_{\rm for} = \{f_1\}$ and $E_{L(\mathbf{G})}(s_2) \cap E_{L(\mathbf{G})}(s_3) \cap \Sigma_{\rm for} = \{f_2\}$. The second condition of relative weak observability does not hold, however, for $s_1.tick \notin L(\mathbf{K})$ and $s_3.tick \in L(\mathbf{K})$. (Notation: we will use the same initial and marker state notation in subsequent figures.).

As with relative observability, the fixed ambient language C, as well as the equivalence relation $\equiv$, renders relative weak observability algebraically well-behaved: an arbitrary union of relatively weakly observable languages is again relatively weakly observable.

Proposition 6: Let $K_\alpha \subseteq C$, $\alpha \in \mathcal{A}$ (some index set), be weakly C-observable. Then $K = \bigcup\{K_\alpha \mid \alpha \in \mathcal{A}\}$ is also weakly C-observable.

Proof: First, by Proposition 2, K is C-observable with respect to $\Sigma_{\rm hib}$. Next, let $s, s' \in \Sigma^*$, $Ps = Ps'$, $s \equiv s'$, $s.tick \in \overline{K}$, $s' \in \overline{C}$, and $s'.tick \in L(\mathbf{G})$; it will be shown that $s'.tick \in \overline{K}$. Since $\overline{K} = \overline{\bigcup_{\alpha\in\mathcal{A}} K_\alpha} = \bigcup_{\alpha\in\mathcal{A}} \overline{K_\alpha}$, there exists $\alpha' \in \mathcal{A}$ such that $s.tick \in \overline{K_{\alpha'}}$. But $K_{\alpha'}$ is weakly C-observable, which yields $s'.tick \in \overline{K_{\alpha'}}$. Hence $s'.tick \in \bigcup_{\alpha\in\mathcal{A}} \overline{K_\alpha} = \overline{K}$. □

Whether or not $K \subseteq C$ is weakly C-observable, write

$$\mathcal{WO}(K, C) := \{K' \subseteq K \mid K' \text{ is weakly } C\text{-observable}\} \quad (15)$$

for the family of weakly C-observable sublanguages of K. Note that the empty language $\emptyset$ is trivially weakly C-observable, thus a member of $\mathcal{WO}(K, C)$. By Proposition 6, moreover, $\mathcal{WO}(K, C)$ has a unique supremal element $\sup \mathcal{WO}(K, C)$ given by

$$\sup \mathcal{WO}(K, C) := \bigcup\{K' \mid K' \in \mathcal{WO}(K, C)\}. \quad (16)$$

This is the supremal weakly C-observable sublanguage of K. In the following, we present an algorithm to compute $\sup \mathcal{WO}(K, C)$.

As noted immediately above Proposition 4, the only difference between relative weak observability and relative observability is the treatment of the event $tick$: in the former, essentially, $tick$ must be treated independently for lookalike strings that do *not* belong to the same equivalence class of $\equiv$. Thus our proposal to compute the supremal relatively weakly observable sublanguage of a language K is as follows: (1) identify equivalence classes of $\equiv$, and relabel $tick$ using distinct event labels $tick_1, tick_2, \ldots$ for distinct equivalent classes; (2) apply the algorithm in [13] to compute the supremal relatively observable sublanguage of K; and finally (3) relabel $tick_1, tick_2, \ldots$ back to $tick$.

Let $\mathbf{G} = (Q, \Sigma, \delta, q_0, Q_m)$, $\mathbf{C}$, and $\mathbf{K}$ be finite-state (trim) TDES [as in (2)] with marked languages $L_m(\mathbf{G})$, C, and K, respectively.

Algorithm 1: (computing the supremal relatively weakly observable sublanguage) Input $\mathbf{G}$, $\mathbf{C}$, $\mathbf{K}$, and $P : \Sigma^* \to \Sigma_o^*$.

---

1. For each $t \in P(L(\mathbf{G}))$, use the subset construction technique (e.g. [4, Section 2.5], [12]) to find the subset

$$Q(t) := \{q \in Q | (\exists s \in P^{-1}(t))\delta(q_0, s) = q\}.$$

For each $q \in Q(t)$, write $E_{L(\mathbf{G})}(q) := \{\sigma \in \Sigma | \delta(q, \sigma)!\}$. Then for each pair $(q, q') \in Q(t) \times Q(t)$, $q \equiv q'$ if either (i) $E_{L(\mathbf{G})}(q) \cap E_{L(\mathbf{G})}(q') \cap \Sigma_{\text{for}} \neq \emptyset$ or (ii) there exist $q_1, \ldots, q_k \in Q(t)$, $k \geq 1$, such that

$$E_{L(\mathbf{G})}(q) \cap E_{L(\mathbf{G})}(q_1) \cap \Sigma_{\text{for}} \neq \emptyset$$

$$\vdots$$

$$E_{L(\mathbf{G})}(q_k) \cap E_{L(\mathbf{G})}(q') \cap \Sigma_{\text{for}} \neq \emptyset.$$

Thus, for each $Q(t)$ we identify the equivalence classes of $\equiv$, say $Q_1(t), Q_2(t), \ldots$. For $tick$ defined at some state in $Q_i(t)$, $i = 1, 2, \ldots$, relabel it by $tick_i$. Do the corresponding relabeling in $\mathbf{C}$ and $\mathbf{K}$, and denote the relabeled generators by $\mathbf{G}'$, $\mathbf{C}'$, and $\mathbf{K}'$.

2. Apply the algorithm in [13] (reviewed in Appendix) with inputs $\mathbf{G}'$, $\mathbf{C}'$, and $\mathbf{K}'$, to compute $\mathbf{K}'_{\text{sup}}$, where $L_m(\mathbf{K}'_{\text{sup}})$ is the supremal C-observable sublanguage of K.

3. Relabel the events $tick_i$ in $\mathbf{K}'_{\text{sup}}$ by $tick$, and denote the result by $\mathbf{K}^W_{\text{sup}}$. Output $L_m(\mathbf{K}^W_{\text{sup}})$.

It follows easily from the preceding discussion that $L_m(\mathbf{K}^W_{\text{sup}})$ is the supremal weakly C-observable sublanguage of K. Also note that Algorithm 1 terminates in finite steps and has double-exponential complexity in the state size (say n) of $\mathbf{K}$. Specifically, Step 1 of Algorithm 1 has worst-case complexity $O(2^{2|Q|})$ due to subset construction and identification of the equivalence relation $\equiv$; Step 2 applies the algorithm in [13] which has worst-case complexity $O(2^{(2^n+1)|Q|})$. Overall, the complexity of Algorithm 1 is $O(2^{(2^n+1)|Q|})$.

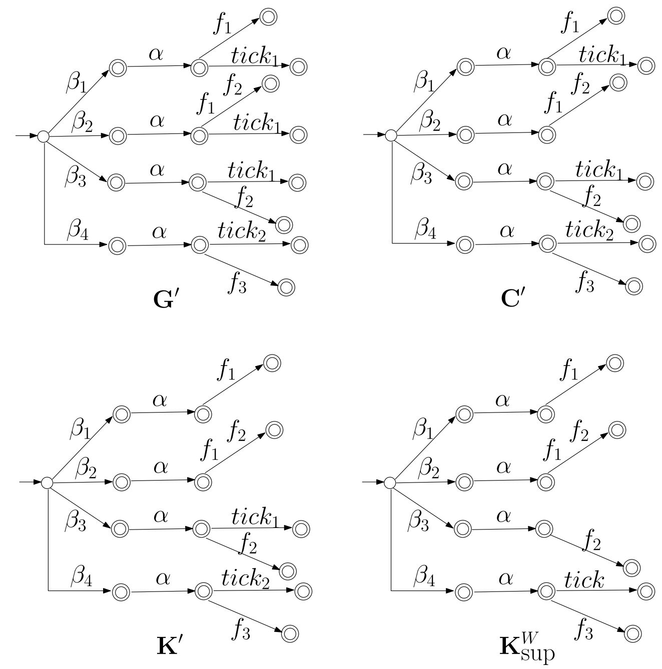


Fig. 3. $L_m(\mathbf{K}^W_{\text{sup}})$ is the supremal weakly $L_m(\mathbf{C})$-observable sublanguage of $L_m(\mathbf{K})$. In Step 1 of Algorithm 2, for $\alpha \in P(L(\mathbf{G}))$, we identify two equivalence classes of $\equiv$ on $Q(\alpha)$: $Q_1(\alpha) = \{\delta(q_0, \beta_1\alpha), \delta(q_0, \beta_2\alpha), \delta(q_0, \beta_3\alpha)\}$, $Q_2(\alpha) = \{\delta(q_0, \beta_4\alpha)\}$. Thus we relabel $tick$ by $tick_1$ for $Q_1(\alpha)$ and $tick_2$ for $Q_2(\alpha)$. Similarly, $tick$ is relabeled in $\mathbf{C}$ and $\mathbf{K}$. Then in Step 2, the algorithm in [13] removes $tick_1$ after $\beta_3\alpha$ in $\mathbf{K}'$. Finally, in Step 3, $tick_2$ after $\beta_4\alpha$ is relabeled back to $tick$, thereby yielding $\mathbf{K}^W_{\text{sup}}$.

