Supervisory Control Synthesis of Discrete-Event Systems using a Coordination Scheme [⋆]

Jan Komenda [a], Tomáš Masopust [a,b], Jan H. van Schuppen [b]

[a] *Institute of Mathematics, Czech Academy of Sciences, Žižkova 22, 616 62 Brno, Czech Republic*

[b] *CWI, P.O. Box 94079, 1090 GB Amsterdam, The Netherlands*

Abstract

Supervisory control of distributed DES with a global specification and local supervisors is a difficult problem. For global specifications the equivalent conditions for local control synthesis to equal global control synthesis may not be met. This paper formulates and solves a control synthesis problem for a generator with a global specification and with a combination of a coordinator and local controllers. Conditional controllability is proven to be an equivalent condition for the existence of such a coordinated controller. A procedure to compute the least restrictive solution within our coordination control architecture is provided and conditions under which the result coincides with the supremal controllable sublanguage are stated.

Key words: Discrete-event systems; Supervisory control; Distributed control; Closed-loop systems; Controllability.

1 Introduction

This paper investigates supervisory control synthesis of discrete-event systems (DES) with a coordinator. Complex DES are formed as a synchronous product of a large number of local components modeled as finite generators [14] and run in parallel. The aim of supervisory control is to ensure that the closed-loop system satisfies the control objectives of safety and of liveness. Safety means the behavior of the system is included in a specification, and liveness means the system cannot deadlock or livelock. As only controllable specifications are achievable, one of the key issues in the supervisory control synthesis is the computation of the supremal controllable sublanguage of a given specification from which the supervisor can be constructed.

The paper addresses control of distributed DES consisting of an interconnection of two or more subsystems. The aim is to find a supervisor for each subsystem so that the composition of controlled subsystems reaches the specification. The issue is that the specification is global because it deals with the interactions of subsystems.

Among the most successful approaches to supervisory control of distributed DES are those that combine decentralized and hierarchical control (both horizontal and vertical abstractions), see [2,4,15], or the approach based on interfaces [11], which restricts the interaction of the subsystems (the communication between the high level and the low level is restricted to these interfaces). A notable difference between our coordination control and the interface-based control of [11] is that the interface is fixed a priori, whereas coordination control is more flexible because we can choose the coordinator depending on the system and the control specification.

The approach of this paper is similar to the above mentioned papers in the sense that coordination control can be seen as an instance of hierarchical control, where the high level is represented by the coordinator and its supervisor. The coordinator receives a part of the observations from local subsystems and its task is to satisfy the global part of the specification and nonblockingness. Hence, the coordinator can be seen as a two-way communication channel, where some events are communicated among subsystems.

Thus, coordination control is a reasonable trade-off between a purely decentralized control synthesis that is in some cases unrealistic, and a global control synthesis that is prohibitive for complexity reasons. Unlike our previous results in decentralized control based on struc-

[⋆] A part of this paper has been presented at WODES 2010. Corresponding author: Jan Komenda. Tel. +420532290444.

Email addresses: `komenda@ipm.cz` (Jan Komenda), `masopust@math.cas.cz` (Tomáš Masopust), `J.H.van.Schuppen@cwi.nl` (Jan H. van Schuppen).

tural, specification independent conditions, e.g. mutual controllability [9], the conditions obtained from the coordination control framework are based on the specification itself rather than on the local plants.

In this paper, we are concerned with the safety issue and propose a necessary and sufficient condition on a specification (called *conditional controllability*) to be achieved in the coordination control architecture consisting of a coordinator, its supervisor, and local supervisors for the subsystems. This condition refines the sufficient condition presented in [8]. In addition, we show that the supremal conditionally-controllable sublanguage of a given specification always exists and is included in the supremal controllable sublanguage. A computational procedure is proposed.

Since the coordinator is chosen as a projected plant computed locally using distributivity of natural projections with the synchronous product, the composition of the plant with the coordinator does not modify the plant. In fact, it turns out that only the event set of the coordinator matters when we are interested in a safety issue: only the coordinator for nonblockingness should be chosen so that the composition with the coordinator restricts the plant to its trim part.

A possible solution for a coordinator that guarantees nonblockingness is sketched in [4, Proposition 4.9], where the coordinator resolves conflicts between local plants without reducing the plant. We return to this problem in a future study.

Unfortunately, concerning the safety issue, the approach of [4] based on abstractions (hierarchical approach) handles the case, where several specifications are given, but no efficient method is proposed for a global specification. Our results then fills this gap by proposing a coordinator for safety that is simple (the plant projected into a suitable coordinator even set) and does not modify the plant, but is equipped with its supervisor that further reduces the plant to achieve the safety specification within our architecture. Hence, our procedure also yields a controllable sublanguage with respect to the original plant. Moreover, additional conditions are found under which the supremal conditionally-controllable sublanguage coincides with the supremal controllable sublanguage.

The organization of this paper is as follows. Next two sections recall supervisory control of DES and motivates the coordination control approach. Section 4 presents the condition on a specification to be exactly achieved in the coordination control architecture and shows that the supremal conditionally-controllable sublanguage always exists. Section 5 proposes a computational procedure and conditions under which the result is optimal. Section 6 summarizes concluding remarks including a discussion on future extensions.

2 Control of discrete-event systems

In this section, the basic elements of supervisory control theory needed in this paper are recalled, see [3,17].

A *generator* is a quintuple $G = (Q, E, f, q_0, Q_m)$, where Q is a finite set of *states*, E is a finite set of *events*, $f : Q \times E \to Q$ is a *partial transition function*, $q_0 \in Q$ is the *initial state*, and $Q_m \subseteq Q$ is a set of *marked states*. As usual, f is extended to $f : Q \times E^* \to Q$. The *language generated* by G is defined as the set $L(G) = \{s \in E^* \mid f(q_0, s) \in Q\}$, and the *marked language* of G as the set $L_m(G) = \{s \in E^* \mid f(q_0, s) \in Q_m\}$.

For event sets $E_0 \subseteq E$, a *natural projection* $P : E^* \to E_0^*$ is a morphism defined by $P(a) = \varepsilon$, $a \in E \setminus E_0$, and $P(a) = a$, $a \in E_0$. The *inverse image* $P^{-1} : E_0^* \to 2^{E^*}$ of P is defined as $P^{-1}(a) = \{s \in E^* \mid P(s) = a\}$. These definitions are naturally extended to languages. Given event sets E_i, E_j, E_k, E_ℓ, we denote by $P^{i+j}_{k\cap\ell}$ the projection from $E_i \cup E_j$ to $E_k \cap E_\ell$. In addition, we use the notation $E_{i+j} = E_i \cup E_j$, and $E_{i,u} = E_u \cap E_i$ for the set of uncontrollable events $E_u \subseteq E$.

A synchronous product of $L_1 \subseteq E_1^*$ and $L_2 \subseteq E_2^*$ is defined as $L_1 \| L_2 = P_1^{-1}(L_1) \cap P_2^{-1}(L_2) \subseteq E^*$, where $P_i : E^* \to E_i^*$ are natural projections, $i = 1, 2$. The synchronous product is also defined for generators, see [3]. For generators G_1 and G_2, it is known that $L(G_1 \| G_2) = L(G_1) \| L(G_2)$ and $L_m(G_1 \| G_2) = L_m(G_1) \| L_m(G_2)$.

A *controlled generator* is a structure (G, E_c, Γ), where G is a generator, $E_c \subseteq E$ is the set of *controllable events*, $E_u = E \setminus E_c$ is the set of *uncontrollable events*, and $\Gamma = \{\gamma \subseteq E \mid E_u \subseteq \gamma\}$ is the set of *control patterns*. A *supervisor* for (G, E_c, Γ) is a map $S : L(G) \to \Gamma$. A *closed-loop system* associated with the controlled generator (G, E_c, Γ) and the supervisor S is defined as the minimal language $L(S/G) \subseteq E^*$ satisfying (i) $\varepsilon \in L(S/G)$ and (ii) if $s \in L(S/G)$, $a \in S(s)$, and $sa \in L(G)$, then $sa \in L(S/G)$.

In the automata framework, where supervisors are represented by generators, the closed-loop system can be recast as a synchronous product of the supervisor and the plant, i.e., $L(S/G) = L(S) \| L(G)$.

The prefix closure $\overline{L}$ of a language L is the set of all prefixes of all its words; L is prefix-closed if $L = \overline{L}$.

