

From Forbidden States to Linear Constraints

M. Zareiee, A. Dideban and P. Nazemzadeh

Abstract—This paper deals with the problem of constructing constraints in non safe Petri Nets and then reducing the number of the constructed constraints. In a system, assigning some linear constraints to forbidden states is possible. Enforcing these constraints on the system prevents it from entering these states. But there is no a systematic method for assigning constraints to forbidden states in non safe Petri Nets. In this paper a useful method is proposed for constructing constraints in non safe Petri Nets. But when the number of these constraints is large enforcing them on the system may complicate the Petri Net model. So, another method is proposed for reducing the number of constructed constraints.

Keywords—discrete event system, Supervisory control, Petri Net, Constraint

I. INTRODUCTION

SUPERVISORY control theory which was presented by Ramadge and Wonham is a theory for the systematic control of discrete event systems [1], [2]. This theory is based on restricting the behavior of system for obtaining the objective function. This restriction can be accomplished by disabling controllable events in special situations [3]. For analyzing the systems by this theory, automata was used as the first tool for modeling of discrete event systems. But when the number of states in the system is large, modeling the systems by automata is difficult or maybe impossible [4]. So, Petri Net (PN) was proposed as an alternative tool for modeling these systems [5]. PNs are made of places and transitions and these are connected to each other by arcs. Compact structure and mathematical properties have made PNs as a useful tool for modeling discrete event systems.

In a system, existence of some states which are called forbidden states is possible. So, the system must be avoided from entering these states. For this avoidance some methods are proposed. In [6], by putting some conditions on the controllable events, the system can be prevented from entering the forbidden states. These conditions lock the controllable transitions in special states. Another method for avoiding the forbidden states is adding some control places to the PN model of the system. In [7], a method is proposed for adding the control places by having the forbidden states. Another method for adding the control places is proposed in [8], which perform it by having linear constraints which are in the form of inequalities.

M. Zareiee is with the Semnan University, Semnan, IRAN, (e-mail: meisamzareiee@Gmail).

A. Dideban, is with Semnan University, Semnan, IRAN. He is with Semnan University, Semnan, IRAN (corresponding author to provide phone:(+98)231-3354123; fax: (+98)231-3354123; e-mail:a_dideban@yahoo.com).

P. Nazemzadeh is with the Semnan University, Semnan, IRAN, (e-mail: payamnazem@gmail.com).

In [9] a method is proposed for assigning linear constraints to forbidden states. This method limits the sum of tokens in some places of PN. When these constraints are enforced on the system by the method in [8], the system is prevented from entering the forbidden states. But this method for assigning constraints to forbidden states is only applicable on safe PNs and there is no a systematic method for assigning constraints to forbidden states in non safe PNs.

In this paper, for constructing linear constraints (also called constraint) related to forbidden states in non safe PNs, a very useful method will be proposed. At first, the constraints are constructed by the relation between the over-states of authorized states and sum of the marks in the places of PN when it is in a forbidden state. This way, like the presented method in [9] for safe PN, assigns an inequality to a forbidden state which in turn guaranties that the sum of marks of some places that are marked in a forbidden state is smaller than this sum when it is in that forbidden state. This constraint can be applicable if and only if it doesn't forbid any authorized states. When this constraint forbids at least one of the authorized states it cannot be applied on the system. So this method must be changed to obtain the constraints which verify all the authorized states. For obtaining these constraints, some inequalities as an integer linear programming problem must be solved. These inequalities are the ones that each one verifies one of the authorized states and one of them is violated by the considered forbidden state. When there is at least one answer that verifies all these inequalities, this constraint can be constructed. By using this method, assigning constraints to forbidden states in many cases is possible.

However, in a system which the number of constraints is large, by using the idea in [8] a large number of control places must be applied on the system. This concept leads to a complicated model of controlled system. But in [9] it is shown that it is possible to reduce the number of the constraints. In [10], a method is proposed that by using the invariant property reduces the number of constraints. In [11], another method for reduction the number of constraints is proposed that uses the concept of over-state. This simplification is done by the relation between the over-states of authorized states and forbidden states. Afterwards, some over-states are selected as the states which forbidding them leads to preventing system from entering the forbidden states. In [12] another method for reducing the number of constraints by using the relation between the set of authorized states and forbidden states and the concept of over-states is proposed. However, all of these methods are applicable on the safe PNs.

