To appear in the *International Journal of Control*
Vol. 00, No. 00, Month 20XX, 1–21

Supervisor Localization for Large-Scale Discrete-Event Systems under Partial Observation

Renyuan Zhang[a], Kai Cai[b]

[a]*School of Automation, Northwestern Polytechnical University, China;*
[b]*Department of Electrical and Information Engineering, Osaka City University, Japan.*

(*Received 00 Month 20XX; accepted 00 Month 20XX*)

Recently we developed *partial-observation supervisor localization*, a top-down approach to distributed control of discrete-event systems (DES) under partial observation. Its essence is the decomposition of the *partial-observation monolithic supervisor* into *partial-observation local controllers* for individual controllable events. In this paper we extend the partial-observation supervisor localization to large-scale DES, for which the monolithic supervisor may be incomputable. Specifically, we first employ an efficient heterarchical supervisor synthesis procedure to compute a heterarchical array of *partial-observation decentralized supervisors* and *partial-observation coordinators*. Then we localize each of these supervisors/coordinators into *partial-observation local controllers*. This procedure suggests a systematic approach to the distributed control of large-scale DES under partial observation. The results are illustrated by a system of automatic guided vehicles (AGV) serving a manufacturing workcell.

Keywords: Discrete-event systems; Supervisory Control; Partial observation.

1. Introduction

Recently we developed in Zhang, Cai, and Wonham (2017) a top-down approach, called *partial-observation supervisor localization*, to the distributed control of multi-agent discrete-event systems (DES) under partial observation. Specifically, we first synthesize a *partial-observation monolithic supervisor* using the concept of *relative observability* in Cai, Zhang, and Wonham (2015, 2016), and then decompose the supervisor into local controllers for individual controllable events, by a *partial-observation localization procedure* adapted from Cai and Wonham (2010a). The derived local controllers have state transitions triggered only by observable events, and they collectively achieve the same controlled behavior as the partial-observation monolithic supervisor does. This approach, however, cannot deal with large-scale system, because the monolithic supervisor synthesis at the first step is NP-hard (Gohari & Wonham, 2000); indeed the state size of the supervisor grows exponentially in the number of individual plant components and specifications.

In this paper, we propose a systematic attack to distributed control of large-scale DES under partial-observation. Just as in Cai and Wonham (2010a, 2010b) for full-observation case, we combine the partial-observation supervisor localization (Zhang et al., 2017) with an efficient heterarchical supervisor synthesis procedure (Feng & Wonham, 2008). Specifically, we first compute a heterarchical array of *partial-observation decentralized supervisors* and *partial-observation coordinators* to achieve globally feasible and nonblocking controlled behavior. In computing these decentralized supervisors and coordinators, we (again) employ relative observability since it is closed under set unions and the supremal sublanguage exists. We then localize each of these partial-observation

Emails: ryzhang@nwpu.edu.cn, kai.cai@eng.osaka-cu.ac.jp.

supervisors and coordinators into partial-observation local controllers by the partial-observation localization procedure in Zhang et al. (2017). As in Zhang et al. (2017), the partial-observation local controllers have only observable events causing state changes.

The contributions of this work are twofold. First, from a theoretical view, the combination of partial-observation supervisor localization procedure with the heterarchical supervisor synthesis procedure supplies a systematic approach to the distributed control of large-scale discrete-event systems under partial observation. The heterarchical supervisor synthesis procedure makes the localization procedure efficient and thus applicable to large systems. By employing relative observability, the derived controlled behavior will be generally more permissive than that derived by normality; the latter is widely used in the literature. Second, from a practical view, this work suggests an effectively computable way to design a distributed control architecture under partial observation for a multi-agent plant with large size and a decomposable specification; all the procedures are implemented by computer algorithms (in the software package TCT (Wonham, 2017a)). The detailed steps are illustrated by an AGV example, in which all the computations are executed by TCT procedures.

The paper is organized as follows. Section 2 reviews the supervisory control problem of DES under partial observation and formulates the partial-observation supervisor localization problem. Section 3 presents the partial-observation localization procedure for large-scale system. Section 4 describes the AGV, and presents the solution to the distributed control of AGV under partial observation. Finally Section 5 states our conclusions.

2. Preliminaries and Problem Formulation

2.1 *Preliminaries on Partial Observation*

The plant to be controlled is modelled by a generator

$$\mathbf{G} = (Q, \Sigma, \delta, q_0, Q_m) \tag{1}$$

where Q is the finite state set; $q_0 \in Q$ is the initial state; $Q_m \subseteq Q$ is the subset of marker states; Σ is the finite event set; $\delta : Q \times \Sigma \to Q$ is the (partial) state transition function. In the usual way, δ is extended to $\delta : Q \times \Sigma^* \to Q$, and we write $\delta(q, s)!$ to mean that $\delta(q, s)$ is defined. Let Σ^* be the set of all finite strings, including the empty string ϵ. The *closed behavior* of $\mathbf{G}$ is the language

$$L(\mathbf{G}) = \{s \in \Sigma^* | \delta(q_0, s)!\}$$

and the *marked behavior* is

$$L_m(\mathbf{G}) = \{s \in L(\mathbf{G}) | \delta(q_0, s) \in Q_m\} \subseteq L(\mathbf{G}).$$

For supervisory control, the event set Σ is partitioned into Σ_c, the subset of controllable events that can be disabled by an external supervisor, and Σ_{uc}, the subset of uncontrollable events that cannot be prevented from occurring (i.e. $\Sigma = \Sigma_c \dot{\cup} \Sigma_{uc}$). For partial observation, Σ is partitioned into Σ_o, the subset of observable events, and Σ_{uo}, the subset of unobservable events (i.e. $\Sigma = \Sigma_o \dot{\cup} \Sigma_{uo}$). Bring in the natural projection $P : \Sigma^* \to \Sigma_o^*$ defined by

$$\begin{aligned} P(\epsilon) &= \epsilon; \\ P(\sigma) &= \begin{cases} \epsilon, & \text{if } \sigma \notin \Sigma_o, \\ \sigma, & \text{if } \sigma \in \Sigma_o; \end{cases} \\ P(s\sigma) &= P(s)P(\sigma), \quad s \in \Sigma^*, \sigma \in \Sigma \end{aligned} \tag{2}$$

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 *supervisory control* for $\mathbf{G}$ is any map $V : L(\mathbf{G}) \to \Gamma$, where $\Gamma := \{\gamma \subseteq \Sigma | \gamma \supseteq \Sigma_{uc}\}$. Then the closed-loop system is $V/\mathbf{G}$, with closed behavior $L(V/\mathbf{G})$ and marked behavior $L_m(V/\mathbf{G})$ (Wonham, 2017b). Under partial observation $P : \Sigma^* \to \Sigma_o^*$, we say that V is *feasible* if

$$(\forall s, s' \in L(\mathbf{G}))\ P(s) = P(s') \Rightarrow V(s) = V(s')$$

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

It is well-known (Lin & Wonham, 1988b) that under partial observation, a feasible and nonblocking supervisory control V exists which synthesizes a (nonempty) sublanguage $K \subseteq L_m(\mathbf{G})$ if and only if K is both controllable and observable (Wonham, 2017b). When K is not observable, however, there generally does not exist the supremal observable (and controllable) sublanguage of K. Recently in Cai et al. (2015), a new concept of *relative observability* is proposed, which is stronger than observability but permits the existence of the supremal relatively observable sublanguage.

Formally, a sublanguage $K \subseteq L_m(\mathbf{G})$ is *controllable* (Wonham, 2017b) if

$$\overline{K}\Sigma_{uc} \cap L(\mathbf{G}) \subseteq \overline{K}.$$

Let $C \subseteq L_m(\mathbf{G})$. A sublanguage $K \subseteq C$ is *relatively observable* with respect to C (or C-observable) if for every pair of strings $s, s' \in \Sigma^*$ that are lookalike under P, i.e. $P(s) = P(s')$, the following two conditions hold (Cai et al., 2015):

$$\text{(i) } (\forall \sigma \in \Sigma) s\sigma \in \overline{K}, s' \in \overline{C}, s'\sigma \in L(\mathbf{G}) \Rightarrow s'\sigma \in \overline{K} \tag{3}$$

$$\text{(ii) } s \in K, s' \in \overline{C} \cap L_m(\mathbf{G}) \Rightarrow s' \in K \tag{4}$$

For $F \subseteq L_m(\mathbf{G})$ write $\mathcal{CO}(F)$ for the family of controllable and C-observable sublanguages of F. Then $\mathcal{CO}(F)$ is nonempty (the empty language $\emptyset$ belongs) and is closed under set union; $\mathcal{CO}(F)$ has a unique supremal element $\sup \mathcal{CO}(F)$ given by

$$\sup \mathcal{CO}(F) = \bigcup \{K | K \in \mathcal{CO}(F)\}$$

which may be effectively computed (Cai et al., 2015).

2.2 *Formulation of Partial-Observation Supervisor Localization Problem for Large-Scale DES*

Let the plant $\mathbf{G}$ be comprised of N (> 1) component agents

$$\mathbf{G}_k = (Q_k, \Sigma_k, \delta_k, q_{0,k}, Q_{m,k}), k = 1, ..., N.$$

Then $\mathbf{G}$ is the *synchronous product* Wonham (2017b) of $\mathbf{G}_k$ (k in the integer range $\{1, ..., N\}$), denoted as $[1, N]$, i.e.

$$\mathbf{G} := \mathop{||}_{k \in [1,N]} \mathbf{G}_k \tag{5}$$

where $||$ denotes synchronous product of generators (Wonham, 2017b). Here Σ_k need not be pair-wise disjoint, and thus $\Sigma = \cup\{\Sigma_k | k \in [1, N]\}$.