Definition 1 *Let $L = \overline{L} \subseteq E^*$ be a language and $E_u \subseteq E$ be a set of uncontrollable events. A language $K \subseteq L$ is* controllable *with respect to L and E_u if $\overline{K}E_u \cap L \subseteq \overline{K}$.*

Given a language $K = \overline{K} \subseteq E^*$, the goal of supervisory control is to find a supervisor S such that $L(S/G) = K$. Such a supervisor exists if and only if K is controllable

[14]. For uncontrollable languages, controllable sublanguages are considered. The notation $\sup \mathrm{C}(K, L, E_u)$ denotes the supremal controllable sublanguage of K with respect to L and E_u, which always exists and equals to the union of all controllable sublanguages of K, see [3].

Distributed control synthesis of a distributed DES is a procedure where control synthesis is carried out separately for each of the two or more local supervisors. The global supervisor then formally consists of the synchronous product of local supervisors, although it is not computed in practice. In terms of behaviors, the optimal global control synthesis is represented by the closed-loop language $\sup \mathrm{C}(K, L, E_u) = \sup \mathrm{C}(\|_{i=1}^n K_i, \|_{i=1}^n L_i, E_u)$. For a rational global specification K, the supremal controllable sublanguage from which the optimal (least restrictive) supervisor is built can be computed. Such a global control synthesis consists in computing the global plant, and then the control synthesis is carried out as above. However, the computational complexity is for most practical problems so high that other approaches need to be developed.

In the decentralized control synthesis, the specification K is replaced by $K_i = K \cap P_i^{-1}(L_i)$ and the synthesis is done as for local specifications or using the notion of partial controllability [6]. Notice the difference with decentralized control of monolithic plants studied in [18], where several control agents have different observations, but the system has no modular structure consisting of subsystems running in parallel. The purely decentralized control synthesis is not always possible because the sufficient conditions under which it can be used are quite restrictive. Therefore, in [8], coordination control is proposed as a trade-off between the purely decentralized control synthesis and the global control synthesis.

3 Concepts

Coordination control for DES is inspired by the concept of conditional independence of the theory of probability and of stochastic processes. Recall from [8] that conditional independence is roughly captured by the event set condition, when every joint action of local subsystems must be accompanied by a coordinator action.

In the coordination scheme, first a supervisor S_k for the coordinator taking care of the part $P_k(K)$ of the specification K is synthesized. Then, supervisors S_i, $i = 1, 2$, are synthesized so that the specifications $P_{i+k}(K)$ are met by the new plant languages $G_i \| (S_k/G_k)$, $i = 1, 2$. The concept of a reachable event set associated with a generator G over E, denoted by $E_r(G) \subseteq E$, is defined so that $e \in E_r(G)$ if and only if there exist $s_1, s_2 \in E^*$ such that $s_1 e s_2 \in L(G)$.

Definition 2 *Consider generators G_1, G_2, G_k. We call G_1 and G_2* conditionally independent *given G_k if in the global system there is no common transition of both G_1 and G_2 without the coordinator G_k being also involved, i.e., $E_r(G_1 \| G_2) \cap E_r(G_1) \cap E_r(G_2) \subseteq E_r(G_k)$.*

Representing languages by generators, we obtain the concept of *conditionally-independent* languages. An important feature of conditionally-independent languages is that the natural projection to E_k (the event set of $L(G_k)$) distributes with the synchronous composition (cf. proof of Lemma 10 below). Another important concept is that of conditional decomposability.

Definition 3 *A language K is* conditionally decomposable *with respect to event sets (E_{1+k}, E_{2+k}, E_k) if $K = P_{1+k}(K) \| P_{2+k}(K) \| P_k(K)$.*

Note that as $P_{i+k}(K) \subseteq (P_k^{i+k})^{-1} P_k(K)$, $i = 1, 2$, $P_{1+k}(K) \| P_{2+k}(K) \| P_k(K) = P_{1+k}(K) \| P_{2+k}(K)$, and K is conditionally decomposable if and only if there exist languages $M_1 \subseteq E_{1+k}^*$, $M_2 \subseteq E_{2+k}^*$, and $M_k \subseteq E_k^*$ such that $K = M_1 \| M_2 \| M_k$ (or $K = M_1 \| M_2$), cf. [7]. Moreover, if $K = M_1 \| M_2 \| M_k$, then $P_{1+k}(K) \subseteq M_1$, $P_{2+k}(K) \subseteq M_2$, and $P_k(K) \subseteq M_k$. This means that even though in general several languages M_i may exist, $i = 1, 2, k$, the triple $P_{1+k}(K)$, $P_{2+k}(K)$, $P_k(K)$ is the smallest decomposition.

Let us mention that for a language $K \subseteq E^*$, there always exists an event set E_k with the corresponding projection $P_k : E^* \to E_k^*$ such that K is conditionally decomposable with respect to (E_{1+k}, E_{2+k}, E_k). In the worst case, $E_k = E_1 \cup E_2$ is the whole global event set.

To give more intuition, for a conditionally-decomposable language K and $E_k \supseteq E_1 \cap E_2$, the order of local events from $E_1 \setminus E_k$ and $E_2 \setminus E_k$ in between two coordinator events from E_k is irrelevant in the sense that if two strings over the global event set differ only in the order of local events, then either both belong to K or both do not. This concept is called shuffle-closedness in [8].

4 Control synthesis: existence conditions

In this section, we propose our coordination control approach consisting of the union of a coordinator and its supervisor and the distributed system as an alternative to the well known modular control setting consisting of local plants and local supervisors. Necessary and sufficient conditions are provided on the global specification for the existence of the three supervisors (the one for the coordinator and the other two for local plants combined with the coordinator) so that the specification is exactly achieved. We consider the case of two local generators because the extension to n local generators with a single coordinator is straightforward.

Problem 4 *Consider generators G_1 and G_2 over event sets E_1 and E_2, respectively, such that the global system* *is given by the synchronous product $G_1\|G_2$. In addition to this usual modular setting we have a coordinator generator G_k over the event set $E_k \supseteq E_1 \cap E_2$. It is assumed that a prefix-closed specification $K \subseteq L(G_1\|G_2)$ is given as the behavior to be imposed on the plant. It should be noted that E_k is chosen depending on K so that K becomes conditionally decomposable as defined in Section 3. Moreover, it is shown that the coordinator may be chosen so that $L(G_1\|G_2\|G_k) = L(G_1\|G_2)$, namely as the generator for $G_k = P_k(G_1\|G_2) = P_k(G_1)\|P_k(G_2)$.*

Assume now that the coordinator G_k makes G_1 and G_2 conditionally independent (cf. Definition 2) and that K is conditionally decomposable with respect to event sets (E_{1+k}, E_{2+k}, E_k). This means that the control task is in fact divided into local subtasks and the coordinator subtask. The coordinator takes care of its part of the specification, namely $P_k(K)$. Otherwise stated, S_k is such that $L(S_k/G_k) \subseteq P_k(K)$. Similarly, supervisors S_1 and S_2 take care of their corresponding parts of the specification, namely $P_{i+k}(K)$, for $i = 1, 2$, i.e., S_i is such that $L(S_i/[G_i\|(S_k/G_k)]) \subseteq P_{i+k}(K)$, $i = 1, 2$. The problem is to determine supervisors S_1, S_2, S_k for the respective generators so that the closed-loop system with the coordinator satisfies

$$\begin{aligned} L(S_1/[G_1\|(S_k/G_k)])\|L(S_2/[G_2\|(S_k/G_k)]) \\ \|L(S_k/G_k) = K\,. \end{aligned} \qquad (*)$$

Note that $(*)$ implies that $K = L(S_1/[G_1\|(S_k/G_k)]) \parallel L(S_2/[G_2\|(S_k/G_k)])$ because the part S_k/G_k is already included in both subsystems.

In this paper, we assume that K is prefix-closed because we focus on controllability issues. An important role of the coordinator to prevent blocking is not considered, a suitable choice of the coordinator for blocking based on abstraction has been discussed in [4].

The solution of Problem 4 requires a concept of conditional controllability which will be proven to be the characterization of the solution of that problem.

Definition 5 *Consider the setting of Problem 4. A language $K \subseteq E^*$ is said to be* conditionally controllable *for generators (G_1, G_2, G_k) and locally uncontrollable event sets $(E_{1+k,u}, E_{2+k,u}, E_{k,u})$ if*

(i) *$P_k(K) \subseteq E_k^*$ is controllable wrt $L(G_k)$ and $E_{k,u}$,*
(ii.a) *$P_{1+k}(K) \subseteq (E_1 \cup E_k)^*$ is controllable wrt $L(G_1)\|P_k(K)\|P_k^{2+k}(L(G_2)\|P_k(K))$ and $E_{1+k,u}$,*
(ii.b) *$P_{2+k}(K) \subseteq (E_2 \cup E_k)^*$ is controllable wrt $L(G_2)\|P_k(K)\|P_k^{1+k}(L(G_1)\|P_k(K))$ and $E_{2+k,u}$.*

Notice that $P_k^{i+k}(L(G_i)\|P_k(K)) = P_k^i(L(G_i))\|P_k(K)$ by results shown in [17].

