

# Marking optimization of deterministic timed weighted marked graphs under infinite server semantics

Zhou He<sup>1</sup>, Zhiwu Li<sup>2</sup>, Isabel Demongodin<sup>3</sup>, and Alessandro Giua<sup>4</sup>

## Abstract

For modelling and analysis of automated production systems as batch or high throughput systems, timed Petri nets are commonly used. In this paper, we deal with the marking optimization problem of a timed weighted marked graphs under infinite server semantics. The problem consists in finding an initial marking to *minimize the weighted sum of tokens in places* while the average cycle time is less than or equal to a given value. We propose two different heuristic approaches to solve the optimization problem.

Published as:

Z. He, Z.W. Li, I. Demongodin, and A. Giua. “Marking optimization of deterministic timed weighted marked graphs under infinite server semantics,” 3rd Int. Conf. on Control, Decision and Information Technologies (Malta), April 6–8, 2016.

<sup>1</sup>Zhou He is with the School of Electro-Mechanical Engineering, Xidian University, Xi’an 710071, China and also with Aix-Marseille Université, CNRS, ENSAM, Université de Toulon, LSIS UMR 7296, Marseille 13397, France. [zhouhe@stu.xidian.edu.cn](mailto:zhouhe@stu.xidian.edu.cn)

<sup>2</sup>Zhiwu Li (corresponding author) is with the Institute of Systems Engineering, Macau University of Science and Technology, Taipa, Macau, and also with School of Electro-Mechanical Engineering, Xidian University, Xi’an 710071, China. [zhwli@xidian.edu.cn](mailto:zhwli@xidian.edu.cn)

<sup>3</sup>Isabel Demongodin is with Aix-Marseille Université, CNRS, ENSAM, Université de Toulon, LSIS UMR 7296, Marseille 13397, France. [isabel.demongodin@lsis.org](mailto:isabel.demongodin@lsis.org)

<sup>4</sup>Alessandro Giua is with Aix-Marseille Université, CNRS, ENSAM, Université de Toulon, LSIS UMR 7296, Marseille 13397, France and also with DIEE, University of Cagliari, Cagliari 09124, Italy. [alessandro.giua@lsis.org](mailto:alessandro.giua@lsis.org), [giua@diee.unica.it](mailto:giua@diee.unica.it)

## I. INTRODUCTION

In real-world systems, activities do not take place instantaneously. Every activity in a system has a time duration which is different from zero. Petri nets (PNs) are well known as efficient tools for modeling discrete event systems, especially manufacturing systems. In this paper, we study a particular class of Petri nets called *timed weighted marked graphs* (TWMGs). The main feature of this class of nets is that each place has only one input and one output transition. Moreover, in *deterministic* TWMGs the steady state performance can be evaluated in terms of the average cycle time of the net.

Timed weighted marked graphs and timed marked graphs (TMGs) find wide applications in manufacturing systems or embedded systems. For instance, transitions represent actions and tokens correspond to physical products in an assembly line or data exchanged between two processes in embedded systems.

Several studies for this class of Petri nets can be found in the literature. For instance, Teruel *et al.* [1] proposed several techniques for the analysis of weighted marked graphs (WMGs). Campos *et al.* [2] developed methods to compute the average cycle time of TMGs for a given initial marking. Munier [3] proposed a method to transform a WMG into a marked graph (MG) under *single server* semantics hypothesis and Nakamura and Silva [4] discussed the same problem under *infinite server* semantics hypothesis.

However, in the literature, few works deal with the optimization problem of TWMGs. Benazouz *et al.* [12] developed an algorithm to minimize the overall buffer capacities with throughput constraint for TWMGs. He *et al.* [8] proposed a mixed integer linear programming problem (MILLP) to solve the cycle time optimization problem of a TWMG. The marking optimization of TWMGs consists in finding an initial marking to minimize the weighted sum of tokens in places while the average cycle time is less than or equal to a given value. Sauer [9] developed a heuristic solution based on an iterative process to solve the marking optimization of TWMGs under single server semantic hypothesis. He *et al.* [10], [11] presented a novel heuristic method to deal with the marking optimization which was shown to be more effective than that of Sauer [9].

In this paper, we consider the marking optimization of a TWMG under infinite server semantics (the degree of self-concurrency of each transition is infinite). While under single server semantics services in a transition are provided sequentially, i.e., there is no self-concurrency, in an infinite server semantics the number of concurrent servers is equal to the enabling degree of the transition. Note that infinite server semantics is more general than single server (or in general  $k$  server) semantics. In fact, single (resp.,  $k$ ) server semantics can be simulated by just adding to each transition a self-loop place with one (resp.,  $k$ ) tokens. For this reason we adopt this more general semantics in this work.

The approach we proposed requires choosing a proper candidate initial marking by solving an MILPP. The cycle time of this selected marking is usually close to the desired value. Starting from this initial marking, we adopt two different heuristic methods from [11] to solve the optimization problem by a greedy approach. Some experimental studies show that the proposed approaches are fast and efficient.

This paper is structured as follows. In the following section, we briefly recall some basic concepts and the main properties. In Section III, we present the problem statement. In Section IV, we propose two different heuristic

approaches to solve the optimization problem. To validate the legitimacy of our approaches, an example is illustrated in Section V. Conclusions and future work are finally drawn in Section VI.