In this paper the second objective is to show that a method which was proposed for reducing the number of constraints in safe PNs is applicable on non safe PNs, too. This method is based on the invariant property and is presented in [10]. In this paper this method is generalized for non safe PN and the capability of it for reducing the number of constraints is shown by an example.

This paper is organized as follows. In section 2, the important definitions and concepts which are required for introducing the new idea, are presented. In section 3, the new idea for assigning constraints to forbidden states in non safe PNs is presented. Afterwards, an example is introduced to illustrate the capability of this method. The method for reducing the number of constraints is introduced in section 4. This method was first proposed for safe PN but in this section it is generalized for non safe PN and the impact of this method will be applied on the example in section 3. Finally conclusion is discussed in section 5.

II. PRELIMINARY PRESENTATION

In this section the goal is to present some definitions and basic concepts that are important for introducing the new idea. Some definitions which were presented in [11] and [12] are necessary for this paper. So, we recall them here.

A PN is represented by a quadruplet $R = \{P, T, W, M_0\}$ where P is the set of places, $T = \{T_C \cup T_U\}$ the set of controllable and uncontrollable transitions, W the incidence matrix and M_0 the initial marking. Places and transitions are connected together by arcs. The relation between places, transitions and arcs are shown by incidence matrix W [11].

In an industrial system, the model of system can be divided into two sections. The first section is the model of process and the next is the model of specification. The process model of system is the model of components of system and the specification is the model of some conditions that must be verified by the system for obtaining the desired behavior.

In a PN model of a system, all of the states which can be obtained by the model compose the set of reachable states and this set is shown by M_R . In the set M_R there may be some states that violate specifications or are deadlock states or the ones that the occurrence of uncontrollable events leads to these states. These states are called forbidden states [11]. So, the set M_R can be divided to two subsets. The first one is the set of forbidden states and is shown by M_F and the other one is the set of authorized states and is shown by M_A . These two subsets don't have any common component.

In the set of forbidden states, there is a very important subset that is called the set of border forbidden states [13]. Forbidding these states leads to forbidding all the forbidden states. These states are defined as follows:

Definition 1 [11]: Let M_B be the set of border forbidden state:

$$\mathcal{M}_B = \{M_i \in \mathcal{M}_F \mid \exists \sigma \in \Sigma_c \text{ and } \exists M_j \in \mathcal{M}_A, M_j \xrightarrow{\sigma} M_i\}$$

Where Σ_c is the set of controllable transitions. \square

From definition 1, it is obvious that by disabling controllable events when the firings of them are leading to the border forbidden states, preventing system from reaching to the border forbidden states is possible and then the system cannot reach to any forbidden states.

Remark 1. In this paper the number of marks in place P_i is shown by m_i . \square

Remark 2. In this paper each state in a PN is shown as $P_1^{m_1} P_2^{m_2} \dots P_n^{m_n}$ where P_1, \dots, P_n are the places which have marks and m_{ik} is the number of marks in place P_{ik} . When $m_{ik}=1$, we only mention it by P_{ik} . \square

For example suppose that in a PN there are 5 places as P_1, P_2, P_3, P_4, P_5 . Suppose that in a state, the places P_1, P_2, P_3 are marked and the places P_4, P_5 are empty. Also suppose that in this state the numbers of tokens in the places P_1, P_2, P_3 are 3, 1, 4, respectively. In this case the state of this PN is shown as $P_1^3 P_2^1 P_3^4$.

In a PN, there may be some properties for the markings that by changing the states of PN remain fixed. It means that when the state of system changes, they don't change. These properties may be invariant or partial invariant. Invariant is a property that the sum of marks of some places in firing of each transition is constant. Partial invariant is a property that is resultant of invariant and is defined in definition 2.

