Automatica 55 (2015) 116–124

Contents lists available at ScienceDirect

Automatica

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

Brief paper

On the enforcement of a class of nonlinear constraints on Petri nets[☆]

YuFeng Chen[a,c], ZhiWu Li[b,a,d], Kamel Barkaoui[c], Alessandro Giua[e,f,a]

[a] School of Electro-Mechanical Engineering, Xidian University, No. 2 South Taibai Road, Xi'an 710071, China
[b] Institute of Systems Engineering, Macau University of Science and Technology, Macau
[c] Cedric Lab and Computer Science Department, Conservatoire National des Arts et Métiers, Paris 75141, France
[d] Faculty of Engineering, King Abdulaziz University, Jeddah 21589, Saudi Arabia
[e] Aix Marseille Université, CNRS, ENSAM, Université de Toulon, LSIS UMR 7296, 13397, Marseille, France
[f] DIEE, University of Cagliari, 09123 Cagliari, Italy

ARTICLE INFO

Article history:
Received 29 July 2013
Received in revised form
17 October 2014
Accepted 29 January 2015
Available online 23 March 2015

Keywords:
Discrete event system
Petri net
Supervisory control
Supervisor
Nonlinear constraint

ABSTRACT

This paper deals with the enforcement of nonlinear constraints on Petri nets. A supervisory structure is proposed for a class of nonlinear constraints. In order to enforce a nonlinear constraint on a Petri net, we propose a transition transformation technique to replace a transition in an original net by a set of transitions. Then, a control place is designed to control the firing of these transitions, aiming to enforce the nonlinear constraint. The proposed supervisor is maximally permissive in the sense that it can make all markings in the admissible-zone reachable and all markings in the forbidden-zone unreachable. The proposed method is applicable to bounded Petri nets. Finally, a number of examples are provided to demonstrate the proposed approach.

1. Introduction

Petri nets (Murata, 1989) are a powerful tool to model and analyze discrete event systems (DESs). They have been widely used for deadlock control, scheduling and planning, and performance evaluation in a variety of resource allocation systems (Barkaoui & Abdallah, 1995; Li, Liu, Hanisch, & Zhou, 2012; Li, Wu, & Zhou, 2012; Zhang et al., 2015). Supervisory control is a suitable mechanism to enforce external constraints on a system to be controlled. In the framework of Petri nets, a supervisor that enforces supervisory control specifications is often represented by a set of control places.

Constraints associated with reachable states in a DES are a typical and important control specification in supervisory control theory of DESs. Many specifications can be converted into linear constraints. For example, deadlock problems in Petri nets are usually dealt with by finding a set of constraints, with respect to the markings, that can prevent the system from reaching deadlock states (Barkaoui, Couvreur, & Klai, 2005; Chen & Li, 2013; Li & Zhao, 2008; Li & Zhou, 2009). Most control requirements in system control design can be directly represented by a set of constraints.

Generally, there are two classes of constraints in Petri nets: linear and nonlinear. Linear constraints, also called generalized mutual exclusion constraints (GMECs) (Giua, DiCesare, & Silva, 1992; Ma, Li, & Giua, in press), play an important role in the development of supervisors for a system modeled by Petri nets. Many efforts have been done to enforce a GMEC by constructing a place invariant (PI) (Banaszak & Krogh, 1990; Chen, Li, & Zhou, 2012; Yamalidou, Moody, Lemmon, & Antsaklis, 1996). The PI-based approach is well-established and widely used by researchers and engineers. Yamalidou et al. (1996) study a variety of GMECs and design control places to enforce them by constructing PIs. Iordache and Antsaklis (2005, 2007) present an approach to the implementation of disjunctive GMECs. The work in Iordache and Antsaklis (2006) provides a good survey on the design of control places by PI based methods. Up to now, a lot of work has been done to deal with deadlocks by Petri nets (Chen, Li, & Barkaoui, 2014; Ghaffari, Rezg, & Xie, 2003; Huang, Jeng, Xie, & Chung, 2006; Li & Zhou, 2004, 2006, 2008; Liu, Li, & Zhou, 2010; Wu, Zhou, & Li, 2008). In fact, almost all of them compute control places by PIs (Chen, Li, & Zhou,