As an illustration of Algorithm 1, consider again the example in Fig. 2. We apply Algorithm 1 to compute the supremal weakly $L_m(\mathbf{C})$-observable sublanguage of $L_m(\mathbf{K})$, as displayed in Fig. 3. Note that the resulting $L_m(\mathbf{K}^W_{\text{sup}})$ is weakly $L_m(\mathbf{C})$-observable but *not* $L(\mathbf{C})$-observable, for the latter requires the further removal of $tick$ after $\beta_4\alpha$.

Let $K \subseteq L_m(\mathbf{G})$ be a nonempty specification language, and let the ambient language $C = K$. Since weak K-observability, controllability, and $L_m(\mathbf{G})$-closedness are all closed under set union, there exists a unique supremal sublanguage of K that satisfies these three properties. Denote this supremal sublanguage by K^{WO}_{sup}; according to Proposition 5, K^{WO}_{sup} is weakly observable, controllable, and $L_m(\mathbf{G})$-closed. Therefore, by Theorem 2, there exists a nonblocking supervisor V such that $L_m(V/\mathbf{G}) = K^{WO}_{\text{sup}}$. In Section V-C we present an algorithm to compute K^{WO}_{sup}.

Remark 2 (tradeoff between timed relative observability and relative weak observability): We have derived two observability concepts for timed supervisory control under partial observation. Timed relative observability is conceptually simpler (since its requirement is imposed only on lookalike strings), allows easier implementation (see Remark 1), but the resulting $tick$-preemption behavior is generally more restrictive. On the other hand, timed relative weak observability requires extra information about the equivalence relation $\equiv$ on lookalike strings. The identification of $\equiv$ is done in Step 1 of Algorithm 1, which has worst-case complexity $O(2^{2|Q|})$; this computation is the price for achieving generally more permissive $tick$ preemption behavior. The decision as to which observability concept to use therefore depends on how much extra information is needed to achieve the corresponding behavior improvement; in practice the latter will be case-dependent. Nevertheless, since we have algorithms for both observability concepts, our suggestion is as follows. First compute the supremal relatively observable sublanguage K_1 (of a given specification language K); if the $tick$ preemption behavior of K_1 is 'satisfactory', then use K_1. Otherwise, compute the supremal relatively weakly observable sublanguage K_2 of K; comparing K_2 with K_1, if the improvement of $tick$ preemption behavior is 'significant', then use K_2.

V. Decentralized Supervisory Control of TDES With Partial Observation

We move on to consider decentralized supervisory control of TDES, where the plant is controlled by multiple decentralized supervisors $i \in \mathcal{I}$ ($\mathcal{I}$ is some finite index set). We shall propose timed relative coobservability and timed relative weak coobservability, as extensions of their centralized counterparts. Both properties are preserved under set union, and the respective supremal sublanguages exist.

A. Relative Coobservability of TDES

Let $\Sigma_{o,i} \subseteq \Sigma$ be the observable event set of the decentralized supervisor $i \in \mathcal{I}$, and $P_i : \Sigma^* \to \Sigma_{o,i}^*$ be the corresponding natural projection. Also let $\Sigma_{\text{hib},i} \subseteq \Sigma_{\text{rem}}$ and $\Sigma_{\text{for},i} \subseteq \Sigma_{\text{act}}$; then the decentralized supervisor $i \in \mathcal{I}$ disables events only in $\Sigma_{\text{hib},i}$, and uses forcible events only in $\Sigma_{\text{for},i}$ to preempt $tick$. Let the controllable event set be $\Sigma_{c,i} := \Sigma_{\text{hib},i} \dot{\cup} \{tick\}$, $i \in \mathcal{I}$; define for a controllable event σ the index set $\mathcal{I}_c(\sigma) := \{i \in \mathcal{I} | \sigma \in \Sigma_{c,i}\}$, and for a forcible σ the set $\mathcal{I}_f(\sigma) := \{i \in \mathcal{I} | \sigma \in \Sigma_{\text{for},i}\}$. Since $tick \in \Sigma_{c,i}$ for all $i \in \mathcal{I}$, there holds $\mathcal{I}_c(tick) = \mathcal{I}$.

The fundamental concept in untimed decentralized supervision is *coobservability* [6], [15], which is easily generalized to the TDES case as follows. Let $K \subseteq L_m(\mathbf{G})$, and $\Sigma_c := \cup_{i \in \mathcal{I}} \Sigma_{c,i}$. We say that K is *coobservable* (with respect to $\mathbf{G}$ and P_i, $i \in \mathcal{I}$) if for every $s \in \overline{K}$ and every $\sigma \in \Sigma_c$ with $s\sigma \in L(\mathbf{G}) \setminus \overline{K}$ there holds

$$(\exists i \in \mathcal{I}_c(\sigma))(\forall\, s' \in \overline{K}) P_i s = P_i s',\ s'\sigma \in L(\mathbf{G}) \Rightarrow s'\sigma \in L(\mathbf{G}) \setminus \overline{K}. \quad (17)$$

Coobservability means that the decision to remove σ after string s must be ratified by at least one decentralized supervisor that owns σ working through its local observation channel. Other variations of coobservability [16], [17] may be similarly extended to the TDES case. Like its untimed counterpart, timed coobservability is not closed under set union, and consequently the supremal coobservable sublanguage of a given language need not exist. This fact motivates us to propose relative coobservability; fix a sublanguage $C \subseteq L_m(\mathbf{G})$ and set C to be the ambient language.

Definition 3: Let $K \subseteq C \subseteq L_m(\mathbf{G})$. We say that K is *relatively coobservable* (with respect to C, $\mathbf{G}$, and P_i, $i \in \mathcal{I}$), or simply *C-coobservable*, if for each $i \in \mathcal{I}$, K is C-observable, i.e., for every pair of strings $s, s' \in \Sigma^*$ with $P_i s = P_i s'$, there holds

$$(\forall\, \sigma \in \Sigma_{c,i})\ s\sigma \in \overline{K}, s' \in \overline{C}, s'\sigma \in L(\mathbf{G}) \Rightarrow s'\sigma \in \overline{K}. \quad (18)$$

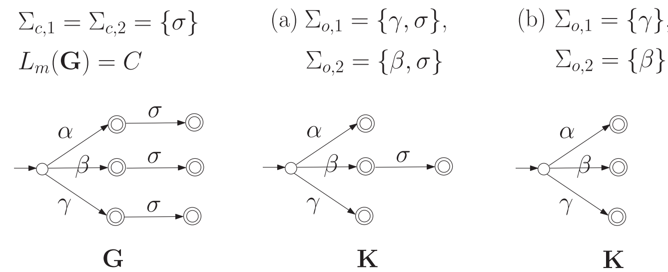


Fig. 4. Case (a), $L_m(\mathbf{K})$ is decomposable but not C-coobservable. First, it is easily verified that $P_1^{-1}P_1(\overline{K}) \cap P_2^{-1}P_2(\overline{K}) \cap L(\mathbf{G}) = \overline{K}$ and hence $L_m(\mathbf{K})$ is decomposable. Then let $s = \beta$, $s' = \alpha$; thus $P_1(s) = P_1(s') = \epsilon$, $s\sigma \in L(\mathbf{K})$, $s' \in \overline{C}$, $s'\sigma \in L(\mathbf{G})$, but $s'\sigma \notin L(\mathbf{K})$. Therefore, $L_m(\mathbf{K})$ is not C-observable with respect to P_1 and consequently not C-coobservable. Case (b), $L_m(\mathbf{K})$ is C-coobservable but not decomposable. A straightforward calculation yields that $P_1^{-1}P_1(\overline{K}) \cap P_2^{-1}P_2(\overline{K}) \cap L(\mathbf{G}) = L(\mathbf{G}) \supsetneqq \overline{K}$ and hence $L_m(\mathbf{K})$ is not decomposable. On the other hand, since the shared controllable event σ is removed after all strings α, β, and γ, it is easily checked that $L_m(\mathbf{K})$ is C-observable with respect to both P_1 and P_2 and therefore C-coobservable.