Definition 2 (Dideban et al., 2009). Let $P = \{P_1, P_2, \dots, P_i\}$ be a place invariant in a PN R , $P_{i1} = \{P_1, P_2, \dots, P_L\}$ for which $\{1, 2, \dots, L\} \subset \{1, 2, \dots, r\}$, is a partial place invariant (also called partial invariant) and it satisfy the following inequality:

$$q_1 m_1 + q_2 m_2 + \dots + q_L m_L \leq k, \forall M \in M(M_0)$$

where $M(M_0)$ is the set of all of states which are obtained by firing of transitions from M_0 . \square

In the PN model of system there is an important concept that is called over-state [11]. Over-states play an important role for introducing the new idea. This concept is defined in definition 3.

Remark 3. Suppose that M_1 and M_2 are two states. The relation $M_2 \leq M_1$ means that all the marked places in M_2 are marked in M_1 and the number of marks of each place in M_2 is smaller or equal than the number of marks of each equivalent place in M_1 . \square

For example, the relation $M_2 \leq M_1$, for the states $M_1 = P_1^3 P_2^4 P_3^2$ and $M_2 = P_1^3 P_2^2$ is true.

Definition 3. Suppose that M_1 is an accessible state. M_2 will be an over-state of M_1 if and only if:

$$M_2 \leq M_1$$

For example, $P_1 P_2 P_3$ is an over-state of the state $P_1^2 P_2 P_3$. All of the over-states of $P_1^2 P_2 P_3$ are presented in the set M_o as follows:

$$M_o = \{P_1, P_2, P_3, P_1 P_2, P_1 P_3, P_2 P_3, P_1 P_2 P_3, P_1^2, P_1^2 P_2, P_1^2 P_3, P_1^2 P_2 P_3\}$$

In some conditions to prevent system from reaching to the forbidden states, it is possible to forbid their over-states. This concept will help us to reduce the number of constraints, [11] and [12].

In the next section, assigning constraints to the forbidden states in safe and non safe PNs will be discussed. Verifying these constraints by the system, leads to preventing system from reaching to the forbidden states.

III. ASSIGNING CONSTRAINTS TO FORBIDDEN STATES

As it is mentioned in section 2, in a system, there may be some states that are called forbidden states. The system must be

avoided from reaching to these states. In [9], a method was proposed that by using it, assigning constraints to forbidden states is possible. These constraints are in the form of inequalities and by the idea presented in [8], the control places for applying them on the system can be calculated. But this idea is only applicable on safe Petri Net. In this section the goal is to construct constraints related to the forbidden states in non safe Petri Nets. So, at first, we remind the idea in [9] for constructing constraints in safe Petri Nets and after that, the new idea for constructing constraints in non safe Petri Nets will be proposed.

1.1. Constructing constraints in safe PN

In this subsection the method for constructing constraints related to forbidden states in safe Petri Nets is discussed. Suppose that in a safe Petri Net there is a forbidden state like $P_1P_2P_3$. When the system is in this state, sum of the marks in the places P_1 , P_2 and P_3 is 3 ($m_1+m_2+m_3=3$, where m_i is the number of marks in the place P_i). One way to prevent system from reaching to this state is to consider a constraint that doesn't permit sum of the marks of these places to be 3. This can be performed by an inequality like $m_1+m_2+m_3 \leq 2$. Verifying this inequality by the system, guaranties that the system cannot reach to this forbidden state. In [9], this concept is generalized for safe Petri Net and is described as follows:

Suppose that in a safe PN, the state $P_{i_1}P_{i_2}...P_{i_n}$ is a forbidden state. So, a linear constraint related to this state can be constructed as follows [9]:

$$\sum_{k=1}^n m_{ik} \leq n-1 \quad (1)$$

Where n is sum of marks of PN when the system is in that forbidden state (or the number of marked places) and m_{ik} is the number of marks in place P_{ik} of forbidden state.

When the constraints related to the forbidden states are verified by the system, the system cannot reach to the forbidden states. by using the idea in [8], control places can be applied on the system to verify these constraints. But this idea is only applicable on safe Petri Net. In the next subsection, the new idea will be proposed that by using it, in non safe Petri Net, constructing constraints related to forbidden states is possible.

