



HAL
open science

Control law for timed marked graph constrained by marking exclusion constraint OR/AND

Maen Atli, Zied Achour, Alexandre Sava, Kondo Hloindo Adjallah

► **To cite this version:**

Maen Atli, Zied Achour, Alexandre Sava, Kondo Hloindo Adjallah. Control law for timed marked graph constrained by marking exclusion constraint OR/AND. Buletinul Institutului Politehnic din Lasi. Sectia IV, Automatica si Calculatoare, 2011, LVII (LXI) (4), pp.75-92. hal-03089833

HAL Id: hal-03089833

<https://hal.univ-lorraine.fr/hal-03089833>

Submitted on 28 Dec 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

BULETINUL INSTITUTULUI POLITEHNIC DIN IAȘI
Publicat de
Universitatea Tehnică „Gheorghe Asachi” din Iași
Tomul LVII (LXI), Fasc. 4, 2011
Secția
AUTOMATICĂ și CALCULATOARE

**CONTROL LAW FOR TIMED MARKED GRAPH
CONSTRAINED BY MARKING EXCLUSION
CONSTRAINT OR/AND**

BY

**MAEN ATLI*, ZIED ACHOUR, ALEXANDRE SAVA
and KONDO H. ADJALLAH**

LGIPM, INRIA COSTEAM
Ecole National d'Ingénieur de Metz,
Metz, France

Received: August 3, 2011

Accepted for publication: September 16, 2011

Abstract. We have already presented a contribution for synthesizing an approach of supervisory control for Discrete Event Systems (DES) modeled by Timed Place Marked Graphs subject to Marking Exclusion Constraint. This paper is a continuation work in this area for building the control law for Discrete Event Systems (DES), where the time is taken in consideration. It solves a forbidden state problems characterized by one of the both types of Marking Exclusion Constraint: MEC-Or and MEC-And. A computationally efficient technique to build the supervisor is proposed, which we take in consideration in the uncontrollable and/or unobservable nature of some events. From the initial state (as known marking), we build time-tables that show us the timeline of tokens distribution over the system. These time-tables help us to build a Control Law Table (CLT), by analyzing this table the supervisor may determine the moment of its intervention.

Key words: component; Discret Event System; control law; Marking Exclusion Constraint; Petri nets; Timed Marked Graph.

2000 Mathematics Subject Classification: 34H05.

*Corresponding authors; *e-mail:* maen.atli@inria.fr

1. Introduction

Our research is concerned on the control synthesis of manufacturing system. A manufacturing system can be modelled by a discrete event system DES. Apart from planning (where people work with ratios of products fabricated per week or per day), any modelling has to be based on the concepts of event and activities. An event corresponds to a state change. An activity is a black-box encapsulating what is occurring between two events. When using Petri Nets, events are associated with transitions, and activities with places. However Petri Net is not the unique tool handling events and activities. For example, algebras and formal models (Ramadge & Wonham, 1987a) are under investigation. This paper deals with the control synthesis problem for Discrete Event System modelled by a class of Timed Petri Net called Timed Marked Graph subject to Marking Exclusion Constraint MEC. The main contribution is to synthesize a supervisor that acts in closed loop with the system, which the whole system/supervisor avoid the forbidden states modelled by MEC. In the context of this synthesis, we propose a technique to build a controller that enforces MEC on partially controllable and/or observable marked graph. The time makes our control policies more permissive by allowing the system to reach more states (marking) which were considered as dangerous marking whether we do not take the time in account. In the sequel, we recall briefly the approaches which are most related to our research work. According to a chronology, we are going firstly to pass on the researches covering the control synthesis for Discrete Event Systems, and the Petri Net models used for this purpose.

1.1. Control Synthesis and Linear Specifications

A Discrete Event System *DES* is a dynamic system that involves in accordance with the abrupt occurrence of physical events at possibly unknown time intervals. Given a discrete event model of the plant and a specification of the desired behaviour, the aim of the control synthesis is to build the appropriate controller that acts in closed-loop with the plant in order to guarantee a desired behaviour. The existence and the characterization of the maximally permissive control were formalized in (Ramadge & Wonham, 1987a; Ramadge & Wonham, 1987b). Finite-state machines and formal languages are the modelling framework considered in these studies and the control action is achieved by event feedback. The drawback of automata models is their lack of structure, which makes difficult the representation of behaviours like synchronization, parallelism and resource sharing. However, to overcome these disadvantages, control synthesis methods based on Petri nets were proposed by (Krogh, 1987; Ishikawa & Hiraishi, 1987) to exploit the modelling power of PN and the rich mathematical results characterize them. Over the last decades *Petri Net* has gained increased usage and acceptance as basic model of DES (Murata, 1989).