☆ This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 61203038 and 61374068, the Fundamental Research Funds for the Central Universities under Grant No. JB140402, the Recruitment Program of Global Experts, and the Science and Technology Development Fund, MSAR, under Grant No. 066/2013/A2. 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: yfchen@mail.xidian.edu.cn (Y.F. Chen), zhwli@xidian.edu.cn (Z.W. Li), kamel.barkaoui@cnam.fr (K. Barkaoui), giua@diee.unica.it, alessandro.giua@lsis.org (A. Giua).

http://dx.doi.org/10.1016/j.automatica.2015.02.014
0005-1098/

2014). Another line to deal with deadlocks in DESs is based on finite state automata, as shown in Nazeem and Reveliotis (2012), Nazeem, Reveliotis, Wang, and Lafortune (2011) and Reveliotis and Nazeem (2013). In this work, we focus on the enforcement of nonlinear constraints on Petri nets.

Uzam and Zhou (2006, 2007) provide an iterative method to design liveness-enforcing supervisors of Petri nets. They divide the reachability graph of a Petri net model into two parts: a live-zone (LZ) and a deadlock-zone (DZ), where the LZ contains all legal markings and the DZ includes all the illegal markings from which no legal marking is reachable. First, they compute the set of first-met bad markings (FBMs) of a net model. An FBM is an illegal marking that represents the very first entry from the LZ to the DZ. At each iteration, an FBM is singled out and a control place is computed to forbid it. The process cannot terminate until all FBMs are forbidden. Then, the controlled net is live since it cannot enter the DZ anymore. The method is intuitive but cannot lead to an optimal supervisor in general. In our previous work (Chen, Li, Khalgui, & Mosbahi, 2011), we improve Uzam and Zhou's results by proposing a method to obtain a maximally permissive supervisor. In Chen and Li (2011, 2012), the control places are computed by solving an integer linear programming problem (ILPP) that makes all legal markings reachable but all FBMs unreachable. Meanwhile, the objective function can minimize the number of the control places.

However, not all specifications can be represented as GMECs. In some cases, the specifications require to enforce a nonlinear constraint on a net model. For GMECs, the control places can be designed by constructing PIs. However, to the best of our knowledge, no work is reported to compute a supervisor by following the clue of handling GMECs if the constraints are nonlinear since we cannot directly construct PIs for the nonlinear constraints. This work focuses on the enforcement of nonlinear constraints. A supervisory structure is developed. It splits a transition in an original net model into a set of transitions. The proposed supervisor can also make all markings in the admissible-zone reachable and all markings in the forbidden-zone unreachable. The proposed approach is applicable to bounded Petri net models.

The rest of the paper is organized as follows. In Section 2, some basics of Petri nets are recalled. Section 3 reports the concepts and properties of nonlinear constraints. Section 4 provides a supervisory structure to implement a nonlinear constraint. Meanwhile, a number of examples are provided to illustrate the performance of the supervisory structure. Finally, conclusions are reached in Section 5.

2. Preliminaries

This section recalls the basics of Petri nets (Murata, 1989) and generalized mutual exclusion constraints (GMECs) (Giua et al., 1992).

2.1. Petri nets

A Petri net is a four-tuple $N = (P, T, F, W)$ where P and T are finite and non-empty sets. P is a set of places and T is a set of transitions with $P \cap T = \emptyset$. $F \subseteq (P\times T)\cup(T\times P)$ is a flow relation of the net, represented by arcs with arrows from places to transitions or from transitions to places. $W : (P \times T) \cup (T \times P) \to \mathbb{N}$ is a mapping that assigns a weight to an arc: $W(x, y) > 0$ if $(x, y) \in F$, and $W(x, y) = 0$, otherwise, where $(x, y) \in (P \times T) \cup (T \times P)$ and $\mathbb{N}$ is the set of non-negative integers. ${}^\bullet x = \{y \in P \cup T \mid (y, x) \in F\}$ is called the preset of x and $x^\bullet = \{y \in P \cup T \mid (x, y) \in F\}$ is called the postset of x. A marking is a mapping $M : P \to \mathbb{N}$. $M(p)$ denotes the number of tokens in place p. The pair (N, M_0) is called a marked Petri net or a net system. A net is pure (self-loop free) if $\forall(x, y) \in (P \times T) \cup (T \times P)$, $W(x, y) > 0$ implies $W(y, x) = 0$. The incidence matrix $[N]$ of a net N is a $|P| \times |T|$ integer matrix with $[N](p, t) = W(t, p) - W(p, t)$.

