Multilevel Coordination Control of Partially Observed Modular DES

Jan Komenda, Tomáš Masopust, and Jan H. van Schuppen

Abstract—Coordination control for multi-level discrete-event systems is generalized to supervisory control with partial observations. The multi-level system architecture is in this paper restricted to three levels, which include two levels of coordination on the two highest levels. At Level 1 a coordinator coordinates several coordinators of Level 2 and at Level 2 each coordinator coordinates a set of subsystems of Level 3. The problem is to synthesize a set of supervisors based on partial observations and on top-down synthesis, which together achieve the control objectives. A necessary and sufficient condition for the existence of such supervisors is three-level conditional controllability and three-level conditional observability. A procedure is formulated and proven (Theorem 16) to construct the supremal three-level conditionally controllable and conditionally normal sublanguage of the specification language.

I. INTRODUCTION

Large-scale technological systems consist nowadays of large networks of subsystems. A realistic model is then a distributed system of a synchronous product of finite automata. Supervisory control synthesis of a distributed system has to overcome the major research issues of complexity and of nonblockingness. Coordination control of distributed systems with synchronous communication has been developed by the authors, see [1] and the references therein, in which a coordinator restricts the behavior of two or more subsystems so that, after further control synthesis, safety and nonblockingness of the distributed system are achieved.

To further limit the complexity of control synthesis, a multi-level system architecture and multi-level control synthesis have been developed. In the multi-level system architecture, there are three levels. At Level 1 a coordinator coordinates several coordinators of Level 2 and at Level 2 each coordinator coordinates a set of subsystems of Level 3, see [2]. This architecture considerably limits the computational complexity due to relatively small event sets at the various levels.

In [3], the authors have proven results on the existence of a set of supervisors based on complete observations at the various levels of a multi-level discrete-event system.

In this paper, supervisory control of multi-level discrete-event systems is generalized to supremal supervisors based on partial observations and for a prefixed-closed specification. The concepts of three-level conditional observability and of three-level conditional normality are introduced. It is proven that such supervisors based on partial observations exist if and only if three-level conditional controllability and three-level conditional observability both hold.

For supremal supervisors the concept of three-level conditional normality is introduced. A procedure is formulated and proven to be correct for the computation of the supremal three-level conditionally controllable and conditionally normal sublanguage. The formulations of the concepts are uniform for all levels and use controllability and normality.

The literature on hierarchical approach to supervisory control of a network of discrete-event systems is rather limited. The hierarchical approach of K. Schmidt, cf. [4] is restricted to two levels of hierarchy.

The content of the paper consists of a summary of coordination control in Section II, the main results on multi-level coordination control of multi-level discrete-event systems in Section III, and concluding remarks in the last section.

II. COORDINATION CONTROL OF PARTIALLY OBSERVED MODULAR DES

In this section elementary notions of supervisory control theory are first recalled. The reader is referred to [5] for more details.

Let $A$ be a finite nonempty set of events, and let $A^*$ denote the set of all finite words over $A$. The empty word is denoted by $e$. Let $|A|$ denote the cardinality of $A$.

A generator is a quintuple $G = (Q, A, f, q_0, Q_m)$, where $Q$ is the finite nonempty set of states, $A$ is the event set, $f : Q \times A \rightarrow Q$ is the partial transition function, $q_0 \in Q$ is the initial state, and $Q_m \subseteq Q$ is the set of marked states. In the usual way, the transition function $f$ can be extended to the domain $Q \times A^*$ by induction.

The behavior of $G$ is described in terms of languages. The language generated by $G$ is the set $L(G) = \{ s \in A^* | f(q_0, s) \in Q \}$ and the language marked by $G$ is the set $L_m(G) = \{ s \in A^* | f(q_0, s) \in Q_m \} \subseteq L(G)$.

A (regular) language $L$ over an event set $A$ is a set $L \subseteq A^*$ such that there exists a generator $G$ with $L_m(G) = L$. The prefix closure of $L$ is the set $\overline{L} = \{ w \in A^* | \exists u \in A^* \text{ such that } wu \in L \}$; $L$ is prefix-closed if $L = \overline{L}$. In this paper it is assumed that all languages are prefix-closed.