Petri Net provides an intuitive graphical representation for DES behaviour. Modelling facilities together with interesting analytical properties made this tool appropriate to controller synthesis for DES. The subclass of Petri nets which each place in the net has exactly one incoming arc and exactly one outgoing arc, is known Marked Graph.

In (Holloway & Krogh, 1990), the authors present an efficient method for synthesizing feedback control logic to solve forbidden-state problems for DES modelled by safe controlled marked graphs. The distributed representation of the DES state in terms of the controlled marked graph marking allows an efficient specification of the forbidden states in terms of individual place markings. The authors synthesize a state feedback control which guarantees that the forbidden states are avoided while allowing a maximal set of state transitions to occur. This approach is extended in (Holloway & Krogh, 1992), by taking into account the liveness property for the closed loop system under the maximally permissive feedback control policies. Using the concept of synchronic distances in Petri nets, they provide sufficient conditions to build a maximally permissive controller. The forbidden state problem has been widely addressed and covers. An interesting class of specifications called generalized mutual exclusion constraints *GMEC* is presented in (Giua *et al.* 1992). These are a natural way of expressing the concurrent use of a finite number of resources, shared among different processes. A control synthesis technique to enforce *GMEC* specifications based on monitor places is also proposed. They present two PN structure capable of enforcing these constraints (Giua *et al.* 1993). The first is a monitor-based solution and the second one is a supervisory-based solution. They deal with the problem of enforcing generalized mutual exclusion constraints on marked graphs with control safe places. In the same way, the paper (Li & Wonham, 1994), propose an algorithm to calculate the optimal solution for nets with acyclic uncontrollable subnets. The controller has to solve online at each step linear integer program. While (Yamalidou *et al.*, 1996) propose a method to build a Petri net controller based on place invariants to enforce linear constraints on the Petri net marking in case of observable and controllable transitions.

In (Ghaffari *et al.*, 2003), the authors develop the significations used in (Giua *et al.*, 1993), and they extended the work of (Holloway & Krogh, 1990; Holloway & Krogh, 1992), to construct controller logic for forbidden-state problems based on marked graphs. They deal with the forbidden-state problem of general marked graphs with uncontrollable transitions. The models need not to be safe or cyclic as for the references (Holloway & Krogh, 1990; Holloway & Krogh, 1992). Control requirements are expressed as the conjunction of *Generalized Mutual exclusion constraints GMEC*. They (Ghaffari *et al.*, 2002), present two supervisory control problems of plant Petri nets models, forbidden state and forbidden state transition. Based on the theory of regions, algebraic characterizations of control places are provided for the both problems.

Another basic specification was presented in (Atli *et al.*, 2010a; Atli *et al.*, 2010b), this basic specification is called Marking Exclusion Constraint MEC. They define MEC as a basic constraint item applied on place markings. Thus a MEC focuses on the marking of one place. For instant a MEC constraint may request the marking of a place p to be different from a given value k . Markings constraints of several places may be modelled by using MEC as a basic item. In this case, they propose three kinds MEC-Or, MEC-And and M-MEC. They define MEC-Or a constraint by a union of MECs, and MEC-And a constraint by a conjunction of MECs. We also extend MEC to the case which a weighted sum of a given Petri net marking must be different from a given number of tokens k , we call it M-MEC. In the same context, they built a control synthesis to restrict a *Marked graph* to respect the specification modelled by a MEC or one of kinds of MEC.

1.2. Control Synthesis of Timed Petri Net

In order to study performance aspects of Petri Nets models, we have to include the duration of activities into the model. The two main extensions of Petri net with time are Time Petri Nets (Merlin & Faber, 1976) and Timed Petri Nets (Ramachandani, 1974). While the transition may fire within a time interval in for Time Petri Net, the transition in Timed Petri Net fires as soon as possible, according a delay associated to places (Timed Transition Petri Net TTPN) or transition (Timed Placer Petri Net TPPN). The two subclasses TTPN and TPPN are expressively equivalent. The firing times may be deterministic, random variables described by a probability distribution function, fuzzy variable described by a fuzzy interval. The authors in (Achour *et al.*, 2004) extended the work of (Ghaffari *et al.*, 2003), by considering unobservable transitions. They proposed approach based on an observer. The same authors in (Achour *et al.*, 2008) have extended their work to take in account the time information. They discuss the influence of observability and time constraints on the controller synthesis and they show that the control law designed using time information is more permissive. The problem of observability has no influence in this case, since even we can not observe the firing, time information help us to deduce the their firing instant. In this paper, we develop our work in (Atli *et al.*, 2010a; Atli *et al.*, 2010b), to take in account the time information as (Achour *et al.*, 2008).