## II. BACKGROUND

### A. Generalities

We assume that the reader is familiar with the structure, firing rules, and basic properties of PNs (see [1], [14], and [15]). In this section, we recall the formalism used in the paper. A *place/transition* net ( $P/T$  net) is a structure  $N = (P, T, \mathbf{Pre}, \mathbf{Post})$ , where  $P$  is a set of  $n$  places;  $T$  is a set of  $m$  transitions;  $\mathbf{Pre} : P \times T \rightarrow \mathbb{N}$  and  $\mathbf{Post} : P \times T \rightarrow \mathbb{N}$  are the *pre-* and *post-incidence* functions that specify the arcs;  $\mathbf{C} = \mathbf{Post} - \mathbf{Pre}$  is the *incidence matrix*, where  $\mathbb{N}$  is a set of non-negative integers.

A vector  $\mathbf{x} = (x_1, x_2, \dots, x_m)^T \in \mathbb{N}^{|T|}$  such that  $\mathbf{x} \neq \mathbf{0}$  and  $\mathbf{C} \cdot \mathbf{x} = \mathbf{0}$  is a *T-semiflow*. A vector  $\mathbf{y} = (y_1, y_2, \dots, y_n)^T \in \mathbb{N}^{|P|}$  such that  $\mathbf{y} \neq \mathbf{0}$  and  $\mathbf{y}^T \cdot \mathbf{C} = \mathbf{0}$  is a *P-semiflow*. The supports of a T-semiflow and a P-semiflow are defined by  $\|\mathbf{x}\| = \{t_i \in T | x_i > 0\}$  and  $\|\mathbf{y}\| = \{p_i \in P | y_i > 0\}$ , respectively. A *minimal* T-semiflow<sup>1</sup> (resp. P-semiflow) is a T-semiflow  $\|\mathbf{x}\|$  (resp. P-semiflow  $\|\mathbf{y}\|$ ) that is not a superset of the support of any other T-semiflow (resp. P-semiflow), and its components are mutually prime.

A *marking* is a vector  $\mathbf{M} : P \rightarrow \mathbb{N}$  that assigns to each place of a  $P/T$  net a non-negative integer number of tokens; we denote the marking of place  $p$  as  $M(p)$ . A  $P/T$  system or *net system*  $\langle N, \mathbf{M}_0 \rangle$  is a net  $N$  with an *initial marking*  $\mathbf{M}_0$ .

A  $P/T$  net is said to be *ordinary* when all of its arc weights are equal to one. A *marked graph* (also called an event graph) is an ordinary Petri net such that each place has exactly one input and one output transition. A *weighted marked graph* (also called a weighted event graph) is a net that also satisfies this structural condition but may not be ordinary, i.e., the weight associated with each arc is a non-negative integer number.

A net is *strongly connected* if there exists a directed path from any node in  $P \cup T$  to every other node. Let us define an *elementary circuit*  $\gamma$  (or elementary cycle) of a net as a directed path that goes from one node back to the same node without passing twice on the same node. In a strongly connected net, it is easy to show that each node belongs to an elementary circuit, and thus the name *cyclic nets* are also used to denote this class.

Given a place  $p$  of a WMG, let  $t_i$  (resp.  $t_j$ ) be its unique input (resp. output) transition as shown in Fig. 1. We denote by  $w(p) = \text{Post}(p, t_i)$  the weight of its input arc and by  $\nu(p) = \text{Pre}(p, t_j)$  the weight of its output arc. For any place  $p \in P$ , we denote by  $\text{gcd}_p$  the greatest common divisor of the integers  $w(p)$  and  $\nu(p)$ .

Every elementary circuit  $\gamma$  of a WMG is neutral, if the following condition holds.

$$\prod_{p \in \gamma} \frac{\nu(p)}{w(p)} = 1$$

In other words, in a neutral circuit the product of the weights of all pre-arcs is equal to that of all post-arcs. This means that if the circuit initially contains enough tokens, it is possible to fire all transitions along the path returning

<sup>1</sup>This is also called a minimal and minimal support semiflow in some references. For the sake of simplicity, we call it a minimal semiflow.

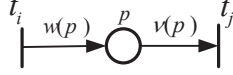


Fig. 1. A place  $p$  between two transitions  $t_i$  and  $t_j$ .

to the same initial marking. A WMG is said to be neutral if all circuits are neutral. It is well known that a neutral WMG has a unique minimal T-semiflow  $\mathbf{x}$  which contains all transitions in its support [1].

A strongly connected WMG in which all circuits are neutral is *bounded*, i.e., there exists an integer  $B$  such that the marking of any place  $p$  is not greater than  $B$  at any reachable marking. In this paper, we limit our study to strongly connected WMGs in which all circuits are neutral.