A (natural) projection $P : A^* \rightarrow A_o^*$, for some $A_o \subseteq A$, is a homomorphism defined so that $P(a) = e$, for $a \in A \setminus A_o$, and $P(a) = a$, for $a \in A_o$. The inverse image of $P$, denoted by $P^{-1} : A_o^* \rightarrow 2^{A^*}$, is defined as $P^{-1}(s) = \{ w \in A^* | P(w) = s \}$.

The definitions can naturally be extended to languages. The projection of a generator $G$ is a generator $P(G)$ whose behavior satisfies $L(P(G)) = P(L(G))$ and $L_m(P(G)) = P(L_m(G))$.

A controlled generator is a structure $(G_c, P, \Gamma)$, where $G$ is a generator over $A$, $A_c \subseteq A$ is the set of controllable events,
A_u = A \setminus A_c$ is the set of uncontrollable events, $P : A^* \rightarrow A_u^*$ is the projection, and $\Gamma = \{ \gamma \subseteq A \mid A_u \subseteq \gamma \}$ is the set of control patterns.

A supervisor for the controlled generator $(G, A_c, P, \Gamma)$ is a map $S : P(L(G)) \rightarrow \Gamma$. A closed-loop system associated with the controlled generator $(G, A_c, P, \Gamma)$ and the supervisor $S$ is defined as the smallest language $L(S/G) \subseteq A^+$ such that

1) $\varepsilon \in L(S/G)$ and
2) if $s \in L(S/G)$, $sa \in L(G)$, and $a \in S(p(s))$, then also $sa \in L(S/G)$.

Let $G$ be a generator over $A$, and let $K = \bar{K} \subseteq L_m(G)$ be a specification. The aim of supervisor control theory is to find a supervisor $S$ such that $L(S/G) = K$. It is known that such a supervisor exists if and only if $K$ is

1) controllable with respect to $L(G)$ and $A_u$:
that is, $KA_u \cap L \subseteq K$, and
2) observable with respect to $L(G)$, $A_o$, and $A_c$:
that is, for all $s \in K$ and $\sigma \in A_c$, if $s\sigma \notin K$ and $s\sigma \in L(G)$, then $P^{-1}(P(s))\sigma \cap K = \emptyset$, where $P : A^* \rightarrow A_u^*$.

The synchronous product (parallel composition) of languages $L_1 \subseteq A_1^*$ and $L_2 \subseteq A_2^*$ is defined by

$$L_1 \parallel L_2 = P_1^{-1}(L_1) \cap P_2^{-1}(L_2) \subseteq A^*,$$

where $P_1 : A^* \rightarrow A_1^*$ and $P_2 : A^* \rightarrow A_2^*$ with $A_2 \subseteq B_1 \subseteq A$, it holds that $P_1(L) \parallel P_2(L) = P_1(L)$.

Let $G$ be a generator over $A$, and let $Q : A^* \rightarrow A^*$ be a natural projection. A language $K \subseteq L(G)$ is normal with respect to $L(G)$ and $Q$ if $K = Q^{-1} Q(K) \cap L(G)$.

For prefix-closed languages that are exclusively considered in this paper nonconflictness holds trivially, hence controllability is preserved by the synchronous product. It is easy to show that the same holds for normality.

Lemma 2: For $i = 1, 2, \ldots, n$, let $K_i \subseteq L_i$ be controllable with respect to $L_i \subseteq A_i^*$ and $A_{i,u}$, and normal with respect to $L_i$ and $Q_i$, where $Q_i : A_i^* \rightarrow A_{i,o}$ are natural projections that define partial observations in subsystems. Then $\bigcap_{i=1}^n K_i$ is controllable with respect to $\bigcap_{i=1}^n L_i$ and $\bigcap_{i=1}^n A_{i,u}$ and normal with respect to $\bigcap_{i=1}^n L_i$ and $Q_i$, where $Q_i : \left( \bigcup_{i=1}^n A_i \right)^* \rightarrow \left( \bigcup_{i=1}^n A_{i,o} \right)^*$ is the natural projection that describes partial observations over the global alphabet.

