Automatica 49 (2013) 3167–3179

Contents lists available at ScienceDirect

Automatica

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

A parameterized liveness and ratio-enforcing supervisor for a class of generalized Petri nets[☆]

Ding Liu[a], ZhiWu Li[a,1], MengChu Zhou[b,c]

[a] School of Electro-Mechanical Engineering, Xidian University, Xian 710071, China

[b] The Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai 201804, China

[c] Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102, USA

ARTICLE INFO

Article history:
Received 16 December 2011
Received in revised form
15 January 2013
Accepted 29 June 2013
Available online 20 August 2013

Keywords:
Flexible manufacturing system
Petri net
Liveness-enforcing supervisor
Ratio-enforcing supervisor
Monitor

ABSTRACT

The work proposes a synthesis method of supervisors for flexible manufacturing systems modeled by a class of generalized Petri nets. A concept of resource usage ratios (RU-ratios) is first presented to describe the occupation degree of a resource by an operation. Next, an intrinsically live structure characterized by a special numerical relationship between arc-weights and initial markings is investigated from a perspective of RU-ratios. Then, a new kind of supervisors is synthesized on the ground of the generic nature of the intrinsically live structure. Such a supervisor can achieve the purposes of both liveness-enforcement and resource usage ratio-enforcement of the system under consideration. Given a plant, it is easy to determine the topological structure of such a supervisor and the number of monitors is bounded by that of resources used in the plant. In addition, when the configuration of the plant model changes, the supervisor can be reusable through adjusting control parameters only without rearrangement of connections. This makes it easy enough and intuitive to be used by industrial practitioners. Instead of maximal behavioral permissiveness, it pursues a precise usage of shared resources that are limited and valuable. Several examples are used to illustrate the proposed methods.

1. Introduction

In this new millennium, the global economy becomes increasingly inseparable. As a paramount foundation of our society's economy system, manufacturing encounters more and more intense global market competition. A new strategy for manufacturing called customizability becomes a popular trend and is extensively adopted by an increasing number of companies, especially those manufacturing high value-added and technology-intensive products. In these fields, the mass production is replaced by a high-mix–low-volume production mode (Mahoney, 1997). It is the very reason that flexible manufacturing systems (FMSs) are built to realize a rapid and mixed production. Moreover, flexibility and agility processed by an FMS make it completely different from the traditional manufacturing systems (Zhou, 1995).

An FMS can thus react swiftly to the rapid changes of production requirements, whether planned or unplanned, to satisfy different market segments with reasonably priced and customized products. As an indispensable component of computer integrated manufacturing, an FMS usually consists of picking and placing robots, machining centers with multiple capacities and functions, logistic systems, and advanced control systems. In theory, an FMS belongs to resource allocation systems (RASs) (Reveliotis, 2007; Tricas, 2003). The resources can be dynamically configured and allocated before and after the system operation according to variable product specifications. The flexibility is gained through allowing multiple processes to run simultaneously and share the limited resources, and multiple types of raw workpieces can be concurrently processed in an FMS to meet a high-mix–low-volume order.

However, due to the competition for limited resources among different production processes, deadlocks arising in an FMS bring about a series of disturbing problems, from degraded system productivity and deteriorated performance to low utilization of some critical and expensive resources and even long system downtime (Zhou & Fanti, 2005).

In addition to automata (Ramadge & Wonham, 1987; Wonham & Ramadeg, 1988) and graph theory (Cho, Kumaran, & Wysk, 1995;

☆ This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 61074035 and 61034004, the National Basic Research Program of China (973 Program) under Grant No. 2011CB302804, and the Fundamental Research Funds for the Central Universities under Grant No. JY10000904001 and K5051304007. The material in this paper was not presented at any conference. This paper was recommended for publication in revised form by Associate Editor Bart De Schutter under the direction of Editor Ian R. Petersen.

