A constructive and modular approach to decentralized supervisory Control problems

Jan Komenda, Hervé Marchand, Sophie Pinchinat

HAL Id: inria-00424857

https://hal.inria.fr/inria-00424857

Submitted on 19 Oct 2009

A CONSTRUCTIVE AND MODULAR APPROACH TO DECENTRALIZED SUPERVISORY CONTROL PROBLEMS

Jan Komenda * **Hervé Marchand** ** **Sophie Pinchinat** **

* *Institute of Mathematics, Czech Academy of Sciences, Brno Branch*
Zizkova 22, 616 62 Brno, Czech Republic
E-mail: komenda@ipm.cz
** *Irisa, Campus universitaire de Beaulieu,*
35042 Rennes Cedex, France.
{herve.marchand,sophie.pinchinat}@irisa.fr

Abstract: We plunge decentralized control problems into modular ones to benefit from the know-how of modular control theory: any decentralized control problem is associated to a natural modular control problem, which over-approximates it. Then, we discuss how a solution of the latter problem delivers a solution of the former.

Keywords: Decentralized supervisory control, Modular discrete-event systems, separability, controllability, coobservability.

1. INTRODUCTION

Over the past years there has been a considerable research effort in decentralized supervisory control (see e.g. Cieslak et al. (1988); Rudie and Wonham (1992); Yoo and Lafortune (2000) and associated references). Decentralized control of discrete-event systems appears as a natural approach to decrease the computational complexity of synthesizing controllers for large scale systems: the overall task of the synthesis is divided into synthetizing individual controllers, each of them reacting according to a partial observation of the system's moves. Most of the works provide only existential results, by exhibiting necessary and sufficient conditions under which a given specification language can be exactly achieved. There are very few results on synthesis of decentralized controllers for specifications that do not fulfill these conditions.

Modular approaches (see Y. and Heymann (1991); Rohloff and Lafortune (2003); Jiang and Kumar (2000); Lee and Wong (2002); deQueiroz and Cury (2000); Akesson et al. (2002); Schmidt et al. (2006)) consider systems that have a structure, and design methods to decompose the specification according to this structure. Under some hypothesis, an optimal solution can be computed efficiently.

In this paper, we plunge decentralized control problems into modular ones to benefit from the know-how of modular control theory. To do so, we first consider the modular over-approximation of the problem where the system and the specification are replaced by their infimal separable super-language. Hypothesis on the status of events lead us to apply the results of Lee and Wong (2002) to derive modular controllers. These controllers provide a solution of the decentralized control problem whenever they realize a language that is contained in the specification of the decentralized problem, without assuming a lot on the specification (neither coobservability, nor decomposability). This solution is by construction controllable and coobservable, but might not be optimal, and neither too trivial. Optimality is however obtained as soon as the system is given by a separable language.

Modular over-approximations have also been considered by Jiang and Kumar (2000). However, for these results to hold, coobservability (actually decomposability) of the specification is a prerequisite, whereas here coobservability is obtained by construction.

The paper is organised as follows: In Section 2 we present preliminaries on models; in Section 3 we present the Basic Decentralised Supervisory Control Problem (BDSCP). Section 4 is dedicated to the study of the fundamental notions of Controllability, Coobservability, Decomposability, and Separability, and their interplay. Section 5 explains how a BDSCP is associated its modular over-approximation, and how modular methods apply to the BDSCP. A short conclusion is given in Section 6.

2. PRELIMINARIES ON MODELS

We follow the automata and language framework (see Ramadge and Wonham (1989); Cassandras and Lafortune (1999) for details). The alphabet of events is written A and the set of finite sequences, that is *words* over A, is denoted A^*; typical words of A^* are $s, s', ...$ of A^*. The empty sequence is written ϵ. A language R over an alphabet is a set of finite sequences of letters in this alphabet. Languages will be written with typical elements $L, K, R, M...$ when the alphabet is A.

It is standard to write ss' for the word obtained by concatenating the two words s and s'. This is extended to the *concatenation* of two languages R and R', written RR' as follows:

$$RR' = \{ss' \mid s \in R \text{ and } s' \in R'\}$$

Finally, a language R is *prefix closed* whenever for each word of the form $ss' \in R$ we also have $s \in R$ (in particulr $\epsilon \in R$). All languages are assumed to be prefix closed in the reminder of the paper.