Now we recall the basic notions of coordination control [1]. A language $K$ over $\bigcup_{i=1}^n A_i^*$ is conditionally decomposable with respect to alphabets $(A_{i,u})_{i=1}^n$ and $A_k$, where $\bigcup_{i=1}^n \bigcup_{j \leq n} (A_i \cap A_j) \subseteq A_k \subseteq \bigcup_{i=1}^n A_i^*$, if

$$K = \bigcap_{i=1}^n P_i(k),$$

for projections $P_i(k)$ from $\bigcup_{i=1}^n A_j$ to $A_j \cup A_k$, for $i = 1, 2, \ldots, n$. The alphabet $A_k$ is referred to as a coordinator alphabet and satisfies the conditional independence property, that is, $A_k$ includes all shared events:

$$A_{sh} = \bigcup_{1 \leq i,j \leq n} (A_i \cap A_j) \subseteq A_k.$$

This has the following well-known impact.

Lemma 3 ([6]): Let $P : A^* \rightarrow A_u^*$ be a projection, and let $L_i$ be a language over $A_i$, for $i = 1, 2, \ldots, n$, and $\bigcup_{1 \leq i,j \leq n} (A_i \cap A_j) \subseteq A_k$. Then $P_i (|_{i=1}^n L_i) = |_{i=1}^n P_i(L_i)$.

The problem of coordination control synthesis is now recalled.

Problem 4: Let $G_i$, for $i = 1, 2, \ldots, n$, be local generators over the event sets $A_i$ of a modular plant $G = \bigcup_{i=1}^n G_i$, and let $G_k$ be a coordinator over an alphabet $A_k$. Consider a prefix-closed specification language $K \subseteq L(G_i[G_k])$. Assume that $A_k \supseteq A_{sh}$ and that the specification language $K$ is conditionally decomposable with respect to event sets $(A_i)_{i=1}^n$ and $A_k$.

The global specification is divided into the local subtasks and the coordinator subtask as in [7]. The coordinator takes care of its “part” of the specification, namely $P_k(K)$, i.e., $L(S_i/G_k) \subseteq P_k(K)$. Similarly, the supervisors $S_i$ take care of their corresponding “parts” of the specification, namely $P_{i+k}(K)$, i.e., $L(S_i/[G_i](S_k/G_k)) \subseteq P_{i+k}(K)$, for $i = 1, 2, \ldots, n$.

The aim is to determine the supervisors $S_1, S_2, \ldots, S_n$, and $S_k$ for the respective generators so that the closed-loop system with the coordinator is such that

$$|_{i=1}^n L(S_i/[G_i](S_k/G_k)) = K.$$

Conditional controllability along with conditional observability form an equivalent condition for a language to be achieved by the closed-loop system within our coordination control architecture, see below.

A language $K \subseteq L(G_i[G_2]\ldots[G_n][G_k])$ is conditionally observable for generators $G_1, G_2, \ldots, G_n$ and a coordinator $G_k$ and uncontrollable alphabets $A_{i,u}$, $i = 1, 2, \ldots, n$, and $A_{k,u}$ if

1) $P_k(K)$ is controllable with respect to $L(G_k)$ and $A_{k,u}$, and
2) $P_{i+k}(K)$ is controllable with respect to $L(G_i) \parallel P_i(K)$ and $A_{i+k,u} = (A_i \cup A_k) \cap A_{i,u}$, for $i = 1, 2, \ldots, n$.

For coordination control with partial observations, the notion of conditional observability is of the same importance as observability for monolithic supervisory control theory with partial observations. We recall that the supervisors $S_i$, $i = 1, 2, \ldots, n$, are supervisors based on partial observations, because they have only information about observable events from $A_{i,o}$ and observable coordinator events $A_{k,o}$, but do not observe events from $A_{i,o} \cup (A_{i,o} \cup A_{k,o})$.