The plant components are implicitly coupled through a control specification language E that imposes behavioral constraints on $\mathbf{G}$. As in the literature (e.g. Lin and Wonham (1988a); Willner and Wonham (1991)), assume that E is *decomposable* into specifications $E_p \subseteq \Sigma_{e,p}^*$ ($p \in \mathcal{P}$, $\mathcal{P}$ an index set), where the $\Sigma_{e,p} \subseteq \Sigma$ need not be pairwise disjoint; namely

$$E = \mathop{||}_{p\in\mathcal{P}} E_p \tag{6}$$

where $||$ denotes synchronous product of languages (Wonham, 2017b). Thus E is defined over $\Sigma_e := \cup\{\Sigma_{e,p} | p \in \mathcal{P}\}$.

Considering partial-observation, let Σ_o be the observable event set. For the plant $\mathbf{G}$ and the specification E described above, let $\alpha \in \Sigma_c$ be an arbitrary controllable event, which may or may not be observable. We say that a generator

$$\mathbf{LOC}_\alpha = (Y_\alpha, \Sigma_\alpha, \eta_\alpha, y_{0,\alpha}, Y_{m,\alpha}),\ \Sigma_\alpha \subseteq \Sigma_o \cup \{\alpha\}$$

is a *partial-observation local controller* for α if (i) $\mathbf{LOC}_\alpha$ enables/disables only the event α, and (ii) if α is unobservable, then α-transitions can only be selfloops in $\mathbf{LOC}_\alpha$.

Condition (i) restricts the control scope of $\mathbf{LOC}_\alpha$ to be only the event α, and condition (ii) defines the observation scope of $\mathbf{LOC}_\alpha$ as Σ_o. The latter is a distinguishing feature of a partial-observation local controller as compared to its full-observation counterpart in Cai and Wonham (2010a); the result is that only observable events may cause a state change in $\mathbf{LOC}_\alpha$, i.e.

$$(\forall y, y' \in Y_\alpha, \forall \sigma \in \Sigma_\alpha)\ y' = \eta_\alpha(y, \sigma)!,\ y' \neq y \Rightarrow \sigma \in \Sigma_o.$$

Note that the event set Σ_α of $\mathbf{LOC}_\alpha$ in general satisfies

$$\{\alpha\} \subseteq \Sigma_\alpha \subseteq \Sigma_o \cup \{\alpha\};$$

in typical cases, both subset containments are strict. The events in $\Sigma_\alpha \setminus \{\alpha\}$ may be viewed as communication events that are critical to achieve synchronization with other partial-observation local controllers (for other controllable events). The event set Σ_α is not fixed *a priori*, but will be determined as part of the localization result presented in the next section. Also note from $\Sigma_\alpha \subseteq \Sigma_o \cup \{\alpha\}$ that the transitions by any unobservable events, except α, are not defined in $\mathbf{LOC}_\alpha$.

We now formulate the *Partial-Observation Supervisor Localization Problem*:

Construct a set of partial-observation local controllers $\{\mathbf{LOC}_\alpha \mid \alpha \in \Sigma_c\}$ such that the collective controlled behavior of these local controllers is safe, i.e.

$$L_m(\mathbf{G}) \cap \Big(\bigcap_{\alpha\in\Sigma_c} P_\alpha^{-1} L_m(\mathbf{LOC}_\alpha) \Big) \subseteq L_m(\mathbf{G}) \cap P_e^{-1} E$$

and nonblocking, i.e.

$$L(\mathbf{G}) \ \cap \ \Big(\bigcap_{\alpha\in\Sigma_c} P_\alpha^{-1} L(\mathbf{LOC}_\alpha) \Big) = \overline{L_m(\mathbf{G}) \ \cap \ \Big(\bigcap_{\alpha\in\Sigma_c} P_\alpha^{-1} L_m(\mathbf{LOC}_\alpha) \Big)}$$

where $P_e : \Sigma^* \to \Sigma_e^*$ and $P_\alpha : \Sigma^* \to \Sigma_\alpha^*$ are the corresponding natural projections.

Having obtained a set of partial-observation local controllers, one for each controllable event, we can allocate each controller to the agent(s) owning the corresponding controllable event. There-

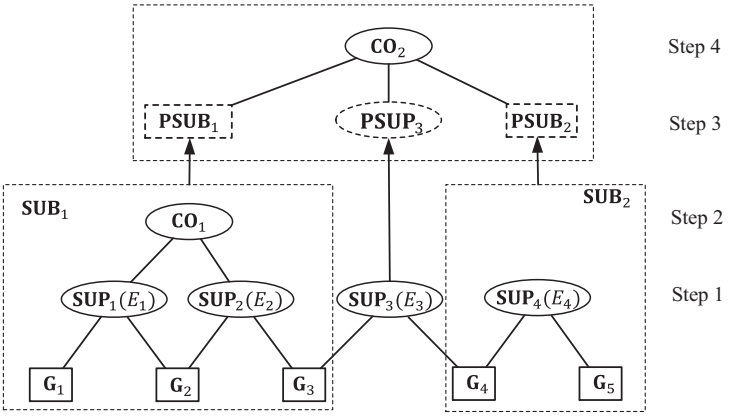

Figure 1. Partial-Observation Supervisor Synthesis

by we build for a multi-agent DES a nonblocking distributed control architecture under partial observation.

3. Partial-Observation Localization Procedure for Large-Scale DES

The partial-observation supervisor localization procedure proposed in Zhang et al. (2017) presents a solution to the problem Partial-Observation Supervisor Localization for small-scale DES, in which the monolithic supervisor is assumed to be feasibly computable. The assumption may no longer hold, however, when the system is large-scale and the problem of state explosion arises. In the literature, there have been several architectural approaches proposed to deal with the computational issue based on *model abstraction* (Feng & Wonham, 2008; Hill & Tilbury, 2006; Schmidt & Breindl, 2011; Su, van Schuppen, & Rooda, 2012).

Just as in Cai and Wonham (2010a), we propose to combine the (partial-observation) localization procedure (Zhang et al., 2017) with an efficient heterarchical supervisor synthesis procedure (Feng & Wonham, 2008) in an alternative top-down manner: first synthesize a heterarchical array of partial-observation decentralized supervisors/coordinators that collectively achieves a globally feasible and nonblocking controlled behavior; then apply the developed localization algorithm to decompose each of the supervisor/coordinator into partial-observation local controllers for the relevant controllable events.

3.1 *Localization Procedure*

Recall that we have:

- The plant to be controlled is given by $\mathbf{G}$ (defined over Σ), consisting of $\mathbf{G}_k$ defined over disjoint Σ_k ($k \in [1, N]$).
- The specification E is decomposable into $E_p \subseteq \Sigma_{e,p}^*$ ($p \in \mathcal{P}$). So E is defined over $\Sigma_e := \bigcup\{\Sigma_{e,p} | p \in \mathcal{P}\}$.
- The subset of unobservable events is $\Sigma_{uo} \subseteq \Sigma$, with the corresponding natural projection $P: \Sigma^* \to \Sigma_o^*$ ($\Sigma_o = \Sigma \setminus \Sigma_{uo}$).

The procedure of this partial-observation heterarchical supervisor localization is outlined as follows; for illustration, we shall use Fig. 1 as a running example.

Step 1) Partial-observation decentralized supervisor synthesis: For each control specification E_p (defined on Σ_p), collect the relevant component agents (e.g. by event-coupling), and denote their

synchronous product by $\mathbf{G}_p$, i.e.

$$\mathbf{G}_p := ||\{\mathbf{G}_k | k \in [1, N], \Sigma_k \cap \Sigma_p \neq \emptyset\} \tag{7}$$

Then the alphabet of $\mathbf{G}_p$ is

$$\Sigma_p := \cup\{\Sigma_k | k \in [1, N], \Sigma_k \cap \Sigma_p \neq \emptyset\}.$$

In this paper we assume that all the component agents are relevant to at least one component specification E_p; thus, $\mathbf{G}$ is exactly the synchronous product of all $\mathbf{G}_p$, i.e.

$$L(\mathbf{G}) = \bigcap_{p \in \mathcal{P}} P_p^{-1} L(\mathbf{G}_p) \tag{8}$$

$$L_m(\mathbf{G}) = \bigcap_{p \in \mathcal{P}} P_p^{-1} L_m(\mathbf{G}_p) \tag{9}$$

Considering partial observation $P : \Sigma^* \to \Sigma_o^*$, first compute using relative observability a controllable and observable sublanguage

$$K_p := \sup \mathcal{CO}(E_p || L_m(\mathbf{G}_p)),$$

and then construct (the construction is based on *uncertainty sets* of the generator representing K_p and the details are referred to Zhang et al. (2017) and Wonham (2017b)) a *partial-observation decentralized supervisor*

$$\mathbf{SUP}_p = (X_p, \Sigma_p, \eta_p, x_{0,p}, X_{m,p}) \tag{10}$$

such that

$$L_m(\mathbf{G}_p) \cap L_m(\mathbf{SUP}_p) = K_p$$
$$L(\mathbf{G}_p) \cap L(\mathbf{SUP}_p) = \overline{K_p}.$$

This is displayed in Fig. 1, "Step 1" where E_p $(p = 1, 2, 3, 4)$ denotes a specification and $\mathbf{SUP}_p$ denotes the corresponding partial-observation decentralized supervisor.