The conditions of Definition 5 can be checked by classical algorithms with low complexities for verification of controllability as is clear from the definition. The complexity of checking conditional controllability is less than that of controllability of the global system $G_1\|G_2\|G_k$. This is because instead of checking controllability with the global specification and the global system, we check it on the corresponding projections, which are smaller when they satisfy the observer property (Definition 8 below).

The following theorem presents the necessary and sufficient condition on a specification to be exactly achieved in the coordination control architecture.

Theorem 6 *Consider the setting of Problem 4. Then, there exist supervisors S_1, S_2, S_k such that*

$$\begin{aligned} L(S_1/[G_1\|(S_k/G_k)]) \parallel L(S_2/[G_2\|(S_k/G_k)]) \\ \parallel L(S_k/G_k) = K \end{aligned} \qquad (1)$$

if and only if K is conditionally controllable for generators (G_1, G_2, G_k) and event sets $(E_{1+k,u}, E_{2+k,u}, E_{k,u})$.

PROOF. To simplify the notation, denote $L_i = L(G_i)$, $i = 1, 2, k$, and $L = L_1\|L_2\|L_k$. By discussion below Definition 3, $P(L) = P_{1+k}(L)\|P_{2+k}(L)\|P_k(L)$.

To prove sufficiency, let $K \subseteq L$ be conditionally controllable. We prove (1). First, $P_k(K) \subseteq L_k$ controllable with respect to L_k means that there exists a supervisor S_k such that $L(S_k/G_k) = P_k(K)$, cf. [13]. Next, $K \subseteq L$ implies $P_{1+k}(K) \subseteq L_1\|L_k\|P_k^2(L_2)$, by [17] (since $P_k(L_1\|L_2) = P_k(L_1)\|P_k(L_2)$, $L_i \subseteq E_i^*$, whenever $E_1 \cap E_2 \subseteq E_k$). This, with $P_{1+k}(K) \subseteq (P_k^{1+k})^{-1}P_k(K)$, $P_k(K) \subseteq L_k$, and $P_k^2(L_2)\|P_k(K) = P_k^{2+k}(L_2\|L(S_k/G_k))$, implies $P_{1+k}(K) \subseteq L_1\|P_k(K)\| P_k^{2+k}(L_2\|P_k(K))$. By conditional controllability of K, there exists a supervisor S_1 such that $P_{1+k}(K) = L(S_1/[G_1\|(S_k/G_k)\|P_k^{2+k}(G_2\|(S_k/G_k))])$, where for a generator H and a projection P, $P(H)$ denotes a generator such that $L(P(H)) = P(L(H))$. Similarly, there exists a supervisor S_2 such that $P_{2+k}(K) = L(S_2/[G_2\|(S_k/G_k)\|P_k^{1+k}(G_1\|(S_k/G_k))])$. Since $L = L\|P_k(L)$, we get $L(G_i\|(S_k/G_k)\|P_k^{i+k}(G_i\|(S_k/G_k))) = L(G_i\|(S_k/G_k))$. Notice that $L(S_1/[G_1\|(S_k/G_k)])\| L(S_2/[G_2\|(S_k/G_k)]) = P_{1+k}(K)\|P_{2+k}(K) = K$ because K is conditionally decomposable. This proves (1).

To prove necessity, projections P_k, P_{1+k}, P_{2+k} are applied to (1). First, note that $K = L(S_1\|S_2\|S_k)\|L$, which yields $P_k(K) \subseteq L(S_k)\|L_k = L(S_k/G_k)$. On the other hand, recall that $L(S_k/G_k) \subseteq P_k(K)$ because S_k is a supervisor that enforces the coordinator part of the specification $P_k(K)$. Hence, $L(S_k/G_k) = P_k(K)$, which means that $P_k(K)$ is controllable with respect to $L(G_k)$, i.e., (i) of Definition 5 is satisfied. Now, we prove (ii.a); (ii.b) is a symmetric condition. As $E_{1+k} \cap E_{2+k} = E_k$, $L(S_2) \| L(G_2 \| (S_k/G_k)) = L(S_2) \cap L(G_2 \| (S_k/G_k))$ because the components are over the same event set E_{2+k}, and $P^{2+k}_{1+k} = P^{2+k}_k$,

$$\begin{aligned} P_{1+k}(K) &\subseteq L(S_1 \| G_1 \| (S_k/G_k) \| P_k^{2+k}(G_2 \| (S_k/G_k))) \\ &\subseteq L(S_1) \| L(S_k) \| L_1 \| L_k \\ &= L(S_1/[G_1 \| (S_k/G_k)]) \subseteq P_{1+k}(K) \,. \end{aligned}$$

Then, $G_1 \| (S_k/G_k) \| P_k^{2+k}(G_2 \| (S_k/G_k))$ is taken as a new plant, i.e., the language $P_{1+k}(K)$ is controllable with respect to $L(G_1 \| (S_k/G_k) \| P_k^{2+k}(G_2 \| (S_k/G_k)))$. Thus, (ii.a) is satisfied. □

The interest in Theorem 6 is in the computational savings in the computation of supervisors. The distributed way of constructing supervisors S_1, S_2, S_k is less complex than the construction of the supervisor for the global system $G_1 \| G_2 \| G_k$ if the coordinator event set E_k is such that the projection P_k satisfies the observer property. Naturally, conditional controllability seems to be a restrictive condition that need not be met by a given specification. However, if the specification is not conditional controllable, conditional controllable sublanguages are considered instead. It is shown in the next section that the supremal conditionally-controllable sublanguage exists and can be computed under fairly weak conditions. Moreover, if these conditions are not met, we can always add new events into E_k so that the conditions are met and the supremal conditionally-controllable sublanguage can be computed.

Note that it is required that $L(S_k/G_k) \subseteq P_k(K)$ and, similarly, $L(S_i/[G_i \| (S_k/G_k)]) \subseteq P_{i+k}(K)$, $i = 1, 2$. Otherwise stated, we are looking for necessary conditions on global specifications for having the maximal permissiveness of the language resulting by the application of the control scheme only in the (reasonable) case where safety can be achieved by the supervisors S_1, S_2, S_k. We have proven that in such a case conditional controllability is necessary for the optimality (maximal permitting). It is clear from the proof that for sufficiency we need not assume the inclusions above (cf. [8]).

In practice, it is interesting to know when safety holds when applying the control scheme combining a coordinator with local supervisors. Similarly as in the monolithic case, it may happen that the maximal acceptable behavior given by the specification K is not achievable using the coordination control scheme. By Theorem 6, such a situation occurs whenever K is not conditionally controllable. A natural question is to find the best approximation from below, i.e., a supremal conditionally-controllable sublanguage.

Theorem 7 *A union of conditionally-controllable sublanguages of a language is conditionally controllable.*

PROOF. Let I be an index set, and let K_i, $i \in I$, be sublanguages of a language $K \subseteq L(G_1 \| G_2 \| G_k)$ conditionally controllable for generators (G_1, G_2, G_k) and event sets $(E_{1+k,u}, E_{2+k,u}, E_{k,u})$. We prove that $\cup_{i \in I} K_i$ is conditionally controllable.

First, note that $P_k(\cup_{i \in I} K_i)$ is controllable with respect to $L(G_k)$ because $P_k(\cup_{i \in I} K_i) E_{k,u} \cap L(G_k) = \cup_{i \in I}(P_k(K_i) E_{k,u} \cap L(G_k)) \subseteq \cup_{i \in I} P_k(K_i) = P_k(\cup_{i \in I} K_i)$ where the inclusion is by controllability of $P_k(K_i)$ with respect to $L(G_k)$, $i \in I$.