### B. Dynamic behavior

There exists two main approaches for introducing the concept of time in PN models, namely, associating a time interpretation with either transitions (T-timed) or places (P-timed) [13]. A deterministic T-timed  $P/T$  net is a pair  $N^\delta = (N, \delta)$ , where  $N = (P, T, \mathbf{Pre}, \mathbf{Post})$  is a standard  $P/T$  net, and  $\delta : T \rightarrow \mathbb{N}$ , called firing delay, assigns a non-negative integer fixed firing duration to each transitions. A transition with a firing delay equal to 0 is said to be immediate. In terms of a deterministic P-timed  $P/T$  net, each place  $p$  is assigned a non-negative integer number  $\delta(p)$  which represents the sojourn time that a token must spend in place  $p$  before it becomes available for its output transitions. In the rest of this paper, we will assume that the timing structure associates delays to transitions, namely, we consider deterministic T-timed WMG.

A transition  $t_i$  is enabled at  $M_j$  if  $M_j \geq \mathbf{Pre}(\cdot, t_i)$  and an enabled transition  $t_i$  may fire yielding a marking  $M'$  with

$$M' = M_j + \mathbf{C}(\cdot, t_i) \quad (1)$$

where  $\mathbf{Pre}(\cdot, t_i)$  (resp.  $\mathbf{C}(\cdot, t_i)$ ) denotes the column of the matrix  $\mathbf{Pre}$  (resp.  $\mathbf{C}$ ) associated with transition  $t_i$ .

The state of a TWMG is defined not only by the marking, as for  $P/T$  nets, but also by the clocks associated with transitions. The enabling degree of  $t_i$  enabled at a marking  $M_j$  denoted by  $\alpha_i(j)$  is the biggest integer number  $k$  such that

$$M_j \geq k \cdot \mathbf{Pre}(\cdot, t_i). \quad (2)$$

Under infinite server semantics, at each time instant  $\tau_j$  the number of clocks  $o_i$  associated with a transition  $t_i$  is equal to its current enabling degree, i.e.,  $o_i = \{o_{i,1}, \dots, o_{i,\alpha_i(j)}\}$ ; this number changes with the enabling degree, thus it can change each time the net evolves from one marking to another one, namely, each time that a transition fires. If transition  $t_i$  is not enabled at marking  $M_j$ , it has no clock. Assuming that

$$o_i^* = \min\{o_{i,1}, \dots, o_{i,\alpha_i(j)}\}$$

and let

$$o^* = \min_{i=1, \dots, m} \{o_i^*\}$$

be the minimum among the values of the clocks  $o_i^*$ . At the time instant  $\tau_{j+1} = \tau_j + o^*$ , transitions whose clocks are equal to  $o^*$  fire yielding a new marking as in Eq. (1). Note that under infinite server semantics, if the minimal value of the clock  $o_i^*$  at marking  $M_j$  holds for more than one clock, as an example  $k$ , in the set  $\{o_{i,1}, \dots, o_{i,\alpha_i(j)}\}$ , this means that if the transition will be the next one to fire, it will fire  $k$  times simultaneously.

### C. Cycle time of a TWMG

The *average cycle time*  $\chi(\mathbf{M})$  of a TWMG system  $\langle N^\delta, \mathbf{M} \rangle$  is the average time to fire once the T-semiflow under the earliest operational model (i.e., transitions are fired as soon as possible). Considering a net consisting only of one circuit, we define  $\chi_\gamma(\mathbf{M})$  as the average cycle time of circuit  $\gamma$ .

Let  $\Gamma$  represent the set of elementary circuits of a cyclic TWMG. It holds that the average cycle time of each circuit is smaller than or equal to the one of the nets [10], i.e.,

$$\chi(\mathbf{M}) \geq \max_{\gamma \in \Gamma} \chi_\gamma(\mathbf{M}).$$

In [17], the authors proved that the earliest execution of a live and strongly connected TMG with integer delays is ultimately repetitive following an execution pattern. Moreover, its throughput is maximum. The period of the pattern is  $\tau$  and the number of firings of every transition within a period is  $f$  (the periodicity). In terms of a strongly connected TWMG, the earliest execution is also ultimately periodic. The number of firings of transition  $t_i$  within the steady period is  $f_i$ . The average cycle time of a TWMG is thus equal to

$$x_i \cdot \frac{\tau}{f_i}. \quad (3)$$

It was shown in [5] and [14] that a lower bound for the average cycle time of a live and bounded TWMG system  $\langle N^\delta, \mathbf{M} \rangle$  can be computed by solving following LPP:

$$\begin{aligned} & \min v \\ & \text{s.t.} \\ & \mathbf{Cz} + v\mathbf{M} \geq \mathbf{Pre}\boldsymbol{\theta} \end{aligned} \quad (4)$$

where  $\boldsymbol{\theta} \in \mathbb{N}^m$  is the vector containing all firing delays of timed transitions (recall that  $m = |T|$ ). Note that for a TMG whose minimal T-semiflow is equal to  $\vec{1}$ , thus the element  $\theta_i$  of vector  $\boldsymbol{\theta}$  is simple equal to the delay time of corresponding transition  $t_i$ , i.e.,

$$\boldsymbol{\theta} = [\delta_1, \delta_2, \dots, \delta_m]^T.$$

Nevertheless, the vector  $\boldsymbol{\theta}$  of a TWMG should be modified as follows:

$$\boldsymbol{\theta} = [x_1 \cdot \delta_1, x_2 \cdot \delta_2, \dots, x_m \cdot \delta_m]^T$$

where  $\mathbf{x}$  is the minimal T-semiflow.