Step 2) Subsystem decomposition and coordination: After Step 1, we view the system as comprised of a set of modules $\mathbf{M}_p (p \in \mathcal{P})$, each consisting of a decentralized supervisor $\mathbf{SUP}_p$ with its associated component agents. We decompose the system into smaller-scale subsystems, through grouping the modules based on their interconnection dependencies (e.g. event-coupling or control-flow net (Feng & Wonham, 2008)).

Having obtained a set of subsystems, we verify the nonblocking property for each of them. If a subsystem $\mathbf{SUB}_q$ (with event set Σ_q) happens to be blocking, we design a *partial-observation coordinator* that removes blocking strings (Feng & Wonham, 2008, Theorem 4). The design of the coordinator must also respect partial observation $P : \Sigma^* \to \Sigma_o^*$ and the construction is similar to that of partial-observation decentralized supervisor: first compute a controllable and observable sublanguage

$$K_q := \sup \mathcal{CO}(L_m(\mathbf{SUB}_q));$$

and then construct a *partial-observation coodinator*

$$\mathbf{CO}_q = (X_q, \Sigma_q, \eta_q, x_{0,q}, X_{m,q}) \tag{11}$$

such that

$$L_m(\mathbf{SUB}_q) \cap L_m(\mathbf{CO}_q) = K_q$$
$$L(\mathbf{SUB}_q) \cap L(\mathbf{CO}_q) = \overline{K_q}.$$

For the example in Fig. 1, in "Step 2", we decompose the system consisting of four modules into two subsystems ($\mathbf{SUB}_1$ and $\mathbf{SUB}_2$), leaving the decentralized supervisor $\mathbf{SUP}_3$ in between. In case $\mathbf{SUB}_1$ is blocking (i.e. the two supervisors $\mathbf{SUP}_1$ and $\mathbf{SUP}_2$ are conflicting), a partial-observation coordinator $\mathbf{CO}_1$ is designed to resolve this conflict.

Step 3) Subsystem model abstraction: After Step 2, the system consists of a set of nonblocking subsystems. Now we need to verify the nonconflicting property among these subsystems. For this we use model abstraction technique with the properties of *natural observer* Feng and Wonham (2008) to obtain an abstracted model of each subsystem,[1] and check the nonconflictingness on the abstracted level, generally with lower computation complexity. The procedure is as follows:

(i) Determine the shared event set, denoted by Σ_{sub}, of these subsystems. Let $P_{sub} : \Sigma^* \to \Sigma_{sub}^*$ be the corresponding natural projection.
(ii) For every subsystem check if the corresponding restriction of P_{sub} is an natural observer. If yes, let $\Sigma'_{sub} = \Sigma_{sub}$, P'_{sub} be the corresponding natural projection, and goto (iii); otherwise, employ the *minimal extension* algorithm in Feng and Wonham (2008) to compute a reasonable extension of Σ_{sub} that does define an observer for every subsystem. Denote the extended alphabet by Σ'_{sub} and the corresponding natural projection by P'_{sub}.
(iii) Compute model abstractions for each subsystem with P'_{sub}.

Note that there is no particular relationship between $P'_{sub} : \Sigma^* \to \Sigma'^*_{sub}$ and the partial-observation P. On the one hand, the projection P'_{sub} guarantees that the control design at the abstracted level is equivalent to that at the non-abstracted level. On the other hand, projection P restricts that the control designs at the both levels must respect to partial-observation.

This step is illustrated in Fig. 1, "Step 3", where $\mathbf{PSUB}_i$ $(i = 1, 2)$ with a dashed box denotes the abstraction of subsystem $\mathbf{SUB}_i$. In addition, for the intermediate supervisor $\mathbf{SUP}_3$, we apply the reduction algorithm (Su & Wonham, 2004) to obtain its (control-equivalent) reduced model, denoted by $\mathbf{RSUP}_3$.

Step 4) Abstracted subsystem decomposition and coordination: This step is similar to Step 2, but for the abstracted models instead of modules. We group the abstracted models based on their interconnection dependencies, and for each group verify the nonblocking property. If a group turns out to be blocking, we design a partial-observation coordinator that removes blocking strings. In Fig. 1, "Step 4", we treat the two subsystem abstractions and the intermediate reduced supervisor as a single group. If this group turns out to be blocking, another coordinator $\mathbf{CO}_2$ is designed to resolve the conflict.

Step 5) Higher-level abstraction: Repeat Steps 3 and 4 until there remains a single group of subsystem abstractions in Step 4.

[1]The natural observer property of a projection $P' : \Sigma^* \to \Sigma'^*$ describes that whenever a string $s \in L(\mathbf{G})$ and $P's$ can be extended to $P'L_m(\mathbf{G})$ by an observable string, s can be extended to $L_m(\mathbf{G})$ by the same projection; this property is important for guaranteeing the nonblockingness of the control design.

The heterarchical supervisor/coordinator synthesis terminates at Step 5; the result is a heterarchical array of partial-observation decentralized supervisors and coordinators. Specifically, Step 1 gives a set of partial-observation decentralized supervisors $\{\mathbf{SUP}_p | p \in \mathcal{P}\}$; and Step 2 to 5 iteratively generate a set of coordinators, denoted by $\{\mathbf{CO}_q | q \in \mathcal{Q}\}$ ($\mathcal{Q}$ an index set). Similar to Feng and Wonham (2008), we prove in Theorem 1 below that these partial-observation supervisors/coordinators together achieve globally feasible and nonblocking (controllable and observable) controlled behavior.

Step 6) Partial-observation localization: In this last step, we apply the partial-observation localization algorithm (Zhang et al., 2017) to decompose each of the obtained decentralized supervisors $\mathbf{SUP}_p$ ($p \in \mathcal{P}$) and coordinators $\mathbf{CO}_q$ ($q \in \mathcal{Q}$) into partial-observation local controllers for their corresponding controllable events. Specifically, for each controllable event $\alpha \in \Sigma_{c,p}$ ($= \Sigma_c \cap \Sigma_p$), we construct by the partial-observation localization procedure a partial-observation local controller $\mathbf{LOC}_{\alpha,p} = (Y_{\alpha,p}, \Sigma_{\alpha,p}, \eta_{\alpha,p}, y_{0,\alpha,p}, Y_{m,\alpha,p})$. By the same procedure, for each $\mathbf{SUP}_p$, we construct a set of partial-observation local controllers $\{\mathbf{LOC}_{\alpha,p} | \alpha \in \Sigma_{c,p}\}$. Similarly, we localize each $\mathbf{CO}_q$ to a set of partial-observation local coordinators $\{\mathbf{LOC}_{\alpha,q} | \alpha \in \Sigma_{c,q}\}$ where $\mathbf{LOC}_{\alpha,q} = (Y_{\alpha,q}$, $\Sigma_{\alpha,q}, \eta_{\alpha,q}, y_{0,\alpha,q}, Y_{m,\alpha,q})$ and $\Sigma_{c,q} = \Sigma_c \cap \Sigma_q$.

We note that the above procedure differs the full-observation one in Cai and Wonham (2010a, 2010b) from: (i) computing *partial-observation decentralized supervisors* and *partial- observation coordinators* in Steps 1-5, and (ii) in Step 6 applying the *partial-observation supervisor localization* developed in Section III. By the following Theorem 1, the resulting local controllers achieve the same controlled behavior as the decentralized supervisors and coordinators did.

3.2 *Main result*

The procedure described above constructs for each controllable event α multiple partial-observation local controllers, because α may belong to different decentralized supervisors or coordinators. In this case, we denote by $\mathbf{LOC}_\alpha := (X_\alpha, \Sigma_\alpha, \xi_\alpha, x_{0,\alpha}, X_{m,\alpha})$ the synchronous product of all the local controllers for α, i.e.

$$L(\mathbf{LOC}_\alpha) = \big(\underset{p\in\mathcal{P}}{||} L(\mathbf{LOC}_{\alpha,p}) \big) \,||\, \big(\underset{q\in\mathcal{Q}}{||} L(\mathbf{LOC}_{\alpha,q}) \big)$$
$$L_m(\mathbf{LOC}_\alpha) = \big(\underset{p\in\mathcal{P}}{||} L_m(\mathbf{LOC}_{\alpha,p}) \big) \,||\, \big(\underset{q\in\mathcal{Q}}{||} L_m(\mathbf{LOC}_{\alpha,q}) \big)$$

It can be easily verified that $\mathbf{LOC}_\alpha$ is also a partial-observation local controller for α, because synchronous product change neither the control authority on α (condition (i)), nor the observation scope Σ_o (condition (ii)).

By the same operation (synchronous product) on the partial-observation local controllers obtained by the localization procedure, we obtain a set of partial-observation local controllers $\mathbf{LOC}_\alpha$, one for each controllable event $\alpha \in \Sigma_c$. We shall verify below that these local controllers collectively achieve a safe and nonblocking controlled behavior.