To prove the other property, note first that, by [17], $P_k^{2+k}(L_2 \| P_k(\cup_{i \in I} K_i)) = P_k^2(L_2) \| P_k(\cup_{i \in I} K_i)$, where $L_i = L(G_i)$, $i = 1, 2, k$. Thus, we need to show that $P_{1+k}(\cup_{i \in I} K_i) E_{1+k,u} \cap L_1 \| P_k(\cup_{i \in I} K_i) \| P_k^2(L_2) \subseteq P_{1+k}(\cup_{i \in I} K_i)$. However,

$$\begin{aligned} & P_{1+k}(\cup_{i \in I} K_i) E_{1+k,u} \cap L_1 \| P_k(\cup_{i \in I} K_i) \| P_k^2(L_2) \\ &= \cup_{i \in I} \, (P_{1+k}(K_i) E_{1+k,u}) \cap \cup_{i \in I}(L_1 \| P_k(K_i) \| P_k^2(L_2)) \\ &= \cup_{i \in I} \cup_{j \in I}(P_{1+k}(K_i) E_{1+k,u} \cap L_1 \| P_k(K_j) \| P_k^2(L_2)) \,. \end{aligned}$$

Assume there are two different indexes $i, j \in I$ such that $P_{1+k}(K_i) E_{1+k,u} \cap L_1 \| P_k(K_j) \| P_k^2(L_2) \not\subseteq P_{1+k}(\cup_{i \in I} K_i)$. Then, there exist $x \in P_{1+k}(K_i)$ and $u \in E_{1+k,u}$ such that $xu \in L_1 \| P_k(K_j) \| P_k^2(L_2)$ and $xu \notin P_{1+k} \left(\cup_{i \in I} K_i\right)$. Thus, $P_k(x) \in P_k^{1+k} P_{1+k}(K_i) = P_k(K_i)$, $P_k(xu) \in P_k(K_j)$, and $P_k(xu) \notin P_k(K_i)$; otherwise, $P_k(xu) \in P_k(K_i)$ implies $xu \in L_1 \| P_k(K_i) \| P_k^2(L_2)$, and controllability of $P_{1+k}(K_i)$ with respect to $L_1 \| P_k(K_i) \| P_k^2(L_2)$ implies $xu \in P_{1+k}(K_i) \subseteq P_{1+k} \left(\cup_{i \in I} K_i\right)$, which is not true. If $u \notin E_{k,u}$, then $P_k(xu) = P_k(x) \in P_k(K_i)$, which does not hold. Thus, $u \in E_{k,u}$. As $P_k(K_i) \cup P_k(K_j) \subseteq L_k$, we get that $P_k(xu) = P_k(x)u \in L_k$. However, controllability of $P_k(K_i)$ with respect to L_k and $E_{k,u}$ implies that $P_k(x)u = P_k(xu)$ is in $P_k(K_i)$; a contradiction. □

5 Supremal sublanguages

We present a procedure for computation of the supremal conditionally-controllable sublanguage of a language K. Given generators G_1, G_2, G_k, we denote $L_i = L(G_i)$, $i = 1, 2, k$. Let $\sup \mathrm{cC}(K, L, (E_{1+k,u}, E_{2+k,u}, E_{k,u}))$ denote the supremal conditionally-controllable sublanguage of K with respect to $L = L_1 \| L_2 \| L_k$ and uncontrollable event sets $(E_{1+k,u}, E_{2+k,u}, E_{k,u})$. The following concepts (see [4,16]) are required in the next result. These concepts are adopted from hierarchical supervisory control [16]. It should not be surprising that they play a role in our study because coordination control is very much related to hierarchical control; the coordinator level can be seen as a high level of hierarchical control.

Definition 8 *A natural projection $P : E^* \to E_k^*$, where $E_k \subseteq E$, is an* L-observer *for $L \subseteq E^*$ if for all $t \in P(L)$ and $s \in \overline{L}$, if $P(s)$ is a prefix of t, then there exists $u \in E^*$ such that $su \in L$ and $P(su) = t$.*

Definition 9 *A natural projection $P : E^* \to E_k^*$, where $E_k \subseteq E$, is* output control consistent *(OCC) for $L \subseteq E^*$ if for every $s \in \overline{L}$ of the form $s = \sigma_1\sigma_2\ldots\sigma_\ell$ or $s = s'\sigma_0\sigma_1\ldots\sigma_\ell$, $\ell \geq 1$, where $\sigma_0, \sigma_\ell \in E_k$ and $\sigma_i \in E \setminus E_k$, for $i = 1, 2, \ldots, \ell - 1$, if $\sigma_\ell \in E_u$, then $\sigma_i \in E_u$, for all $i = 1, 2, \ldots, \ell - 1$.*

If L is represented by a generator with n states, then the time and space complexities to verify whether P is an L-observer are $O(n^3)$ and $O(n)$, respectively, see [4], or both complexities are $O(n^2)$ as shown in [12]. The time and space complexities of the verification whether P is OCC for L are $O(n^2)$ and $O(n)$, respectively, see [4].

Lemma 10 *Let E_1, E_2, and E_k be event sets such that $E_1 \cap E_2 \subseteq E_k$. Let $L_1 \subseteq E_1^*$ and $L_2 \subseteq E_2^*$, and let $P_k : (E_1 \cup E_2)^* \to E_k^*$ be a natural projection. Then, $P_k(L_1 \| L_2) = P_k^{1+k}(P_1^{1+k})^{-1}(L_1) \cap P_k^{2+k}(P_2^{2+k})^{-1}(L_2)$.*

PROOF. It follows from the fact that $P_k(L_1\|L_2) = P^1_{1\cap k}(L_1)\|P^2_{2\cap k}(L_2)$ shown in [17], definition of the synchronous product, and [4, Proposition 4.2(6)] showing that $(P^k_{i\cap k})^{-1}P^i_{i\cap k} = P_k^{i+k}(P_i^{i+k})^{-1}$, for $i = 1, 2$. □

The next theorem gives a computational procedure for the construction of supremal conditionally-controllable sublanguages.

Theorem 11 *Let $K \subseteq L = L_1\|L_2\|L_k$ be two prefix-closed languages over an event set $E = E_1 \cup E_2 \cup E_k$, where $L_i \subseteq E_i^*$, $i = 1, 2, k$, assume that K is conditionally decomposable, and define the languages*

$$\begin{aligned}
\sup \mathrm{C}_k &= \sup \mathrm{C}(P_k(K), L_k, E_{k,u})\,,\\
\sup \mathrm{C}_{1+k} &= \sup \mathrm{C}(P_{1+k}(K), L_1\|\sup \mathrm{C}_k, E_{1+k,u})\,,\\
\sup \mathrm{C}_{2+k} &= \sup \mathrm{C}(P_{2+k}(K), L_2\|\sup \mathrm{C}_k, E_{2+k,u})\,.
\end{aligned}$$

Let the projection P_k^{i+k} be an $(P_i^{i+k})^{-1}(L_i)$-observer and OCC for $(P_i^{i+k})^{-1}(L_i)$, for $i = 1, 2$. Then,

$$\begin{aligned}
&\sup \mathrm{C}_k\|\sup \mathrm{C}_{1+k}\|\sup \mathrm{C}_{2+k}\\
&\qquad = \sup \mathrm{cC}(K, L, (E_{1+k,u}, E_{2+k,u}, E_{k,u}))\,.
\end{aligned}$$

PROOF. Define $M := \sup \mathrm{C}_k\|\sup \mathrm{C}_{1+k}\|\sup \mathrm{C}_{2+k}$ and $\sup \mathrm{cC} := \sup \mathrm{cC}(K, L, (E_{1+k,u}, E_{2+k,u}, E_{k,u}))$. To prove $M \subseteq \sup \mathrm{cC}$, we show that (1) $M \subseteq K$ and (2) M is conditionally controllable with respect to L and $(E_{1+k,u}, E_{2+k,u}, E_{k,u})$.

1) First, notice that $M = \sup \mathrm{C}_k\|\sup \mathrm{C}_{1+k}\|\sup \mathrm{C}_{2+k} \subseteq P_k(K)\|P_{1+k}(K)\|P_{2+k}(K) = K$ because K is conditionally decomposable.

2) To prove that M is conditionally controllable with respect to L and $(E_{1+k,u}, E_{2+k,u}, E_{k,u})$, we show the properties of Definition 5.

(i) To prove that $P_k(M)E_{k,u} \cap L_k \subseteq P_k(M)$ note that $P_k(M) = \sup \mathrm{C}_k \cap P_k^{1+k}(\sup \mathrm{C}_{1+k}) \cap P_k^{2+k}(\sup \mathrm{C}_{2+k})$, which follows from [17] by $P_k(L_1\|L_2) = P_k(L_1)\|P_k(L_2)$ whenever $E_1 \cap E_2 \subseteq E_k$, and by definition of the synchronous product. Let $x \in P_k(M)$, then there exists $w \in M$ such that $P_k(w) = x$. Assume that $a \in E_{k,u}$ is such that $xa \in L_k$. We show that $xa \in P_k(M)$. As $x \in P_k(M) \subseteq \sup \mathrm{C}_k$, it follows by controllability of $\sup \mathrm{C}_k$ with respect to L_k that

$$xa \in \sup \mathrm{C}_k\,. \tag{2}$$

Thus, it remains to show that

$$xa \in P_k^{i+k}(\sup \mathrm{C}_{i+k})\,, \tag{3}$$

for $i = 1, 2$. To this end, note first that by the properties of natural projections we have

$$P_{1+k}(w) \in P_{1+k}(M) \subseteq \sup \mathrm{C}_{1+k}\,, \tag{4}$$

and $a \in E_{k,u} \subseteq E_{1+k,u}$. By definition of the synchronous product we obtain

$$L_1\|\sup \mathrm{C}_k = (P_1^{1+k})^{-1}(L_1) \cap (P_k^{1+k})^{-1}(\sup \mathrm{C}_k)\,. \tag{5}$$