Let $G_i$ be generators over the event sets $A_i$, $i = 1, 2, \ldots, n$, and let $G_k$ be a coordinator over $A_k$. A language $K \subseteq L(G_i[G_2]\ldots[G_n][G_k])$ is conditionally observable with respect to the generators $G_i$ and $G_k$, controllable sets $A_{i,c}$ and $A_{k,c}$, and projections $Q_{i+k} : A_{i+k} \rightarrow A_{i+k,o}$, for $i = 1, 2, \ldots, n$ and $Q_k : A_k \rightarrow A_{k,o}$, if

1) $P_k(K)$ is observable with respect to $L(G_k)$, $A_{k,c}$, $Q_k$,
2) \( P_{1+k}(K) \) is observable with respect to \( L(G) \parallel P_k(K) \), \( A_{1+k} = A_r \cap (A_i \cup A_k) \), and \( Q_{1+k} \), for \( i = 1, 2, \ldots, n \).

We can now formulate the main existential result for coordination control with partial observation.

**Theorem 5:** Consider the setting of Problem 4. There exist supervisors \( S_1, S_2, \ldots, S_n \) and \( S_k \) based on partial observations such that

\[
\|\bigcap_{i=1}^n L(S_i/G_i) (S_k/G_k) \| = K
\]

if and only if \( K \) is

1) conditionally controllable with respect to the generators \( G_i \) and \( G_k \) and the uncontrollable sets \( A_{i,u} \) and \( A_{k,u} \), for \( i = 1, 2, \ldots, n \), and

2) conditionally observable with respect to \( G_i \) and \( G_k \), event sets \( A_{i,c} \) and \( A_{k,c} \), and projections \( Q_{i+k} \) and \( Q_k \), for \( i = 1, 2, \ldots, n \).

### III. Multi-level Coordination Control with Partial Observations

In this section coordination control of DES with partial observations recalled in the previous section is generalized to the multi-level setting. Similarly as in the complete observation case [3], [2], the subsystems are organized into groups on the lowest level of the hierarchy and a high-level coordinator will be assigned to each group. The high-level coordinator ensures communication among these groups.

**A. Existential results of three-level coordination control**

We first recall existential results for the three-level coordination control with complete observations from [2] and constructive results from [3].

It is assumed that \( G = G_1 \parallel G_2 \parallel \ldots \parallel G_n \) and that the subsystems are organized into \( m \leq n \) groups \( I_j \), for \( j = 1, 2, \ldots, m \). The multi-level structure of the subsystems and their coordinators is displayed on Fig. 1. The notation

\[
A_{k} = \bigcup_{i \in I_k} A_i
\]

is used in the paper. Here \( P_I \) denotes the projection \( P_I : A^* \rightarrow A_I \). The notation \( P_{j+k} : A^* \rightarrow (A_j \cup A_k)^* \) is then used for the projection to group events extended by the high-level coordinator events. Similarly, \( P_{j+k} : A^* \rightarrow (A_j \cup A_k \cup A_k)^* \) denotes the projection to the event set \( A_j \) of an automaton \( G_j \) belonging to the group \( I_j \), extended by the event set \( A_k \) of the group coordinator for the low-level group \( I_j \), and by the event set \( A_h \) of the high-level coordinator. We have introduced the corresponding notion of conditional decomposability in [2].

**Definition 6 (three-level conditional decomposability):** A language \( K \subseteq A^* \) is called three-level conditionally decomposable with respect to the alphabets \( A_1, A_2, \ldots, A_n \), the high-level coordinator alphabet \( A_k \), and the low-level coordinator alphabets \( A_{k_1}, A_{k_2}, \ldots, A_{k_m} \)

if \( K = \bigcap_{i=1}^n P_{i+k}(K) \) and \( P_{h+k}(K) = \bigcap_{j \in I_h} P_{j+k}(K) \) for \( r = 1, 2, \ldots, m \).

It should be noted that on the right-hand side of the second equation in Definition 6, the natural projection includes events from both the group coordinator \( A_{k_r} \) and the high-level coordinator \( A_h \). Note that on the left-hand side there can be events outside the group \( I_r \), because the high-level coordinator alphabet includes shared events between different low-level groups. Therefore, these events should also be included on the right-hand side.