1.3. Summary of Content

Following the algebraic model of Petri nets, a brief summary of the basic linear specifications *Marking Exclusion Constraint MEC* and *Generalized Mutual Exclusion Constraints GMEC* are reviewed in section 2. The supervisory control review and our primary contribution is presented of this paper appears in section 3, through 4 study cases. An Example is introduced in

section 4. Finally, we summarized our contributions in this paper and the future research regarding these contributions.

2. Background

2.1. Modelling with Petri Net and Marked Graph

A Petri Net PN is a graphical and mathematical modelling tool particularly adapted for modelling and analysing of discrete event systems. A complete presentation can be found in (Murata, 1989). Similarly, the set of input (output) places of a transition t is denoted ${}^{\circ}t$ (t°). A Petri net is said to be ordinary if the weight associated to each arc is equal to 1. A Petri net marking is a mapping $M : P \rightarrow N$ that assigns to each place of a Petri Net a non-negative integer number of tokens, where the tokens are represented by black dots (Murata, 1989). The following notations will be used throughout this paper:

P	Set of m places
T	Set of n transitions
Pre	Pre-incidence function that defines weighted arcs from places to transitions.
$Post$	Post-incidence function that defines weighted arcs from transitions to places.
${}^{\circ}p$ (respectively p°)	Set of input (respectively, output) transitions of place p .
${}^{\circ}t$ (respectively t°)	Set of input (respectively, output) places of transition t .
$m(p)$	Marking of one place p .
M	Petri net marking (M_0 is the initial marking) represent the markings courant for all places.
$M[t >$	Transition t is friable at marking M .
$M[t > M'$	Firing transition t at marking M leads to marking M' .
$\sigma(t)$	Number of firing's times of transition t .
$R(N, M_0)$	Set of reachable markings from M_0 .
$R^u(N, M_0)$ (respectively $R^{u0}(N, M_0)$)	The set of reachable markings from a marking M_0 by firing only uncontrollable transitions (respectively unobservable transitions).

A *Marked Graph* is an ordinary PN in which all weights associated to arcs are 1 and every place has exactly one input transition and one output transition, *i.e.*, $|{}^{\circ}p| = |p^{\circ}| = 1$ for all $p \in P$. The marking of place $m(p)$ in

marked graph is changed by firing the input and output transition (Ghaffari *et al.*, 2003), as the following equation:

$$M(p) = M_0(p) - \sigma({}^{\circ}p) + \sigma(p^{\circ}) \quad (1)$$

Firing transition ${}^{\circ}p$ or p° at marking M_0 leads to marking M .

2.2. Marking Exclusion Constraint MEC

A *Marking Exclusion Constraint* (Atli *et al.*, 2010a; Atli *et al.*, 2010b) is defined as a condition that forbids the existence of a given number of tokens in one critical place. Let a Petri net with reachable marking $R(N, M_0)$. A single *Marking Exclusion Constraint* MEC defines a set of legal markings described as following:

$$M_L = \{M \in R(N, M_0), \quad q \in Q_{mc} / M(q) \neq k\} \quad (2)$$

where q is called *critical place* where there is a MEC constraint (ω, k) such $\omega(q) \neq 0$.

On distinguish two principal kinds of this specification for more than one critical place, MEC-Or and MEC-And.

Definition: Let a Petri net with reachable marking $R(N, M_0)$. A *Marking Exclusion Constraint* of type MEC-Or defines a set of legal markings described as following:

$$M_L = \left\{ M \in R(N, M_0) / \bigvee_{q_i \in Q_{mc}} M(q_i) \neq k_i \right\}$$

Definition: Let a Petri net with reachable marking $R(N, M_0)$. A *Marking Exclusion Constraint* of type And MEC-And defines a set of legal markings described as following:

$$M_L = \left\{ M \in R(N, M_0) / \bigwedge_{q_i \in Q_{mc}} M(q_i) \neq k_i \right\}$$

2.3. Timed Marked Graph TMG

Dealing with time in a MG is accomplished by assigning time elapses to transitions or places. The time can be assigned to both nodes but the analysis of TMG models is simpler when the time is attached to one kind of node. In this

paper, we assign the time to transition; the resulted Marked Graph in this case called Timed Transition Marked Graph TTMG. TTMG is defined as (N, H, θ) where N is a normal Petri Net and H is a set of delays (durations); $H = \{hi\}$ for $i = 1, \dots, m$, each $hi \in \mathbb{N}$ (Natural number). Each hi represents the time that the arriving tokens to place p_i stay unavailable. θ_i specifies the firing instant of transition t_i . A token becomes available after hi time units from the instant of its arrival to p_i . Only the available tokens enable the transitions firing. In other words, a token in a place p_i , may have two possible states: available or unavailable. In any time delay h , the present marking $M = Ma + Mu$, such that Ma comprises available tokens and Mu comprises unavailable tokens. A transition is enabled if $Ma(p_i) \geq Pre(p_i, t)$, where Ma encompasses the available tokens. When a token arrives in a timed place p_i , it must remain there for a time delay hi , before becoming available.