The above timed relative coobservability is an extension of the untimed counterpart studied in [19], by accounting for the special event $tick$ which may be preempted by a decentralized supervisor $i \in \mathcal{I}$. This is in contrast with direct disablement of the decentralized supervisor's prohibitible events in $\Sigma_{\text{hib},i}$. Indeed, $tick$ is a common event that each decentralized supervisor must deal with using its local subset of forcible events.

According to the definition, timed relative coobservability is $\mathcal{I}$-fold timed relative observability. It is proved, similar to the untimed case [19], that timed relative coobservability is stronger than timed coobservability (and any of its variations), but enjoys the property that it is closed under set union. Therefore, there exists the supremal relatively coobservable sublanguage of a given language. This supremal sublanguage may be computed by an algorithm presented in [19].

Timed relative coobservability is on the other hand weaker than *conormality* (see a proof in [19]). A language $K \subseteq L_m(\mathbf{G})$ is conormal [15] if

$$\bigcup_{\forall\, i \in \mathcal{I}} P_i^{-1} P_i(\overline{K}) \cap L(\mathbf{G}) = \overline{K}. \quad (19)$$

Conormality is an extension of normality to the decentralized case. Conormality may be overly restrictive because it requires that for each decentralized supervisor $i \in \mathcal{I}$, only observable (under P_i), prohibitible events may be disabled. Relative coobservability, by contrast, does not impose this restriction, i.e., control may be exercised by each decentralized supervisor over its unobservable prohibitible events.

Another concept related to (and weaker than) conormality is *decomposability* [15]: A language $K \subseteq L_m(\mathbf{G})$ is decomposable if

$$\bigcap_{\forall\, i \in \mathcal{I}} P_i^{-1} P_i(\overline{K}) \cap L(\mathbf{G}) = \overline{K}.$$

In general, decomposability and relative coobservability do not imply each other; this is illustrated by the example in Fig. 4. Decomposability, like conormality, does not allow disabling any unobservable prohibitible events, which is nevertheless permitted by relative coobservability. Moreover, decomposability

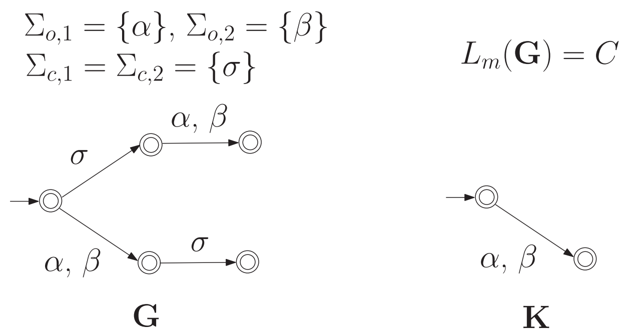


Fig. 5. $L_m(\mathbf{K})$ is controllable and observable with respect to both P_1 and P_2; thus it is controllable and coobservable. On the other hand, $L_m(\mathbf{K})$ is neither C-observable with respect to P_1 nor P_2; the supremal controllable and C-coobservable sublanguage of $L_m(\mathbf{K})$ is the empty language.

is not closed under union, and consequently there need not exist the supremal decomposable sublanguage of a given language.

We also note that a *weak conormality* concept is studied in [17]. As pointed out in [19], relative coobservability is generally weaker than weak conormality.

Now for control, let $K \subseteq L_m(\mathbf{G})$ be a nonempty specification language and fix the ambient language $C = K$. Since timed K-coobservability, controllability, and $L_m(\mathbf{G})$-closedness are all closed under set union, there exists a unique supremal sublanguage of K that satisfies these three properties. Denote this supremal sublanguage by K_{sup}^{CO}; we present an algorithm in Section V-C to compute K_{sup}^{CO}.

We note that for a prefix-closed language K, K_{sup}^{CO} may be empty even when there is a nonempty controllable and coobservable sublanguage of K. See Fig. 5 for an example.

B. Relative Weak Coobservability of TDES

To achieve more permissive controlled behavior than is allowed by timed coobservability, in [18] the authors studied the following conditions by exploiting choices of local forcible events of decentralized supervisors to preempt the $tick$ event. Again let $\Sigma_{o,i} \subseteq \Sigma$, $\Sigma_{\text{hib},i} \subseteq \Sigma_{\text{rem}}$, and $\Sigma_{\text{for},i} \subseteq \Sigma_{\text{act}}$ be the observable, prohibitible, and forcible event sets of the decentralized supervisor $i \in \mathcal{I}$. Also let $\Sigma_{\text{hib}} := \cup_{i \in \mathcal{I}} \Sigma_{\text{hib},i}$ and $\Sigma_{\text{for}} := \cup_{i \in \mathcal{I}} \Sigma_{\text{for},i}$. For a language $K \subseteq L_m(\mathbf{G})$, the two conditions in [18] are the following.

(1) For each $s \in \overline{K}$ and each $\sigma \in \Sigma_{\text{hib}}$ with $s\sigma \in L(\mathbf{G}) \setminus \overline{K}$, there holds

$$(\exists i \in \mathcal{I}_c(\sigma))(\forall s' \in \overline{K}) P_i s = P_i s',\ s'\sigma \in L(\mathbf{G}) \Rightarrow s'\sigma \in L(\mathbf{G}) \setminus \overline{K}. \quad (20)$$

(2) For each $i \in \mathcal{I}$ and each $t \in P_i(\overline{K})$, there exists a subset $F_i(t) \subseteq \Sigma_{\text{for},i}$ such that for each $s \in \overline{K}$ with $tick \in E_{L(\mathbf{G})}(s)$, there holds

$$tick \in E_K(s) \Leftrightarrow (\forall \sigma \in E_K(s) \cap \Sigma_{\text{for}})(\exists j \in \mathcal{I}_f(\sigma)) \sigma \notin F_j(P_j s). \quad (21)$$

These two conditions extend those of weak observability (see Section IV) to the decentralized setup, and for this reason we call the above *weak coobservability* of K. Other variations of weak coobservability are also presented in [18] and [24].

Weak coobservability (or any of its variations) is, however, not closed under set union and consequently the supremal weakly coobservable sublanguage of a given language need not exist. This problem motivates us to propose relative weak coobservability; fix a sublanguage $C \subseteq L_m(\mathbf{G})$ and set C to be the ambient language.

Definition 4: Let $K \subseteq C \subseteq L_m(\mathbf{G})$. We say that K is *relatively weakly coobservable* (with respect to C, $\mathbf{G}$, and P_i, $i \in \mathcal{I}$), or simply weakly *C-coobservable*, if for each $i \in \mathcal{I}$, K is weakly C-observable, i.e., the following two conditions hold:

(1) for every pair of strings $s, s' \in \Sigma^*$ with $P_i s = P_i s'$ there holds

$$(\forall \sigma \in \Sigma_{\text{hib}})\ s\sigma \in \overline{K}, s' \in \overline{C}, s'\sigma \in L(\mathbf{G}) \Rightarrow s'\sigma \in \overline{K}.$$

(2) For every pair of strings $s, s' \in \Sigma^*$ with $P_i s = P_i s'$ there holds

$$s.tick \in \overline{K}, s' \in \overline{C}, s'.tick \in L(\mathbf{G}), s \equiv s' \Rightarrow s'.tick \in \overline{K}$$

where the equivalence relation $\equiv$ is defined in (14).