A transition $t \in T$ is enabled at marking M if $\forall p \in {}^\bullet t$, $M(p) \geq W(p, t)$. This fact is denoted as $M[t\rangle$. Once an enabled transition t fires, it yields a new marking M', denoted as $M[t\rangle M'$, where $M'(p) = M(p) - W(p, t) + W(t, p)$. The set of reachable markings of net N with initial marking M_0 is denoted by $R(N, M_0)$. It can be graphically expressed by a reachability graph, denoted as $G(N, M_0)$. It is a directed graph whose nodes are markings in $R(N, M_0)$ and arcs are labeled by the fired transitions.

Let (N, M_0) be a net system with $N = (P, T, F, W)$. A transition $t \in T$ is live if $\forall M \in R(N, M_0)$, $\exists M' \in R(N, M)$, $M'[t\rangle$. (N, M_0) is live if $\forall t \in T$, t is live. It is dead if $\nexists t \in T$, $M_0[t\rangle$.

2.2. Generalized mutual exclusion constraint

A GMEC (Giua et al., 1992) is a control requirement that limits a weighted sum of tokens contained in a subset of places. Let $[N]$ be the incidence matrix of a plant with n places and m transitions. A GMEC can be expressed as:

$$\sum_{i=1}^{n} w_i \cdot \mu_i \leq k \tag{1}$$

where μ_i denotes the number of tokens in place p_i at any reachable marking, and w_i and k are non-negative integers. Eq. (1) can be represented as a vector form, i.e.,

$$\vec{w}^T \cdot \vec{\mu} \leq k \tag{2}$$

where $\vec{w}$ is a weight vector of nonnegative integers with $\vec{w}(i) = w_i$, $\vec{\mu}$ is a vector of nonnegative integers with $\vec{\mu}(i) = \mu_i$ and k is a positive integer. A GMEC is usually denoted as $(\vec{w}, k)$.

By introducing a non-negative slack variable μ_s, Eq. (2) becomes

$$\vec{w}^T \cdot \vec{\mu} + \mu_s = k \tag{3}$$

where μ_s represents the marking of control place p_s, generally called a monitor. The firing of a transition t modifies the tokens in p_s by a constant:

$$\Delta(t) = -\vec{w}^T \cdot [N](\bullet, t). \tag{4}$$

In fact, $\forall M_1, M_2 \in R(N, M_0)$ with $M_1 = M_2 + [N](\bullet, t)$, we have $\Delta(t) = M_1(p_s) - M_2(p_s)$. Thus, the incidence vector $[N_s]$ of p_s can be computed by:

$$[N_s] = -\vec{w}^T \cdot [N]. \tag{5}$$

The initial marking $M_0(p_s)$ of p_s can be calculated as follows:

$$M_0(p_s) = k - \vec{w}^T \cdot M_0. \tag{6}$$

3. Generalizations of arbitrary marking constraints

In this section, we present basic concepts of nonlinear constraints in Petri nets in the sense of reachability graph analysis. A constraint for a Petri net is in general a predicate with respect to the states (markings) of the Petri net. Let c be a constraint that restricts the tokens contained in a subset of places of a Petri net model (N, M_0). In this work, the constraints are only associated with markings while no firing vectors of transitions are considered.

Definition 1. Let c be a constraint and $M \in R(N, M_0)$ a marking of a net (N, M_0). The function $F(c, M)$ is defined as $F(c, M) = 1$ if M satisfies c and $F(c, M) = 0$ otherwise.

Given a constraint c, the reachable markings of a net are classified into two groups: admissible ones that satisfy c and inadmissible ones that do not satisfy c, as defined below:

Definition 2. Let c be a constraint of a Petri net model (N, M_0). A marking $M \in R(N, M_0)$ is said to be admissible with respect to c if $F(c, M) = 1$. The set of admissible markings of c is denoted by $\mathcal{M}_c$. A reachable marking M of (N, M_0) is said to be inadmissible with respect to c if $F(c, M) = 0$. The set of inadmissible markings of c is denoted by $\mathcal{M}_{\bar{c}}$.