3. Supervisor Synthesis

3.1. Controllability Problem

A forbidden marking is a marking which violates at least one MEC constraint. The set of forbidden markings is

$$M_f = \{M \in R(N, M_0) / \omega M(q_i) = k_i\} \quad (3)$$

The transition set is partitioned into two disjoint subsets: $T = T_c \cup T_u$ & $T_c \cap T_u = \emptyset$, where T_c is the set of controllable transitions and T_u is the set of uncontrollable transitions. A direct consequence of the presence of uncontrollable transitions is the existence of dangerous markings. A marking is called dangerous if the starting from this marking, the system may evolve uncontrollably to a forbidden marking. The set of dangerous markings is:

$$M_D = \{M \in R(N, M_0) / M[U_{zero}, \sigma > M', M' \in M_f]\} \quad (4)$$

The set of admissible markings, denoted M_a , is defined by the markings which are neither dangerous nor forbidden. The set M_a is the maximal set of states under which there exists a control policy which will prevent the system from reaching a forbidden state $M' \in M_f$.

An *influence path* $\pi(q)$ of a critical place is an elementary path joining t to q such that $t \in T_c$ is a controllable transition and all other transitions in the path are uncontrollable. The controllable transition t of the influence path is

called an *influence transition*. The control synthesis approach that we propose relies on the concept of binary control of controllable transition given in (Holloway & Krogh, 1992). A binary control is a mapping $U : T_c \rightarrow \{0,1\}$, which assigns a binary value count to each controllable transition. A controllable transition t is allowed fire if $U_{(t)} = 1$.

Let $D(q, M)$ denote the maximum number of times (${}^\circ q$) may fire without firing any influence transition in $Z(q)$, the influence zone of q , *i.e.* under U_{zero} .

$$D(q, M) = \min_{t \in T_c, {}^\circ q} td(M, t) \quad (5)$$

where $td(M, t \in T_c, {}^\circ q)$ be the token distance (the sum of tokens) between an influence transition t' and ${}^\circ q$ (the input transition of a critical place q) at marking M . The influence zone $Z(q)$ of a critical place q is the subnet containing all nodes for which there exists a directed path from an influence transition to q . In this paper, we denote the set of influence transitions by $\Gamma(q)$, it is formally defined by:

$$\Gamma_{(q)} = \{t \in (Z(q) \cap T^c)\} \quad (6)$$

Let $Y(q, M)$ (Ghaffari *et al.*, 2003) be the marking having the maximal number of tokens in q which may be reached uncontrollably from marking M . Then $Y(q, M)$ is updated according to transitions firing.

$$Y(q, M) = M_0(q) + D(q, M) + \sigma({}^\circ q) - \sigma(q^\circ) \quad (7)$$

where $\sigma({}^\circ q)$ (respectively $\sigma(q^\circ)$) defines the number of times of transition (${}^\circ q$) (respectively (q°)) is fired.

3.2. Supervisory Synthesis for Timed Marked Graph

In this section, we present the idea discussed in (Achour *et al.*, 2008; Atli *et al.*, 2011). The supervisor must evaluate the worst case noted by $Y(q, M, \theta)$ which represents the number of tokens that place q could have after a global time θ . by starting from a known marking M and firing only uncontrollable transitions. The marking of q is influenced by the marking of the path $\pi(q)$ such as $\pi(q) = t_n^c p_1 t_2 p_2 \dots p_n t_{n+1} q$, where t_c represents the only controllable transition (influence transition) in the path π .

When considering the time information, tokens can no longer stay indefinitely in places. A token cannot get to a given place before a certain period of time and has to live it before another given period of time. Consequently, the behaviour of the system is restricted regarding the untimed case and some potentially dangerous situations do not need to be considered anymore when building the controller. Thus, using time information enables to build more permissive control laws.