Timed relative weak coobservability is $\mathcal{I}$-fold relative weak observability, and therefore weaker than $\mathcal{I}$-fold relative observability (Proposition 4), i.e., relative coobservability. In turn, relative weak coobservability is weaker than conormality. On the other hand, relative weak coobservability is stronger than weak coobservability [18], as asserted in the following.

Proposition 7: If $K \subseteq C$ is weakly C-coobservable and controllable, then K is also weakly coobservable.

Proof: First, since K is weakly C-coobservable, it is C-coobservable with respect to Σ_{hib} by condition (1) of Definition 4; thus in turn K is coobservable with respect to Σ_{hib}, i.e., the first condition (20) of weak coobservability holds. Now let $i \in \mathcal{I}$, $t \in P_i(\overline{K})$, and

$$F_i(t) = \bigcup \{E_K(s) \cap \Sigma_{\text{for},i} | s \in \overline{C} \cap P_i^{-1}(t), (\exists s' \in P_i^{-1} P_i s)(s' \equiv s \ \&\ s'.tick \notin \overline{K})\}.$$

Moveover let $s_1 \in \overline{K}$ with $s_1.tick \in L(\mathbf{G})$. Then $s_1 \in \overline{C}$. Suppose that $s_1.tick \in \overline{K}$. Let $\sigma \in E_K(s_1) \cap \Sigma_{\text{for}}$; then there exists $j \in \mathcal{I}_f(\sigma)$ such that $\sigma \in E_K(s_1) \cap \Sigma_{for,j}$. It follows from K being weakly C-coobservable that for every $s_1' \in P_j^{-1} P_j s_1$ with $s_1 \equiv s_1'$, $s_1' \in \overline{C}$, and $s_1'.tick \in L(\mathbf{G})$, there holds $s_1'.tick \in \overline{K}$. This implies that $\sigma \notin F_j(P_j s_1)$ owing to the definition of the equivalence relation $\equiv$ in (14).

Conversely, suppose that $s_1.tick \notin \overline{K}$. By controllability of K we have $E_K(s_1) \cap \Sigma_{\text{for}} \neq \emptyset$. Let $\sigma \in E_K(s_1) \cap \Sigma_{\text{for}}$ and $j \in \mathcal{I}_f(\sigma)$; then $\sigma \in E_K(s_1) \cap \Sigma_{\text{for},i}$, and again by K being weakly C-coobservable we derive that $\sigma \in F_j(P_j s_1)$. Therefore the second condition (21) of weak coobservability holds, as required. □

Timed relative weak coobservability is closed under set union, i.e., if $K_\alpha \subseteq C \subseteq L_m(\mathbf{G})$, $\alpha \in \mathcal{A}$ (some index set), are weakly C-coobservable, then $K = \bigcup \{K_\alpha \mid \alpha \in \mathcal{A}\}$ is also weakly C-coobservable. Indeed, for each $i \in \mathcal{I}$, K_α is weakly C-observable; by Proposition 6, K is also weakly

C-observable. The latter holds for every $i \in \mathcal{I}$, and therefore K is weakly C-coobservable. Thus there exists the supremal relatively coobservable sublanguage of a given language.

Now for control, let $K \subseteq L_m(\mathbf{G})$ be a nonempty specification language and fix the ambient language $C = K$. Since timed weak K-coobservability, controllability, and $L_m(\mathbf{G})$-closedness are all closed under set union, there exists a unique supremal sublanguage of K that satisfies these three properties. Denote this supremal sublanguage by $K_{\sup}^{WCO}$; we present in Section V-C an algorithm to compute $K_{\sup}^{WCO}$.

C. Algorithm

The following algorithm computes the supremal relatively (weakly) coobservable (with ambient K), controllable, and $L_m(\mathbf{G})$-closed sublanguage. As will be seen, in a special case this algorithm computes the supremal relatively (weakly) observable, controllable, and $L_m(\mathbf{G})$-closed sublanguage. Let $\mathbf{G}$ and $\mathbf{K}$ be finite-state (trim) TDES [as in (2)] with marked languages $L_m(\mathbf{G})$ and K, respectively.

Algorithm 2 (computing the supremal relatively (respectively, weakly) coobservable, controllable, and $L_m(\mathbf{G})$-closed sublanguage) : Input $\mathbf{G}$, $\mathbf{K}$, and $P_i : \Sigma^* \to \Sigma_{o,i}^*$, $i \in \mathcal{I} := \{1, \ldots, N\}$.

1. Set $\mathbf{K}_0 = \mathbf{K}$.
2. For $j \geq 0$, apply the algorithm in [3] (reviewed in Appendix) with inputs $\mathbf{G}$ and $\mathbf{K}_j$ to obtain $\mathbf{H}_j$ such that $L_m(\mathbf{H}_j)$ is the supremal controllable and $L_m(\mathbf{G})$-closed sublanguage of $L_m(\mathbf{K}_j)$.
3. Compute $\mathbf{K}_{j+1} = \text{RCO}(\mathbf{G}, \mathbf{K}, \mathbf{H}_j, P_i)$. If $\mathbf{K}_{j+1} = \mathbf{K}_j$, then output $K_{\sup}^{CO} := L_m(\mathbf{K}_{j+1})$ (respectively, $K_{\sup}^{WCO} := L_m(\mathbf{K}_{j+1})$). Otherwise, advance j to $j+1$ and go to Step 2.

$\text{RCO}(\mathbf{G}, \mathbf{K}, \mathbf{H}_j, P_i)$

4. Set $\mathbf{M}_0 := \mathbf{H}_j$.
5. For $p \geq 0$, set $\mathbf{M}_{p,1} := \mathbf{M}_p$.
6. For $i \geq 1$, apply the algorithm in [13] (respectively, Algorithm 1) with inputs $\mathbf{G}$, $\mathbf{K}$, $\mathbf{M}_{p,i}$, and P_i to obtain $\mathbf{M}_{p,i+1}$ such that $L_m(\mathbf{M}_{p,i+1})$ is the supremal (respectively, weakly) $L(\mathbf{K})$-observable sublanguage of $L_m(\mathbf{M}_{p,i})$ with respect to P_i. Proceed until $\mathbf{M}_{p,N}$ is computed, and set it to be $\mathbf{M}_{p+1}$. If $\mathbf{M}_{p+1} = \mathbf{M}_p$, then return $\mathbf{M}_{p+1}$. Otherwise, advance p to $p+1$ and go to Step 5.

Proposition 8: The output $K_{\sup}^{CO}$ (respectively, $K_{\sup}^{WCO}$) of Algorithm 2 is the supremal relatively (respectively, weakly) coobservable (with ambient K), controllable, and $L_m(\mathbf{G})$-closed sublanguage of K.

Proof: We prove that $K_{\sup}^{CO}$ is the supremal relatively coobservable (with ambient $K = L_m(\mathbf{K})$), controllable, and $L_m(\mathbf{G})$-closed sublanguage of K. The conclusion for $K_{\sup}^{WCO}$ follows similarly.

First, the subroutine RCO (Steps 4–6) generates a sequence of sublanguages

$$L_m(\mathbf{M}_0) \supseteq L_m(\mathbf{M}_1) \supseteq L_m(\mathbf{M}_2) \supseteq \cdots$$

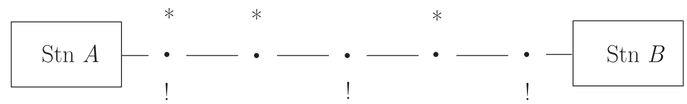


Fig. 6. Guideway: stations A and B are connected by a single one-way track from A to B. The track consists of 4 sections, with stoplights ($*$) and detectors (!) installed at various section junctions as displayed.