In this paper, we will be interested in decentralized control for which partial observation plays a central role. Given a finite set I of locations, for each $i \in I$, we assume given a set $A_{o,i} \subseteq A$, of the events that are *observable* from location i. Words in $A_{o,i}^*$ will be noted $s_i, s'_i, ...$ and languages over $A_{o,i}^*$ are noted R_i, L_i, K_i.

Now, given a set I as defined above and a set of alphabets $A_{o,i} \subseteq A$, we denote by $A_o = \cup_{i \in I} A_{o,i}$ the set of globally observable events and $A_{uo} = A \setminus A_o$ the set of globally unobservable events.

Based on these notations, the classical natural projection P_i from A^* to $A_{o,i}^*$, for each $i \in I$ is $P_i(\epsilon) = \epsilon$ and $P_i(s\sigma) = P_i(s)\sigma$ if $\sigma \in A_{o,i}$, and $P_i(s)$ otherwise. The projection P_i simply erases the unobservable events from location i. For $R \subseteq A^*$, $P_i(R) = \{P_i(s) \mid s \in R\}$. Given $R_i \in A_{o,i}^*$, the inverse projection of R_i is defined by $P_i^{-1}(R_i) = \{s \in A^* \mid P_i(s) \in R_i\}$.

Now, because in general $A_o, A_{uo} \neq A$, we write P for the natural projection from A^* to A_o^* and P_{uo} for the natural projection from A^* to A_{uo}^*.

An important operation in language theory is the parallel composition of languages.

Definition 1. Let $(R_i)_{i \in I}$ be languages respectively over $A_{o,i}$. The parallel composition *of the $(R_i)_{i \in I}$ is the language over A defined by*

$$\|_{i \in I} R_i = \bigcap_{i \in I} P_i^{-1}(R_i)$$

By definition $\|$ is associative and commutative. Note that in general, we only have $R \subseteq \|_{i \in I} P_i(R)$. Also, $\|_{i \in I}$ is monotone in the sense that, whenever $R_i \subseteq R'_i$ for each $i \in I$ then $\|_{i \in I} R_i \subseteq \|_{i \in I} R'_i$.

3. DECENTRALIZED BASIC CONTROL PROBLEMS

We consider the most commonly used architecture for decentralized control, namely the *conjunctive and permissive* architecture considered by Rudie and Wonham (1992),Yoo and Lafortune (2002), and others. It is very standard, and we simply recall the principles. We assume that the plant is observed through different locations, denoted by the set I.

- For each $i \in I$, $A_{c,i} \subseteq A_i$ represents the set of events that are controllable in location i.
- $A_c = \cup_{i \in I} A_{c,i}$. An event is controllable whenever it is controllable in at least one location.
- Finally the set of globally uncontrollable events is given by $A_{uc} = A \setminus A_c$.

Since Ramadge and Wonham (1989) followed by Rudie and Wonham (1992) for the decentralized setting, it is standard to represent a (local) supervisor in location i as a mapping $S_i : P_i(L) \to 2^A$, where we recall that $P_i(L)$ is a language on $A_{o,i}$; $S_i(s_i)$ represents the set of events that the supervisor in location i enables after observing the string $s_i \in A_{o,i}^*$. De facto, we must have $A_{uc} \subseteq S_i(s_i)$ for all $s_i \in A_{o,i}^*$.

Given supervisors $(S_i)_{i \in I}$ and a language L, the language controlled by the S_i's is a sublanguage of L, written $(\bigwedge_{i \in I} S_i)/L$ defined as the least language containing ϵ (the empty word), and satisfying : if $s \in (\bigwedge_{i \in I} S_i)/L$ and if $a \in \bigcap_{i \in I} S_i(P_i(s))$ then $sa \in (\bigwedge_{i \in I} S_i)/L$.

Now, assume given a prefix closed language over A, say $K \subseteq L$, the *Basic Decentralized Supervisory Control Problem (BDSCP)* consists in finding local supervisors $(S_i)_{i \in I}$ so that $(\bigwedge_{i \in I} S_i)/L = K$. We call K the *specification* of the BDSCP.