They define:

$$Y(q, M, \theta) = Y^+(q, M, \theta) - Y^-(q, M, \theta) \quad (8)$$

where $Y^+(q, M, \theta)$ (*resp.* $Y^-(q, M, \theta)$) represents the number of tokens which arrive in the place q in a period θ (*resp.* represents the number of tokens which leave the place q in a period θ). Thus

$$Y^+(q, M, \theta) = M(q) + M(p_n)_{if \theta \geq h(\pi_n)} + \dots + M(p_1)_{if \theta \geq h(\pi_1)} + \sigma(t_1, \theta - e - h_{(\pi_1)}) \quad (9)$$

$$Y^-(q, M, \theta) = M(q)_{if \theta \geq h(q)} + M(p_n)_{if \theta \geq h(\pi_n) + h(q)} + \dots + M(p_1)_{if \theta \geq h(\pi_1) + h(q)} + \sigma(t_1, \theta - e - (h_{(\pi_1)} + h_{(q)})) \quad (10)$$

where π_i represent the paths $\pi(pi \rightarrow q)$, $\sigma(t_i, \tau)$ is the firing times of t_i during the period τ , and e is the moment of influence transition firing.

The supervisor will not tamper with the evolution of system behaviour whether this behaviour respect always (8), otherwise the supervisor will tamper with the controllable transition firing (either influence transition or output transition). The control law for this case is described as following:

$$U_{(t \in Tc)} = \begin{cases} 1, & \text{if (8)} \\ 0, & \text{otherwise} \end{cases} \quad (11)$$

In the sequel of this section, we recall the significations and rules (Atli *et al.*, 2011), which are indispensable in this paper:

Supervisor work rule 1: The task of supervisor is to observe the work of system and to control the firing of controllable transition (to fire or to not fire) *i.e.* the supervisor prevents only the firing when it is necessary. From the marking M , the supervisor builds the CLT for the period $[0, h(\pi) + h(q)]$.

The supervisor prevents the output controllable transition q° to fire:

1. At the moment θ_i , if the logic value of control law is false at θ_i .
2. At the moment θ_j , if at least one of the three following conditions is not validated:
 - There is an available token or more in the critical place at θ_i .
 - The logic value of control law is true at θ_j .
 - $Y(q, M, \theta_j) < k$, where $\theta_i > \theta_j$, $\theta_j \in [0, \theta_i]$ and θ_j : the moment before θ_i when the value of control law is false.

Supervisor work rule 2: the supervisor allows the influence transition to fire at $\theta = e$, if all values control law in CLT over $[e, e + h_{(\pi_1)} + h_{(q)}]$ are true.

Supervisor work rule 3: if we want to fire the influence transition at $\theta = e$, and the two conditions are validates:

1. There is at least a value false in CLT over $[e, e + h_{(\pi_1)} + h_{(q)}]$ at θ_i .
2. $Y(q, M, \theta_i - 1) > Y(q, M, \theta_i)$.

The supervisor has two choices:

- Either, the supervisor prevents the influence transition to fire at $\theta = e$.
- Or, the supervisor permits the influence transition to fire at $\theta = e$, provided that it prevents the controllable output transition to fire at θ_i .

For more information about these three rules, you may review the paper (Atli *et al.*, 2011).

3.3. Control Law Tables

The following is an algorithm to compute the control law tables for building the supervisory control for the discrete event systems DESs. We distinguish two, the first one is without firing any controllable transition (at the initial marking), and the second one whether the system is going to fire one controllable transition.

A. At the *initial marking* M_0 :

1. For each critical place: build a *Control Law Table* CLT as shown in Table 1 over $[0, h_{(\pi_1)} + h_{(q)}]$. Use the *supervisor work rule 1* to compute the control law value (Tables 1, 2 and 3).

Table 1

Calculating $Y^+(q_i, M_0, \theta)$

θ	$M(q_i)$	$Y^+(q_i, M_0, \theta)$
0			
...			
$h_{(\pi_1)} + h_{(q)}$			

Table 2

Calculating $Y^-(q_i, M_0, \theta)$

θ	$M(q_i)_{if \theta \geq h(q)}$	$Y^-(q_i, M_0, \theta)$
0			
...			
$h_{(\pi_1)} + h_{(q)}$			

Table 3

Control Law Table CLT

θ	$Y^+(q_i, M_0, \theta)$	$Y^-(q_i, M_0, \theta)$	$Y(q_i, M_0, \theta)$	U_i
0				
...				
$h_{(\pi_1)} + h_{(q)}$				

2. Build the *Control Law Table General CLTG* from as shown in Table 2 for MEC-Or specification, and as shown in Table 3 for MEC-And specification (Tables 4 and 5).