Theorem 1: *The set of partial-observation local controllers* $\{\mathbf{LOC}_\alpha | \alpha \in \Sigma_c\}$ *is a solution to the Partial-Observation Supervisor Localization Problem (for large-scale DES), i.e.*

$$L_m(\mathbf{G}) \cap L_m(\mathbf{LOC}) \subseteq L_m(\mathbf{G}) \cap P_e^{-1} E \tag{12}$$
$$L(\mathbf{G}) \cap L(\mathbf{LOC}) = \overline{L_m(\mathbf{G}) \cap L_m(\mathbf{LOC})} \tag{13}$$

where $L_m(\mathbf{LOC}) = \bigcap_{\alpha\in\Sigma_c} P_\alpha^{-1} L_m(\mathbf{LOC}_\alpha)$ *and* $L(\mathbf{LOC}) = \bigcap_{\alpha\in\Sigma_c} P_\alpha^{-1} L(\mathbf{LOC}_\alpha)$.

This theorem asserts that the local controllers and coordinators achieve a global nonblocking controlled behavior, that may not be feasibly computable for large-scale systems in a monolithic way (comparison on the complexities of the two approaches will be presented in the next subsection). Instead, by the proposed heterarchical approach, the partial-observation decentralized supervisors and coordinators are easier to be obtained, reducing the computational effort of the localization procedure. Moreover, such local controllers exist, as long as the monolithic partial-observation supervisor exists (the existence of the monolithic supervisor depends on a variety of factors, including plant dynamics, specifications, choices of controllable and observable events). This theorem also confirms that the proposed localization procedure supplies a computable way to the distributed control problem for large-scale DES under partial observation; to the best of our knowledge, no result is found in the literature to deal with this problem.

Proof of Theorem 1: The first five steps of the procedure generate a heterarchical array of partial-observation decentralized supervisors $\{\mathbf{SUP}_p | p \in \mathcal{P}\}$ and coordinators $\{\mathbf{CO}_q | q \in \mathcal{Q}\}$. We first prove that the collectively controlled behavior of these decentralized supervisors and coordinators is safe and nonblocking, and then show that the partial-observation local controllers are control equivalent to the decentralized supervisors and coordinators.

(i) (*safe and nonblocking*) Let **SYS** represent the collective behavior of these decentralized supervisors and coordinators, i.e.

$$\begin{aligned} L_m(\mathbf{SYS}) &:= L_m(\mathbf{G}) \cap \big(\bigcap_{p\in\mathcal{P}} P_p^{-1} L_m(\mathbf{SUP}_p) \big) \cap \big(\bigcap_{q\in\mathcal{Q}} P_q^{-1} L_m(\mathbf{CO}_q) \big) \\ L(\mathbf{SYS}) &:= L(\mathbf{G}) \cap \big(\bigcap_{p\in\mathcal{P}} P_p^{-1} L(\mathbf{SUP}_p) \big) \cap \big(\bigcap_{q\in\mathcal{Q}} P_q^{-1} L(\mathbf{CO}_q) \big) \end{aligned}$$

where $P_p : \Sigma^* \to \Sigma_p^*$ and $P_q : \Sigma^* \to \Sigma_q^*$ are the corresponding natural projections. First, it is easy to verify that $L_m(\mathbf{SYS}) \subseteq L_m(\mathbf{G}) \cap P_e^{-1}E$, because for each decentralized supervisor, by (10) $L_m(\mathbf{G}_p) \cap L_m(\mathbf{SUP}_p) = K_p \subseteq E_p || L_m(\mathbf{G}_p)$ and thus

$$\begin{aligned} L_m(\mathbf{SYS}) &\subseteq L_m(\mathbf{G}) \cap \big(\bigcap_{p\in\mathcal{P}} P_p^{-1} L_m(\mathbf{SUP}_p) \big) \\ &= \bigcap_{p\in\mathcal{P}} P_p^{-1} \big(L_m(\mathbf{G}_p) \cap L_m(\mathbf{SUP}_p) \big) \\ &\subseteq \bigcap_{p\in\mathcal{P}} P_p^{-1} (E_p || L_m(\mathbf{G}_p)) \\ &= P_e^{-1} (\mathop{||}_{p\in\mathcal{P}} E_p) \cap L_m(\mathbf{G}) \\ &= P_e^{-1} E \cap L_m(\mathbf{G}) \end{aligned}$$

Hence, the collective behavior is safe.

Then it follows from (Feng & Wonham, 2008, Theorem 4) that

$$L(\mathbf{SYS}) = \overline{L_m(\mathbf{SYS})}$$

i.e. the collective behavior is nonblocking.

(ii) (*control-equivalence*) In Step 6, each decentralized supervisor $\mathbf{SUP}_p$ ($p \in \mathcal{P}$) is decomposed into a set of local controllers $\mathbf{LOC}_{\alpha,p}$, one for each controllable event $\alpha \in \Sigma_{c,p}$, thus by (Zhang et

al., 2017, Theorem 1),

$$\begin{aligned} L(\mathbf{G}_p) \cap \big(\mathop{\|}_{\alpha\in\Sigma_{c,p}} L(\mathbf{LOC}_{\alpha,p})\big) =& L(\mathbf{G}_p) \cap L(\mathbf{SUP}_p) \\ L_m(\mathbf{G}_p) \cap \big(\mathop{\|}_{\alpha\in\Sigma_{c,p}} L_m(\mathbf{LOC}_{\alpha,p})\big) =& L_m(\mathbf{G}_p) \cap L_m(\mathbf{SUP}_p) \end{aligned}$$

So,

$$\begin{aligned} L(\mathbf{G}) \cap \Big(\bigcap_{\alpha\in\Sigma_c} P_\alpha^{-1}\big(\mathop{\|}_{p\in\mathcal{P}} L(\mathbf{LOC}_{\alpha,p})\big)\Big) =& \Big(\bigcap_{p\in\mathcal{P}} P_p^{-1} L(\mathbf{G}_p)\Big) \cap \Big(\bigcap_{p\in\mathcal{P}} \big(\mathop{\|}_{\alpha\in\Sigma_{c,p}} L(\mathbf{LOC}_{\alpha,p})\big)\Big) \\ =& \bigcap_{p\in\mathcal{P}} P_p^{-1}\Big(L(\mathbf{G}_p) \cap \big(\mathop{\|}_{\alpha\in\Sigma_{c,p}} L(\mathbf{LOC}_{\alpha,p})\big)\Big) \\ =& \bigcap_{p\in\mathcal{P}} P_p^{-1}\big(L(\mathbf{G}_p) \cap L(\mathbf{SUP}_p)\big) \\ =& L(\mathbf{G}) \cap \bigcap_{p\in\mathcal{P}} P_p^{-1}\big(L(\mathbf{SUP}_p)\big) \end{aligned}$$

and

$$L_m(\mathbf{G}) \cap \Big(\bigcap_{\alpha\in\Sigma_c} P_\alpha^{-1}\big(\mathop{\|}_{p\in\mathcal{P}} L_m(\mathbf{LOC}_{\alpha,p})\big)\Big) = L_m(\mathbf{G}) \cap \bigcap_{p\in\mathcal{P}} P_p^{-1}\big(L_m(\mathbf{SUP}_p)\big)$$

Similarly, for the coordinators $\mathbf{CO}_q$ ($q \in \mathcal{Q}$), we have

$$\begin{aligned} L(\mathbf{G}) \cap \Big(\bigcap_{\alpha\in\Sigma_c} P_\alpha^{-1}\big(\mathop{\|}_{q\in\mathcal{Q}} L(\mathbf{LOC}_{\alpha,q})\big)\Big) &= L(\mathbf{G}) \cap \bigcap_{q\in\mathcal{Q}} P_q^{-1}\big(L(\mathbf{CO}_q)\big) \\ L_m(\mathbf{G}) \cap \Big(\bigcap_{\alpha\in\Sigma_c} P_\alpha^{-1}\big(\mathop{\|}_{q\in\mathcal{Q}} L_m(\mathbf{LOC}_{\alpha,q})\big)\Big) &= L_m(\mathbf{G}) \cap \bigcap_{q\in\mathcal{Q}} P_q^{-1}\big(L_m(\mathbf{CO}_q)\big) \end{aligned}$$

Hence,

$$\begin{aligned} L(\mathbf{G}) \cap L(\mathbf{LOC}) =& L(\mathbf{G}) \cap \bigcap_{\alpha\in\Sigma_c} P_\alpha^{-1} L(\mathbf{LOC}_\alpha) \\ =& \Big[L(\mathbf{G}) \cap \Big(\bigcap_{\alpha\in\Sigma_c} P_\alpha^{-1}\big(\mathop{\|}_{p\in\mathcal{P}} L(\mathbf{LOC}_{\alpha,p})\big)\Big)\Big] \\ & \cap \Big[L(\mathbf{G}) \cap \Big(\bigcap_{\alpha\in\Sigma_c} P_\alpha^{-1}\big(\mathop{\|}_{q\in\mathcal{Q}} L(\mathbf{LOC}_{\alpha,q})\big)\Big)\Big] \\ =& L(\mathbf{G}) \cap \big(\bigcap_{p\in\mathcal{P}} P_p^{-1} L(\mathbf{SUP}_p)\big) \cap \big(\bigcap_{q\in\mathcal{Q}} P_q^{-1} L(\mathbf{CO}_q)\big) \\ =& L(\mathbf{SYS}) \end{aligned}$$

and

$$\begin{aligned} L_m(\mathbf{G}) \cap L_m(\mathbf{LOC}) =& L_m(\mathbf{G}) \cap \bigcap_{\alpha\in\Sigma_c} P_\alpha^{-1} L_m(\mathbf{LOC}_\alpha) \\ =& L_m(\mathbf{G}) \cap \big(\bigcap_{p\in\mathcal{P}} P_p^{-1} L_m(\mathbf{SUP}_p)\big) \cap \big(\bigcap_{q\in\mathcal{Q}} P_q^{-1} L_m(\mathbf{CO}_q)\big) \\ =& L_m(\mathbf{SYS}) \end{aligned}$$

which means that the partial-observation local controllers achieve the same controlled behavior **SYS** with the decentralized supervisors and coordinators. By the results in (i), i.e. **SYS** is safe and nonblocking, the conditions (12) and (13) hold. □