Furthermore, $P_k^{1+k}(P_{1+k}(w)a) = xa \in \sup \mathrm{C}_k$ implies that $P_{1+k}(w)a \in (P_k^{1+k})^{-1}(\sup \mathrm{C}_k)$. This and

$$\begin{aligned}
\sup \mathrm{C}_k &\subseteq P_k(K)\|P_k(L_1\|L_2)\\
&= P_k(K) \cap P_k^{1+k}(P_1^{1+k})^{-1}(L_1) \qquad (6)\\
&\qquad \cap P_k^{2+k}(P_2^{2+k})^{-1}(L_2)\,, \text{ by Lemma 10}\,,
\end{aligned}$$

imply $P_k^{1+k}(P_{1+k}(w)a) \in P_k^{1+k}(P_1^{1+k})^{-1}(L_1)$. By (4) and definition of $\sup \mathrm{C}_{1+k}$, $P_{1+k}(w) \in (P_1^{1+k})^{-1}(L_1)$. As $P_k^{1+k}(P_{1+k}(w))$ is a prefix of $P_k^{1+k}(P_{1+k}(w)a)$, and P_k^{1+k} is an $(P_1^{1+k})^{-1}(L_1)$-observer, there exists $u \in E_{1+k}^*$ such that

$$P_{1+k}(w)ua \in (P_1^{1+k})^{-1}(L_1) \tag{7}$$

and $P_k^{1+k}(P_{1+k}(w)ua) = P_k^{1+k}(P_{1+k}(w)a)$, which means that $u \in (E_1 \setminus E_k)^*$. As L_1 is prefix-closed, so is $(P_1^{1+k})^{-1}(L_1)$. Thus, $P_{1+k}(w)u \in (P_k^{1+k})^{-1}(L_1)$. Notice that $P_k^{1+k}(P_{1+k}(w)u) = x \in \sup \mathrm{C}_k$, i.e., $P_{1+k}(w)u \in (P_k^{1+k})^{-1}(\sup \mathrm{C}_k)$. By (5), $P_{1+k}(w)u \in$

$L_1 \| \sup \mathrm{C}_k$. As P_k^{1+k} is OCC for $(P_1^{1+k})^{-1}(L_1)$ and $P_{1+k}(w)ua$ satisfies $a \in E_k$, $u \in (E_1 \setminus E_k)^*$, and $a \in E_u$, it follows that $u \in E_u^*$. As $P_{1+k}(w) \in \sup \mathrm{C}_{1+k}$, $\sup \mathrm{C}_{1+k}$ is controllable with respect to $L_1 \| \sup \mathrm{C}_k$ and $E_{1+k,u}$, and $P_{1+k}(w)u \in L_1 \| \sup \mathrm{C}_k$, it holds that $P_{1+k}(w)u \in \sup \mathrm{C}_{1+k}$, see [1]. Recall that $P_{1+k}(w)ua \in (P_1^{1+k})^{-1}(L_1)$ is satisfied by (7). As it also holds that $P_k^{1+k}(P_{1+k}(w)ua) = xa \in \sup \mathrm{C}_k$ by (2), we obtain by (5) that $P_{1+k}(w)ua \in L_1 \| \sup \mathrm{C}_k$, which implies, by controllability of $\sup \mathrm{C}_{1+k}$ with respect to $L_1 \| \sup \mathrm{C}_k$, that $P_{1+k}(w)ua \in \sup \mathrm{C}_{1+k}$, i.e., $xa = P_k^{1+k}(P_{1+k}(w)ua) \in P_k^{1+k}(\sup \mathrm{C}_{1+k})$. Analogously, we can prove that $xa \in P_k^{2+k}(\sup \mathrm{C}_{2+k})$, which proves (3). Thus, $xa \in P_k(M)$, which was to be shown.

(ii.a) $P_{1+k}(M)E_{1+k,u} \cap L_1 \| P_k(M) \| P_k^{2+k}(L_2 \| P_k(M)) \subseteq P_{1+k}(M)$. By [4] (showing that for $L_1 \subseteq E_1^*$, $L_2 \subseteq E_2^*$, with $E_k = E_1 \cap E_2$, and natural projections $P_i : (E_1 \cup E_2)^* \to E_i^*$, $P_k^j : E_j^* \to E_k^*$, $i = 1, 2, k$, $j = 1, 2$, $\{i, j\} = \{1, 2\}$, $P_i(L_1 \| L_2) = L_i \cap (P_k^i)^{-1} P_k^j(L_j)$) and definition of the synchronous product,

$$P_{1+k}(M) = (P_k^{1+k})^{-1}(\sup \mathrm{C}_k) \cap \sup \mathrm{C}_{1+k} \\ \cap (P_k^{1+k})^{-1} P_k^{2+k}(\sup \mathrm{C}_{2+k}) \,.$$

Assume that $x \in P_{1+k}(M)$, which is if and only if there exists $w \in M$ such that $P_{1+k}(w) = x$. Let there be $a \in E_{1+k,u}$ such that

$$xa \in L_1 \| P_k(M) \| P_k^{2+k}(L_2 \| P_k(M)) \,. \tag{8}$$

We show that $xa \in P_{1+k}(M)$. As $P_k(M) \subseteq \sup \mathrm{C}_k$, it holds that $L_1 \| P_k(M) \| P_k^{2+k}(L_2 \| P_k(M)) \subseteq L_1 \| \sup \mathrm{C}_k \| P_k^{2+k}(L_2 \| \sup \mathrm{C}_k)$. Then, by controllability of $\sup \mathrm{C}_{1+k}$ with respect to $L_1 \| \sup \mathrm{C}_k$, and by the inclusion $L_1 \| \sup \mathrm{C}_k \| P_k^{2+k}(L_2 \| \sup \mathrm{C}_k) \subseteq L_1 \| \sup \mathrm{C}_k$, we obtain that $xa \in \sup \mathrm{C}_{1+k}$. Moreover, it holds that $P_k(w) \in P_k(M) \subseteq \sup \mathrm{C}_k$ and $P_{2+k}(w) \in P_{2+k}(M) \subseteq \sup \mathrm{C}_{2+k}$ (see above).

(A) If $a \in E_1 \setminus E_k$, then $P_k^{1+k}(xa) = P_k(wa) = P_k(w)$ implies $P_k^{1+k}(xa) \in \sup \mathrm{C}_k$, and $P_k^{1+k}(xa) = P_k^{2+k} P_{2+k}(wa) = P_k^{2+k} P_{2+k}(w)$ implies $P_k^{1+k}(xa) \in P_k^{2+k}(\sup \mathrm{C}_{2+k})$. Hence, $xa \in P_{1+k}(M)$.

(B) If $a \in E_1 \cap E_k$, then $xa \in L_1 \| P_k(M)$ implies that $P_k^{1+k}(xa) \in P_k(M) \subseteq \sup \mathrm{C}_k$. Therefore, $xa \in (P_k^{1+k})^{-1}(\sup \mathrm{C}_k)$ is satisfied. It remains to show

$$xa \in (P_k^{1+k})^{-1} P_k^{2+k}(\sup \mathrm{C}_{2+k}) \,. \tag{9}$$

However, by (8) and Lemma 10 it follows that

$$P_k^{1+k}(xa) \in P_k^{2+k}(P_2^{2+k})^{-1}(L_2) \cap P_k(M) \,. \tag{10}$$

In addition, it holds, by definition of $\sup \mathrm{C}_{2+k}$, that $P_{2+k}(w) \in (P_2^{2+k})^{-1}(L_2)$. As $P_k^{2+k}(P_{2+k}(w))$ is a prefix of $P_k^{2+k}(P_{2+k}(w)a)$, $P_k^{2+k}(P_{2+k}(w)a) = P_k^{1+k}(x)a \in P_k(M) \subseteq \sup \mathrm{C}_k \subseteq P_k^{2+k}(P_2^{2+k})^{-1}(L_2)$, and P_k^{2+k} is an $(P_2^{2+k})^{-1}(L_2)$-observer, there exists $u \in E_{2+k}^*$ such that