The decision variables are  $v \in \mathbb{R}^+$  and  $\mathbf{z} \in \mathbb{R}^m$ : the optimal value of  $v$  is a lower bound of the average cycle time of the TWMG system  $\langle N^\delta, \mathbf{M} \rangle$ , i.e.,

$$\chi(\mathbf{M}) \geq v \quad (5)$$

Chao *et al.* [18] proposed a method to compute the cycle time of a TWMG but under restrictive conditions at initial marking. One way to analytically compute the average cycle time of a TWMG is to convert it into an equivalent P-timed marked graphs (PMGs). In fact, Munier [3] showed that a TWMG system under single server semantics can be transformed into an *equivalent* TMG system which describes the same precedence constraints on the firing of transitions. Nakamura and Silva [4] proved that a TWMG system under infinite server semantics can be transformed into an *equivalent* PMG system and the PN language of the PMG system is the same as that of the TWMG system. The main drawback of this approach is that the structure of equivalent PMG can exponentially increase with respect to the structure of original TWMG. Thus, computing the equivalent PMG may not be possible for a wide class of graphs. Otherwise, we can use simulation to compute the average cycle time which is faster and easier [19].

### III. PROBLEM STATEMENT

In this paper, the *marking optimization problem* of a neutral TWMG under infinite server semantics is considered. We aim to find an initial marking  $\mathbf{M}$  at which the average cycle time is less than or equal to a given value. Among all feasible solutions, we look for those that minimize weighted sum of tokens in places, i.e., minimize the cost of resources.

In other words we look for an initial marking  $\mathbf{M}$  that provides the optimal solution of the following problem:

$$\begin{aligned} \min \quad & f(\mathbf{M}) = \mathbf{y}^T \cdot \mathbf{M} \\ \text{s.t.} \quad & \\ & \chi(\mathbf{M}) \leq b \end{aligned}$$

where

- $\chi(\mathbf{M})$  is the average cycle time of the TWMG with initial marking  $\mathbf{M}$ .
- $\mathbf{y}^T = (y_1, \dots, y_n)$  is a non-negative weight vector.
- $b$  is a given positive integer number, representing the upper bound on the average cycle time, i.e., the lower bound on the throughput.

We choose the weight vector  $\mathbf{y}$  as a P-semiflow since the value of  $\mathbf{y}^T \cdot \mathbf{M}$  for every reachable marking  $\mathbf{M}' \in R(N, \mathbf{M})$  is an invariant. In particular, we choose the P-semiflow  $\mathbf{y}$  equals to the weighted sum of all minimal P-semiflows, i.e.,  $\mathbf{y} = \sum_{\gamma \in \Gamma} \lambda_\gamma \cdot \mathbf{y}_\gamma$ , where  $\mathbf{y}_\gamma$  represent the minimal P-semiflow corresponding to circuit  $\gamma$  and  $\lambda_\gamma$  represent the cost of the resources modeled by tokens in the support of  $\mathbf{y}_\gamma$ .

### IV. MARKING OPTIMIZATION FOR TWMGs: A HEURISTIC SOLUTION

We propose here two different heuristic solutions to solve the marking optimization problem of a neutral TWMG. A candidate live initial marking is firstly computed by an analytical method. The average cycle time of this selected

marking is usually greater than the desired value  $b$ . Thus, more tokens should be added to decrease the average cycle time until it satisfies the constraint  $\chi(\mathbf{M}) \leq b$ . In the following Subsection, an MILLP method to compute a candidate live initial marking is proposed. In Subsections IV-B and IV-C, two different heuristic approaches adopted from [11], [6], [7] are presented to solve the optimization problem.

#### A. Selection of a candidate marking: an MILLP approach

**Useful tokens:** Firstly, we present some notations on useful tokens taken from literature [16]. For a TWMG, the initial marking  $M(p_i)$  of any place  $p_i$  can be replaced by  $M^*(p_i) = \left\lfloor \frac{M(p_i)}{\gcd_{p_i}} \right\rfloor \cdot \gcd_{p_i}$  tokens without any influence on the precedence constraints induced by  $p_i$ .

If  $M(p_i)$  is not a multiple of  $\gcd_{p_i}$ , there will always be  $M(p_i) - M^*(p_i)$  tokens remaining in place  $p_i$  that will never be used in the firing of the output transition of place  $p_i$ . As a result, we can deduce that the average cycle time at  $\mathbf{M}$  and  $\mathbf{M}^*$  are the same.

**Selection of a live initial marking:** To start our heuristic solution, we present an analytical method to select a live initial marking  $\mathbf{M}$  based on Eq. (4). The average cycle time of this marking usually satisfy  $\chi(\mathbf{M}) \leq b$ . Nevertheless, we find that in practical examples, the average cycle time  $\chi(\mathbf{M})$  is very close to the desired value  $b$ . Let us first recall some basic results regarding liveness of a WMG. Note that Propositions 1 and 2 are valid for both infinite server semantics and single server semantics.