3.3 *Complexity study*

This subsection discusses the time complexities of algorithms employed by the newly proposed heterarchical (partial-observation) localization procedure. The discussion will follow the steps described in Subsection 3.1. For a concrete exposure we analyze the case displayed in Fig. 1; analysis of the general case can be done in a similar fashion. For simplicity, we assume that the largest state size of the plant components $\mathbf{G}_k$ $(k = 1, ..., 5)$ is n, and the largest state size of the specification components E_p $(p = 1, ..., 4)$ is m.

At Step 1), for each $p = 1, ..., 4$, there are two plant components $\mathbf{G}_k$ satisfying $\Sigma_k \cap \Sigma_p \neq \emptyset$; thus the state number of $\mathbf{G}_p$ (as in (7)) is upper bounded by

$$|\mathbf{G}_p|_u := n^2.$$

Notation: in this subsection, we shall use $|\cdot|_u$ to represent the upper bound of the state number of the argument generator. According to Algorithm 3 in Cai et al. (2015), the complexity of computing the partial-observation decentralized supervisors $\mathbf{SUP}_p$ is

$$T_p := O\big(2^{m*n^2}(1 + |\Sigma|) * 2^{(2^{m*n^2}+1)*n^2}\big) \tag{14}$$

and the state number of $\mathbf{SUP}_p$ is upper bounded by

$$|\mathbf{SUP}_p|_u = 2^{m*n^2}.$$

At Step 2), the system is decomposed into two subsystems $\mathbf{SUB}_1$ and $\mathbf{SUB}_2$, and a decentralized supervisor $\mathbf{SUP}_3$. $\mathbf{SUB}_1$ is comprised of two modules, each consisting of a partial-observation decentralized supervisor $\mathbf{SUP}_p$ and its associated plant components, i.e. $\mathbf{SUB}_1 = ||_{p=1}^2 (\mathbf{G}_p||\mathbf{SUP}_p) = ||_{p=1}^2 \mathbf{SUP}_p$ (because $L(\mathbf{SUP}_p) \subseteq L(\mathbf{G}_p)$ and $L_m(\mathbf{SUP}_p) \subseteq L_m(\mathbf{G}_p)$). Thus the state number of $\mathbf{SUB}_1$ is upper bounded by

$$|\mathbf{SUB}_1|_u := |\mathbf{SUP}_1|_u * |\mathbf{SUP}_2|_u = (2^{m*n^2})^2 = 2^{2m*n^2}.$$

Similarly, the state number of $\mathbf{SUB}_2$ (containing only one module) is upper bounded by

$$|\mathbf{SUB}_2|_u := |\mathbf{SUP}_4|_u = 2^{m*n^2}.$$

$\mathbf{SUB}_1$ is blocking, thus a partial-observation coordinator $\mathbf{CO}_1$ is designed with complexity

$$\begin{aligned} T_{co_1} :=& O\big(|\mathbf{SUB}_1|_u * (1+|\Sigma|) * 2^{(|\mathbf{SUB}_1|_u+1)*|\mathbf{SUB}_1|_u}\big) \\ =& O\big((2^{2m*n^2}) * (1+|\Sigma|) * 2^{(2^{2m*n^2}+1)*2^{2m*n^2}}\big). \end{aligned} \tag{15}$$

Note that the above complexity T_{co_1} is different from T_p in (14), i.e. the complexity need not be double exponential in the state number of $\mathbf{SUB}_1$; the reason is that $\mathbf{SUP}_1$ and $\mathbf{SUP}_2$ in $\mathbf{SUB}_1$ are *normal* and we need not compute the normal form of $\mathbf{SUB}_1$ as in (14), which is exponential in the number of $\mathbf{SUB}_1$ (the readers are referred to Cai et al. (2015) for the definition of normal generators and the associated computational complexity). The state number of $\mathbf{CO}_1$ is upper bounded by

$$|\mathbf{CO}_1|_u := 2^{2m*n^2}.$$

The coordinator $\mathbf{CO}_1$ removes blocking states from $\mathbf{SUB}_1$; since the number of the blocking states is unknown, the upper bound of the state number of the $\mathbf{SUB}_1$ is unchanged.

By Step 3), we obtained abstractions $\mathbf{PSUB}_1$, $\mathbf{PSUB}_2$ and $\mathbf{PSUP}_3$ for the nonblocking subsystems $\mathbf{SUB}_1$, $\mathbf{SUB}_2$ and $\mathbf{SUP}_3$ respectively. According to Feng and Wonham (2008), the algorithm for computing abstracted models is to find *natural observers* for the subsystems, and thus its complexity is polynomial in the state number of the subsystems. So the complexity of this step can be neglected from the overall complexity estimation, and the upper bounds of state numbers of the abstracted subsystems are unchanged, i.e. $|\mathbf{PSUB}_1|_u = 2^{2m*n^2}$, $|\mathbf{PSUB}_2|_u = 2^{m*n^2}$ and $|\mathbf{PSUP}_3|_u = 2^{m*n^2}$.

At Step 4), a coordinator $\mathbf{CO}_2$ is designed for resolving the conflicting among the abstracted subsystems $\mathbf{PSUB}_1$, $\mathbf{PSUB}_2$ and $\mathbf{PSUP}_3$. Similar to estimation of T_{co_1}, the complexity of designing $\mathbf{CO}_2$ is

$$\begin{aligned} T_{co_2} :=& O\big((|\mathbf{PSUB}_1|_u * |\mathbf{PSUB}_2|_u * |\mathbf{PSUP}_3|_u) * (1+|\Sigma|) * \\ & 2^{(|\mathbf{PSUB}_1|_u*|\mathbf{PSUB}_2|_u*|\mathbf{PSUP}_3|_u+1)*(|\mathbf{PSUB}_1|_u*|\mathbf{PSUB}_2|_u*|\mathbf{PSUP}_3|_u)}\big) \\ =& O\big((2^{4m*n^2}) * (1+|\Sigma|) * 2^{(2^{4m*n^2}+1)*2^{4m*n^2}}\big) \end{aligned} \tag{16}$$

and the state number of $\mathbf{CO}_2$ is upper bounded by

$$|\mathbf{CO}_2|_u := 2^{4m*n^2}.$$

The heterarchical decentralized supervisor/coordinator synthesis terminates at Step 5), because at Step 4) we have already obtained a single group of subsystems. (In the general case one needs to analyze the complexity of Step 5); since this step is to repeat Steps 3) and 4), the analysis may be carried out similarly.)

Step 6) employs the partial-observation localization algorithm to compute partial-observation local controllers and coordinators. According to Zhang et al. (2017), the complexity of computing $\mathbf{LOC}_{\alpha,p}$ $(p = 1, 2, ..., 5)$ from $\mathbf{SUP}_p$ is

$$T_{loc,p} := O(2^{4*|\mathbf{SUP}_p|_u}) = O(2^{2^{2m*n^2}}),$$

that of computing $\mathbf{LOC}_{\alpha,1}$ from $\mathbf{CO}_1$ is

$$T_{loc,co_1} := O(2^{4*|\mathbf{CO}_1|_u}) = O(2^{2^{4m*n^2}}),$$

and that of computing $\mathbf{LOC}_{\alpha,2}$ from $\mathbf{CO}_2$ is

$$T_{loc,co_2} := O(2^{4*|\mathbf{CO}_2|_u}) = O(2^{2^{8m*n^2}}).$$

Hence, the overall complexity of the proposed procedure for the case in Fig. 1 is

$$T_{het,loc} := O\Big(\sum_{p=1}^{2}(T_p + |\Sigma_c| * T_{loc,p}) + \sum_{q=1}^{2}(T_{co_q} + |\Sigma_c| * T_{loc,co_q})\Big) \tag{17}$$

On the other hand, the complexity of synthesizing the partial-observation *monolithic* supervisor for the case in Fig. 1 is

$$T_{mon} := O\Big((2^{(m^4*n^5)} * (1 + |\Sigma|) * 2^{(2^{(m^4*n^5)}+1)*n^5}\Big)$$

and the complexity of computing a partial-observation local controller from the monolithic supervisor is

$$T_{loc} = O(2^{4*2^{(m^4*n^5)}}) = O(2^{2^{(2m^4*n^5)}}).$$

In total the overall complexity of the monolithic partial-observation localization procedure is

$$T_{mon,loc} := O\big(T_{mon} + |\Sigma_c| * T_{loc}\big). \tag{18}$$

On comparing the complexities of the heterarchical approach in (17) with those of the monolithic approach in (18), we have the following observations:

(i) For all $p = 1, 2, ..., 5$, $T_p < T_{mon}$ and for all $q = 1, 2$, $T_{co_q} < T_{mon}$; thus the partial-observation decentralized supervisors/coordinators can be obtained more efficiently than the partial-observation monolithic supervisor.