$$P_{2+k}(w)ua \in (P_2^{2+k})^{-1}(L_2) \tag{11}$$

where $P_k^{2+k}(P_{2+k}(w)ua) = P_k^{2+k}(P_{2+k}(w)a)$, i.e., $u \in (E_2 \setminus E_k)^*$. As L_2 is prefix-closed, so is $(P_2^{1+k})^{-1}(L_2)$. Therefore, $P_{2+k}(w)u \in (P_2^{2+k})^{-1}(L_2)$ is satisfied. Furthermore, $P_k^{2+k}(P_{2+k}(w)u) = P_k^{1+k}(x) \in P_k(M) \subseteq \sup \mathrm{C}_k$ means that $P_{2+k}(w)u \in (P_k^{2+k})^{-1}(\sup \mathrm{C}_k)$. Together and with definition of the synchronous product, $P_{2+k}(w)u \in L_2 \| \sup \mathrm{C}_k$. As P_k^{2+k} is OCC for $(P_2^{2+k})^{-1}(L_2)$, and $P_{2+k}(w)ua$ satisfies $a \in E_k$, $u \in (E_2 \setminus E_k)^*$, and $a \in E_u$, it holds that $u \in E_u^*$. As $P_{2+k}(w) \in \sup \mathrm{C}_{2+k}$, $\sup \mathrm{C}_{2+k}$ is controllable with respect to $L_2 \| \sup \mathrm{C}_k$, and $P_{2+k}(w)u \in L_2 \| \sup \mathrm{C}_k$ is satisfied, $P_{2+k}(w)u \in \sup \mathrm{C}_{2+k}$. Finally, as $P_k^{2+k}(P_{2+k}(w)ua) = P_k^{1+k}(x)a \in P_k(M) \subseteq \sup \mathrm{C}_k$, by (10), it follows by this, (11), and definition of the synchronous product that $P_{2+k}(w)ua \in L_2 \| \sup \mathrm{C}_k$. By this and controllability of $\sup \mathrm{C}_{2+k}$ with respect to $L_2 \| \sup \mathrm{C}_k$, $P_{2+k}(w)ua \in \sup \mathrm{C}_{2+k}$, i.e., $P_k^{1+k}(x)a = P_k^{2+k}(P_{2+k}(w)ua) \in P_k^{2+k}(\sup \mathrm{C}_{2+k})$, which proves (9). Thus, $xa \in P_{1+k}(M)$.

As condition (ii.b) of Definition 5 is analogous, we have shown that M is conditionally controllable with respect to $L = L_1 \| L_2 \| L_k$ and $(E_{1+k,u}, E_{2+k,u}, E_{k,u})$, i.e., $M \subseteq \sup \mathrm{cC}$.

To prove the opposite inclusion, $\sup \mathrm{cC} \subseteq M$, it is sufficient to show that (1) $P_k(\sup \mathrm{cC}) \subseteq \sup \mathrm{C}_k$ and (2) $P_{i+k}(\sup \mathrm{cC}) \subseteq \sup \mathrm{C}_{i+k}$, for $i = 1, 2$. To prove this note that $P_k(\sup \mathrm{cC}) \subseteq P_k(K) \subseteq L_k$. As $P_k(\sup \mathrm{cC})$ is controllable with respect to L_k and $E_{k,u}$, $P_k(\sup \mathrm{cC}) \subseteq \sup \mathrm{C}_k$ is satisfied. Furthermore, $P_{1+k}(\sup \mathrm{cC}) \subseteq P_{1+k}(K) \subseteq L_1 \| L_k$. Moreover, $P_{1+k}(\sup \mathrm{cC})$ is controllable with respect to $L_1 \| P_k(\sup \mathrm{cC}) \| P_k^{2+k}(L_2 \| P_k(\sup \mathrm{cC}))$ and $E_{1+k,u}$. By (6), $P_k(\sup \mathrm{cC}) \subseteq \sup \mathrm{C}_k \subseteq P_k^{2+k}(P_2^{2+k})^{-1}(L_2)$. The following holds.

$$\begin{aligned} & L_1 \| P_k(\sup \mathrm{cC}) \| P_k^{2+k}(L_2 \| P_k(\sup \mathrm{cC})) \\ & = L_1 \| P_k(\sup \mathrm{cC}) \| P_k(\sup \mathrm{cC}) \cap P_k^{2+k}(P_2^{2+k})^{-1}(L_2) \\ & = L_1 \| P_k(\sup \mathrm{cC}) \,. \end{aligned}$$

As $P_k(\sup \mathrm{cC})$ is controllable with respect to L_k, it is also controllable with respect to $\sup \mathrm{C}_k \subseteq L_k$. As $P_{1+k}(\sup \mathrm{cC})$ is controllable with respect to $L_1 \| P_k(\sup \mathrm{cC})$, and $L_1 \| P_k(\sup \mathrm{cC})$ is controllable with respect to $L_1 \| \sup \mathrm{C}_k$ by [4, Proposition 4.6] (as all the languages under consideration are prefix-closed), it follows by Lemma 12 (see below) that $P_{1+k}(\sup \mathrm{cC})$ is controllable with respect to $L_1 \| \sup \mathrm{C}_k$, which implies that $P_{1+k}(\sup \mathrm{cC}) \subseteq \sup \mathrm{C}_{1+k}$. The case of property (ii.b) is proven analogously. Hence, we have proven that $\sup \mathrm{cC} \subseteq M$ and the proof is complete. □

Lemma 12 (Transitivity of controllability) *Let $K \subseteq L \subseteq M$ be languages over an event set E such that K is controllable with respect to L and E_u, and L is controllable with respect to M and E_u. Then, K is controllable with respect to M and E_u.*

PROOF. From $KE_u \cap L \subseteq K$ and $LE_u \cap M \subseteq L$, we show that $KE_u \cap M \subseteq K$. Assume that $s \in K$, $a \in E_u$, and $sa \in M$. As $K \subseteq L$, $s \in L$, which implies that $sa \in L$ by controllability of L with respect to M. However, $sa \in L$ implies $sa \in K$, by controllability of K with respect to L. □

Another consequence is of interest. Namely, under the conditions of Theorem 11, $\sup \mathrm{cC}$ is conditionally decomposable (cf. the discussion below Definition 3). Moreover, $\sup \mathrm{cC}$ is controllable with respect to the global plant.

Corollary 13 *In the setting of Theorem 11, the language* $\sup \mathrm{cC} := \sup \mathrm{cC}(K, L, (E_{1+k,u}, E_{2+k,u}, E_{k,u}))$ *is controllable with respect to L and E_u.*

PROOF. It is sufficient to show that $\sup \mathrm{cC}$ is controllable with respect to $L = L_1 \| L_2 \| L_k$. There exist $\sup \mathrm{C}_k \subseteq E_k^*$, $\sup \mathrm{C}_{1+k} \subseteq E_{1+k}^*$, and $\sup \mathrm{C}_{2+k} \subseteq E_{2+k}^*$ as defined in Theorem 11 so that $\sup \mathrm{cC} = \sup \mathrm{C}_k \| \sup \mathrm{C}_{1+k} \| \sup \mathrm{C}_{2+k}$. In addition, $\sup \mathrm{C}_k$ is controllable with respect to L_k, $\sup \mathrm{C}_{1+k}$ is controllable with respect to $L_1 \| \sup \mathrm{C}_k$, $\sup \mathrm{C}_{2+k}$ is controllable with respect to $L_2 \| \sup \mathrm{C}_k$. By [4, Proposition 4.6] (as the languages under consideration are prefix-closed) $\sup \mathrm{C}_k \| \sup \mathrm{C}_{1+k} \| \sup \mathrm{C}_{2+k}$ is controllable with respect to $L_k \| (L_1 \| \sup \mathrm{C}_k) \| (L_2 \| \sup \mathrm{C}_k) = L \| \sup \mathrm{C}_k$. Analogously, $L \| \sup \mathrm{C}_k$ is controllable with respect to $L \| L_k = L$. By Lemma 12, $\sup \mathrm{cC}$ is controllable with respect to L. □

We show that if some additional conditions are satisfied, the resulting supremal conditionally-controllable sublanguage constructed in Theorem 11 is optimal. The conditions of Theorem 11 imply that P_k is OCC for L.