E-mail addresses: dliu@xidian.edu.cn (D. Liu), zhwli@xidian.edu.cn (Z. Li), zhou@njit.edu (M. Zhou).

[1] Tel.: +86 29 88201986; fax: +86 29 88204338.

0005-1098/$ – see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.automatica.2013.07.023

Fanti, Maione, & Turchiano, 2000; Fanti & Zhou, 2004), Petri nets (Girault & Valk, 2003; Petri & Reisig, 2008; Wu & Zhou, 2010; Zhou & DiCesare, 1993; Zhou & Venkatesh, 1998) are an effective formalism and used extensively for modeling and analyzing deadlock problems in FMSs. The relationship between shared resources and deadlocks is fully investigated and reported by many researchers (Ezpeleta, García-Vallés, & Colom, 1998). The essential reason of deadlocks in an FMS is identified as improper allocation of limited resources, which leads to the occurrence of circular waits. Liveness in Petri nets, which implies deadlock-freedom but not vice versa, is a significant behavioral property in the research realm of FMSs. Under the tremendous influence of supervisory control theory (SCT) (Ramadge & Wonham, 1987), a variety of supervisors are designed to impose control on the plant Petri net model of an FMS to achieve a liveness-enforcing system. Both reachability graphs (Badouel & Darondeau, 1998; Chen & Li, 2011; Chen, Li, Khalgui, & Mosbahi, 2011) and structural analysis (Barkaoui, Chaoui, & Zouari, 1997; Barkaoui, Couvreur, & Klai, 2005; Barkaoui & Pradat-Peyre, 1996; Cho et al., 1995; Ezpeleta, Colom, & Martínez, 1995; Hu, Zhou, Li & Tang, 2013) are the most frequently employed methodologies to synthesize supervisors for plant models.

The theory of region (Ghaffari, Rezg, & Xie, 2003; Li, Zhou, & Jeng, 2008; Uzam, 2002; Uzam & Zhou, 2007) based on a reachability graph is a reliable, effective, and accurate method to analyze and verify a plant model. Every single state can be probed by inspecting a complete state space to yield a well-designed supervisor ensuring liveness and even maximal behavioral permissiveness. However, the cost to achieve the above goal is computationally prohibitive. The enumeration of all states is computationally expensive or infeasible in practice when dealing with a sizable plant model. The number of states of a Petri net model grows exponentially with respect to its size and initial marking. Therefore, structural theory (Lewis, Gürel, Bogdan, Doğanalp, & Pastravanu, 1998; Li & Zhou, 2009; Liu, Li, & Zhou, 2010, 2012, 2013; Liu, Li, Zhou & Xiong, 2013; Ohta & Tsuji, 2003; Xing, Zhou, Wang, Liu, & Tian, 2011; Zhong & Li, 2010b) is widely used to circumvent the state explosion problem. In order to thoroughly investigate and fully utilize the structure information of a Petri net model, a special structural object, called siphons, is frequently and extensively used to reveal the relationship between resource allocation and liveness. Siphon-based characterization of both deadlocks and liveness becomes a mainstream trend of methodologies that deal with the deadlock control and liveness-enforcement issues.

