Automatica 52 (2015) 322–333

Contents lists available at ScienceDirect

Automatica

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

A branch and bound approach for the design of decentralized supervisors in Petri net models☆

Francesco Basile [a,1], Roberto Cordone [b], Luigi Piroddi [c]

[a] Dipartimento di Ingegneria dell'Informazione, Ingegneria elettrica e Matematica applicata, Università di Salerno, Italy

[b] Dipartimento di Informatica, Università degli Studi di Milano, Milano, Italy

[c] Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Milano, Italy

ARTICLE INFO

Article history:
Received 13 February 2014
Received in revised form
5 August 2014
Accepted 11 November 2014
Available online 7 January 2015

Keywords:
Petri nets
Supervisory control
Monitor places
Decentralized control
Discrete-event systems

ABSTRACT

The paper addresses the design of compact and maximally permissive decentralized supervisors for Petri nets, based on generalized mutual exclusion constraints. Decentralization constraints are formulated with respect to the net transitions, instructing each local supervisor to detect and disable transitions of its own control site only. A solution is characterized in terms of the states it allows and its feasibility is assessed by means of two separate tests, one checking the required behavioral properties (*e.g.*, liveness, reversibility and controllability) of the induced reachability subgraph and the other ensuring the existence of a decentralized supervisor enforcing exactly the considered set of allowed states. The second test employs an integer linear programming formulation. Maximal permissivity is ensured by efficiently exploring the solution space using a branch and bound method that operates on the reachable states. Particular emphasis is posed on the obtainment of the controllability property, both in the structural and the behavioral interpretation.

© 2014 Elsevier Ltd. All rights reserved.

1. Introduction

Supervisory control (SC) concerns the design of an agent (called the *supervisor*) that enforces forbidden state specifications on a discrete event system (DES). In the Petri net (PN) framework forbidden state specifications are often expressed in terms of linear state inequalities, called Generalized Mutual Exclusion Constraints (GMECs), which are amenable to a straightforward PN implementation, in the form of *monitor* places suitably connected to the transitions of the PN model of the plant and enforcing conservative conditions on the state evolution (through corresponding P-invariants), (Giua, DiCesare, & Silva, 1992; Moody & Antsaklis, 2000).

The supervisor design problem faces various objectives at the same time, namely the enforcement of specific properties (liveness, reversibility, controllability, etc.) in a maximally permissive way (*i.e.*, enabling as many reachable states as possible), and introducing the minimum number of monitors possible. Recent developments have shown that this problem can be optimally and efficiently solved in two steps, *i.e.* by calculating first the maximal subset of reachable states that guarantees the obtainment of the required properties, denoted $\mathcal{L}$ (the *legal* set), and then the monitor-based supervisor that restricts the reachability set of the plant net in closed loop exactly to $\mathcal{L}$.

Regarding the first step, Basile, Cordone, and Piroddi (2013) introduce a technique to calculate the legal set enforcing multiple specifications, both *static* and *behavioral*, the former being associated directly to individual states, while the latter depend on the structure of the reachability graph of the PN. Bounds on job and resource usage fall in the first category, whereas deadlock prevention (DP), liveness enforcement (LE), reversibility, controllability, etc. are behavioral specifications. The approach is particularly useful when multiple behavioral specifications, such as liveness and controllability, are formulated. Indeed, in such cases, it is inconvenient to enforce separately each behavioral property, since enforcing one may jeopardize the other.

As for the second step of the methodology, Nazeem, Reveliotis, Wang, and Lafortune (2010, 2011) provide a complete framework for the characterization of the existence of optimal supervisors and their synthesis, formulating an ILP problem where the decision variables are the GMEC parameters and the constraints are

☆ The research leading to these results has received funding from Regione Campania (Legge n.5/2002) under the preject ''Automazione orientata ai servizi per linee di produzione iperflessibile''. The material in this paper was not presented at any conference. This paper was recommended for publication in revised form by Associate Editor Jan Komenda under the direction of Editor Ian R. Petersen.

E-mail addresses: fbasile@unisa.it (F. Basile), roberto.cordone@unimi.it (R. Cordone), luigi.piroddi@polimi.it (L. Piroddi).

1 Tel.: +39 089 964400; fax: +39 06 233227957.

http://dx.doi.org/10.1016/j.automatica.2014.12.004
0005-1098/© 2014 Elsevier Ltd. All rights reserved.

expressed in terms of the legal and illegal markings. On similar lines, Chen, Li, Khalgui, and Mosbahi (2011) and Chen and Li (2011) concentrate the attention on the so-called First met Bad Markings (FBMs) and propose an iterative greedy ILP approach to find a GMEC that forbids one FBM at a time. A more efficient solution to the same problem, that systematically addresses the structural optimality of the supervisor, is suggested in Cordone and Piroddi (2013), where a simpler ILP formulation (addressing the prevention of a subset of illegal states with an individual GMEC) is used as the core element of a Branch and Bound (B&B) approach that solves the set covering problem of assigning optimally the illegal states to a minimum number of GMECs. Later developments extend these approaches to problems where a plain GMEC-based supervisor does not exist and more complex (nonlinear) supervisors are required, (Cordone, Nazeem, Piroddi, & Reveliotis, 2013, 2012; Nazeem & Reveliotis, 2012). The monitor redundancy issue has attracted much attention in the recent literature, with specific focus on the reduction of the number of control places as well as the supervisor structure. In Dideban and Alla (2008) the concept of over-state is introduced for safe PNs, and exploited to reduce the constraints for a given set of forbidden states, and this approach has been recently improved introducing the concept of quasi partial invariants and semi quasi partial invariants in Dideban, Zareiee, and Alla (2013). In Zareiee, Dideban, and Orouji (2014) ILP problems are used to obtain a small number of control places with small number of arcs. Another interesting approach for supervisor design enforcing behavioral properties, such as reversibility, is discussed in Reveliotis and Choi (2006). This work can also be extended to accommodate uncontrollable transitions.