**Problem 7 (three-level coordination control problem):** Consider the modular system \( G = G_1 \parallel G_2 \parallel \ldots \parallel G_n \) along with the three-level hierarchical structure of the subsystems organized into groups \( I_j \), \( j = 1, 2, \ldots, m \), on the low level. The synchronous products \( \bigcap_{i \in I_j} G_i \), \( j = 1, 2, \ldots, m \), then represent the \( m \) high-level systems. The coordinators \( G_{k_j} \) are associated to groups of subsystems \( \{ G_i \mid i \in I_j \} \), \( j = 1, 2, \ldots, m \). The three-level coordination control problem consists in synthesizing the supervisor \( S_j \) for each low-level system \( G_i \), \( i = 1, 2, \ldots, n \), and the high-level supervisor \( S_k \) supervising the group coordinator \( G_{k_j} \), \( j = 1, 2, \ldots, m \), such that the specification \( K = \bar{K} \subseteq L(G) \) is met by the closed-loop system, i.e.,

\[
\bigcap_{j=1}^m \bigcap_{i \in I_j} L(S_i/G_i) (S_k/G_{k_j}) = K.
\]

In this paper the distinguishing feature is that all supervi- sors have only partial observations of their respective event sets. Moreover, we will present a generalization of sufficient conditions in the constructive part of the paper and it is at the same time a generalization of sufficient conditions in the complete-observations case presented in [3].

**Remark 8:** In [2] we have proposed a simplification. We increase the low-level coordinator alphabets \( A_{k_j} \) that contain shared events among subsystems of the group \( I_j \), \( j = 1, 2, \ldots, m \), by making its union with the high-level coordinator alphabet \( A_h \), i.e., we put \( A_{k_j} := A_{k_j} \cup A_h \), \( j = 1, 2, \ldots, m \). We recall that \( A_h \) contains only events shared between different groups of subsystems, that is, \( A_h \supseteq \bigcup_{k \in 1, 2, \ldots, m} (A_{k_j} \cap A_{k_j}) \), which is typically a much smaller set than the set of shared events (between two or more subsystems). Otherwise stated, we include into the alphabets \( A_{k_j} \) of the group coordinators also events from the global coordinator set (if this is nonempty). We recall that we first construct \( A_{k_j} \) by extending the set of events that are shared among the low-level groups, i.e.,

\[
A_{sh} = \bigcup_{k \in 1, 2, \ldots, m} (A_{k_j} \cap A_{h_j})
\]

This set is typically much smaller than the set of all shared events, because many events are shared only among subsystems belonging to a given low-level group and these do not count for \( A_{sh} \). We find \( A_h \) as an extension of \( A_{sh} \) using a method described in [8] such that the first equation of Definition 6 holds true.

We recall that in the prefix-closed case the coordinators (both the high-level and the group coordinators) are actually determined by the corresponding alphabets from Definition 6 as projections of the plant to these alphabets. The simplification described above enables us to use only the group coordinators \( G_{k_j} \) in all definitions below, which is more concise than using \( G_h \| G_{k_j} \), but we have to bear in mind that \( G_{k_j} \) may also contain the high-level coordinator events belonging to other groups than \( I_j \).
Formally, the group coordinators \( G_k, k = 1, 2, \ldots, m \), are computed using Algorithm 1 below. We emphasize that \( G_k \) are computed in a distributed way and \( A_k \) might be extended further so that \( P_k \) have the observer property [6], which makes \( P_k \) smaller than \( G_k \). We recall that with the definition that \( A_k \subseteq A_k \) described in Remark 8, we can simplify \( L(G_k) \) to \( L(G_k) \) of \([2]\) to \( L(G_k) \). Indeed, by our choice of the coordinators, \( L(G_k) = P_k(L) = P_k(L) = P_k(L) = L(G_k) \), where \( L = \bigcap_{i=1}^{n}(G_i) \) is the global plant language and the second equality holds by Lemma 1. Therefore, instead of the low-level coordinators \( G_k \), \( j = 1, 2, \ldots, m \), for subsystems belonging to the individual groups \( \{G_i | i \in I_j\} \) and the high-level coordinators \( G_k \) that coordinate the different groups, we are using only the low-level (group) coordinators \( G_k \), but over larger alphabets compared to \([2]\).