Table 4

Control Law Table CLTG for MEC-Or Specification

θ	U_1	U_{mc}	$U = U_1 \vee \dots \vee U_{mc}$
0				
...				
$h_{(\pi_1)} + h_{(q)}$				

Table 5

Control Law Table CLTG for MEC-And Specification

θ	U_1	U_{mc}	$U = U_1 \wedge \dots \wedge U_{mc}$
0				
...				
$h_{(\pi_1)} + h_{(q)}$				

3. All the controllable transitions are fireable whether the control law value (last column in CLTG) is true (one).

4. If the control law value takes zero at the moment θ_i :

a) The specification is MEC-Or: it is sufficient to suspend one of the controllable output transition ($q_i^\circ \in T_c$) ;

b) The specification is MEC-And: All the controllable transitions are suspended at the moment when the value is false (zero).

B. Firing $t \in T_c$ at the moment e :

1. For each critical place: build a *Control Law Table* CLT as shown in table 1 over $[e, e + h_{(\pi_1)} + h_{(q)}]$.

2. Build the *Control Law Table General* CLTG.

3. All the controllable transitions are fireable whether all the value in CLTG (last column in CLTG) is true (one) (rule 2).

4. If the control law value takes zero at the moment θ_i :

a) The specification is MEC-Or: it is sufficient to check the rule 3 for one critical place.

b) The specification is MEC-And: it must check the rule 3 for all critical places.

3.4. Illustrative Example

The production line in our example (Fig. 1) consists of two cellules, the first one is consisted of a tank of liquid with unlimited capacity, and a pump that fills a litre for each once, and the second one is a conveyor to transport bottles with a capacity of one litre. The timed marked graph model for this example is presented in Fig. 2, and the Table 6 provides us the description of places and transitions, as well as the associated time delays. The initial state of this example is: $M_0 = [3 \ 1 \ 0 \ 3 \ 1 \ 0 \ 0]$.

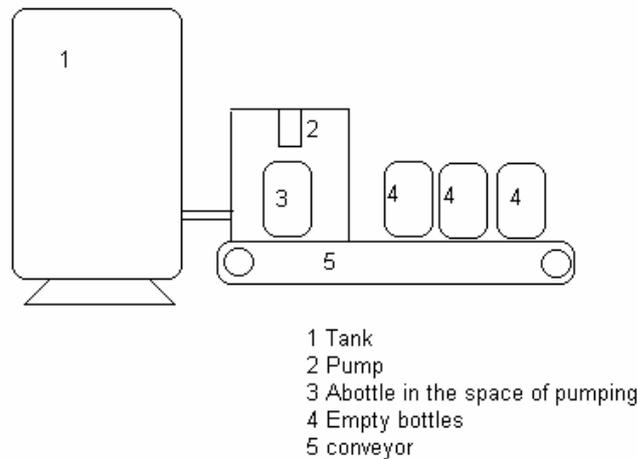


Fig. 1 – Machine for filling bottles with liquid.

Table 6
Interpretation and Delays of Places and Transitions

Place	Interpretation	Delay
p_1	empty bottle (bottles) is (are) on the conveyer	Untimed
p_2	The empty bottle is exactly under the valve of pump (in the space of pumping).	2 time units
p_3	Capacity the space of pumping.	Untimed
p_4	Tank	Untimed
p_5	Liquid of one litre is ready to be pumped.	1time unit
p_6	Capacity of pump.	Untimed
p_7	Pumping the liquid	2 time unit
<i>Transition</i>		
t_1	Arriving a new bottle to the conveyer.	Untimed
t_2	Arriving a bottle to the space of pumping.	Untimed
t_3	Exiting the bottle from the space of pumping.	Untimed
t_4	Fill the tank with liquid.	Untimed
t_5	Empty one litre from the tank.	Untimed
t_6	The pump is ready to work	Untimed
t_7	Finish the work of pump.	Untimed

Through the evolution of the system, it is necessary to confirm that the pump won't work whether there is no empty bottle in the space of pumping, on the other side; it is possible to have a bottle in this area although the pump has no liquid to fill. In other words, all markings having one token in p_7 and no token in p_2 are forbidden. As well as, the set of markings where there is a token in p_2 and no token in p_7 *i.e.*, the set of critical places is: $Q_{mc} = \{p_2, p_7\}$, the set of forbidden markings is: $M_f = \{M \in R(N, M_0) / M(p_2) \neq 1 \wedge M(p_7) \neq 0\}$.