Of course, there is no solution in general, but since Rudie and Wonham (1992), we know necessary and sufficient conditions which guarantee the existence of a solution (see next section). Moreover provided these conditions are fulfilled, an optimal (supremal) solution can be effectively computed by defining each local supervisor according to : first of all, $A_{uc} \subseteq S_i(s_i)$, and

(P) $a \in S_i(P_i(s)) \cap A_c$ whenever there exists $s' \in K$ s.t. $P_i(s') = P_i(s)$ and $s'a \in K$.

Thus, $a \in A$ is enabled by S_i after observing s_i either because a is uncontrollable from location i, or because from location i some sequence which observation is s_i does not leave K, when extended by a. Optimality of this solution is discussed in the next section.

4. CONTROLLABILITY, COOBSERVABILITY, DECOMPOSABILITY, AND SEPARABILITY

This section concerns properties of languages. We assume a given input to the BDSCP, in particular with two prefix-closed languages L and K. Properties of *controllability* and *coobservability* concerns K with

respect to L (and the alphabets). They form the necessary and sufficient conditions to get a solution to the BDSCP.

Controllability means that what is required (to stay inside the specification) is compatible with what is uncontrolled.

Definition 2. Let $A' \subseteq A$, and let R and M be two languages over A. We say that $R \subseteq M$ is controllable w.r.t. M and A' whenever $RA' \cap M \subseteq R$.

The set of all languages that are controllable w.r.t. M and A' is denoted $\mathcal{C}(M, A')$. The set $\mathcal{C}(M, A')$ being closed under arbitrary union, for every *specification* language R there exists a unique *supremal controllable sublanguage* of R w.r.t. M and A': $SupC(R, M, A') := \bigcup\{H \in \mathcal{C}(M, A')| \ H \subseteq R\}$.

Lemma 1. Let $A' \subseteq A$, and let M_1 and M_2 be two languages over A s.t. $M_1 \subseteq M_2$. Then $\mathcal{C}(M_2, A') \subseteq \mathcal{C}(M_1, A')$.

Notice that the following is trivial, but useful later on.

Lemma 2. Let $R \subseteq M \subseteq A^*$. If $R \neq \emptyset$ is controllable w.r.t. M and A then $R = M$.

Coobservability is a key notion in decentralized control; it is due to Rudie and Wonham (1992), but also deeply studied by Yoo and Lafortune (2002), and others. It is formally defined as follows.

Definition 3. $R \subseteq L$ is coobservable with respect to L and $(A_{o,i})_{i \in I}$ if for each $s \in R$ and for any $a \in A_c$, if $sa \in L \setminus R$, there must exist some $i \in I$ s.t. for all s' with $P_i(s) = P_i(s')$, $s'a \notin R$.

Coobservability guarantees the correctness of the protocol **(P)** between the local supervisors regarding the BDSCP, as in the previous section. According to Rudie and Wonham (1992) :

Theorem 1. Given $K \subseteq L$, there exist supervisors S_i such that $(\bigwedge_{i \in I} S_i)/L = K$ if and only if K is controllable and coobservable.

Because coobservability is not preserved under unions, there is no hope to always have an optimal solution. For this reason, Rudie and Wonham (1992) considered *decomposability* of languages, or as proposed here the more general weaker notion of *half decomposability* (which need not assume $K \subseteq L$).

Definition 4. (Half decomposability and decomposability) R is half decomposable with respect to L and $(A_{o,i})_{i \in I}$ if

$$L \cap \|_{i \in I} P_i(R) \subseteq R \qquad \text{(HD)}$$

Now, R is decomposable with respect to L and $(A_{o,i})_{i \in I}$ if Equation (HD) is an equality.

A direct consequence of Definition 4 is :

Proposition 1. If R is half decomposable w.r.t. $(A_{o,i})_{i \in I}$, then $R \cap L$ is decomposable w.r.t. $(A_{o,i})_{i \in I}$.

The next result will be usefeul in the next sections

Proposition 2. Assuming that $P_{uo}(R) = P_{uo}(L)$, then R is half decomposable w.r.t. L and $(A_{o,i})_{i \in I}$ if and only if R is half decomposable w.r.t. L and $A_{uo}, (A_{o,i})_{i \in I}$.

Proof Since $L \subseteq P_{uo}^{-1}(P_{uo}(L))$

$$L \cap \cap_i P_i^{-1}(P_i(R))$$
$$= L \cap P_{uo}^{-1}(P_{uo}(L)) \cap \cap_i P_i^{-1}(P_i(R))$$
$$= L \cap P_{uo}^{-1}(P_{uo}(R)) \cap \cap_i P_i^{-1}(P_i(R))$$

$\diamond$

By Rudie and Wonham (1992), (half) decomposability and coobservability are incomparable in general, unless the additional assumption that controllability of events is consistent between locations.