Since the only known condition ensuring that the projected generator is smaller than the original one is the observer property \([9]\) we might need to further extend the alphabets \( A_k \) so that the projection \( P_k \) is an \( L(G_i) \)-observer, for all \( i \in I_j \).

The key concept is the following.

**Definition 9 ([2]):** Consider the setting and notations of Problem 7, and let \( G_k \) be a coordinator. A language \( K \subseteq L(\bigcap_{i=1}^{n}(G_i)) \) is three-level conditionally controllable with respect to the generators \( G_1, G_2, \ldots, G_n \), the local alphabets \( A_1, A_2, \ldots, A_n \), the low-level coordinator alphabets \( A_{k_1}, A_{k_2}, \ldots, A_{k_m} \), and the uncontrollable alphabet \( A_u \) if for all \( j = 1, 2, \ldots, m \)

1) \( P_k(K) \) is controllable with respect to \( L(G_{k_j}) \) and \( A_{k_j,u} \).
2) \( P_j+1(K) \) is controllable with respect to \( L(G_{j+1}) \) and \( A_{j+1,u} \), for all \( i \in I_j \).

The original version of three-level conditional controllability from \([2]\) is simplified by replacing the composition \( L(G_k) \parallel L(G_k) \) by \( L(G_k) \) as discussed in Remark 8. For the sake of brevity \( K \) will be called three-level conditionally controllable with respect to \( G_i', i \in I_j \), and \( G_k \), where some sets are not referenced.

For modular DES with partial observations and three-level hierarchy the following concept is needed.

**Definition 10:** A language \( K \subseteq L(\bigcap_{i=1}^{n}(G_i)) \) is three-level conditionally observable with respect to the generators \( G_1, G_2, \ldots, G_n \), the local alphabets \( A_1, A_2, \ldots, A_n \), the low-level coordinator alphabets \( A_{k_1}, A_{k_2}, \ldots, A_{k_m} \), and the corresponding natural projections if for \( j = 1, 2, \ldots, m \)

1) \( P_k(K) \) is observable with respect to \( L(G_{k_j}) \) and \( Q_{k_j} \).
2) \( P_j+1(K) \) is observable with respect to \( L(G_{j+1}) \parallel P_{j+1}(K) \) and \( Q_{j+1} \), for all \( i \in I_j \).

The main existential result of multi-level coordination control stated below is a generalization of the one from \([2]\).

**Theorem 11:** Let \( K \) be three-level conditionally decomposable with respect to the local alphabets \( A_1, A_2, \ldots, A_n \), the high-level coordinator alphabet \( A_k \), and the low-level coordinator alphabets \( A_{k_1}, A_{k_2}, \ldots, A_{k_m} \), and let the multi-level structure of the groups of subsystems, coordinators and corresponding supervisors under partial observations be as described in Problem 7. There exist supervisors \( S_i, i \in I_j \), for the low-level systems within any group of low-level systems \( \{G_i | i \in I_j\} \), \( j = 1, 2, \ldots, m \), and supervisors \( S_{k_j}, j = 1, 2, \ldots, m \), for the low-level coordinators such that

\[
\bigwedge_{j=1}^{m}(L(S_j/G_i) \parallel (S_{k_j}/G_{k_j})) = K
\]

if and only if \( K \) is three-level conditionally controllable and three-level conditionally observable.
B. Generalized constructive results

If the specification fails to satisfy the necessary and sufficient conditions for being achievable then it is customary in supervisory control theory to construct a maximal achievable sublanguage. This applies also to the multi-level coordination control architecture for modular DES. We recall from [3] that in the case specification $K$ fails to be three-level conditionally controllable, the supremal three-level conditionally controllable sublanguage always exists and can be computed in a distributive way.

It is not surprising that three-level conditional observability is not closed under language unions as it is the case of (one-level) conditional observability as well as observability in the monolithic framework. Therefore, we propose three-level conditional normality that is stronger, but closed under language unions.