(ii) For all $p = 1, 2, ..., 5$, $T_{loc,p} \leq T_{loc}$, and for all $q = 1, 2$, $T_{loc,co_q} < T_{loc}$; thus the computations of partial-observation local controllers/coordinators by the heterarchical approach have lower cost than the ones by the monolithic approach.

Based on (i) and (ii), we conclude that the overall time complexity of the proposed heterarchical approach is lower than that of the monolithic approach. To further demonstrate this point, we present an AGV case study in the next section.

4. Case Study: AGVs

In this section we apply the proposed heterarchical localization procedure to study the distributed control of AGV serving a manufacturing workcell under partial observation. As displayed in Fig. 2, the plant consists of five independent AGV

$$\mathbf{A1}, \mathbf{A2}, \mathbf{A3}, \mathbf{A4}, \mathbf{A5}$$

and there are nine imposed control specifications

$$\mathbf{Z1}, \mathbf{Z2}, \mathbf{Z3}, \mathbf{Z3}, \mathbf{WS13}, \mathbf{WS14S}, \mathbf{WS2}, \mathbf{WS3}, \mathbf{IPS}$$

Table 1. Physical interpretation of unobservable events

Event	Physical interpretation
13	**A1** re-enters Zone 1
23	**A2** re-enters Zone 1
31	**A3** re-enters Zone 2
42	**A4** exists Zone 4 and loads from WS3
53	**A5** re-enters Zone 4

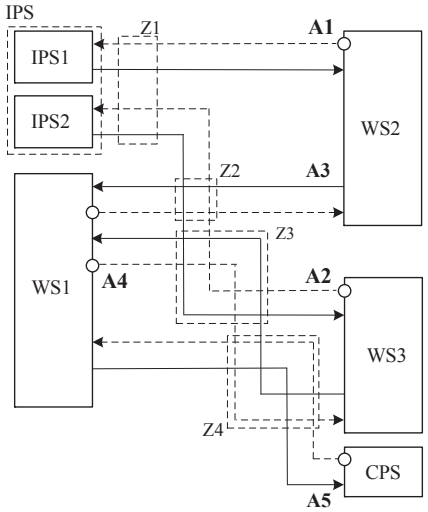

Figure 2. AGV system configuration. Rectangular dashed boxes represent shared zones of the AGV's traveling routes.

Table 2. State sizes of partial-observation decentralized supervisors

Supervisor	State size	Supervisor	State size
Z1SUP	13	**Z2SUP**	11
Z3SUP	26	**Z4SUP**	9
WS13SUP	15	**WS14SUP**	19
WS2SUP	15	**WS3SUP**	26
IPSSUP	13		

which require no collision of AGV in the shared zones and no overflow or underflow of buffers in the workstations. The generator models of the plant components and the specification are displayed in Figs. 3 and 4 respectively; the detailed system description and the interpretation of the events are referred to (Wonham, 2017b, Section 4.7).

Consider partial observation and let the unobservable event set be $\Sigma_{uo} = \{13, 23, 31, 42, 53\}$; thus each AGV has an unobservable event and the corresponding physical interpretation is listed in Table 1. Our control objective is to design for each AGV a set of local strategies subject to partial observation such that the overall system behavior satisfies the imposed specifications and is nonblocking.

Step 1) Partial-observation decentralized supervisor synthesis: For each specification displayed in Fig. 4, we group its event-coupled AGV as the decentralized plant (see Fig. 5), and synthesize as in (10) a partial-observation decentralized supervisor. The state sizes of these decentralized supervisors are displayed in Table 2, in which the supervisors are named correspondingly to the specifications, e.g. **Z1SUP** is the decentralized supervisor corresponding to the specification **Z1**.

Step 2) Subsystem decomposition and coordination: We have nine decentralized supervisors, and thus nine modules (consisting of a decentralized supervisor with associated AGV components). Under full observation, the decentralized supervisors for the four zones (**Z1SUP**, ..., **Z4SUP**) are

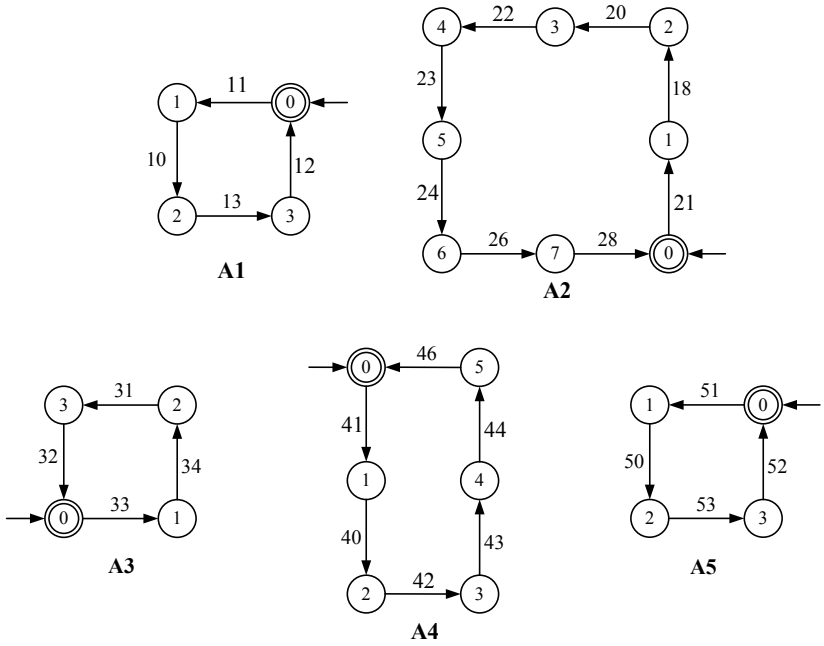

Figure 3. AGV: Generators of plant components

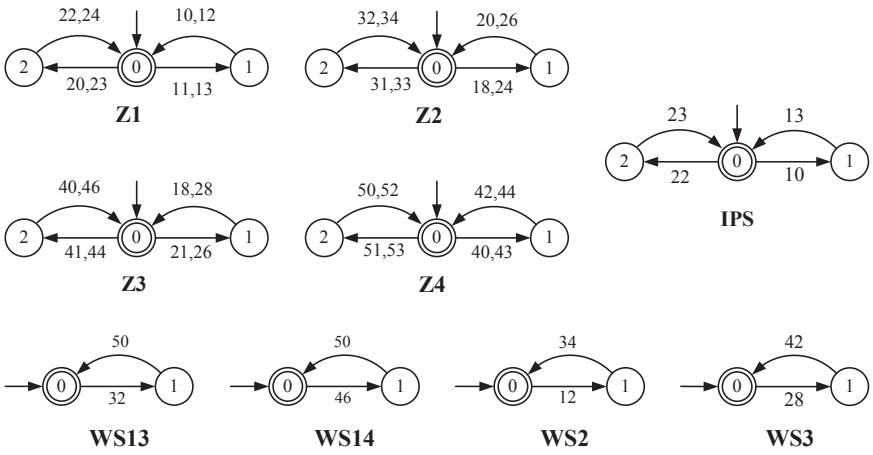

Figure 4. AGV: Generators of specifications

harmless to the overall nonblocking property (Feng & Wonham, 2006, Proposition 5), and thus can be safely removed from the interconnection structure; then the interconnection structure of these modules are simplified by applying *control-flow net* (Feng & Wonham, 2008). Under partial observation, however, the four decentralized supervisors are not harmless to the overall nonblocking property (also by (Feng & Wonham, 2006, Proposition 5), the necessary conditions are not satisifed due to partial observation) and thus cannot be removed. As displayed in Fig. 6, we decompose the overall system into two subsystems:

$$\textbf{SUB1} := \textbf{A2}||\textbf{A4}||\textbf{A5}||\textbf{WS3SUP}||\textbf{WS14SUP}||\textbf{Z3SUP}||\textbf{Z4SUP}$$
$$\textbf{SUB2} := \textbf{A1}||\textbf{A3}||\textbf{A5}||\textbf{WS2SUP}||\textbf{WS13SUP}$$

Between the two subsystems are decentralized supervisors **Z1SUP**, **Z2SUP**, and **IPSSUP**. It is verified that **SUB2** is nonblocking, but **SUB1** is blocking. Hence we design a coordinator **CO1**