Theorem 2. Assume $A_{o,i} \cap A_c \subseteq A_{c,i}$ ($i \in I$). If R is (half) decomposable w.r.t. L and $(A_{o,i})_{i \in I}$ then $R \cap L$ is coobservable w.r.t. $(A_{o,i})_{i \in I}$.

We now consider the property of separability which is a special case of decomposability for $L = A^*$. This property is useful in the next section.

Definition 5. R is separable w.r.t. $(A_{o,i})_{i \in I}$ whenever

$$R = \|_{i \in I} P_i(R).$$

As for decomposability and coobservability, separability is not preserved by union. Separability has another characterisation :

Proposition 3. R is separable with respect to $(A_{o,i})_{i \in I}$ if and only if there exists $R_i \subseteq A_{o,i}^*$ for each i, s.t. $R = \|_{i \in I} R_i$.

Proof ($\Rightarrow$) By Definition 5. ($\Leftarrow$) We want to prove that $R = \|_i P_i(R)$:
($\subseteq$) : Obviously $R \subseteq \cap_{i \in I} P_i^{-1}(P_i(R))$, because R is contained in each term.
($\supseteq$) We establish that $P_i(R) \subseteq R_i$, and conclude by the monotony of $\|_i$: Since $\|_{i \in I} R_i = \cap_i P_i^{-1}(R_i)$, from $R = \|_{i \in I} R_i$, we obtain $P_i(R) \subseteq R_i \cap [\cap_{l \neq i} P_i P_l^{-1}(R_l)]$, for each $i \in I$. $\diamond$

Let us now emphasize properties of separability:

Proposition 4. $\|_{i \in I} P_i(R)$ is infimal among the super-languages of R that are separable with respect to $(A_{o,i})_{i \in I}$.

Proof By construction, $\|_{i \in I} P_i(R)$ is separable w.r.t. $(A_{o,i})_{i \in I}$. Now, let M be a separable language w.r.t. $(A_{o,i})_{i \in I}$ with $R \subseteq M$; hence $P_i(R) \subseteq P_i(M)$, for each $i \in I$. Since $\|_i$ is monotone, $\|_i P_i(R) \subseteq \|_i P_i(M)$. As M is separable $\|_{i \in I} P_i(M) = M$, which concludes. $\diamond$

Proposition 5. If R is separable w.r.t. $(A_{o,i})_{i \in I}$, then $P_{uo}(R) = A_{uo}^*$.

Proof Clearly $P_{uo}(R) \subseteq A_{uo}^*$. Moreover, we have $A_{uo}^* \subseteq P_i^{-1}(\{\epsilon\}) \subseteq P_i^{-1}(P_i(R))$, for all i as $\epsilon \in P_i(R)$. Thus $A_{uo}^* \subseteq \cap_i P_i^{-1}(P_i(R)) = R$ and finally $A_{uo}^* = P_{uo}(A_{uo}^*) \subseteq P_{uo}(R)$. $\diamond$

In particular, the following property holds

Proposition 6. If R is separable w.r.t. $(A_{o,i})_{i \in I}$, then it is also separable w.r.t. $A_{uo}, (A_{o,i})_{i \in I}$.

Proof

$$R = \|_{i \in I} P_i(R) = \bigcap_{i \in I} P_i^{-1}(P_i(R))$$

$$= \bigcap_{i \in I} P_i^{-1}(P_i(R) \cap P_{uo}^{-1}(A_{uo}^*)) \ (as \ P_{uo}^{-1}(A_{uo}^*) = A^*)$$

Now $A_{uo}^* = P_{uo}(R)$ (Proposition 5). Therefore,

$$R = \bigcap_{i \in I} P_i^{-1}(P_i(R) \cap P_{uo}^{-1}(P_{uo}(R)))$$

$\diamond$

We now relate separability and decomposability.

Lemma 3. If R is separable with respect to $(A_{o,i})_{i \in I}$, then $R \cap L$ is decomposable with respect to L and $(A_{o,i})_{i \in I}$.