*Proposition 1:* (Teruel *et al.* [1]) *A strongly connected WMG is live iff each elementary circuit is live.*

Teruel *et al.* [1] proposed a sufficient condition for the liveness of a neutral circuit. They defined a marking  $\mathbf{M}_D = (v(p_1) - 1, v(p_2) - 1, \dots, v(p_n) - 1)^T$  and a weighted function  $W(\mathbf{M}) = \mathbf{y}^T \cdot \mathbf{M}$  of marking  $\mathbf{M}$ .

*Proposition 2:* (Teruel *et al.* [1]) *If  $W(\mathbf{M}) > W(\mathbf{M}_D)$ , then the neutral circuit is live.*

*Proposition 3:* *Let  $\mathbf{M}$  be a marking which satisfies the following condition:*

$$\begin{aligned} & \min v \\ & \text{s.t.} \\ & \begin{cases} \mathbf{Cz} + v\mathbf{M} \geq \mathbf{Pre}\boldsymbol{\theta}, & (a) \\ v = b, & (b) \\ \mathbf{y}_\gamma^T \cdot \mathbf{M} > W(\mathbf{M}_D^\gamma), \forall \gamma \in \Gamma, & (c) \\ \frac{M(p_i)}{\gcd(p_i)} \in \mathbb{N}, \forall p_i \in P. & (d) \end{cases} \end{aligned} \quad (6)$$

*Then, the TWMG system  $\langle N, \mathbf{M} \rangle$  will be live and  $b$  is a lower bound of the average cycle time  $\chi(\mathbf{M})$ .*

**Proof.** The constraint (a) is adopted from Eq. (4) and can provide a marking  $\mathbf{M}$  whose lower bound of the average cycle time is equal to  $b$  if  $\mathbf{C}$ ,  $\mathbf{Pre}$ ,  $\boldsymbol{\theta}$ , and  $b$  are given. The constraint (b) specifies that the lower bound of the average cycle time should equal to  $b$ , i.e.,  $\chi(\mathbf{M}) \geq b$ .

As we mentioned in Eq. (2), the constraint (c) ensures that the TWMG system  $\langle N^\delta, \mathbf{M} \rangle$  will be live. The number of tokens in each place  $p_i$  should be a multiple of  $\gcd_{p_i}$  which is guaranteed by constraint (d).  $\square$

As it is stated in Proposition 3, the average cycle time of  $M$  is usually greater than or equal to the upper bound  $b$ . Thus, tokens should be added to the net until the requirement on the average cycle time is satisfied.

### B. Heuristic approach 1

In this Subsection, we present heuristic approach 1 which is adopted from [10]. The main idea underlying this heuristic approach is the following: at each iteration step, we add tokens to some circuits until the average cycle time is less than or equal to the upper bound of the cycle time. In the rest of this paper, the following notations will be used.

- $\Gamma_c$ : the set of selected circuits to add tokens.
- $P_c$ : the set of places belonging to  $\Gamma_c$ .

After we obtain a candidate initial marking  $M$ , we can compute the average cycle time  $\chi(M)$  of the TWMG and  $\chi_\gamma(M)$  for every elementary circuit. Tokens should be added to the circuits which satisfy the following condition:

$$\Gamma_c = \{\gamma \in \Gamma | \chi_\gamma > b\}. \quad (7)$$

And for each circuit  $\gamma$  belongs to  $\Gamma_c$ , we select one place  $p_r$  and add  $\text{gcd}_{p_r}$  tokens to it. We choose the one that increases  $f(M_0)$  as little as possible, i.e., the increment of the criteria value  $f(M_0)$  should be the least after adding  $\text{gcd}_{p_r}$  tokens. We define an  $n$ -dimensional vector  $I$  of zeros and ones.

$$I^T = (I_{p_1}, I_{p_2}, \dots, I_{p_n}) \quad (8)$$

where

$$I_{p_r} = \begin{cases} 1, & \text{add } \text{gcd}_{p_r} \text{ tokens to place } p_r \\ 0, & \text{add } 0 \text{ tokens to place } p_r \end{cases} \quad (9)$$

In other words, we add tokens to places with the coefficient  $I_{p_r} = 1$ . Let  $G_d = (\text{gcd}_{p_1} \cdot y_1, \text{gcd}_{p_2} \cdot y_2, \dots, \text{gcd}_{p_n} \cdot y_n)$ , where  $y$  is a P-semiflow and  $\text{gcd}_{p_r} \cdot y_r$  represents the increment of  $f(M)$  after adding  $\text{gcd}_{p_r}$  tokens to place  $p_r$ . We denote by  $\Delta f(M)$  the total increment of  $f(M)$ , where

$$\Delta f(M) = I^T \cdot G_d \quad (10)$$

Then, we can select places by solving the following problem:

$$\begin{aligned} \min \quad & \Delta f(M) \\ \text{s.t.} \quad & \\ & \sum_{p \in \gamma} I_p = 1, \quad \forall \gamma \in \Gamma_c \end{aligned} \quad (11)$$

and we denote  $P_c$  by following equation:

$$P_c = \{p | I_p = 1\} \quad (12)$$

The constrains in Eq. (11) ensures that only one place should be selected for each circuit that belongs to  $\Gamma_c$ .