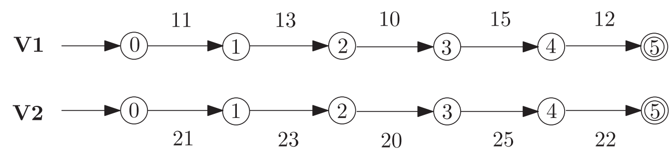


Fig. 7. Vehicle untimed DES models. Notation: a circle with $\rightarrow$ denotes the initial state, and a double circle denotes a marked state; this notation will be used henceforth.

From $L_m(\mathbf{M}_p)$ to $L_m(\mathbf{M}_{p+1})$ (for each $p \geq 0$), the algorithm in [13] is applied N times, one for each P_i. Since the algorithm in [13] is finitely convergent, so is the above sequence. When the sequence converges, i.e., $\mathbf{M}_{p+1} = \mathbf{M}_p$ for some p, $L_m(\mathbf{M}_{p+1})$ is the supremal $L(\mathbf{K})$-observable sublanguage for each P_i, $i \in \mathcal{I}$, and therefore is the supremal $L(\mathbf{K})$-coobservable sublanguage.

The main routine (Steps 1–3) generates a sequence of sublanguages

$$L_m(\mathbf{K}_0) \supseteq L_m(\mathbf{H}_0) \supseteq L_m(\mathbf{K}_1) \supseteq L_m(\mathbf{H}_1) \supseteq \cdots$$

Since the algorithm in [3] and the subroutine RCO are both finitely convergent, so is the above sequence. When the main routine converges, i.e., $\mathbf{K}_{j+1} = \mathbf{K}_j$ for some j, $K_{\sup}^{CO} := L_m(\mathbf{K}_{j+1})$ is the supremal $L(\mathbf{K})$-coobservable, controllable, and $L_m(\mathbf{G})$-closed sublanguage. □

Algorithm 2 terminates in finite steps, and has double-exponential complexity in the state size of $\mathbf{K}$ inasmuch as the algorithm in [13] (or Algorithm 1) is of this complexity.

Specialize Algorithm 2 to the case $\mathcal{I} = \{1\}$, and denote the output by $K_{\sup}^{O}$ (respectively, $K_{\sup}^{WO}$). The following result is immediate.

Corollary 1: For $\mathcal{I} = \{1\}$, the output $K_{\sup}^{O}$ (respectively, $K_{\sup}^{WO}$) of Algorithm 2 is the supremal relatively (respectively, weakly) observable (with ambient K), controllable, and $L_m(\mathbf{G})$-closed sublanguage of K.

VI. Guideway Example

We demonstrate Algorithm 2 in Section V-C and the concepts of (weak) relative (co)observability with a Guideway example under partial observation, adapted from [4, Section 6.6]. As displayed in Fig. 6, stations A and B on a Guideway are connected by a single one-way track from A to B. The track consists of 4 sections, with stoplights ($*$) and detectors (!) installed at various section junctions. Two vehicles, $\mathbf{V}_1$ and $\mathbf{V}_2$, use the Guideway simultaneously. Their untimed DES models are displayed in Fig. 7; $\mathbf{V}_i$, $i = 1, 2$, is at state 0 (station A), state j (while travelling in section $j = 1, \ldots, 4$), or state 5 (station B).

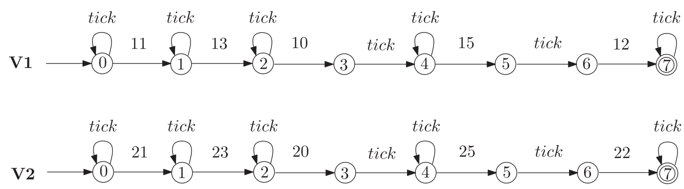


Fig. 8. Vehicle TDES models.

Assign lower and upper time bounds to each event as follows: $(i = 1, 2)$

	$i0$	$i1$	$i2$	$i3$	$i5$
l_σ	0	0	1	0	1
u_σ	∞	∞	1	∞	∞

Thus, prospective events are $i2$, and remote events are $i0$, $i1, i3, i5$. As in (2), the TDES models of $\mathbf{V}_1$ and $\mathbf{V}_2$ are generated; see Fig. 8. Here, state 4 (respectively, state 6) of $\mathbf{V}_i$ means that the vehicle has left Section III (respectively, Section IV) but not yet reached Section IV (respectively, station B). The plant $\mathbf{G}$ to be controlled is then $\mathbf{G} = \mathbf{V}_1||\mathbf{V}_2$, the *synchronous product* (e.g., [4]) of $\mathbf{V}_1$ and $\mathbf{V}_2$.[1]

To prevent collision, control of the stoplights must ensure that $\mathbf{V}_1$ and $\mathbf{V}_2$ never travel on the same section of track simultaneously, i.e., ensure mutual exclusion of the state pairs (j, j), $j = 1, \ldots, 6$. Let $\mathbf{K}$ be a generator enforcing this specification.

First, consider a centralized supervisory control problem under partial observation. Let the prohibitible events be $i1, i3, i5$, forcible events $i5$, and unobservable events $i3, i5$, $i = 1, 2$. The latter define a natural projection P.

Applying Algorithm 2 with inputs $\mathbf{G}$, $\mathbf{K}$, and P $(\mathcal{I} := \{1\})$ to compute the supremal relatively observable sublanguage, we obtain the generator displayed in Fig. 9. The resulting controlled behavior is verified to be $L_m(\mathbf{K})$-observable (thus also observable by Proposition 1), controllable, and $L_m(\mathbf{G})$-closed. Moreover, it is strictly larger than the supremal normal, controllable, and $L_m(\mathbf{G})$-closed sublanguage represented by the generator displayed in Fig. 10. The reason is as follows. Referring to Fig. 8, after a string $s \in (tick)^*.11.(tick)^*.13.(tick)^*.10$, $\mathbf{V}_1$ is at state 3 (track section 3) and $\mathbf{V}_2$ at 0 (station A). With relative observability, either $\mathbf{V}_2$ executes 21 (moving to state 1) or a $tick$ occurs (note that event 15, namely $\mathbf{V}_1$ moving to state 4, has lower bound 1). In the former case, event 23 is disabled after execution of 21 to ensure mutual exclusion at (3, 3) because event 20 is uncontrollable. With normality, however, event 23 cannot be disabled because it is unobservable; thus, 21 is disabled after the string s, and the only possibility is that a $tick$ occurs, following which $\mathbf{V}_1$ executes 15 (more $tick$ events may occur before 15). In fact, 21 is kept disabled until the observable event 12 occurs, i.e., $\mathbf{V}_1$ arrives at station B.

Next, apply Algorithm 2 with inputs $\mathbf{G}$, $\mathbf{K}$, and P $(\mathcal{I} := \{1\})$ to compute the supremal relatively weakly observable sublanguage; we obtain the generator displayed in Fig. 11. The resulting controlled behavior is verified to be weakly $L_m(\mathbf{K})$-observable, controllable, and $L_m(\mathbf{G})$-closed. Moreover, it is strictly larger than the supremal relatively observable, controllable, and $L_m(\mathbf{G})$-closed sublanguage represented by the generator in Fig. 9. The reason is as follows. After a string $s \in (tick)^*.11.(tick)^*.13.(tick)^*.10.21.(tick)^*.23.(tick)^*.20$, the $tick$ event is preempted by forcible event 15 to ensure mutual exclusion specification. But since 15 is unobservable, $tick$ after $s.15$ must also be removed to satisfy relative observability. This removal of $tick$ is avoided in the case of relative weak observability because there is no common forcible event defined after the lookalike strings s and $s.15$, and thus the respective $tick$ events are relabeled to be distinct events. Referring to Fig. 8 for the TDES models of the two vehicles, the more permissive controlled behavior in Fig. 11 allows one vehicle to arrive at track section 3 when the other has just vacated it and has not yet reached section 4.

Now let us consider a decentralized supervisory control problem described as follows. Suppose that the Guideway is to be controlled by two decentralized supervisors, with unobservable event sets $\Sigma_{uo,1} = \{13, 15, 23\}$, $\Sigma_{uo,2} = \{13, 23, 25\}$; these define the corresponding natural projections P_1, P_2. Since $\Sigma_{uo,1} \cap \Sigma_{uo,2} = \{13, 23\}$, no supervisor can observe events 13, 23. In addition let the prohibitible and forcible event sets be $\Sigma_{\text{hib},1} = \Sigma_{\text{for},1} = \{11, 13, 23, 15\}$, $\Sigma_{\text{hib},2} = \Sigma_{\text{for},2} = \{21, 13, 23, 25\}$; thus the shared prohibitible/forcible events are the unobservable 13, 23.