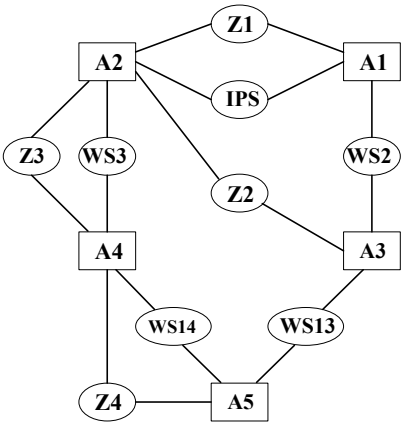

Figure 5. Event-coupling relations

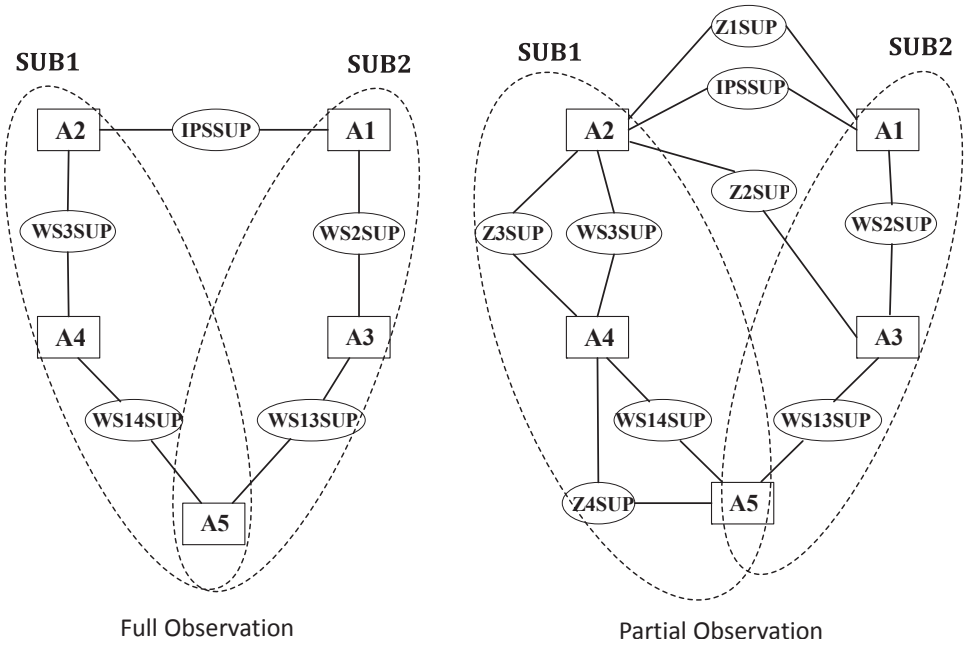

Figure 6. Subsystem decomposition

(as in (11)) which makes **SUB1** nonblocking. This coordinator **CO1** has 36 states, and we refer to this nonblocking subsystem **NSUB1**.

Step 3) Subsystem model abstraction: Now we need to verify the nonconflicting property among the nonblocking subsystems **NSUB1**, **SUB2** and the decentralized supervisors **IPSSUP**, **Z1SUP** and **Z2SUP**. First, we determine their shared event set, denoted by Σ_{sub}. Subsystems **NSUB1** and **SUB2** share all events in **A5**: 50, 51, 52 and 53. For **IPSSUP**, **Z1SUP** and **Z2SUP**, we use their reduced generator models **IPSSIM**, **Z1SIM** and **Z2SIM** by supervisor reduction Su and Wonham (2004), as displayed in Fig. 7. By inspection, **IPSSUP** and **Z1SIM** share events 21 and 24 with **NSUB1**, and events 11 with **SUB2**; **Z2SUP** shares events 24 and 26 with **NSUB1**, and events 32, 33 with **SUB2**. Thus

$$\Sigma_{sub} = \{11, 12, 21, 24, 26, 32, 33, 50, 51, 52, 53\}.$$

It is then verified that $P_{sub} : \Sigma^* \to \Sigma_{sub}^*$ satisfies the natural observer property Feng and

Table 3. State sizes of model abstractions

	NSUB1	**QC_NSUB1**	**SUB2**	**QC_SUB2**
State size	50	19	574	56

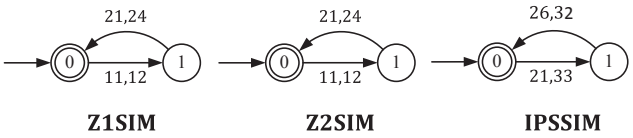

Figure 7. Reduced generator models of decentralized supervisors **Z1SUP**, **Z2SUP** and **IPSSUP**

Table 4. State sizes of partial-observation local controllers/coordinators

Supervisors/ coordinators	Local controllers of **A1(state size)**	Local controllers of **A2(state size)**	Local controllers of **A3(state size)**	Local controllers of **A4(state size)**	Local controllers of **A5(state size)**
Z1SUP	**Z1_11(2)**	**Z1_21(2)**			
Z2SUP		**Z2_21(2)**	**Z2_33(2)**		
Z3SUP		**Z3_21(2)**,**Z3_23(3)**		**Z3_41(2)**,**Z3_43(3)**	
Z4SUP				**Z4_41(2)**	**Z4_51(2)**
WS13SUP			**WS13_31(2)**		**WS13_51(2)**
WS14SUP				**WS14_43(2)**	**WS14_51(2)**
WS2SUP	**WS2_13(2)**		**WS2_33(2)**		
WS3SUP		**WS3_21(2)**		**WS3_41(2)**	
IPSSUP	**IPS_11(2)**	**IPS_21(2)**			
CO1				**CO1_41(2)**	
CO2	**CO2_11(6)**		**CO2_33(4)**		

Wonham (2008). With P_{sub}, therefore, we obtain the subsystem model abstractions, denoted by **QC_NSUB1** $= P_{sub}$(**NSUB1**) and **QC_SUB2** $= P_{sub}$(**SUB2**), with state sizes listed in Table 3.

Step 4) Abstracted subsystem decomposition and coordination: We treat **QC_NSUB1**, **QC_SUB2**, **IPSSIM**, **Z1SIM** and **Z2SIM** as a single group, and check the nonblocking property. This group turns out to be blocking, and a coordinator **CO2** is then designed (as in (11)) to make the group nonblocking. This coordinator **CO2** has 123 states.

Step 5) Higher-level abstraction: The modular supervisory control design terminates with the previous Step 4.

We have obtained a heterarchical array of nine partial-observation decentralized supervisors and two partial-observation coordinators. These supervisors and coordinators together achieve a globally feasible and nonblocking controlled behavior.

Step 6) Partial-observation localization: We finally apply the partial-observation supervisor localization procedure (Zhang et al., 2017) to decompose the obtained decentralized supervisors/coordinators into partial-observation local controllers. The generator models of the local controllers are displayed in Fig. 8-12; they are grouped with respect to the individual AGV and their state sizes are listed in Table 4. By inspecting the transition structures of the local controllers, only observable events lead to states changes.

Partial observation affects the control logics of the controllers/coordinators and thus affects the controlled system behavior. For illustration, consider the following case: assuming that event sequence 11.10.13.12.21.18.20.22 has occurred, namely **A1** has loaded a type 1 part to workstation **WS2**, and **A2** has moved to input station **IPS2**. Now, **A2** may load a type 2 part from **IPS2** (namely, event 23 may occur). Since event 24 (**A2** exits Zone 1 and re-enter Zone 2) is uncontrollable, to prevent the specification on Zone 2 (**Z2**) not being violated, AGV **A3** cannot enter Zone 2 if 23 has occurred, i.e. event 33 must be disabled. However, event 33 is eligible to occur if event 23 has occurred. So, under the full observation condition (event 23 is observable) event 33 would occur safely if event 23 has not occurred. However the fact is that event 23 is unobservable; so

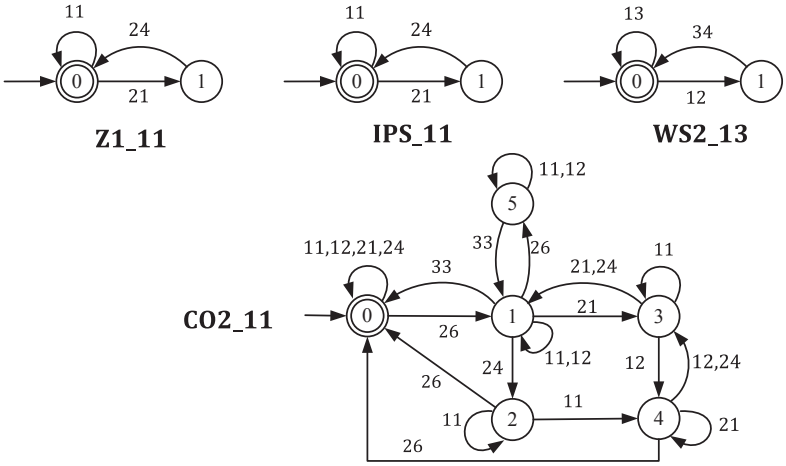

Figure 8. Partial-observation local controllers and coordinators for **A1** with controllable events 11 and 13 (the local controllers are named in the format of 'specification_event')

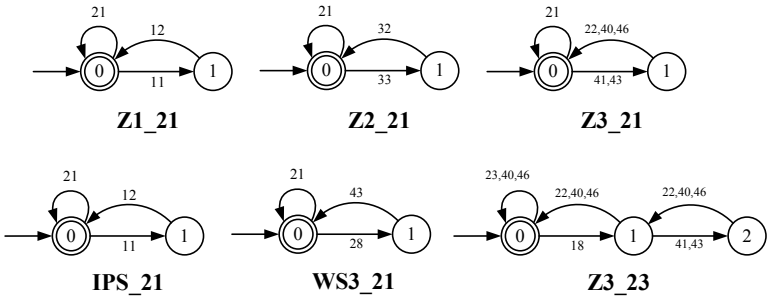

Figure 9. Partial-observation local controllers for **A2** with controllable events 21 and 23