### C. Heuristic approach 2

We propose here another heuristic approach to solve the optimization problem. The basic idea of the heuristic process is to allocate tokens, which reduces the average cycle time  $\chi(\mathbf{M})$  as much as possible while increases the objective function  $f(\mathbf{M})$  (i.e., weighted sum of tokens) as less as possible. At each step, we choose one circuit which has the maximal average cycle time (also called critical circuit) among all circuits and add tokens to this circuit. Thus, the selected circuit in Eq. (7) should be redefined as follows.

$$\Gamma_c = \{\gamma \in \Gamma \mid \chi_\gamma(\mathbf{M}) = \chi^*(\mathbf{M})\}, \quad (13)$$

where

$$\chi^*(\mathbf{M}) = \max_{\gamma \in \Gamma} \chi_\gamma(\mathbf{M}). \quad (14)$$

If there exists more than one critical circuit, we choose one. After we choose a critical circuit, we select one place  $p$  and add  $k$  tokens to it. The number  $k$  is a multiple of  $\text{gcd}_p$  which represents the minimal number of tokens that we should add to decrease the average cycle time of the critical circuit. It can be computed by using simulation. We denote the decrease in the average cycle time by  $\Delta_\gamma \chi(\mathbf{M})$  after allocating  $k$  tokens to place  $p$ . We have

$$\Delta_\gamma \chi(\mathbf{M}) = \chi_\gamma(\mathbf{M}') - \chi_\gamma(\mathbf{M}) \quad (15)$$

where  $\mathbf{M}'$  is the marking such that  $\mathbf{M}'(p) = \mathbf{M}(p) + k$  and  $\mathbf{M}'(p') = \mathbf{M}(p')$  if  $p' \neq p$ . Let  $\Delta f(\mathbf{M})$  be the gain in criterion value, i.e., the resources that we add, where

$$\Delta f(\mathbf{M}) = y_p \cdot k \quad (16)$$

We introduce a criterion  $\Delta_p$  in which  $p$  takes into account both the decreasing of the average cycle time and the gain in criterion value, i.e.,

$$\Delta_p = \frac{\Delta f(\mathbf{M})}{\Delta_\gamma \chi(\mathbf{M})}. \quad (17)$$

Tokens will be allocated to the place such that

$$P_c = \{p^* \mid \Delta_{p^*} = \min_{p \in \Gamma_c} \Delta_p\}. \quad (18)$$

Note that, the computation of  $\Delta_p$  is simple: the amount of computation is proportional to the number of places which belong to the critical circuit. At each iteration step, if there is more than one place with minimal value of  $\Delta_p$ , we keep all the optimal allocations to next iteration step.

## V. ILLUSTRATIVE EXAMPLE

Let us illustrate the proposed approaches through an example taken from [16]. It combines cyclic assembly process, buffers, WIP, and batch operations. Two parallel machines (machine one and machine two) are working on items. Machine three loads two parts produced by machine one and three parts produced by machine two and assembles them to get one product. The assembly process is finished by machine four. The batching transportation device removes three finished products from the workshop and brings six items to machine one and nine items

to machine two, respectively. The TWMG model of the assembly process is depicted by Fig. 2. Transitions  $t_1$ ,  $t_2$ ,  $t_3$ ,  $t_4$ , and  $t_5$  represent machine one, machine two, machine three, machine four, and transportation device, respectively.

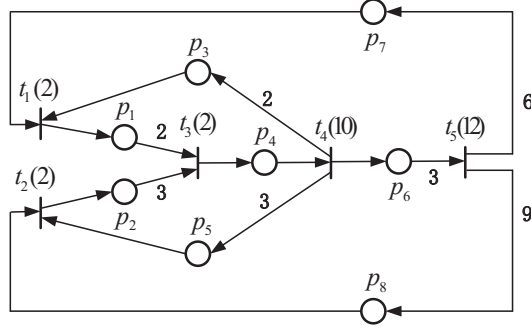


Fig. 2. The TWMG model of an assembly line.

The minimal T-semiflow of the TWMG is  $\mathbf{x}_1 = (6, 9, 3, 3, 1)$  and the minimal P-semiflows are  $\mathbf{y}_1 = (1, 0, 0, 2, 0, 2, 1, 0)^T$ ,  $\mathbf{y}_2 = (0, 1, 0, 3, 0, 3, 0, 1)^T$ ,  $\mathbf{y}_3 = (1, 0, 1, 2, 0, 0, 0, 0)^T$ , and  $\mathbf{y}_4 = (0, 1, 0, 3, 1, 0, 0, 0)^T$ . Thus, the weighted elementary circuits corresponding to all minimal P-semiflows are  $\gamma_1 = p_1 t_3 p_4 t_4 p_6 t_5 p_7 t_1$ ,  $\gamma_2 = p_2 t_3 p_4 t_4 p_6 t_5 p_8 t_2$ ,  $\gamma_3 = p_1 t_3 p_4 t_4 p_3 t_1$ , and  $\gamma_4 = p_2 t_3 p_4 t_4 p_5 t_2$ .