Applying Algorithm 2 with inputs $\mathbf{G}$, $\mathbf{K}$, and P_i $(i \in \mathcal{I} := \{1, 2\})$ to compute the supremal relatively coobservable sublanguage, we obtain the generator displayed in Fig. 12. The resulting controlled behavior is confirmed to be $L_m(\mathbf{K})$-coobservable, controllable, and $L_m(\mathbf{G})$-closed. Moreover, it is strictly larger than the supremal conormal, controllable, and $L_m(\mathbf{G})$-closed sublanguage, which is the same as the supremal normal counterpart and thus represented again by the generator displayed in Fig. 10. This is because, with conormality, the first (respectively, second) decentralized supervisor cannot disable its unobservable prohibitible events 13, 15, 23 (respectively, 13, 23, 25); by contrast, relative coobservability does not impose this constraint.

Finally we apply Algorithm 2 with inputs $\mathbf{G}$, $\mathbf{K}$, and P_i $(i \in \mathcal{I} := \{1, 2\})$ to compute the supremal relatively weakly coobservable sublanguage; the resulting generator is the same as the one displayed in Fig. 11. We see that the controlled behavior is strictly larger than the supremal $L_m(\mathbf{K})$-coobservable, controllable, and $L_m(\mathbf{G})$-closed sublanguage represented by the generator in Fig. 12. This is owing to the flexibility of suitably treating $tick$ as distinct events, so that the first (respectively, second) decentralized supervisor may use its unobservable forcible events 13, 15, 23 (respectively, 13, 23, 25) to preempt different $tick$s while satisfying relative weak coobservability.

VII. Conclusion

We have studied new observability concepts in monolithic and decentralized supervisory control of TDES under partial observation. In monolithic supervisory control, timed relative

[1] To compose two TDES, an operation called *composition* [4, Section 9.6] is used in general. In the special case where the two TDES have *disjoint* event sets except for $tick$ (as $\mathbf{V}_1$ and $\mathbf{V}_2$ in this example), it is known [4, Section 9.6] that composition is equivalent to synchronous product in the untimed case.

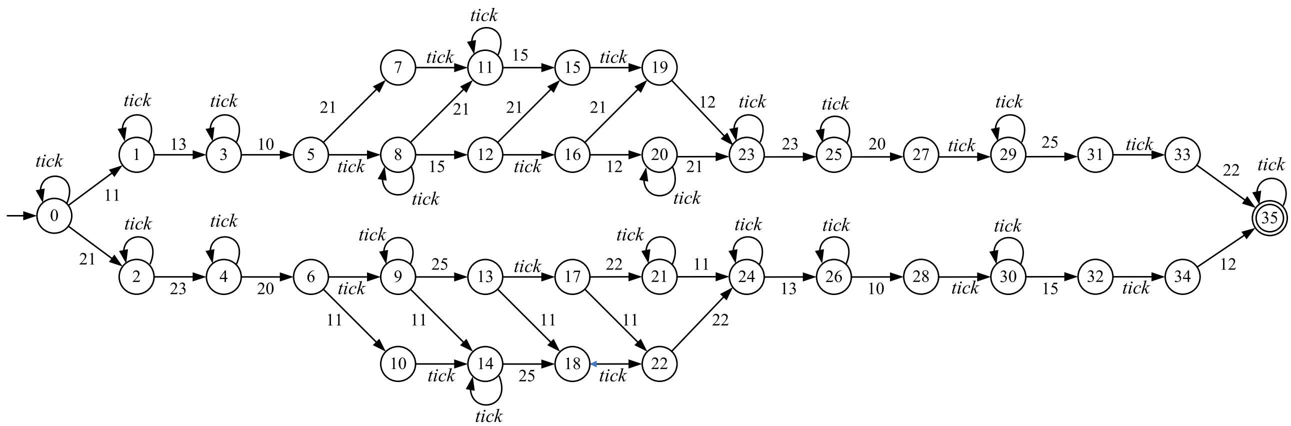


Fig. 9. Supremal timed relatively observable, controllable, and $L_m(\mathbf{G})$-closed sublanguage.

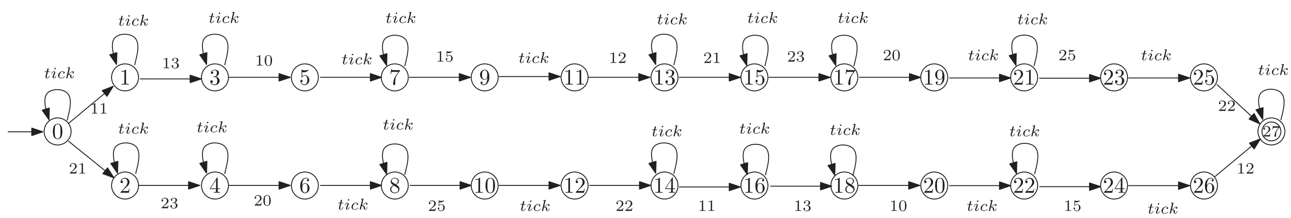


Fig. 10. Supremal (co)normal, controllable, and $L_m(\mathbf{G})$-closed sublanguage.

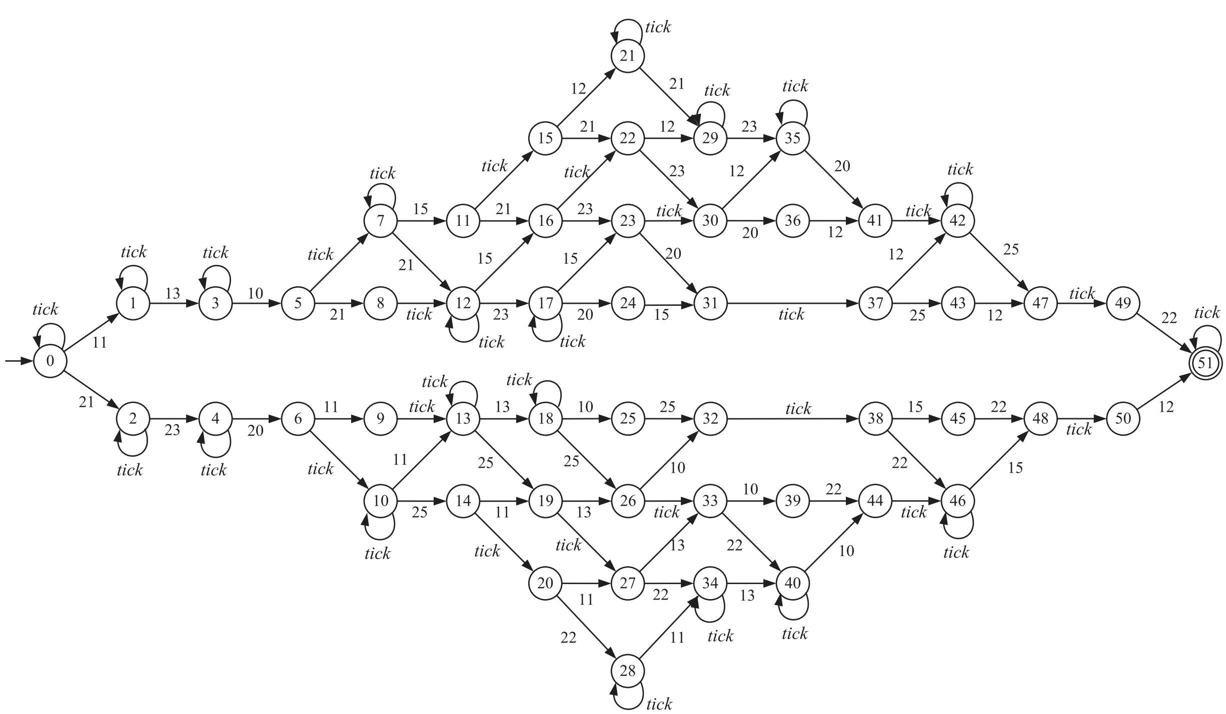


Fig. 11. Supremal timed relatively weakly observable, controllable, and $L_m(\mathbf{G})$-closed sublanguage.

observability and timed relative weak observability have been introduced, and proved to be closed under set union. In decentralized control, we have proposed timed relative coobservability and timed relative weak coobservability. These properties again have been shown to be closed under set union. Algorithms have been designed to compute the supremal sublanguages, which have been applied to synthesizing partially-observed monolithic and decentralized supervisory control for a Guideway example; the derived controlled behaviors have been compared and trade-offs discussed.