1.2. assigning constraints to forbidden states in non safe PN

In section 3.1, we saw how to assign constraints to forbidden states in safe PN. But that method is not applicable for non safe PN. To show the deficiency of this method in non safe PNs, let us consider a simple example. Suppose that in a non safe PN there is a forbidden state like $P_1P_2P_3^2$. When the system is in this state we have:

$$m_1+m_2+m_3=4$$

Where m_i is the number of marks in the place P_i . Applying (1) on this state leads to the inequality as follows:

$$m_1+m_2+m_3 \leq 3 \quad (2)$$

But this inequality may forbid some authorized states. For example this inequality forbids the states $P_1^4, P_2^4, P_3^4, P_1P_2^3, P_1^2P_2^2, P_1^3P_2, P_1P_3^3, P_1^2P_3^2, P_1^3P_3, P_2P_3^3, P_2^2P_3^2, P_2^3P_3, P_1^2P_2P_3, P_1P_2^2P_3, P_1P_2P_3^2$ which some of them can be authorized states. So it is not correct to use the inequality (2) for preventing system from reaching to the forbidden state $P_1P_2P_3^2$. As a result, it is not always possible to apply relation

(1) on non safe PNs. But what happens if the inequality (2) doesn't forbid any authorized state? In section 3.2.1, a method is introduced to construct constraints related to forbidden states. This method is generalization of the method that was presented for safe PN.

1.2.1. assigning constraints to forbidden states by generalization of the method in safe PN

As it is obvious, the problem with applying the relation (1) on non safe PNs was the possibility of forbidding some authorized states by the constructed constraint. But it is possible that the constructed constraint by this method doesn't forbid any authorized state. In this case, it is possible to apply this constraint on the system for preventing system from reaching to this forbidden state. This concept is generalized in property 1.

Property 1. Suppose that $P_1^{k_1}P_2^{k_2}...P_n^{k_n}$ is a forbidden state. If all the admissible states deduced from the set of places (P_1, P_2, \dots, P_n) that the sum of their markings is $(k_1+k_2+\dots+k_n)$, doesn't exist in the set of over-states of authorized states, then, for preventing system from reaching to this state a constraint can be applied on the system as follows:

$$m_1+m_2+\dots+m_n \leq k_1+k_2+\dots+k_n-1 \quad (3)$$

Proof. For the proof of this property, it is necessary to show that this constraint forbids the considered forbidden state and doesn't forbid any authorized states.

It is clear that the considered forbidden state violates the constraint (3) since we have $m_1+m_2+\dots+m_n = k_1+k_2+\dots+k_n$, that is not lower or equal to $k_1+k_2+\dots+k_n-1$. Then by applying this constraint on the system, the considered forbidden state is not reachable.

Now, it must be proved that this constraint is not violated by any authorized states. If it is not true, applying this inequality on the system must forbid at least one of the authorized states. It means that there is at least one over-state of authorized states that its places are in the set (P_1, P_2, \dots, P_n) and the sum of its marking is $k_1+k_2+\dots+k_n$. But this is not true because this condition is evaluated in the hypothesis of property and if it is verified, we can use the constraint. \square

For example suppose that the state $P_1P_2P_3^2$ is a forbidden state. If the states $P_1^4, P_2^4, P_3^4, P_1P_2^3, P_1^2P_2^2, P_1^3P_2, P_1P_3^3, P_1^2P_3^2, P_1^3P_3, P_2P_3^3, P_2^2P_3^2, P_2^3P_3, P_1^2P_2P_3, P_1P_2^2P_3, P_1P_2P_3P_4$ are not in the set of over-states of authorized states, then, the inequality $m_1+m_2+m_3 \leq 3$ can be applied on the system for preventing system from reaching to this forbidden state.

The advantage of this property is that in some conditions it is possible to assign constraints to forbidden states in non safe PN. Then these inequalities can be applied on the system by adding the control places.

To explain this property, a simple example is introduced.