Proof Since R is separable, $R = \|_{i \in I} P_i(R)$, and $L \cap \|_{i \in I} P_i(R) = L \cap R \subseteq R$. Hence, R is half decomposable and according to Proposition 2, $R \cap L$ is decomposable with respect to L and $(A_{o,i})_{i \in I}$. $\diamond$

We complete this section with Theorem 3 relating separability and coobservability :

Theorem 3. Assume $A_c \subseteq A_o$ and $A_{o,i} \cap A_c \subseteq A_{c,i}$ (for $i \in I$). If R is separable with respect to $A_{o,i}$ ($i \in I$) then $R \cap L$ is coobservable with respect to L and $(A_{o,i})_{i \in I}$.

Proof Use Theorem 2 and Lemma 3.

Given L the system and K the expected specification, the following corollary is central to our approach :

*Corollary 1. Assume $A_c \subseteq A_o$ and $A_{o,i} \cap A_c \subseteq A_{c,i}$ (for $i \in I$). If K is separable with respect to $(A_{o,i})_{i \in I}$ and if $K \cap L$ is controllable with respect to L and A_{uc}, then the BDSCP has a solution which is effectively computable according to the protocol (**P**).*

Now provided $K \subseteq L$, by Lemma 3, the assumption that K is separable is equivalent to the assumption that K is decomposable. Consequently, if $K \subseteq L$, assumptions of Corollary 1 become necessary and sufficient conditions for the BDSCP. Notice that this is actually the very results of Jiang and Kumar (2000) (Corollary 2 and Lemma 2).

In the case where controllability of $K \cap L$ does not hold, one considers $SupC(K \cap L, L, A_{uc})$ instead of K, and then checks for its separability. Unfortunately, failure of separability is the limit of the approach. In the next section, we investigate conditions and algorithms in order to circumvent this limit.

5. OVER-APPROXIMATION OF BDSCP IN A MODULAR FRAMEWORK

We exploit the control theory of modular plants (see e.g. Jiang and Kumar (2000); Lee and Wong (2002); Akesson et al. (2002); Schmidt et al. (2006)) by considering a very natural modular over-approximation of the BDSCP. We gain effective computations of solutions that are sound for the BDSCP under hypothesis we exhibit.

Assume given an input to the BDSCP :

- A plant $L \subseteq A^*$.
- For each i in a given finite set I, a set of observable events $A_{o,i} \subseteq A$ and a local set of controllable events $A_{c,i} \subseteq A_{o,i}$.
- A prefix-closed language $K \subseteq A^*$.

where we additionally assume that

$$A_c = \cup_{i \in I} A_{c,i} \subseteq A_o \qquad \text{(C1)}$$

$$A_{o,i} \cap A_c \subseteq A_{c,i}, \forall i \in I \qquad \text{(C2)}$$

We transform the BDSCP into a modular control of structured plant problem. To do so, we first consider an extension of the original BDSCP by adding a virtual location as follows : we consider a fresh element $\emptyset \notin I$, to denote this virtual location. We set $A_{o,\emptyset} = A_{uo}$. To maintain assumption (C2), $A_{c,\emptyset} = \emptyset$, then $A_{o,\emptyset} \setminus A_{c,\emptyset} = A_{o,\emptyset}$. The associated projection $P_\emptyset : A^* \to A_{uo}^*$ is P_{uo} as in the previous section. Now I is extended to $J = I \cup \{\emptyset\}$.

To the avoid confusion in the following, we introduce new names for alphabets, relying on letter "B" instead of letter "A", since previously dedicated to formalize the BDSCP.

- $B = A$.
- $B_j = A_{o,j}$ ($j \in J$); hence $B_\emptyset = A_{uo}$.
- The sub-alphabets of controllable events are represented, for each $j \in J$, by $B_{c,j} = B_j \cap A_{c,j}$. We also define $B_{uc,j} = B_j \setminus B_{c,j}$ for all $j \in J$.

In the rest of the section, we take the following convention of notations : for each $j \in J$, we write $L_j = P_j(L)$ and $L' = \|_{j \in J} L_j$, on the one hand, $K_j = P_j(K)$, $K' = \|_{j \in J} K_j$, on the other hand. Each B_j is then the alphabet of L_j (and of K_j), whereas $B_{c,j}$ contains the controllable events of L_j, and $B_{uc,j}$ contains the uncontrollable events of L_j.