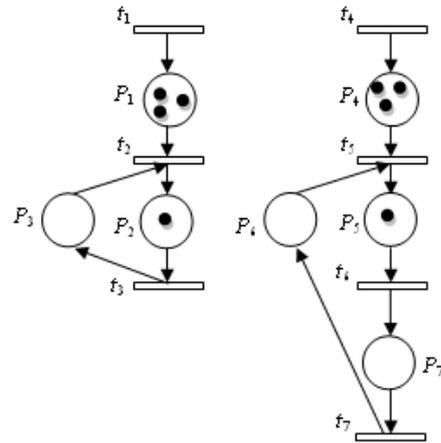


Fig. 2 – Timed Marked Graph model for example shown in Fig. 1.

Tables 7,...,12 present the calculations and their results.

Table 7

Calculating $Y^+(p_2, M_0, \theta)$

θ	$M(p_2)$	$Y^+(p_2, M_0, \theta)$
0	1	1
1	1	1
2	1	1

Table 8

Calculating $Y^-(p_2, M_0, \theta)$

θ	$M(p_2)_{if \theta \geq 2}$	$Y^-(p_2, M_0, \theta)$
0	0	0
1	0	0
2	1	1

Table 9

Control Law Table CLT for p_2

θ	$Y^+(p_2, M_0, \theta)$	$Y^-(p_2, M_0, \theta)$	$Y(p_2, M_0, \theta)$	U_1
0	1	0	1	1
1	1	0	1	1
2	1	1	0	0

Table 10Calculating $Y^+(p_7, M_0, \theta)$

θ	$M(p_5)_{if \theta \geq 1}$	$M(p_7)$	$Y^+(p_7, M_0, \theta)$
0	0	0	0
1	1	0	1
2	1	0	1
3	1	0	1

Table 11Calculating $Y^-(p_7, M_0, \theta)$

θ	$M(p_5)_{if \theta \geq 3}$	$M(p_7)_{if \theta \geq 2}$	$Y^-(p_7, M_0, \theta)$
0	0	0	0
1	0	0	0
2	0	0	0
3	1	0	1

Table 12Control Law Table CLT for p_7

θ	$Y^+(p_7, M_0, \theta)$	$Y^-(p_7, M_0, \theta)$	$Y(p_7, M_0, \theta)$	U_2
0	0	0	0	1
1	1	0	1	0
2	1	0	1	0
3	1	1	0	1

Table 13Control Law Table CLTG for
MEC-Or Specification

θ	U_1	U_2	$U = U_1 \vee U_2$
0	1	1	1
1	1	0	1
2	0	0	0
3	0	1	1

From Table 13, we may see that our specification won't be respected at $\theta = 2$, the pump will let a part of one litre of liquid to go out without a bottle under the valve of this pump, therefore the supervisor must take many procedures at this moment. It is sufficient that the value of U_1 become true, therefore the supervisor will prevent the transition t_3 to fire at $\theta = 2$, and let it to fire at the next step.

4. Conclusions

This paper is considered as continuation of our work presented in (Atli *et al.*, 2011). We present a control synthesis for DES modelled by *Timed Place Marked Graph* TPMG subject to a specification modelled by one of two kind (Or/And) of a *Marking Exclusion Constraint* MEC. We deal with uncontrollable and/or unobservable transition, where there is no deference in this case between the two kinds of transitions because the time information helps us to determine the marking current. In our control policy, we depend on the marking initial as known marking to build time tables that show us the timeline of marking distribution over the system. These time-tables help to build a Control law Table CLT, through analysis the information in this table the supervisor may determine the moment of its intervention. The goal of supervisor in this paper is to observe the system behaviour and to tamper with this behaviour when it is necessary. We present three *Supervisor work rules*, the first one to describe the conditions to control the output transition, and two rules for the influence transitions. Future work is to develop the control synthesis to include Time Marked Graph (the node is attached with time interval).

REFERENCES

- Achour Z., Rezg N., Xie X., *Supervisory Control of Partially Observable Marked Graphs*. IEEE Transaction on Automatic Control, Vol. **49**, 11, 2004.
- Achour Z., Sava A., Rezg N., *On the Supervisory Control of Marked Graph*. IEEE Singapore, 2122–2126, 2008.
- Atli M., Achour Z., Sava A., Adjallah K.H., Rezg N., *Supervisory Control of Timed-Place Marked Graph based on Marking Exclusion Constraint*. IEEE, CCCA, 2011.
- Atli M., Sava A., Achour Z., Rezg N., *Control Synthesis for Partially Controlled Marked Graph Subject to Marking Exclusion Constraints*. IFAC, 12th LSS Symposium: Large Scale Systems: Theory and Applications, 2010.
- Atli M., Sava A., Achour Z., Rezg N., *Supervisory Control of Partially Observable Marked Graph based on Marking Exclusion Constraint*. IFAC, MCPL, 2010.
- Ghaffari A., Rezg N., Xie X., *Algebraic and Geometric Characterization of Petri Net Controllers Using the Theory of Regions*. Discrete Event Systems, Proceedings Sixth International Workshop, 219–224, 2002.
- Ghaffari A., Rezg N., Xie X., *Feedback Control Logic for Forbidden-State Problems of Marked Graphs: Application to a Real Manufacturing System*. IEEE Trans. Automat. Contr., Vol. **48**, 18–29, 2003.
- Giua A., DiCesare F., Silya M., *Generalized Mutual Exclusion Constraints on Nets with Uncontrollable Transitions*. in Proc. IEEE Int. Conf. on Systems, Man and Cybernetics, Chicago, USA, 974–799, 1992.