**Definition 12:** A language $K \subseteq L(\prod_{i=1}^n G_i)$ is three-level conditionally normal with respect to the generators $G_1, G_2, \ldots, G_n$, the local alphabets $A_1, A_2, \ldots, A_n$, the low-level coordinator alphabets $A_{1,k}, A_{2,k}, \ldots, A_{n,k}$, and the corresponding natural projections if for all $j = 1, 2, \ldots, m$

1. $P_{k,j}(K)$ is normal with respect to $L(G_{k,j})$ and $Q_{k,j}$,
2. $P_{i+k,j}(K)$ is normal with respect to $L(G_i) \parallel P_{k,j}(K)$ and $Q_{i+k,j}$, for all $i \in I_j$.

**Theorem 13:** Three-level conditional normality is preserved under arbitrary unions, hence the supremal three-level conditionally normal language always exists.

The computation of conditionally controllable and conditionally normal sublanguages is an important issue if the specification fails to satisfy these conditions. By Theorem 13, the supremal three-level conditionally normal sublanguage of a specification $K$ always exists. Below we propose a procedure to compute the supremal three-level conditionally controllable and conditionally normal sublanguage of $K$ with respect to the three level hierarchical structure of the system, denoted by $\text{sup}_2c\text{CN}(K,L,A,Q)$ that always exists as well, since the three-level conditional-controllability is already known to preserve language unions [3].

Similarly as in the centralized coordination control we introduce the following notation. For all $j = 1, 2, \ldots, m$ and $i \in I_j$,

$$\text{supCN}_{k,j} = \text{supCN}(P_{k,j}(K), L(G_{k,j}), \text{A}_{k,j}, Q_{k,j})$$

$$\text{supCN}_{i+k,j} = \text{supCN}(P_{i+k,j}(K), L(G_i), \text{supCN}_{k,j}, \text{A}_{i+k,j}, Q_{i+k,j})$$

where $\text{supCN}(K,L,A,Q)$ denotes the supremal sublanguage of $K$ controllable with respect to $L$ and $A$ and normal with respect to $L$ and the natural projection $Q$, see [5].

As in the centralized coordination, the following inclusion always holds true.

**Lemma 14:** For all $j = 1, 2, \ldots, m$ and for all $i \in I_j$, we have that $P_{i+k,j}(\text{supCN}_{i+k,j}) \subseteq \text{supCN}_{k,j}$.

Transitivity of controllability and normality is needed later.

**Lemma 15** (i)): Let $K \subseteq L \subseteq M$ be languages over $A$ such that $K$ is controllable with respect to $L$ and $A$ and normal with respect to $L$ and $Q$, and $L$ is controllable with respect to $M$ and $A$, and normal with respect to $M$ and $Q$. Then $K$ is controllable with respect to $M$ and $A$, and normal with respect to $M$ and $Q$.

The notation $\text{sup}_2c\text{CN}_j = \|_{i \in I_j} \text{supCN}_{i+k,j}$ is chosen for the resulting language of the (centralized) coordination control applied in the low-level group $I_j$. Then we have the following result.

**Theorem 16:** Consider Problem 7 and the languages defined in (2). For $j = 1, 2, \ldots, m$ and $i \in I_j$, let the languages $P_{k,j}(\text{supCN}_{i+k,j})$ be controllable with respect to $L(G_{k,j})$ and $A_{k,j}$, and normal with respect to $L(G_{k,j})$ and $Q_{k,j}$, and let $P_{i+k,j}(\text{supCN}_j)$ be controllable with respect to $L(G_i)$ and $A_{i+k,j}$, and normal with respect to $L(G_i)$ and $Q_i$. Then

$$\text{sup}_2c\text{CN}(K,L,A,Q) = \|_{j=1}^{m} \|_{i \in I_j} \text{supCN}_{i+k,j}.$$
for imposing controllability and normality with respect to $L(G_k)$. It can be shown that $\sup_{j \in I_j} \supCN_{i+k_j}^P$ further restricted by these supervisors will always satisfy all controllability and normality conditions required in Theorem 16. Moreover, it is easy to check that $\supCN_k^P$ can be computed in the modular way as follows:

$$\supCN_k^P = \supCN(P_k(\|i \in I_i \supCN_{i+k_i}^P), L(G_k), A_{k_i}, Q_k) = \cap_{i \in I_i} \supCN(P_i(\supCN_{i+k_i}^P), L(G_k), A_{k_i}, Q_k).$$

This is because it is a special case of modular control with multiple prefix-closed specifications for a single plant $G_k$.