By construction, L' and K' are over-approximations of L and K. Moreover, by Proposition 4, L' (resp. K') is infimal among the super-languages of L (resp. K) that are separable with respect to $A_{o,j}$ ($j \in J$).

It is clear that $L = L'$ if and only if L is separable with respect to $A_{o,j}$ ($j \in J$).

We now turn to computing a solution for inputs L' and K' but by using the theory of control of modular systems. The problem called *Basic Modular Supervisory Control Problem (BMSCP)* consists now in the following : Given a separable language $L' = \|_{j \in J} L_j$

and a prefix-closed language K', can we compute compute supervisors S_j, one for each L_j, such that $\|_{j \in J} (S_j / L_j) = K'$.

In the following we plunge BDSCP in the BMSCP, and see how we can benefit from the know-how of e.g. Lee and Wong (2002) in the latter framework.

Lee and Wong (2002) established powerful results for the BMSCP. In this work, two hypothesis are predominant : first, the proposed setting is the total observation, that is $\bigcup_{i \in J} B_j = B$; this is given for free, by the definitions of the B_j's. Second, any event that is shared by two different locations has the same status of controllability in both locations, formalised as $B_j \cap B_{uc,l} = B_{uc,j} \cap B_l, \forall j, l \in J$ (or equivalently that $B_j \cap B_{c,l} = B_{c,j} \cap B_l$). Proposition 7 below shows that we fit these two fundamental hypothesis.

Proposition 7. If Assumption (C2) holds, then $B_j \cap B_{c,l} = B_{c,j} \cap B_l$ ($j, l \in J$).

Proof We actually prove that If $A_{o,i} \cap A_c \subseteq A_{c,i}$ ($i \in I$), then $B_j \cap B_{c,l} \subseteq B_{c,j}$ ($j, l \in J$), which is sufficient to get $B_j \cap B_{c,l} = B_{c,j} \cap B_l$ ($j, l \in J$).

By definition, $A_{o,\emptyset} \cap A_c = \emptyset$, so that $A_{o,j} \cap A_c \subseteq A_{c,j}$ for all $j \in J$. Since $A_c = \bigcup_{j \in J} A_{c,j}$ (remember $A_c = \bigcup_{i \in I} A_{c,i}$ by definition and $A_{c,\emptyset} = \emptyset$), this entails $\bigcup_{l \in J} (A_{o,j} \cap A_{c,l}) \subseteq A_{c,j}$, that is $A_{o,j} \cap A_{c,l} \subseteq A_{c,j}$, for any l. By intersection with $A_{o,j}$, we get $A_{o,j} \cap A_{c,l} \subseteq A_{c,j} \cap A_{o,j}$. Replacing relevant terms, we obtain $B_j \cap A_{c,l} \subseteq B_{c,j}$. Because $B_{c,l} \subseteq A_{c,l}$, $B_j \cap B_{c,l} \subseteq B_j \cap A_{c,l}$. Hence, $B_j \cap B_{c,l} \subseteq B_{c,j}$, which concludes. ⬦

We now recall the key notion of *mutual controllability* introduced by Lee and Wong (2002) : mutual controllability is a binary relation between languages which is reflexive and symmetrical. Given two locations $j, l \in J$, and a language R_j over B_j, we write $\lfloor R_l \rfloor_j$ for the set $P_j(P_l)^{-1}(R_l)$. Intuitively, $\lfloor R_l \rfloor_j$ is R_l seen from location j.

Definition 6. R_j and R_l are *mutually controllable*, if $R_j(B_{uc,l} \cap B_j) \cap \lfloor R_l \rfloor_j \subseteq R_j$ and $R_l(B_{uc,j} \cap B_l) \cap \lfloor R_j \rfloor_l \subseteq R_l$.

Lee and Wong (2002) have proved that

Theorem 4. If $B_j \cap B_{c,l} = B_{c,j} \cap B_l$ ($j, l \in J$), and

$\quad$ L_j's are pairwise mutually controllable $\quad$ (C3)

then
$$SupC(K' \cap L', L', B_{uc})$$
$$= \qquad (1)$$
$$\|_{j \in J} SupC(K_j \cap L_j, L_j, B_{uc,j})$$

where $B_{uc} = \bigcup_{j \in J} B_{uc,j}$.