TABLE I  
HEURISTIC PROCESS OF APPROACH 1 FOR THE ASSEMBLY LINE.

step	$M$	$b$	$\chi$	$\chi_{\gamma_1}$	$\chi_{\gamma_2}$	$\chi_{\gamma_3}$	$\chi_{\gamma_4}$	$\Gamma_c$	$P_a$	$f(M)$	CPU time [s]
0	$(0, 0, 0, 10, 0, 0, 0, 0)^T$	8	8.7	8.7	8.7	4.2	4.2	$\{\gamma_1, \gamma_2\}$	$\{p_7, p_8\}$	150	–
1	$(0, 0, 0, 10, 0, 0, 1, 1)^T$	8	8.7	8.7	8.7	4.2	4.2	$\{\gamma_1, \gamma_2\}$	$\{p_7, p_8\}$	154	–
2	$(0, 0, 0, 10, 0, 0, 2, 2)^T$	8	8.7	8.7	8.7	4.2	4.2	$\{\gamma_1, \gamma_2\}$	$\{p_7, p_8\}$	158	–
3	$(0, 0, 0, 10, 0, 0, 3, 3)^T$	8	8.7	8.7	8.7	4.2	4.2	$\{\gamma_1, \gamma_2\}$	$\{p_7, p_8\}$	162	–
4	$(0, 0, 0, 10, 0, 0, 4, 4)^T$	8	8.7	6.5	8.7	4.2	4.2	$\{\gamma_2\}$	$\{p_8\}$	166	–
5	$(0, 0, 0, 10, 0, 0, 4, 5)^T$	8	8.7	6.5	8.7	4.2	4.2	$\{\gamma_2\}$	$\{p_8\}$	168	–
6	$(0, 0, 0, 10, 0, 0, 4, 6)^T$	8	6.5	6.5	6.5	4.2	4.2	–	–	170	92.7

TABLE II  
HEURISTIC PROCESS OF APPROACH 2 FOR THE ASSEMBLY LINE.

Step	$M$	$b$	$\chi$	$\chi_{\gamma_1}$	$\chi_{\gamma_2}$	$\chi_{\gamma_3}$	$\chi_{\gamma_4}$	$\Gamma_c$	$P_a$	Number of tokens $k$	$f(M)$	CPU time [s]
0	$(0, 0, 0, 10, 0, 0, 0, 0)^T$	8	8.7	8.7	8.7	4.2	4.2	$\gamma_1$	$p_7$	4	150	–
1	$(0, 0, 0, 10, 0, 0, 4, 0)^T$	8	8.7	6.5	8.7	4.2	4.2	$\gamma_2$	$p_8$	6	158	–
2	$(0, 0, 0, 10, 0, 0, 4, 6)^T$	8	6.5	6.5	6.5	4.2	4.2	–	–	–	170	156.5

TABLE III  
SIMULATION RESULTS FOR DIFFERENT VALUE OF  $b$ .

$b$	Heuristic approach 1			Heuristic approach 2		
	Iteration steps	Objective function $f(\mathbf{M})$	CPU time [s]	Iteration steps	Objective function $f(\mathbf{M})$	CPU time [s]
2	0	585	12.1	0	585	12.1
5	6	261	101.4	2	261	127.6
12	6	125	94.5	2	125	107.4
18	3	86	42.4	2	86	81.3

The number of tokens in  $\gamma_1$  and  $\gamma_2$  represent the number of items proceed (i.e., WIP) by machines one and two, respectively. Thus, we initialize the cost of the tokens in these circuits to two, and that of circuits three and four to one, i.e.,  $\lambda_1 = 2$ ,  $\lambda_2 = 2$ ,  $\lambda_3 = 1$ , and  $\lambda_4 = 1$ . Thus the P-semiflow we used in the criteria is

$$\mathbf{y} = \sum_{\gamma \in \Gamma} \lambda_{\gamma} \cdot \mathbf{y}_{\gamma} = (3, 3, 1, 15, 1, 10, 2, 2)^T.$$

Let us consider the following optimization problem

$$\begin{aligned} \min \quad & \mathbf{y}^T \cdot \mathbf{M} \\ \text{s.t.} \quad & \\ & \chi(\mathbf{M}) \leq 8 \end{aligned}$$

By using the technique introduced in Eq. (6), we can obtain an initial marking  $\mathbf{M}_0 = (0, 0, 0, 10, 0, 0, 0, 0)^T$ . This marking has an average cycle time which is very close to the desired value. Thus, the optimization problem can be efficiently solved by the proposed heuristic approaches. We use HYPENS [19] to implement the heuristic algorithms 1 and 2 and the simulation results are shown in Tables I and II, respectively. As one can see, it takes six iteration steps for approach 1 and two iteration steps for approach 2. Note that at each iteration step, heuristic approach 2 needs compute more information than heuristic 1. Both the solutions of approaches 1 and 2 have the same value of objective function  $f(\mathbf{M})$ , i.e., the total cost of the resources of the assembly line is 170.

To better verify the effectiveness of the two heuristic approaches, we test the example for different value of  $b$  and the simulation results are shown in Table III. We can observe that in all the test cases, heuristic approach 1 is slightly faster than heuristic approach 2, while the obtained objective functions  $f(\mathbf{M})$  are the same. Note that in the case that  $b = 2$ , the marking  $\mathbf{M}$  obtained by the MILPP (6) is a heuristically good solution, i.e.,  $\chi(\mathbf{M}) \leq b$ . Thus, we do not need to add more tokens to the system. The simulation studies show that the two approaches produce comparable results. They always find the same optimal solution and the execution time is very similar. While the presented results which approach 1 are always faster than approach 2 may depend on the particular example considered.