The supervisor design problem becomes more involved in a decentralized setting. In that context, it is assumed that several local supervisors operate, each having authority only on a portion of the system (*i.e.*, on a subset of the transitions), in the absence of central coordination and with mutual communication inhibited. Such control architecture becomes of crucial importance for plants having a wide geographic extension or a large number of devices such as in modern communication systems. In these cases, communication with all plant sensors or actuators is infeasible because of economic reasons or bandwidth limitations. Even where centralized control is possible, it is of interest to study decentralized control solutions to address temporary failures that prevent communication with a certain area of the plant, in order to robustify the design.

While there is a large literature on decentralized control with formal languages and automata (Barret & Lafortune, 2000; Lin & Wonham, 1990; Rudie & Wonham, 1992), relatively fewer works address this problem in the PN framework. In Guan and Holloway (1997) global specifications are implemented by local supervisors with communication. In Chen and Hu (1991) a central coordinator is also present but specifications are given from the beginning in a distributed form. The approach of Basile, Giua, and Seatzu (2007, 2008) proposes an algorithm to optimize the permissiveness of the closed loop behavior under decentralized control by selecting with a heuristic rule the decentralized specifications that find a compromise between fairness among variables and the maximal cardinality of the set of legal markings under decentralized control. The controlled system is not guaranteed to be live or to satisfy any particular behavioral property. The mentioned works of Basile et al. (2007, 2008) employ a formalization of the decentralization specifications similar to the one adopted here, but for the fact that the control sites are expressed in terms of subsets of places rather than transitions. This design choice appears to be less intuitive and significant in practice since, while transitions are generally associated to events, places do not always have a clear physical meaning. In Iordache and Antsaklis (2006) global specifications without central coordination are considered and a sufficient condition is given for a set of GMECs to be enforced in a decentralized setting (d-admissibility). In addition, the transformation of inadmissible decentralized constraints into admissible ones is posed either in terms of the minimization of communication costs or in terms of the transformation of the constraints into a set of more restrictive – but d-admissible – ones. D-admissible constraints can be implemented by supervisors that detect and disable transitions of a single site.

The decentralized supervisor design problem is formulated here in the framework of the two-step supervisory control methodology described above. The main idea is to look for legal state sets (*i.e.*, compatible with all the requirements in the centralized setting) that are also exactly enforceable by decentralized supervisors. An optimization method is designed to find the maximal such set. Notice that, differently from Iordache and Antsaklis (2006), this paper focuses on the decentralized implementation of a set of legal markings by means of monitors, rather than the decentralization of a given set of constraints. The main difficulty in extending the two-step approach to the decentralized case lies in the fact that the two steps are interdependent. Indeed, not all sets of legal states that are compatible with a centralized supervisor implementation are also enforceable by a decentralized one. In fact, the decentralization requirement typically results in a reduction of the maximal legal set that can be actually allowed, compared to the centralized control case. Consequently, one cannot completely decouple the determination of the legal set $\mathcal{L}$ from the assessment of the existence of a decentralized supervisor that exactly enforces it.

This difficulty is here overcome by adopting a proposal-acceptance mechanism, where a candidate legal set $\mathcal{L}$ (by construction, included in or equal to the maximal set of legal states that can be allowed by a centralized supervisor), is first selected so as to guarantee the obtainment of all the desired static and behavioral requirements, and then tested for the existence of a decentralized supervisor that can exactly enforce it. In case of failure alternative smaller candidate legal sets are generated by a B&B algorithm by subsequent reductions of the global legal state set, guaranteeing a full exploration of its subsets. The B&B algorithm searches for the maximal such subset that provides all the required properties and is also enforceable in a decentralized way. Notice in passing that any existing decentralized controller can also be implemented in a centralized way, so that the existence of a centralized supervisor is in fact a pre-requisite for the existence of a decentralized one.

Two different procedures are proposed to deal with controllability from a *structural* and *behavioral* point of view, respectively. More in detail, structural controllability can be taken into account in the supervisor design phase alone by simply constraining the monitors introduced by the local supervisors not to have arcs directed towards uncontrollable transitions. On the other hand, behavioral controllability impacts on both the reachability pre-processing phase and the supervisor design. Indeed, behavioral controllability allows the existence of arcs directed from a local controller to an uncontrollable transition, as long as the latter is never disabled by an exclusive action of the former. In other words, whenever the control place of the local supervisor connected with an arc to the uncontrollable transition is insufficiently marked to enable the transition, there must always exist another place (not belonging to the local supervisor) that disables the transition. To enforce this property, a specific condition is added to the supervisor design phase, concerning every reachable marking where a partially controllable transition[2] must be disabled. This additional constraint ensures the presence of arcs disabling such a transition under the above mentioned marking only from local supervisors acting on sites where the transition is controllable. The set of such markings must be determined in the reachability pre-processing phase. Observability is also considered in the design process, but only from a structural point of view, for reasons explained in the paper.

[2] A *partially controllable* transition is a transition that can be used by multiple control sites, but is not controllable by all of them.

Download English Version:

https://daneshyari.com/en/article/695762

Download Persian Version:

https://daneshyari.com/article/695762

Daneshyari.com