We now study the properties of the language

$$M = \|_{j \in J} SupC(K_j \cap L_j, L_j, A_{o,j} \setminus A_{c,j})$$

M has several properties that we list and prove now :

(*L'*-Cont) $M = SupC(K' \cap L', L', A_{uc})$: first $A_{o,j} \setminus A_{c,j}$ is $B_{uc,j}$, by definition. It is now enough to apply Theorem 4 to obtain $M = SupC(K' \cap L', L', A_{uc})$. Finally, by (C1), $B_{uc} = A_{uc}$.

(*L*-Cont) $M \cap L$ is controllable w.r.t. L and A_{uc} : by (*L'*-Cont), M is controllable w.r.t. L' and A_{uc} ; we conclude by Lemma 1 as $L \subseteq L'$.

Next we focus on the coobservability of $M \cap L$. To do so we wirst need some usful lemmas:

Lemma 4. If $M \neq \emptyset$, then $P_\emptyset(M) = L_\emptyset$.

Proof As $M \neq \emptyset$, it entails in particular that $SupC(K_\emptyset \cap L_\emptyset, L_\emptyset, A_{o,\emptyset}) \neq \emptyset$. By writing $K_j^{\uparrow c} = SupC(K_j \cap L_j, L_j, A_{o,j} \setminus A_{c,j})$, due to Lemma 2, $K_\emptyset^{\uparrow c} = L_\emptyset$.

Hence $M = P_\emptyset^{-1}(L_\emptyset) \cap \|_{i \in I} K_i^{\uparrow c}$, and

$$P_\emptyset(M) = P_\emptyset(P_\emptyset^{-1}(L_\emptyset)) \cap P_\emptyset(\|_{i \in I} K_i^{\uparrow c})$$
$$= L_\emptyset \cap P_\emptyset(\|_{i \in I} K_i^{\uparrow c})$$
$$= L_\emptyset \cap \bigcap_{i \in I} P_\emptyset(P_i^{-1}(K_i^{\uparrow c}))$$

Now, as $A_\emptyset^* \subseteq P_i^{-1}(K_i^{\uparrow c})$, $\forall i \in I$, $P_\emptyset(P_i^{-1}(K_i^{\uparrow c}))$ is $A_\emptyset^*$, which concludes. ⬦

Lemma 5. If $M \neq \emptyset$, then $M \cap L$ is decomposable w.r.t. L and $(A_{o,i})_{i \in I}$.

Proof M is separable w.r.t. $(A_{o,j})_{j \in J}$ by definition, it is then half decomposable w.r.t. $(A_{o,j})_{j \in J}$. As $P_\emptyset(M) = L_\emptyset$ (Lemma 4), by Proposition 2, M is also half decomposable w.r.t. L and $(A_{o,i})_{i \in I}$ and we conclude with Proposition 1. ⬦

We are now able to prove that $M \cap L$ is coobservable w.r.t. $(A_{o,j})_{j \in J}$:

(**Coobs**) $M \cap L$ is coobservable w.r.t. $(A_{o,i})_{i \in I}$: By lemma 5, if $M \neq \emptyset$, $M \cap L$ is decomposable w.r.t. L and $(A_{o,i})_{i \in J}$ and as the assumption (C2) holds, because of Theorem 2, it coobservable w.r.t. $(A_{o,i})_{i \in J}$.

From the above and from Theorem 1, we obtain :

Theorem 5. Assume (C1), (C2) and (C3) hold. If $\emptyset \subset M \cap L \subseteq K \cap L$ then the effectively computable supervisors $(S_i)_{i \in I}$ implementing the $P_i^{-1}(SupC(K_i \cap L_i, L_i, A_{o,i} \setminus A_{c,i})$'s constitute a solution of the BDSCP.

It is worthwile noting that the solution $M \cap L$ given by Theorem 5, which by construction is controllable and coobservable, only satisfies $M \cap L \subseteq SupC(K \cap L, L, A_{uc})$. Hence, it might be non-optimal ; there might exist a language that is either not comparable with $M \cap L$ or that contains $M \cap L$ while being controllable and coobservable.

However, if L is separable with respect to $(A_{o,j})_{j \in J}$, we have :

Theorem 6. Assume (C1), (C2) and (C3) hold. If $\emptyset \subset M \cap L \subseteq K \cap L$ and if L is separable with respect to $(A_{o,j})_{j \in J}$, then M is coobservable with respect to L and $(A_{o,i})_{i \in I}$, and is equal to $SupC(K \cap L, L, A_{uc})$.