## VI. CONCLUSION

This paper deals with the marking optimization of deterministic TWMGs under infinite server semantics, which is a more general case than previous one [11]. The problem consists in finding an initial marking to minimize the

weighted sum of tokens in places while the average cycle time is less than or equal to a given value. We propose two different heuristic approaches to solve the optimization problem. Future work will pertain to find an analytical solution to solve the marking optimization of a TWMG under infinite server hypothesis.

## VII. ACKNOWLEDGEMENT

This work was supported by the National Natural Science Foundation of China under Grant Nos. 61374068 and 61472295, the Recruitment Program of Global Experts, the Science and Technology Development Fund, MSAR, under Grant No. 066/2013/A2, and the Fundamental Research Funds for the Central Universities under Grant No. JB142001-15, Xidian University.

## REFERENCES

- [1] E. Teruel, P. Chrzastowski-Wachtel, J.M. Colom and M. Silva. "On Weighted T-Systems," *Appl. Theory Petri Nets*, vol. 616, pp. 348–367, 1992.
- [2] J. Campos, G. Chiola, and M. Silva. "Ergodicity and throughput bounds of Petri nets with unique consistent firing count vector," *IEEE Trans. Software Eng.*, vol. 17, no. 2, pp. 117–125, 1991.
- [3] A. Munier. Régime asymptotique optimal d'un graphe d'événements temporisé généralisé : Application à un problème d'assemblage. *APII*, vol. 27, pp. 487–513, 1993.
- [4] M. Nakamura and M. Silva. "Cycle time computation in deterministically timed weighted marked graphs," in *Proc. 7th IEEE Int. Conf. Emerg. Technol. Factory Autom.*, vol. 2, pp. 1037–1046, 1999.
- [5] A. Giua, A. Piccaluga and C. Seatzu. "Firing rate optimization of cyclic timed event graph," *Automatica*, vol. 38, no. 1, pp. 91–103, 2002.
- [6] A.S. Oshaba, E.S. Ali and S.M. Abd-Elazim, "ACO Based Speed Control of SRM Fed by Photovoltaic System," *Int. J. of Electrical Power and Energy Systems*, Vol. 67, pp. 529-536, May 2015.
- [7] A.S. Oshaba, E.S. Ali and S.M. Abd-Elazim, "MPPT Control Design of PV System Supplied SRM Using BAT Search Algorithm," *Sustainable Energy, Grids and Networks*, Vol. 2C, pp. 51-60, 2015.
- [8] Z. He, Z.W. Li, and A. Giua. "Cycle time optimization of deterministic timed weighted marked graphs," in *Proc. 11th IEEE Int. Conf. Autom. Sci. Eng.*, pp. 274–279, Gothenburg, Sweden, August 2015.
- [9] N. Sauer. "Marking optimization of weighted marked graphs," *Discrete Event Dyn. Syst.*, vol. 13, no. 3, pp. 245–262, 2003.
- [10] Z. He, Z.W. Li, and A. Giua. "Marking optimization of deterministic timed weighted marked graphs," in *Proc. 10th IEEE Int. Conf. Autom. Sci. Eng.*, pp. 413–418, Taipei, Taiwan, August 2014.
- [11] Z. He, Z.W. Li, and A. Giua. "Optimization of deterministic timed weighted marked graphs," *IEEE Trans. Autom. Sci. Eng.*, 2016. (To appear).
- [12] M. Benazouz, O. Marchetti, A. Munier, and U. Pascal. "New approach for minimizing buffer capacities with throughput constraint for embedded system design," in *Proc. IEEE Int. Conf. Comp. Syst. Appl.*, pp. 1–8, 2010.
- [13] J.H. van Schuppen, M. Silva, and C. Seatzu. Control of discrete-event systems-Automata and Petri Net perspectives, *Lect. Notes Control Inf. Sci.*, Springer, Vol. 433, pp. 319-340, 2012.
- [14] J. Campos, G. Chiola, J.M. Colom and M. Silva. "Properties and performance bounds for timed marked graphs," *IEEE Trans. Fundam. Theory Appl.*, vol. 39, no. 5, pp. 386–401, 1992.
- [15] T. Murata. "Petri nets: Properties, analysis and applications," in *Proc. IEEE*, vol. 77, no. 4, pp. 541-580, 1989.
- [16] A. Benabid-Najjar, C. Hanen, O. Marchetti, and A. Munier. "Periodic schedules for bounded timed weighted event graphs," *IEEE Trans. Autom. Control*, vol. 57, no. 5, pp. 1222–1232, 2012.
- [17] J.V. Millo and R. De Simone. "Periodic scheduling of marked graphs using balanced binary words," *Theor. Comput. Sci.*, vol. 458, pp. 113–130, 2012.
- [18] D.Y. Chao, M.C. Zhou, and D.T. Wang. "Multiple weighted marked graphs," in *Proc. 12th IFAC Triennial World Congress*, vol. 4, pp. 259–263, 1993.
- [19] HYPENS: <http://www.diee.unica.it/automatica/hypens/>