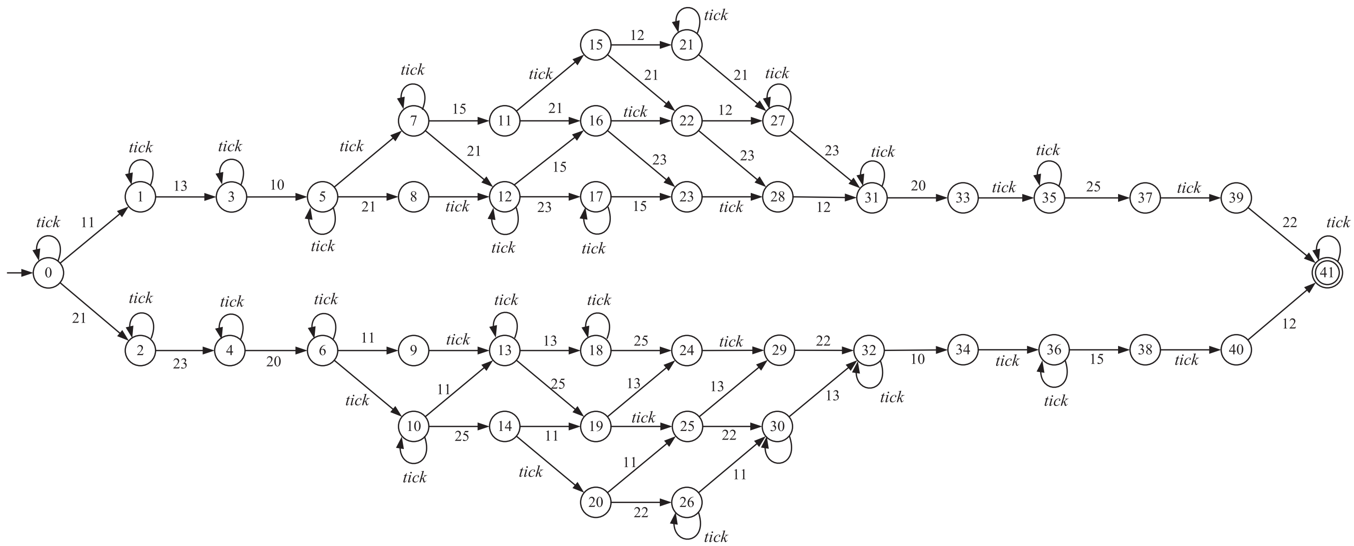


Fig. 12. Supremal timed relatively coobservable, controllable, and $L_m(\mathbf{G})$-closed sublanguage.

Finally, we note from [13] that although the designed algorithms have double-exponential complexity in general, if the involved natural projections satisfy the *natural observer* property, then the complexity of these algorithms is in fact polynomial. Alternatively, in future work we aim to develop efficient algorithms for online synthesis of timed monolithic/decentralized supervisors under partial observation. In addition, we are interested in combining the proposed observability concepts with *supervisor localization* [25] for partially-observed distributed control of TDES.

Appendix

First, we review the algorithm in [13] that computes the supremal relatively observable sublanguage (with ambient language $C \subseteq L_m(\mathbf{G})$) of a given language $K \subseteq L_m(\mathbf{G})$.

Algorithm in [13] : Input $\mathbf{G} = (Q, \Sigma, \delta, q_0, Q_m)$, $\mathbf{C} = (Y^C, \Sigma, \eta^C, y_0^C, Y_m^C)$, $\mathbf{K} = (Y, \Sigma, \eta, y_0, Y_m)$ (representing $L_m(\mathbf{G})$, C, K respectively), and $P : \Sigma^* \to \Sigma_o^*$.

1. Set $\mathbf{K}_0 := (Y_0, \Sigma, \eta_0, y_0, Y_{m,0}) = \mathbf{K}$.
2. For $i \geq 0$ let $\tilde{\mathbf{K}}_i := (\tilde{Y}_i, \Sigma, \tilde{\eta}_i, y_0, Y_{m,i})$, with $\tilde{Y}_i = Y_i \cup \{y_d\}$, the *dump state* $y_d \notin Y_i$, and $\tilde{\eta}_i(y_0, s) = \eta_i(y_0, s)$ if $s \in L(\mathbf{K}_i)$ and $\tilde{\eta}_i(y_0, s) = y_d$ otherwise. Then calculate

$$T_i(s) := \{(q, y) \in Q \times \tilde{Y}_i | (\exists s') Ps' = Ps \;\&\; q = \delta(q_0, s') \;\&\; y = \tilde{\eta}_i(y_0, s') \;\&\; \eta^C(y_0, s')!\}$$

and let $\mathcal{T}_i := \{T_i(s) | s \in \Sigma^*, |T_i(s)| \geq 2\}$.
3. For each $T \in \mathcal{T}_i$, check if the following two conditions are satisfied for all $(q, y), (q', y') \in T$:

(i) $(\forall \sigma \in \Sigma)\ \tilde{\eta}_i(y, \sigma) \neq y_d \;\&\; \delta(q', \sigma)! \Rightarrow \tilde{\eta}_i(y', \sigma) \neq y_d$
(ii) $q' \in Q_m \;\&\; y \in Y_{m,i} \Rightarrow y' \in Y_{m,i}$.;

If so, then output $\mathbf{K}_i$. Otherwise, let $R_i := \bigcup_{T \in \mathcal{T}_i} R_T$ and $M_i := \bigcup_{T \in \mathcal{T}_i} M_T$, where

$$R_T := \bigcup_{\sigma \in \Sigma} \{(y, \sigma, \eta_i(y, \sigma)) \mid \eta_i(y, \sigma)! \;\&\; (\exists s) T = T(s) \;\&\; (q, y) \in T \;\&\; (\exists (q', y') \in T)(\delta(q', \sigma)! \;\&\; \tilde{\eta}_i(y', \sigma) = y_d)\}$$

$$M_T := \{y \in Y_{m,i} \mid (\exists s) T = T(s) \;\&\; (q, y) \in T \;\&\; (\exists (q', y') \in T)(q' \in Q_m \;\&\; y' \notin Y_{m,i})\}.$$

Then set $\eta_i' := \eta_i - R_i$ and $Y_{m,i}' := Y_{m,i} - M_i$; let $\mathbf{K}_{i+1} := (Y_{i+1}, \Sigma, \eta_{i+1}, y_0, Y_{m,i+1}) = \text{trim}((Y_i, \Sigma, \eta_i', y_0, Y_{m,i}'))$, where trim$(\cdot)$ removes all non-reachable and non-coreachable states and corresponding transitions of the argument generator. Advance i to $i+1$, and go to Step 2.

Next, we review the algorithm in [3] which computes the supremal controllable and $L_m(\mathbf{G})$-closed sublanguage of a given language K.

Algorithm in [3] : Input $\mathbf{G} = (Q, \Sigma, \delta, q_0, Q_m)$ and $\mathbf{K} = (Y, \Sigma, \eta, y_0, Y_m)$ representing $L_m(\mathbf{G})$ and K, respectively.