Lemma 14 *Let $L_i \subseteq E_i^*$, $i = 1, 2$, be two (prefix-closed) languages, and let $P_i : (E_1 \cup E_2)^* \to E_i^*$, $i = 1, 2, k$ and $E_k \subseteq E_1 \cup E_2$, be natural projections. Let $E_u \subseteq E_1 \cup E_2$ be the set of uncontrollable events. If $E_1 \cap E_2 \subseteq E_k$ and P_k^{i+k} is OCC for $(P_i^{i+k})^{-1}(L_i)$, $i = 1, 2$, then P_k is OCC for $L = L_1 \| L_2 \| L_k$.*

PROOF. For $s \in L$ of the form $s = s'\sigma_0\sigma_1 \ldots \sigma_{k-1}\sigma_k$, for some $k \geq 1$, assume that $\sigma_0, \sigma_k \in E_k$, $\sigma_i \in E \setminus E_k$, for $i = 1, 2, \ldots, k-1$, and $\sigma_k \in E_u$. We show that $\sigma_i \in E_u$, for $i = 1, 2, \ldots, k-1$. However, $P_{i+k}(s) = P_{i+k}(s')\sigma_0 P_{i+k}(\sigma_1 \ldots \sigma_{k-1})\sigma_k \in (P_i^{i+k})^{-1}(L_i)$ and the OCC property imply that $P_{i+k}(\sigma_1 \ldots \sigma_{k-1}) \in E_u^*$, for $i = 1, 2$. Let $\sigma \in \{\sigma_1, \sigma_2, \ldots, \sigma_{k-1}\}$. Then, $\sigma \in (E_1 \cup E_2) \setminus E_k$. Without loss of generality, assume that $\sigma \in E_1$. Then, $P_{1+k}(\sigma) = \sigma \in E_u$ and $P_{2+k}(\sigma) = \varepsilon \in E_u^*$. Thus, $\{\sigma_1, \sigma_2, \ldots, \sigma_{k-1}\} \subseteq E_u$. □

Theorem 15 *Consider the setting of Theorem 11. If, in addition, $L_k \subseteq P_k(L)$ and P_{i+k} is OCC for $P_{i+k}^{-1}(L_i \| L_k)$, for $i = 1, 2$, then $\sup \mathrm{C}(K, L, E_u) = \sup \mathrm{cC}(K, L, (E_{1+k,u}, E_{2+k,u}, E_{k,u}))$.*

PROOF. One inclusion is shown in Corollary 13. We prove the other inclusion. By the assumptions P_k^{i+k} is the $(P_i^{i+k})^{-1}(L_i)$-observer, $i = 1, 2$, and P_k^k is an L_k-observer, as the observer property holds for identities. Proposition 4.5 in [4] applied to P_k^{1+k} and P_k^{2+k} implies that P_k is an $(P_1^{1+k})^{-1}(L_1) \| (P_2^{2+k})^{-1}(L_2) = L_1 \| L_2$-observer. Another application of this result to P_k and P_k^k implies that P_k is an $(L_1 \| L_2) \| L_k = L$-observer. By Lemma 14, P_k is OCC for L. Denote $\sup \mathrm{C} := \sup \mathrm{C}(K, L, E_u)$. We prove that $P_k(\sup \mathrm{C})$ is controllable with respect to L_k. Assume $t \in P_k(\sup \mathrm{C})$, $a \in E_{k,u}$, and $ta \in L_k \subseteq P_k(L)$. Then, there is $s \in \sup \mathrm{C}$ such that $P_k(s) = t$. As P_k is the L-observer, there is $v \in E^*$ such that $sv \in L$ and $P_k(sv) = P_k(s)P_k(v) = ta$, i.e., $v = ua$, for some $u \in (E \setminus E_k)^*$. By the OCC property of P_k, $u \in E_u^*$. By controllability of $\sup \mathrm{C}$ with respect to L, $sua \in \sup \mathrm{C}$, i.e., $P_k(sua) = ta \in P_k(\sup \mathrm{C})$. Thus, (i) of Definition 5 holds.

Note that the identities P_{i+k}^{i+k} are the $(P_i^{i+k})^{-1}(L_i)$-observers, $i = 1, 2$, that $P_{j+k}^{i+k} = P_k^{i+k}$ is the $(P_i^{i+k})^{-1}(L_i)$-observer, $\{i, j\} = \{1, 2\}$, and that $P_k^k = P_{i+k}^k$ is the L_k-observer, $i = 1, 2$. Similarly as above, Proposition 4.5 in [4] applied to projections P_{i+k}^{i+k}, P_{i+k}^{j+k}, $j \neq i$, and P_{i+k}^k implies that P_{i+k} are L-observers, for $i = 1, 2$. To prove (ii) of Definition 5, assume that, for some $1 \leq i \leq 2$, $t \in P_{i+k}(\sup \mathrm{C})$, $a \in E_{i+k,u}$, and $ta \in L_i \| P_k(\sup \mathrm{C}) \| P_k^{j+k}(L_j \| P_k(\sup \mathrm{C}))$, for $j \neq i$. Then, there exists $s \in \sup \mathrm{C}$ such that $P_{i+k}(s) = t$. As P_{i+k} is the L-observer, and $L_i \| P_k(\sup \mathrm{C}) \| P_k^{j+k}(L_j \| P_k(\sup \mathrm{C}))$ is a subset of $P_{i+k}(L) = L_i \| L_k \| P_k^{j+k}(L_j \| L_k)$, $j \neq i$, because $P_k(\sup \mathrm{C}) \subseteq P_k(K) \subseteq P_k(L) \subseteq L_k$, there exists $v \in E^*$ such that $sv \in L$ and $P_{i+k}(sv) = P_{i+k}(s)P_{i+k}(v) = ta$, i.e., $v = ua$, for some $u \in (E \setminus E_{i+k})^*$. Since P_{i+k} is OCC for $P_{i+k}^{-1}(L_i \| L_k)$ and $sua \in L \subseteq P_{i+k}^{-1}(L_i \| L_k)$, we obtain that $u \in E_u^*$. Then, controllability of $\sup \mathrm{C}$ with respect to L and E_u implies that $sua \in \sup \mathrm{C}$. This means that $P_{i+k}(sua) = ta \in P_{i+k}(\sup \mathrm{C})$. □

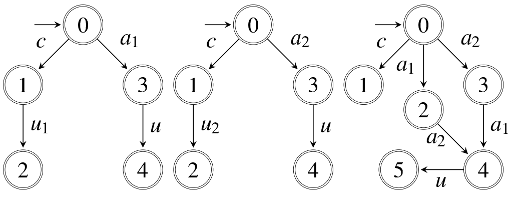

Fig. 1. Generators G_1, G_2, and the coordinator.

It is sufficient to assume that P_{i+k} is OCC for L, which is less restrictive than the used assumption. However, it is an open problem how to verify this assumption without computing the whole plant. On the other hand, if P_{i+k} is OCC for $P_i^{-1}(L_i)$, $i = 1, 2$, then the theorem holds as well. In addition, the assumptions do not imply that all controllable events are contained in the event set of the coordinator as shown in the following examples.

The complexity of the computation of the supremal controllable sublanguage of a language K with respect to L with n and m states in their minimal generator representations, respectively, is $O(mn)$ for prefix-closed languages [10]. Denote the number of states of minimal generators for $L(G_1)$, $L(G_2)$, $L(G_k)$ by m_1, m_2, m_k, respectively. As the language K is conditionally decomposable, $K = P_{1+k}(K) \| P_{2+k}(K) \| P_k(K)$, we denote the number of states of minimal generators for $P_{1+k}(K)$, $P_{2+k}(K)$, $P_k(K)$ by n_1, n_2, n_k, respectively. Then, in the worst case, $m = O(m_1m_2m_k)$ and $n = O(n_1n_2n_k)$. The computational complexity of $\sup \mathrm{C}_k$, $\sup \mathrm{C}_{1+k}$, $\sup \mathrm{C}_{2+k}$ results in the formula $O(m_kn_k + m_1n_1m_kn_k + m_2n_2m_kn_k)$, which is better than $O(mn) = O(m_1m_2m_kn_1n_2n_k)$ in the monolithic case.

Recall that both the output control consistency and the observer property, standard notions used in hierarchical supervisory control, can be checked in polynomial time, cf. [17]. However, if E_k does not satisfy these properties it does not mean that we cannot compute a controllable sublanguage without building the global plant. It suffices to extend E_k by adding new events so that P_k satisfies the observer and OCC properties required in Theorem 11 to compute the supremal conditionally-controllable sublanguage. Let us recall from [5] that finding a minimal extension of an event set that satisfies the observer property is NP-hard, but there is a polynomial-time algorithm that finds a satisfactory extension.