Example 1. Suppose a system composed of two machines, one robot and a buffer between them. The start command of each machine is done by the controllable events c_1 and c_2 and the end of task of each machine is accomplished by the occurrence of uncontrollable events f_1 and f_2 . In this system the specification is that the second machine can do its work when the first machine completes its task twice. The synchronized model of this system is illustrated in fig.1.

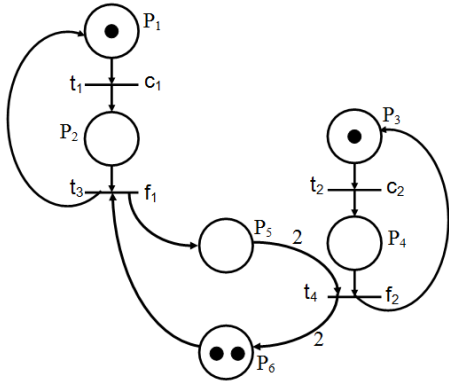


Fig. 1. PN model of the system in example 1

The marking graph of this system is shown in fig. 2.

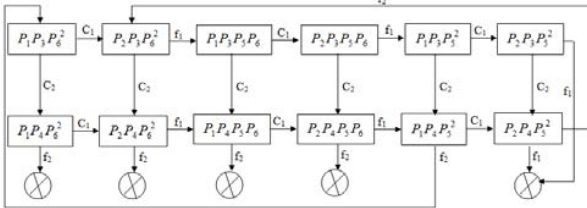


Fig. 2. Marking graph of the system in example 1

In this figure, the uncertain states are shown by \otimes . These states are those which violate specification. For example, when the system is in the state $P_1P_4P_6^2$ firing of the transition f_2 violates the specification. Since machine 1 hasn't completed its task.

As it is obvious from the marking graph, the set of authorized states is:

$$M_A = \{P_1P_3P_6^2, P_2P_3P_6^2, P_1P_3P_5P_6, P_2P_3P_5P_6, P_1P_3P_5^2, P_1P_4P_5^2\}.$$

And the set of forbidden states which are also border forbidden states is as follows:

$$M_B = \{P_1P_4P_6^2, P_2P_4P_6^2, P_2P_4P_5^2, P_2P_3P_5^2, P_2P_4P_5P_6, P_1P_4P_5P_6\}.$$

So, to prevent system from reaching to these forbidden states, the constraints related to them must be extracted and then the control places corresponding to these constraints must be added to PN model of system. Now we must see if it is possible to assign inequalities to forbidden states by using the method proposed in property 1.

Remark 4. In this paper we don't write the set of over-states of authorized states since for applying property 1, the set of authorized states is enough. It means that the set of over-states of authorized states are recognized from the set of authorized states. \square

At first, we consider the set of forbidden states to construct constraints. For the forbidden state $P_1P_4P_6^2$ we must check the set B_1 as follows:

$$B_1 = \{P_1^4, P_4^4, P_6^4, P_1P_4^3, P_1^2P_4^2, P_1^3P_4, P_1P_6^3, P_1^2P_6^2, P_1^3P_6, P_4P_6^3, P_4^2P_6^2, P_4^3P_6\}.$$

The states in B_1 don't belong to the set of over-states of M_A . So, according to property 1, a constraint can be assigned to the forbidden state $P_1P_4P_6^2$ as follows:

$$m_1 + m_4 + m_6 \leq 3 \quad (4)$$

Similar to this case, for the forbidden states $P_2P_4P_6^2$, $P_2P_4P_5^2$, $P_2P_3P_5^2$, $P_2P_4P_5P_6$, the related constraints are constructed as follows:

$$P_2P_4P_6^2 \rightarrow m_2 + m_4 + m_6 \leq 3 \quad (5)$$

$$P_2P_4P_5^2 \rightarrow m_2 + m_4 + m_5 \leq 3 \quad (6)$$

$$P_2P_3P_5^2 \rightarrow m_2 + m_3 + m_5 \leq 3 \quad (7)$$

$$P_2P_4P_5P_6 \rightarrow m_2 + m_4 + m_5 + m_6 \leq 3 \xrightarrow{m_5+m_6=2} m_2 + m_4 \leq 1 \quad (8)$$