Proof If L is separable w.r.t. $(A_{o,j})_{j \in J}$, then $L = L'$. Henceforth, $M = SupC(K' \cap L, L, A_{uc}) \subseteq L$ and thus $M \cap L = M$. Now, as the conditions of Theorem 5 are fulfilled, we have that M is controllable w.r.t. L and A_{uc} and coobservable w.r.t. $(A_{o,i})_{i \in I}$.

Now, since $K \subseteq K'$, by monotony of the $SupC()$ operator, $SupC(K \cap L, L, A_{uc}) \subseteq SupC(K' \cap L, L, A_{uc})(= M)$. The reverse inclusion $M \subseteq SupC(K \cap L, L, A_{uc})$ follows from the assumption that $M \subseteq K \cap L$ and Item (L-Cont). $\diamond$

One can note that, because of Proposition 6) the separability of L w.r.t. $(A_{o,i})_{i \in i}$ is enough in Theorem 6.

6. CONCLUSION

We have proposed a natural over-approximation of the Basic Decentralised Supervisory Control Problem (BDSCP), enabling us to exhibit hypothesis to compute a controllable and coobservable solution of the BDSCP (Theorem 5). Concerning those hypothesis, Assumptions (C1) and (C2) are obviously inherited from Rudie and Wonham (1992), but (C3), expressing mutual controllaility, comes up from the need to exploit the results of Lee and Wong (2002) in the theory of modular control. Notice that (C3) can be equivalently stated on the set I instead of $I \cup \{\emptyset\}$, as $A_{o,\emptyset} \cap A_{o,i}$ is empty for each $i \in I$.

The result of Theorem 5 is important: it says that the solution can be efficiently computed, and implemented. However, testing inclusion of the solution in $K \cap L$ can be of high cost. Futur work would then consist in finding subclasses of BDSCP that are easily solvable, as provided with criteria that ensure inclusion of our candidate solution in the specification.

References

K. Akesson, H. Flordal, and M. Fabian. Exploiting modularity for synthesis and verification of supervisors. In *Proc. of the IFAC*, barcelona, Spain, July 2002.

C. Cassandras and S. Lafortune. *Introduction to Discrete Event Systems*. Kluwer Academic Publishers, 1999.

R. Cieslak, C. Desclaux, A. Fawaz, and P. Varaiya. Supervisory control of discrete–event processes with partial observations. *IEEE Trans. Autom. Control*, 33(3):249–260, March 1988.

M.H. deQueiroz and J.E.R. Cury. Modular supervisory control of large scale discrete-event systems. In *Discrete Event Systems: Analysis and Control. Proc. WODES'00*, pages 103–110. Kluwer Academic, 2000.

S. Jiang and R. Kumar. Decentralized control of discrete event systems with specializations to local control and concurrent systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 30(5): 653–660, October 2000.

S.-H. Lee and K.C. Wong. Structural decentralized control of concurrent discrete-event systems. *European Journal of Control*, 8(5), 2002.

P. J. Ramadge and W. M. Wonham. The control of discrete event systems. *Proceedings of the IEEE; Special issue on Dynamics of Discrete Event Systems*, 77(1):81–98, 1989.

K. Rohloff and S. Lafortune. The control and verification of similar agents operating in a broadcast network environment. In *42nd IEEE Conference on Decision and Control*, Hawaii, USA, December 2003.

K. Rudie and W.M. Wonham. Think globally, act locally: decentralized supervisory control. *IEEE Transaction on Automatic Control*, 31(11):1692–1708, November 1992.

K. Schmidt, H. Marchand, and G. Gaudin. Modular and decentralized supervisory control of concurrent discrete event systems using reduced system models. In *Workshop on Discrete Event Systems, WODES'06*, July 2006.

Willner Y. and M. Heymann. Supervisory control of concurrent discrete-event systems. *International Journal of Control*, 54(5):1143–1169, 1991.

T. Yoo and S. Lafortune. A general architecture for decentralized supervisory control of discrete-event systems. In *Proc of 5th Workshop on Discrete Event Systems, WODES 2000*, Ghent, Belgium, August 2000.

T.S. Yoo and S. Lafortune. General architecture for decentralized supervisory control of discrete-event systems. *Discrete Event Dynamic Systems: Theory and Applications*, 12:335–377, 2002.