Automatica 48 (2012) 247–254

Contents lists available at SciVerse ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

# Supervisory control synthesis of discrete-event systems using a coordination scheme☆

Jan Komenda [a,1], 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

ARTICLE INFO

*Article history:*
Received 5 July 2010
Received in revised form
18 April 2011
Accepted 14 July 2011
Available online 17 December 2011

*Keywords:*
Discrete-event systems
Supervisory control
Distributed control
Closed-loop systems
Controllability

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.



## 1. Introduction

This paper investigates the 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 (Ramadge & Wonham, 1989) 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 Cai and Wonham (2010), Feng (2007), Schmidt, Moor, and Perk (2008), or the approach based on interfaces (Leduc, Pengcheng, & Raoguang, 2009), 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 Leduc et al. (2009) 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

☆ The research received funding from the 7th 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. The material in this paper was partially presented at the 10th International Workshop on Discrete Event Systems (WODES 2010), August 30 – September 1, 2010, Berlin, Germany. This paper was recommended for publication in revised form by the Associate Editor, Bart De Schutter under the direction of the Editor, Ian R. Petersen.

*E-mail addresses:* komenda@ipm.cz (J. Komenda), masopust@math.cas.cz (T. Masopust), J.H.van.Schuppen@cwi.nl (J.H. van Schuppen).

[1] Tel.: +420 532290444; fax: +420 541218657.

0005-1098/$ – see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.automatica.2011.07.008

control based on structural, specification independent conditions, e.g. mutual controllability (Komenda, van Schuppen, Gaudin, & Marchand, 2008), 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 Komenda and van Schuppen (2008). 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 Feng (2007, 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 Feng (2007) 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 Cassandras and Lafortune (2008) and Wonham (2009).

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 $E_i, E_j, E_k, E_\ell$, 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 Cassandras and Lafortune (2008). 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 $(G, E_c, \Gamma)$, 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 $(G, E_c, \Gamma)$ is a map $S : L(G) \to \Gamma$. A *closed-loop system* associated with the controlled generator $(G, E_c, \Gamma)$ 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 (Ramadge & Wonham, 1989). For uncontrollable languages, controllable sublanguages are considered. The notation $\sup C(K, L, E_u)$ 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 Cassandras and Lafortune (2008).

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 $\sup C(K, L, E_u) = \sup C(\|_{i=1}^n K_i, \|_{i=1}^n L_i, E_u)$. 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 (Gaudin & Marchand, 2004). Notice the difference with decentralized control of monolithic plants studied in Yoo and Lafortune (2002), 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 Komenda and van Schuppen (2008), 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