We note that the equality in Lemma 14 implies the sufficient conditions of Theorem 16. Indeed, if $P_j(\supCN_{i+k_i}) \subseteq \supCN_{j+k_j}$, then in particular $P_k(\supCN_{i+k_i})$ is controllable and normal with respect to $L(G_k)$. Hence, $\cap_{i \in I_i} P_i(\supCN_{i+k_i})$ is controllable and normal with respect to $L(G_k)$, for all $j = 1, 2, \ldots, m$.

Similarly, there is a sufficient condition for controllability in terms of observer and OCC conditions (that can also be weakened to LCC of [4]). These conditions are well known in hierarchical supervisory control and an interested reader can find the definitions of these conditions in [3].

**Proposition 18**: [3] Consider the setting in the case of complete observations with $\supCN_{i+k_i}$ replaced by $\supCN_{i+k_i}^P$. If for all $j = 1, 2, \ldots, m$, $P_j^P$ is an $L(G_k)$-observer and OCC for $L(G_k)$, then $\supCN(P_j(\supCN_{i+k_i}^P), L(G_k), A_{k_i}, Q_k)$, then $\supCN(P_j(\supCN_{i+k_i}^P), L(G_k), A_{k_i}, Q_k)$.

Finally, we emphasize that even without the above sufficient conditions $\sup_{j \in I_j} \supCN_{i+k_j}^P$ is controllable and normal with respect to $L(G)$, but we cannot prove that it is three-level conditionally controllable and conditionally normal. These conditions are, however, necessary for the language $\sup_{j \in I_j} \supCN_{i+k_j}^P$ to be achievable in our multi-level coordination control architecture under partial observations as we have shown in Theorem 11. Similarly, we cannot guarantee the maximal permissiveness with respect to the three-level coordination control architecture. Therefore, in the case these conditions are not satisfied, it is reasonable to synthesize supervisors that make them hold, which is a natural approach knowing that these conditions are formulated in terms of controllability and observability for both level interfaces.

**IV. CONCLUDING REMARKS**

We have extended multi-level coordination control to partially observed discrete-event systems. A constructive algorithm for the computation of the supremal three-level conditionally controllable and conditionally normal sublanguages have been presented. Moreover, we have generalized the sufficient condition for the computation of the supremal three-level conditionally controllable language that has now the same form for both high-level and low-level coordinators. It should be noted that recently a weaker condition than normality, called relative observability, has been proposed for monolithic partially observed DES, cf. [11]. It is then possible to introduce a distributed version of relative observability, conditional relative observability [12].

This work opens the way to combine top-down and bottom-up approach. It turns out that bottom-up approach is better suited for handling nonblockingness, because non-blocking is best to be guaranteed first within low-level groups and then in the higher level between different groups. Moreover, it follows from the main constructive theorem that the sufficient conditions on normality between the different levels of the hierarchy can be naturally met by computing the appropriate supervisors that will guarantee the normality conditions. It appears that the best way to do it is in the bottom-up way: first a supervisor on low-level coordinator alphabets are computed and then the supervisors on the high-level coordinator alphabet is computed. Unlike previous approaches this means that we do not need to extend the respective coordinator alphabets to meet the sufficient conditions, but we can design supervisors to meet the conditions on the current coordinator alphabets.

**V. ACKNOWLEDGMENTS**

The authors would like to thank S. Lafortune and F. Lin for a fruitful discussion. Supported by RVO 67985840, by the MSMT grant LH13012 (MUSIC), and partially by the DFG project DIAMOND (Emmy Noether grant KR 4381/1-1).

**REFERENCES**