For the forbidden state $P_1P_4P_5P_6$, the set B_6 must be considered as follows:

$$B_6 = \{P_1^4, P_4^4, P_5^4, P_6^4, P_1P_4^3, P_1^2P_4^2, P_1^3P_4, P_1P_5^3, P_1^2P_5^2, P_1^3P_5, P_1P_6^3, P_1^2P_6^2, P_1^3P_6, P_4P_5^3, P_4^2P_5^2, P_4^3P_5, P_4P_6^3, P_4^2P_6^2, P_4^3P_6, P_5P_6^3, P_5^2P_6^2, P_5^3P_6, P_1P_4P_5^2, P_1P_4^2P_5, P_1^2P_4P_5, P_1P_4P_6^2, P_1P_4^2P_6, P_1^2P_4P_6, P_1P_5P_6^2, P_1P_5^2P_6, P_1^2P_5P_6, P_4P_5P_6^2, P_4P_5^2P_6, P_4^2P_5P_6\}$$

As it is obvious from this set, the state $P_1P_4P_5^2$ belongs to the set of over-states of authorized states (it is an authorized state). So it is not possible to assign a constraint to the forbidden state $P_1P_4P_5P_6$ by using the method presented in property 1.

As it was obvious, by using the method in property 1 assigning constraints was possible for 5 forbidden states, but for one of them it was impossible. So, to prevent system from reaching to the 5 forbidden states $\{P_1P_4P_6^2, P_2P_4P_6^2, P_2P_4P_5^2, P_2P_3P_5^2, P_2P_4P_5P_6\}$, five control places can be applied on the system. But by using the method in property 1, for the 6th forbidden state ($P_1P_4P_5P_6$), it's not possible to assign a constraint. But how can we prevent the system from reaching to the 6th forbidden state ($P_1P_4P_5P_6$)? Is it possible to assign an inequality to it? In section 3.2.2, property 1 is developed and a general method for assigning constraint to forbidden states in non safe PN is proposed.

3.2.2. Assigning constraints to forbidden states in a general way

As mentioned before, when the constructed constraint by the method in property 1 forbids at least one of the authorized states, applying this property is impractical. This concept was seen in the case of 6th forbidden state ($P_1P_4P_5P_6$) in example 1. But how can we assign another inequality to this forbidden state? If this inequality exists, it must be the one that verifies all of the authorized states and prevent system from reaching to this forbidden state. For this reason we can consider a general inequality and check verifying all the authorized states and violating by the forbidden state. So, we can consider this inequality as follows:

$$k_1m_1 + k_2m_2 + \dots + k_6m_6 \leq x \quad (9)$$

Where k_1, k_2, \dots, k_6 and x are some constants and m_i is the number of marks in place P_i . If this inequality verifies all the authorized states and is violated by the forbidden state ($P_1P_4P_5P_6$), it can be applied on the system for preventing system from reaching to this forbidden state. To verify the authorized states by the inequality (9), we can put them into this inequality as follows:

$$P_1P_3P_6^2 \rightarrow k_1 + k_3 + 2k_6 \leq x \quad (10)$$

$$P_2P_3P_6^2 \rightarrow k_2 + k_3 + 2k_6 \leq x \quad (11)$$

$$P_1P_3P_5P_6 \rightarrow k_1 + k_3 + k_5 + k_6 \leq x \quad (12)$$

$$P_2P_3P_5P_6 \rightarrow k_2 + k_3 + k_5 + k_6 \leq x \quad (13)$$

$$P_1P_3P_5^2 \rightarrow k_1 + k_3 + 2k_5 \leq x \quad (14)$$

$$P_1P_4P_5^2 \rightarrow k_1 + k_4 + 2k_5 \leq x \quad (15)$$

For Violating the inequality (9) by the forbidden state $P_1P_4P_5P_6$, we must put this state into the inequality (9) and convert the smaller equal sign to greater sign as follows:

$$P_1P_4P_5P_6 \rightarrow k_1+k_4+k_5+k_6 > x \quad (16)$$

If there is a solution that verifies (10) to (16), at least one inequality like $k_1m_1+k_2m_2+\dots+k_6m_6 \leq x$ can be obtained that verifies all the authorized states and prevent system from reaching to the forbidden state $P_1P_4P_5P_6$. So (10) to (16) must be solved. One answer for them is $k_1=0, k_2=0, k_3=0, k_4=2, k_5=0, k_6=1, x=2$. So, the inequality (9) will be as follows:

$$2m_4+m_6 \leq 2 \quad (17)$$

This inequality is a constraint that applying it on the system verifies all the authorized states and prevents system from reaching to the forbidden state $P_1P_4P_5P_6$.

So, this concept can be generalized for constructing constraints for the forbidden states. Algorithm 1 shows this concept in a general way.

Algorithm 1. Let $\mathcal{M}_A = \{P_{v1}^{s_{v1}} P_{v2}^{s_{v2}} \dots P_{vr}^{s_{vr}}, \dots, P_{r1}^{s_{r1}} P_{r2}^{s_{r2}} \dots P_{rq}^{s_{rq}}\}$ be the set of authorized states and $P_{v1}^{s_{v1}} P_{v2}^{s_{v2}} \dots P_{vu}^{s_{vu}}$ a forbidden state. Follow these steps to obtain a constraint related to this forbidden state:

- 1- Consider an inequality as follows:

$$k_1m_1+k_2m_2+\dots+k_nm_n \leq x \quad (18)$$

Where n is the number of places and m_i is the number of marks in place P_i and k_i and x are constants.

- 2- Put the markings of all the authorized states in the inequality (18) and construct inequalities as follows:

$$P_{v1}^{s_{v1}} P_{v2}^{s_{v2}} \dots P_{vr}^{s_{vr}} \rightarrow k_{v1}s_{v1} + k_{v2}s_{v2} + \dots + k_{vr}s_{vr} \leq x \quad (1-19)$$

.

.

.

$$P_{r1}^{s_{r1}} P_{r2}^{s_{r2}} \dots P_{rq}^{s_{rq}} \rightarrow k_{r1}s_{r1} + k_{r2}s_{r2} + \dots + k_{rq}s_{rq} \leq x \quad (r-19)$$

Where r is the number of authorized states.

- 3- Put the marking of the forbidden state in the inequality (18) and convert the smaller equal sign to greater sign as follows:

$$P_{v1}^{s_{v1}} P_{v2}^{s_{v2}} \dots P_{vu}^{s_{vu}} \rightarrow k_{v1}s_{v1} + k_{v2}s_{v2} + \dots + k_{vu}s_{vu} > x \quad (20)$$

- 4- Solve the set of relations (1-19) to (r-19) and (20) and obtain the minimum values for k_i s and x that verifies these relations.

- 5- If step 4 has an answer, then put the k_i s and x in the inequality (18). The resultant inequality is a constraint for the forbidden state $P_{v1}^{s_{v1}} P_{v2}^{s_{v2}} \dots P_{vu}^{s_{vu}}$.

□

Remark 5. When the algorithm 1 has an answers, it can have many answers, so we can choose one of them with minimum of borne and weight (this concept is considered in step 4).

□

It is clear that if the algorithm gives an answer, it verifies all the authorized states and forbid the considered forbidden state.

As it is obvious, the algorithm 1 gives a very efficient method for assigning constraints to forbidden states and then these constraints can be applied on the system by control places using the idea in [8].

In example 1, there are six constraints that can be applied on the system by control places. But these constraints are so many and in [9], it is mentioned that they can be reduced. This reduction can be performed by the method presented for safe PN. These methods can be generalized for non safe PN. In the next section this concept will be discussed.

IV. REDUCTION THE NUMBER OF CONSTRAINTS

In the previous section, an example was presented that had 6 forbidden states, so, 6 constraints were constructed for preventing system from reaching to them. In [9], it is shown that the number of constraints can be reduced. In [10-12], some methods are proposed that using them reducing the number of constraints in safe PN is possible.