- Giua A., DiCesare F., Silya M., *Petri Net Supervisors for Generalized Mutual Exclusion Constraints*. In Proc. 12th IFAC World Congress, Sidney, Australia, Vol. 1, 267–270, 1993.
- Holloway L.E., Krogh B., *On Closed-Loop Liveness of Discrete-Event Systems Under Maximally Permissive Control*. IEEE Transactions on Automatic Control, Vol. 37, 5, 692–697, 1992.
- Holloway L.E., Krogh B., *Synthesis of Feedback Control Logic for a Class of Controlled Petri Nets*. IEEE Transactions on Automatic Control, Vol. 35, 5, 514–523, 1990.
- Ishikawa A., Hiraishi K., *Analysis and Control of Discrete Event Systems Represented by Petri Nets*. In P. VARAIYA and A.B. KURZHANSKI (Eds.), *Discrete Event Systems: Models and Applications*, Berlin: Springer, 115–134, 1987.
- Krogh B.H., *Controlled Petri Net and Maximally Permissive Feedback Logic*. Proceedings of 25th annual Allerton Conference, University of Illinois, Urbana, IL, USA, 317–326, 1987.
- Li Y., Wonham W.M., *Control of Vector Discrete Event Systems II-Controller Synthesis*. IEEE trans. Automat. Contr., Vol. 39, 512–531, 1994.
- Merlin P., Faber D.J., *Recoverability of Communication Protocols*. IEEE Trans. Commun. COM, 24, 9, 1976.
- Moody J.O., Antsaklis P.J., *PN Supervisors for DES with Uncontrollable and Unobservable Transitions*. Technical Report ISIS-99-004, ISIS Group at the University of Notre Dame, 1999.
- Murata T., *Petri Nets: Properties, Analysis and Applications*. Proceedings of the IEEE, Vol. 77, 4, 541–580, 1989.
- Ramachandani C., *Analysis of Asynchronous Concurrent Systems by Timed Petri Net*. Massachusetts Inst. Tech. Project MAC, Tech. Rep., 120, 1974.
- Ramadge P., Wonham M., *Modular Feedback Logic for Discrete Event Systems*. SIAM J., Control Optim, Vol. 25, 5, 1202–1218, 1987b.
- Ramadge P., Wonham M., *Supervisory Control of a Class of Discrete Event Processes*, SIAM J., Control Optim, Vol. 25, 1, 206–230, 1987a.
- Ramadge P., Wonham M., *The Control of Discrete Event Systems*. Proceedings of the IEEE. Vol. 77, 1, 81–98, 1989.
- Yamalidou K., Moody J., Lemmon M., Antsaklis P., *Feedback Control of PNs Based on Place Invariants*. Automatica, Vol. 32, 1, 15–28, 1996.

CONTROLUL GRAFURILOR MARCATE TEMPORIZATE PE POZIȚII CU
MARCAJ SUPUS LA RESTRICȚII DE TIP EXCLUDERE OR/AND

(Rezumat)

Ca o continuare a unor contribuții anterioare privind sinteza unor module de supervizare și control a unor sisteme cu evenimente discrete, modelate cu rețele Petri de tip *Timed Place Marked Graphs* în care sunt marcate constrângerile de excludere, lucrarea tratează problema stabilirii legii de control pentru un sistem cu evenimente

discrete la care trebuie să se ia în considerare și influența timpului. Se propune o rezolvare a unei probleme dificile în care intervin cele două tipuri de constrângeri la marcarea excluziunii: MEC-OR și MEC-AND. De asemenea, se propune o tehnică eficientă de implementare prin program a metodei de construire a unui modul supervisor pentru evenimente necontrolabile și/sau neobservabile. Plecând din starea inițială specificată prin marcaj, se construiesc tabele de timp care arată evoluția distribuției marcajelor în toată rețeaua Petri. Aceste tabele de timp permit construirea tabelară a legii ce control, pe baza căreia se poate stabili momentul în care supervisorul trebuie să intervină.