Example 16 *Let $G = G_1 \| G_2$ be a system over the event set $E = \{a_1, a_2, c, u, u_1, u_2\}$, where G_1 and G_2 are defined in Fig. 1, $E_u = \{u, u_1, u_2\}$. The specification K is defined in Fig. 2. The coordinator event set E_k has to contain shared events c and u, and to make K conditionally decomposable, at least one of a_1, a_2 has to be in E_k. The projections must satisfy observer and OCC properties. If a_1 or a_2 is not in E_k, P_k^{i+k} is not OCC for $(P_i^{i+k})^{-1}(L_i)$. Thus, $E_k = \{a_1, a_2, c, u\}$. The coordinator is now defined as $G_k = P_k(G_1) \| P_k(G_2)$, see Fig. 1. Then, $P_k(K) = \overline{\{a_2a_1, c, a_1a_2u\}}$, $P_{1+k}(K) = \overline{\{a_1a_2u, a_2a_1, cu_1\}}$, and $P_{2+k}(K) = \overline{\{a_1a_2u, a_2a_1, cu_2\}}$. As the assumptions of Theorem 11 are satisfied, we can compute $\sup \mathrm{C}_k = \overline{\{a_2, c, a_1a_2u\}}$, $\sup \mathrm{C}_{1+k} = \overline{\{a_1a_2u, a_2, cu_1\}}$, $\sup \mathrm{C}_{2+k} = \overline{\{a_1a_2u, a_2, cu_2\}}$, whose synchronous product results in the supremal conditionally-controllable sublanguage $\overline{\{a_1a_2u, a_2, cu_1u_2, cu_2u_1\}}$ of K that coincides with the supremal controllable sublanguage of K. Note that the subsystems are not mutually controllable, thus the approach of [9] cannot be used.*

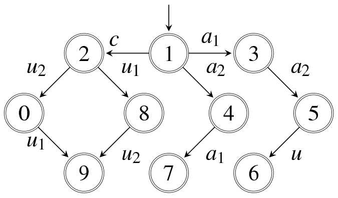

Fig. 2. Generator for the specification K.

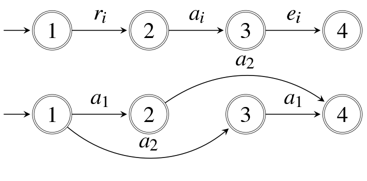

Fig. 3. Generators G_i, $i = 1, 2$, and the coordinator G_k.

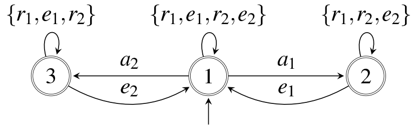

Fig. 4. The specification K.

Example 17 (Concurrent access to a database) *Database transactions are typical examples of DES that should be controlled to avoid incorrect behaviors. Our model of a transaction to the database is a sequence of request, access (read), and exit (write) operations. Often, several users access the database, which can lead to inconsistencies when executed concurrently because not all interleaving of operations gives a correct behavior. Consider two users and events r_i, a_i, e_i, $i = 1, 2$. All possible schedules are given by the language of the plant $G = G_1 \| G_2$ over the event set $E = \{r_1, r_2, a_1, a_2, e_1, e_2\}$, where G_1 and G_2 are defined as in Fig. 3, and $E_c = \{a_1, a_2, e_1, e_2\}$. The specification K, depicted in Fig. 4, describes the correct behavior consisting in finishing the transaction in the write stage before another transaction can proceed to the write phase. For $E_k = \{a_1, a_2\}$ and the coordinator $G_k = P_k(G_1) \| P_k(G_2)$, the assumptions of Theorem 11 are satisfied. Thus, we compute $\sup \mathrm{C}_k$, $\sup \mathrm{C}_{1+k}$, $\sup \mathrm{C}_{2+k}$, see Fig. 5. The solution is optimal: the supremal conditionally-controllable sublanguage of K coincides with the supremal controllable sublanguage of K. Note that $K \not\subseteq L(G)$, which is no problem because the computation of supremal sublanguages of K with respect*

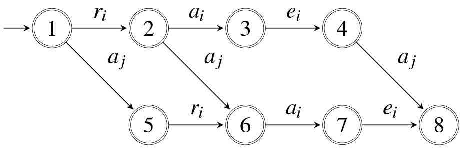

Fig. 5. $supC_{i+k}$, $i = 1, 2$, where $j = 1, 2$, $j \neq i$; $supC_k = G_k$.

to $L(G)$ results in sublanguages of $K \cap L(G)$. The benefit of this representation is a space saving. Moreover, extending the example to three or more subsystems means to add new events a_i and e_i into E_c, a_i into E_k, for $i \geq 3$, and modify the specification in a natural way. The required space results in a square root of the number of states needed by the global plant.

6 Conclusion and discussion

We have considered supervisory control of distributed DES with global specifications. A coordination control framework has been adopted where, unlike the purely decentralized setting, a global layer with a coordinator acting on a subset of the global event set has been added. Two main results have been presented—a necessary and sufficient condition on a specification to be exactly achieved, and the synthesis of the supremal conditionally-controllable sublanguage and its relation to the supremal controllable sublanguage. The approach is general, any distributed plant and any specification can be treated within this control architecture, which is a considerable difference with earlier approaches to control of distributed DES with global specifications.

As blocking is not considered, it is sufficient to choose a suitable coordinator event set and the coordinator itself need not impose any restriction on the behavior because its supervisor takes care of the required restriction. We will address the blocking issue in the future. In particular, the synthesis of coordinators for nonblocking and an extension to partially observed distributed plants.

The approach can be extended to more subsystems with one central coordinator, whose event set contains all shared events (which is a common assumption, see [2,15]). Specifically, Condition (ii) of Definition 5 results in $P_{i+k}(K)$ is controllable with respect to $P_{i+k}(\|_{i=1}^{n} L(G_i) \| P_k(K)) = \|_{i=1}^{n} P_{i+k}(L(G_i) \| P_k(K))$ and $E_{i+k,u}$. In the future work, multi-level coordination architectures depending on coupling local components will be studied.

Finally, note that it is clear that there have to be procedures and algorithms for the exceptional circumstances like breakdown of communication and other failures.

Acknowledgements

Comments by K. Schmidt are gratefully acknowledged. The research received funding from the 7^{th} Framework Programme grant INFSO-ICT-224498, from the Czech Acad. Sci., Inst. Research Plan AV0Z10190503, and from the GAČR grants 103/11/0517 and P202/11/P028.

References

[1] R.D. Brandt, V. Garg, R. Kumar, F. Lin, S.I. Marcus, and W.M. Wonham. Formulas for calculating supremal controllable and normal sublanguages. *Systems Control Lett.*, 15(2):111–117, 1990.

[2] K. Cai and W.M. Wonham. Supervisor localization: A top-down approach to distributed control of discrete-event systems. *IEEE Trans. Automat. Control*, 55(3):605–618, 2010.

[3] C.G. Cassandras and S. Lafortune. *Introduction to discrete event systems, second edition.* Springer, 2008.

[4] L. Feng. *Computationally Efficient Supervisor Design for Discrete-Event Systems.* PhD thesis, Univ. of Toronto, 2007.

[5] L. Feng and W.M. Wonham. On the computation of natural observers in discrete-event systems. *Discrete Event Dyn. Syst.*, 12(3):63–102, 2008.

[6] B. Gaudin and H. Marchand. Supervisory control of product and hierarchical discrete event systems. *Eur. J. Control*, 10(2):131–145, 2004.

[7] J. Komenda, T. Masopust, and J.H. van Schuppen. Synthesis of safe sublanguages satisfying global specification using coordination scheme for discrete-event systems. In *Proc. of WODES 2010*, pages 436–441, 2010. http://www.ifac-papersonline.net/.

[8] J. Komenda and J.H. van Schuppen. Coordination control of discrete event systems. In *Proc. of WODES 2008*, pages 9–15, 2008.

[9] J. Komenda, J.H. van Schuppen, B. Gaudin, and H. Marchand. Supervisory control of modular systems with global specification languages. *Automatica*, 44(4):1127–1134, 2008.

[10] R. Kumar, V. Garg, and S.I. Marcus. On controllability and normality of discrete event dynamical systems. *Systems Control Lett.*, 17(3):157–168, 1991.

[11] R.J. Leduc, D. Pengcheng, and S. Raoguang. Synthesis method for hierarchical interface-based supervisory control. *IEEE Trans. Automat. Control*, 54(7):1548–1560, 2009.

[12] P.N. Pena, J.E.R. Cury, and S. Lafortune. Polynomial-time verication of the observer property in abstractions. In *Proc. of ACC 2008*, pages 465–470, Seattle, USA, 2008.

[13] P.J. Ramadge and W.M. Wonham. Supervisory control of a class of discrete event processes. *SIAM J. Control Optim.*, 25(1):206–230, 1987.

[14] P.J. Ramadge and W.M. Wonham. The control of discrete event systems. *Proc. of IEEE*, 77(1):81–98, 1989.

[15] K. Schmidt, T. Moor, and S. Perk. Nonblocking hierarchical control of decentralized discrete event systems. *IEEE Trans. Automat. Control*, 53(10):2252–2265, 2008.

[16] K.C. Wong and W.M. Wonham. Hierarchical control of discrete-event systems. *Discrete Event Dyn. Syst.*, 6(3):241–273, 1996.

[17] W.M. Wonham. Supervisory control of discrete-event systems. Lecture notes, Department of electrical and computer engineering, Univ. of Toronto, 2009.

[18] T.S. Yoo and S. Lafortune. A general architecture for decentralized supervisory control of discrete-event systems. *Discrete Event Dyn. Syst.*, 12(3):335–377, 2002.