A system of simple sequential processes with resources (S^3PR) (Ezpeleta et al., 1995, 1998) is a class of ordinary Petri nets[2] widely used to model and analyze the deadlock and liveness problems in FMSs. Generalized extensions to S^3PR, such as a system of simple sequential processes with general resource requirements (S^3PGR^2) (Chao, 2007; Park & Reveliotis, 2001), a weighted system of simple sequential processes with several resources (WS^3PSR) (Tricas & Martínez, 1995), a system of simple sequential processes with weighted resources allocation (WS^3PR) (Liu et al., 2010; Zhong & Li, 2010a), a system of sequential systems with shared resources (S^4R) (Tricas, García-Vallés, Colom, & Ezpeleta, 2000), and G-systems (Barkaoui et al., 1997; Barkaoui & Petrucci, 1998; Barkaoui & Pradat-Peyre, 1996; Li & Zhao, 2008), are subsequently proposed in the literature. They all focus on the deadlock control and liveness-enforcement of generalized Petri nets. Note that generalized and ordinary Petri nets have the same modeling power, but the former can provide a more compact expression. Different from the latter, variable arc-weights are an important structural information and should be focused with more attention. The numerical relationship among weights has been examined to analyze robustness of non-ordinary Petri nets for flexible assembly processes (Hsieh, 2010, 2011), implement a proportion of different productions (Hu, Zhou, & Li, 2009, 2010, 2011), study ratio-driven FMS (Koh & DiCesare, 1991; Lee & Korbaa, 2004; Ohl, Camus, Castelain, & Gentina, 1995), and deal with liveness problems (Marchetti & Munier-Kordon, 2009). However, the structural information behind weights and its influence over behavioral properties need to be further investigated.

In our previous work (Liu et al., 2010, 2012, 2013), the structure of WS^3PR is explored. A new structural object, called weighted simple directed circuits (WSDCs), is defined to describe the structure of augmented simple circular waits among resources, instead of the traditional siphons. A siphon is defined as a set of places and does not carry the weight information of arcs. The previous work focuses on the competition relationship between an upstream activity place and a downstream one of a competition path contained by a WSDC. A restriction is proposed to identify this kind of intrinsically live structures and develop a WSDC-based method that enforces liveness by reconfiguring the initial marking of a plant model. Without external monitors, the liveness-enforcement of a WS^3PR is achieved through taking the full advantage of an intrinsically live structure in Liu et al. (2010, 2012). However, the existence of this kind of global special structures limits the application scope of the WSDC-based method. Therefore, we combine it with elementary siphons (Li & Zhao, 2008; Li & Zhou, 2009) to generate a hybrid liveness-enforcing policy (Liu et al., 2013) that can simplify the supervisor for a system, improve the permissiveness, and finally, reduce the control implementation cost. In addition, the min-marked and minimally controlled siphons are also analyzed with the assistance of WSDCs in Liu et al. (2013).

Since the seminal work by Ezpeleta et al. (1995), a variety of deadlock prevention and liveness-enforcing methods are implemented by adding monitors to an original plant model, most of which are designed aiming at siphon control. In this work, a new kind of monitors is proposed to control resource places directly. A so-called control path containing such a monitor is imposed on a plant model and actually replaces a competition path containing a resource place to be controlled. This results in a new WSDC containing the control path that is designed to satisfy a proposed restriction. The behavior of the original WSDC that contains the competition path is manipulated by the new one. In order to design such a monitor, a concept of resource usage ratios (RU-ratios) is first proposed to describe the occupation degree of a resource by an operation. The competition for a resource between two operations is expressed by a relationship between their RU-ratios. Next, the generic nature of an intrinsically live structure is analyzed from a viewpoint of the RU-ratio. Then, the ratio information and a numerical restriction are combined to synthesize a liveness and ratio-enforcing supervisor.

The new supervisor proposed in the paper achieves liveness and ratio-enforcement at the same time. In particular, the liveness-enforcement is implemented by imposing a proper resource usage ratio-enforcement on a plant model. It is an attempt to pursue a precise usage of resources instead of maximal behavioral permissiveness. The number of monitors in such a supervisor is bounded by that of resources used in the plant model. Hence, the control cost can be estimated without difficulty. In addition, the topology of the supervisor can be easily determined when a plant model is given and is simple enough to be understood and adopted by industrial practitioners. When the initial configuration and weights of the plant model are changed, the proposed supervisor can adapt quickly by adjusting control parameters only. The connection of monitors remains unchanged. It is just the work style of today's programmable logic controllers, i.e., the re-hard-wired job can be avoided. This greatly saves control cost.

[2] Compared with generalized Petri nets, all arc weights are equal to one in ordinary Petri nets.