1. Set $\mathbf{K}_0 := (Y_0, \Sigma, \eta_0, y_0, Y_{m,0}) = \mathbf{K}$.
2. For $i \geq 0$, calculate $\mathbf{K}_i' = (Y_i', \Sigma, \eta_i', y_0, Y_{m,i}')$ where

$$Y_i' = \{y \in Y_i \mid (\forall q \in Q)(\exists s \in L(\mathbf{K}_i))\ y = \eta(y_0, s) \;\&\; q = \eta(q_0, s) \;\&\; \Sigma(q) \cap \Sigma_u \subseteq \Sigma(y)\}$$

where $\Sigma(\cdot)$ is the set of events defined at the argument state

$$Y_{m,i}' = Y_{m,i} \cap Y_i'$$

$\eta_i' = \eta_i|_{Y_i'}$, the restriction of η_i to Y_i'.

3. Set $\mathbf{K}_{i+1} = \text{trim}(\mathbf{K}_i') = (Y_{i+1}, \Sigma, \eta_{i+1}, y_0, Y_{m,i+1})$. If $\mathbf{K}_{i+1} = \mathbf{K}_i$, then output $\mathbf{K}_{i+1}$. Otherwise, advance i to $i+1$ and go to Step 2.

References

[1] I. Lee, J. Leung, and S. Son, Eds., *Handbook of Real-Time and Embedded Systems*. Boca Raton, FL: Chapman & Hall/CRC, 2007.

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

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

[4] W. M. Wonham, Supervisory Control of Discrete-Event Systems 2015, Systems Control Group, Department of Electrical and Computer Engineering, University of Toronto, updated annually 1998–2015. [Online]. Available: http://www.control.toronto.edu/DES

[5] F. Lin and W. M. Wonham, "On observability of discrete-event systems," *Inform. Sci.*, vol. 44, no. 3, pp. 173–198, 1988.

[6] R. Cieslak, C. Desclaux, A. S. Fawaz, and P. Varaiya, "Supervisory control of discrete-event processes with partial observations," *IEEE Trans. Autom. Control*, vol. 33, no. 3, pp. 249–260, Mar. 1988.

[7] H. Cho and S. I. Marcus, "Supremal and maximal sublanguages arising in supervisor synthesis problems with partial observations," *Math. Syst. Theory*, vol. 22, no. 3, pp. 177–211, 1989.

[8] S. Takai and T. Ushio, "Effective computation of an Lm(G)-closed, controllable, observable sublanguage arising in supervisory control," *Syst. Control Lett.*, vol. 49, no. 3, pp. 191–200, 2003.

[9] X. Yin and S. Lafortune, "Synthesis of maximally permissive nonblocking supervisors for partially observed discrete event systems," in *Proc. 53rd IEEE Conf. Decision Control*, Los Angeles, CA, USA, 2014, pp. 5156–5162.

[10] F. Lin and W. M. Wonham, "Supervisory control of timed discrete-event systems under partial observation," *IEEE Trans. Autom. Control*, vol. 40, no. 3, pp. 558–562, Mar. 1995.

[11] B. Brandin and W. Wonham, "Supervisory control of timed discrete-event systems," *IEEE Trans. Autom. Control*, vol. 39, no. 2, pp. 329–342, 1994.

[12] S. Takai and T. Ushio, "A new class of supervisors for timed discrete event systems under partial observation," *Discrete Event Dynam. Syst.*, vol. 16, no. 2, pp. 257–278, 2006.

[13] K. Cai, R. Zhang, and W. M. Wonham, "Relative observability of discrete-event systems and its supremal sublanguages," *IEEE Trans. Autom. Control*, vol. 60, no. 3, pp. 659–670, Mar. 2015.

[14] K. Cai, R. Zhang, and W. M. Wonham, "On relative observability of timed discrete-event systems," in *Proc. Int. Workshop Discrete Event Syst.*, Cachan, France, 2014, pp. 208–213.

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

[16] T. S. Yoo and S. Lafortune, "A general architecture for decentralized supervisory control of discrete-event systems," *Discrete Event Dynam. Syst.*, vol. 12, no. 3, pp. 335–377, 2002.

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

[18] M. Nomura and S. Takai, "Decentralized supervisory control of timed discrete event systems," *IEICE Trans. Fundam.*, vol. 94, no. 12, pp. 2802–2809, 2011.

[19] K. Cai, R. Zhang, and W. M. Wonham, "On relative coobservability of discrete-event systems," in *Proc. American Control Conf.*, Chicago, IL, USA, 2015, pp. 371–376.

[20] S. Tripakis, "Undecidable problems of decentralized observation and control on regular languages," *Inform. Process. Lett.*, vol. 90, no. 1, pp. 21–28, 2004.

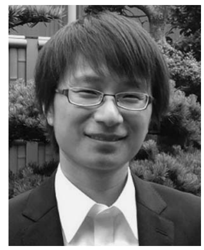
[21] Y. Brave and M. Heymann, "Formulation and control of real time discrete event processes," in *Proc. 27th IEEE Conf. Decision Control*, Austin, TX, USA, 1988, pp. 1131–1132.

[22] J. S. Ostroff, "Deciding properties of timed transition models," *IEEE Trans. Parallel Distrib. Syst.*, vol. 1, no. 2, pp. 170–183, 1990.

[23] D. Cofer and V. Garg, "Supervisory control of real-time discrete-event systems using lattice theory," *IEEE Trans. Autom. Control*, vol. 41, no. 2, pp. 199–209, Feb. 1996.

[24] M. Nomura and S. Takai, "Decentralized supervisory control of timed discrete event systems using a partition of the forcible event set," *IEICE Trans. Fundam.*, vol. 95, no. 5, pp. 952–960, 2012.

[25] K. Cai and W. M. Wonham, *Supervisor Localization: A Top-Down Approach to Distributed Control of Discrete-Event Systems*. Switzerland: Springer-Verlag, 2015.



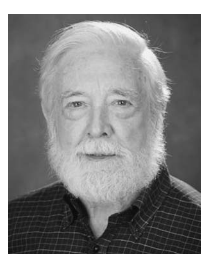
Kai Cai (M'08) received the B. Eng. degree in electrical engineering from Zhejiang University, Hangzhou, China, in 2006, the M.A.Sc. degree in electrical and computer engineering from the University of Toronto, Toronto, ON, Canada, in 2008, and the Ph.D. degree in systems science from the Tokyo Institute of Technology, Tokyo, Japan, in 2011.

He is currently an Associate Professor at Osaka City University. Previously, he was an Assistant Professor at the University of Tokyo (2013–2014), and a postdoctoral Fellow at the University of Toronto (2011–2013). His research interests include distributed control of multi-agent systems, distributed control of discrete-event systems, and control architecture of complex networked systems. He is the co-author (with W. M. Wonham) of Supervisor Localization (New York: Springer-Verlag: 2015).

Dr. Cai received the Best Paper Award of SICE in 2013 and in 2010, the Best Student Paper Award of the IEEE Multi-Conference on Systems and Control, and the Young Author's Award of SICE.

Renyuan Zhang (M'15) received the B. Eng. degree and Ph.D. degrees in electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 2007 and 2013, respectively.

He is currently a Lecturer at Northwestern Polytechnical University, Xi'an. He was an international visiting student in the University of Toronto, Toronto, ON, Canada, during 2011–2012. His research interests include distributed control of discrete-event systems and hybrid control of urban traffic systems.



W. M. Wonham (LF'77) received the B. Eng. degree in engineering physics from McGill University in 1956 and the Ph.D. degree in control engineering from the University of Cambridge, Cambridge, U.K., in 1961.

From 1961 to 1969, he was associated with several U.S. research groups in control. Since 1970, he has been a faculty member in Systems Control with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada. His research interests have included stochastic control and filtering, geometric multivariable control, and discrete-event systems. He is the author of Linear Multivariable Control: A Geometric Approach (3rd edition) (New York: Springer-Verlag, 1985), co-author (with C. Ma) of Nonblocking Supervisory Control of State Tree Structures (New York: Springer-Verlag: 2005), and co-author (with K. Cai) of Supervisor Localization (New York: Springer-Verlag: 2015).

Dr. Wonham is a Fellow of the Royal Society of Canada and a Foreign Member of the (U.S.) National Academy of Engineering. In 1987, he received the IEEE Control Systems Science and Engineering Award and in 1990 was Brouwer Medallist of the Netherlands Mathematical Society. In 1996 he was appointed University Professor in the University of Toronto, and in 2000 University Professor Emeritus.