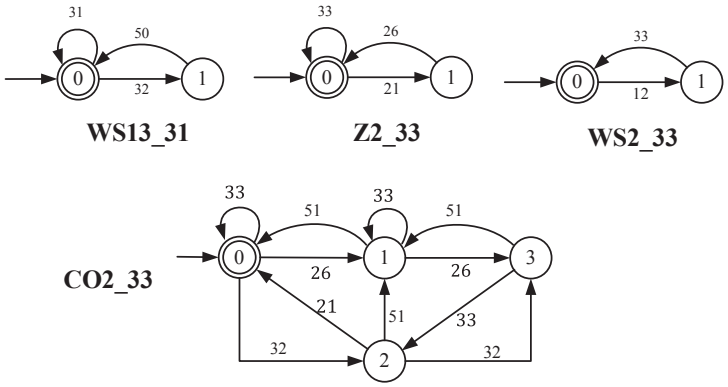

Figure 10. Partial-observation local controllers for **A3** with controllable events 31 and 33

due to (relative) observability, 33 must also be disabled even if 23 has not occurred, namely the controllers will not know whether or not event 23 has occurred, so it will disable event 33 in both cases, to prevent the possible illegal behavior. This control strategy coincides with local controller **Z2_33**: event 33 must be disabled if event 21 has occurred, and will not be re-enabled until event 26 has occurred (**A2** exits Zone 2 and re-enter Zone 3).

Finally, the heterarchical supervisor localization has effectively generated a set of partial-observation local controllers with small state sizes (between 2 and 6 states). Grouping these local controllers for the relevant AGV, we obtain a distributed control architecture for the system where each AGV is controlled by its own controllers while observing certain observable events of other

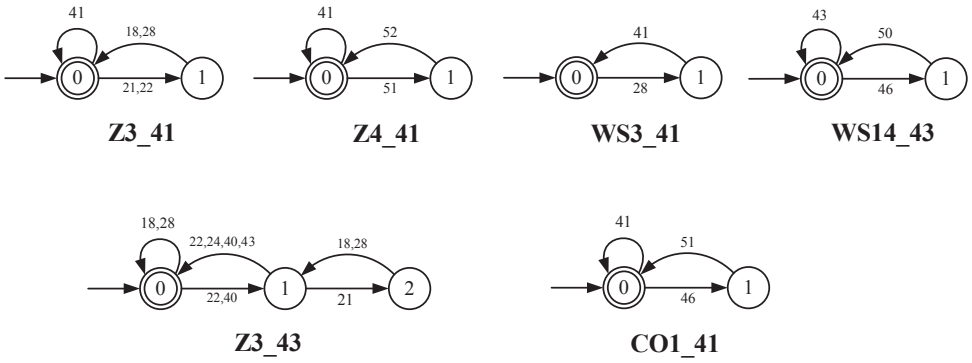

Figure 11. Partial-observation local controllers for **A4** with controllable events 41 and 43

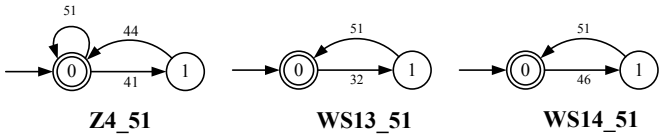

Figure 12. Partial-observation local controllers for **A5** with controllable events 51 and 53 (event 53 is not disabled and thus there is no corresponding local controller)

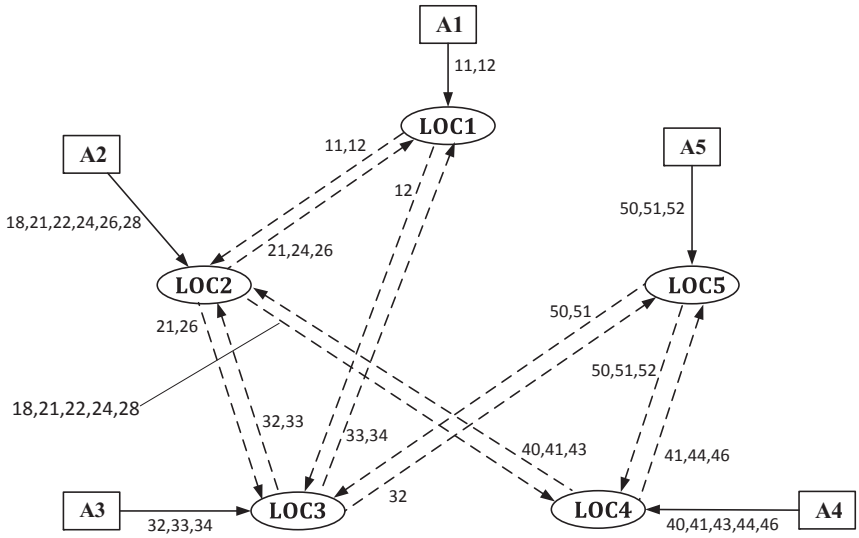

Figure 13. AGV: communication diagram of partial-observation local controllers. For $i = 1, ..., 5$, **LOCi** represents the local controllers corresponding to **Ai**.

AGV; according to the transition diagrams of the local controllers, we obtain a communication diagram, as displayed in Fig. 13, which shows the events to be observed (denoted by solid lines) or communicated (denoted by dashed lines) to local controllers.

5. Conclusions

We have developed a systematic top-down approach to solve the distributed control of large-scale multi-agent DES under partial observation. This approach first employs relative observability and an efficient heterarchical synthesis procedure to compute a heterarchical array of partial-observation decentralized supervisors and partial-observation coordinators, and then decomposes the decentralized supervisor/coordinators into a set of partial-observation local controllers whose

state changes are caused only by observable events. Moreover, we have proved that these local controllers collectively achieve a globally nonblocking behavior. An AGV example has been presented for illustration.

Funding

This work was supported in part by the National Nature Science Foundation of China, Grant no. 61403308, 11772264; the Natural Science Foundation of Shaanxi Province, China, Grant no. 2017JM5061; JSPS KAKENHI Grant no. JP16K18122.

References

Cai, K., & Wonham, W. (2010a). Supervisor localization: a top-down approach to distributed control of discrete-event systems. *IEEE Transactions on Automatic Control*, *55*(3), 605–618.

Cai, K., & Wonham, W. (2010b). Supervisor localization for large discrete-event systems: case study production cell. *International Journal of Advanced Manufacturing Technology*, *50*(9-12), 1189–1202.

Cai, K., Zhang, R., & Wonham, W. (2015). Relative observability of discrete-event systems and its supremal sublanguages. *IEEE Transactions on Automatic Control*, *60*(3), 659-670.

Cai, K., Zhang, R., & Wonham, W. (2016). Relative observability and coobservability of timed discrete-event systems. *IEEE Transactions on Automatic Control*, *61*(11), 3382-3395.

Feng, L., & Wonham, W. (2006). Computational efficient supervisory design: control flow decomposition. In *Proc. 8th international workshop discrete event systems* (pp. 9–14). Ann Arbor, MI.

Feng, L., & Wonham, W. (2008). Supervisory control architecture for discrete-event systems. *IEEE Transactions on Automatic Control*, *53*(6), 1449-1461.

Gohari, P., & Wonham, W. (2000). On the complexity of supervisory control design in the RW framework. *IEEE Transactions on Systems, Man and Cybernetics Part B: Cybernetics (Special Issue on Discrete Systems and Control)*, *30*(5), 643-652.

Hill, R., & Tilbury, D. (2006, July). Modular supervisory control of discrete-event systems with abstraction and incremental hierarchical construction. In *Proc. 8th international workshop on discrete event systems* (pp. 399–406). Ann Arbor, MI.

Lin, F., & Wonham, W. (1988a). Decentralized supervisory control of discrete-event systems. *Information Sciences*, *44*(3), 199-224.

Lin, F., & Wonham, W. (1988b). On observability of discrete-event systems. *Information Sciences*, *44*(3), 173-198.

Schmidt, K., & Breindl, C. (2011). Maximally permissive hierarchical control of decentralized discrete event systems. *IEEE Transactions on Automatic Control*, *56*(4), 723–737.

Su, R., van Schuppen, J. H., & Rooda, J. E. (2012). Maximum permissive coordinated distributed supervisory control of nondeterministic discrete-event systems. *Automatica*, *48*(7), 1237-1247.

Su, R., & Wonham, W. (2004). Supervisor reduction for discrete-event systems. *Discrete Event Dynamic Systems*, *14*(1), 31–53.

Willner, Y., & Wonham, W. (1991). Supervisory control of concurrent discrete-event systems. *International Journal of Control*, *54*(5), 1143-1169.

Wonham, W. (2017a). *Design software: Tct.* Systems Control Group, ECE Dept., University of Toronto, Toronto, ON, Canada. (Available at http://www.control.utoronto.ca/DES)

Wonham, W. (2017b). *Supervisory control of discrete-event systems.* Systems Control Group, ECE Dept., University of Toronto, Toronto, ON, Canada. (Available at http://www.control.utoronto.ca/DES)

Zhang, R., Cai, K., & Wonham, W. (2017). Supervisor localization of discrete-event systems under partial observation. *Automatica*, *81*, 142-147.