International Journal of Innovative
Computing, Information and Control
Volume 7, Number 12, December 2011

ICIC International ©2011 ISSN 1349-4198
pp. 6677–6690

MONITOR DESIGN FOR SIPHON CONTROL IN S^4R NETS: FROM STRUCTURE ANALYSIS POINTS OF VIEW

Chunfu Zhong[1], Zhiwu Li[1,2,*] and Kamel Barkaoui[3]

[1]School of Electro-Mechanical Engineering
Xidian University
No. 2, South Taibai Road, Xi'an 710071, P. R. China
cfuzhong@gmail.com

[2]College of Engineering
King Saud University
P. O. Box 800, Riyadh 11421, Saudi Arabia
*Corresponding author: zhiwu.li@informatik.uni-halle.de

[3]Laboratoire CEDRIC
Conservatoire National des Arts et Métiers
292 rue Saint Martin, Paris Cedex 03, France
kamel.barkaoui@cnam.fr

Received July 2010; revised November 2010

Abstract. *Petri nets are widely used to design a liveness-enforcing Petri net supervisor for deadlock prevention in a flexible manufacturing system. A monitor-based supervisor performs better, i.e., with more permissive behavior, if its monitors are well designed. Deadlock control policies developed by max-controlled siphons are shown to be overly conservative from permissive behavior points of view. On the basis of the concept of max'-controlled siphons, this paper provides three methods, from structural points of view, to design monitors for siphons in S^4R nets, systems of simple sequential processes with shared resources. A deadlock prevention policy is developed accordingly, leading to liveness-enforcing Petri net supervisors with more permissive behavior. The proposed approaches and algorithm are then illustrated through an example.*

Keywords: Petri net, Flexible manufacturing system, Deadlock prevention, Siphon

1. **Introduction.** In a flexible manufacturing system (FMS), a deadlock means that two or more precesses keeps waiting indefinitely for the other to release resources [1]. Deadlocks can greatly decrease the productivity of an FMS. Therefore, they are extremely undesirable in a highly automated FMS and should be dealt with.

A Petri net (PN) is a graphical and mathematical modeling tool and has been well developed. There are many extensions to Petri nets, e.g., colored Petri net [2], timed Petri net [3] and extended Petri net with condition information [4]. As the development of Petri nets, they have been used in many areas such as software design [5], data analysis and manufacturing systems [6, 7].

Petri nets can efficiently characterize the behavior and properties of an FMS, such as concurrency, conflict, causal dependency, liveness and boundedness [8, 9]. Therefore, they are adopted to deal with deadlock problems in an FMS. Deadlock prevention, as one of the well-defined approaches to tackle deadlock problems, has drawn much attention in the past twenty years [8, 10, 11, 12, 13, 14, 15]. It can be achieved by designing a live system [16] or by using an offline computational mechanism to control the requests for resources and ensure that deadlocks never occur. The latter is usually implemented by designing a supervisor that consists of monitors and related arcs to be added to the original net model