The idea in [10] uses the invariant and partial invariant for reducing the number of constraints. This method was proposed for safe and conservative PN. But it can be applied to non safe PN. In the following in property 2 and 3, this method is generalized for non safe PN.

Property 2. Let be $C_i = \{(m_1+m_{i1}+\dots+m_{i(n-1)} \leq k), \dots, (m_r+m_{r1}+\dots+m_{r(n-1)} \leq k)\}$ r constraints related to the forbidden states. If there is an invariant in the system as follows:

$$m_1+m_2+\dots+m_r=1$$

Then the r constraints will be converted to one constraint as follows:

$$m_1+\dots+m_{i(n-1)} \leq k-1$$

Where n is the number of marked places and k is the borne of each constraint. □

Proof. The proof of this property is like the one for safe PN in [10] and we recall it here.

Necessary condition:

The sum of all the constraints gives the constraint as follow:

$$(m_1+m_{i1}+\dots+m_{i(n-1)})+(m_2+m_{i1}+\dots+m_{i(n-1)})+\dots+(m_r+m_{i1}+\dots+m_{i(n-1)}) \leq r(k-1)$$

Since there is an invariant ($m_1+m_2+\dots+m_r = 1$), this constraint is changed as follows:

$$1+r(m_{i1}+\dots+m_{i(n-1)}) \leq r(k-1) \rightarrow m_{i1}+\dots+m_{i(n-1)} \leq k-1-1/r \quad (m_i \text{ is integer number}) \rightarrow m_{i1}+\dots+m_{i(n-1)} \leq n-2$$

Sufficient condition:

$$m_{i1}+\dots+m_{i(n-1)} \leq n-2$$

$$\forall \{1, \dots, r\} \quad m_j = 0 \text{ or } 1 \rightarrow m_j + m_{i1} + \dots + m_{i(n-1)} \leq n-1$$

□

Property 3. Let be $C_p = \{(m_1+m_{i1}+\dots+m_{i(n-1)} \leq k), \dots, (m_r+m_{r1}+\dots+m_{r(n-1)} \leq k)\}$ the equivalent constraints to forbidden states. If there is a partial invariant in the system as follows:

$$m_1+m_2+\dots+m_r \leq 1$$

Then these r constraints can be reduced to a constraint as follows:

$$m_1+m_2+\dots+m_r+m_{i1}+\dots+m_{i(n-1)} \leq k$$

Where n is the number of marked places and k is the borne of constraints. □

Proof. The proof of this property is like the one for safe PN in (Dideban & Alla, 2005), so we recall it as follow.

Necessary condition:

$$\forall j \in \{1, \dots, r\}, m_j + m_{i1} + \dots + m_{i(n-1)} \leq k \text{ and } m_1 + m_2 + \dots + m_r \leq 1 \Rightarrow (m_1 + m_2 + \dots + m_r) + m_{i1} + \dots + m_{i(n-1)} \leq k+1$$

We want to show that the limit $k+1$ is never reached. If not, it is necessary that $m_1 + m_2 + \dots + m_r = 1$ and $m_{i1} + \dots + m_{i(n-1)} = k$. But if $m_{i1} + \dots + m_{i(n-1)} = k$, it is necessary that for $\forall j \in \{1, \dots, r\}, m_j = 0$ since from the constraints we have $m_j + m_{i1} + \dots +$

$m_{i(n-1)} \leq k$. then $m_1 + m_2 + \dots + m_r = 0$. Thus limit $k+1$ never is reached.

Sufficient condition:

$(m_1 + m_2 + \dots + m_r) + m_{i_1} + \dots + m_{i(n-1)} \leq k$ and $\forall i \in \{1, \dots, r\}, m_i \geq 0$

Then

$\forall i \in \{1, \dots, r\} m_i + m_{i_1} + \dots + m_{i(n-1)} \leq k$

□

Using these properties, reducing the number of constraints in non safe PN is reachable. Now we want to apply these properties on the constraints in example 1.

The constraints in example 1 were as follow:

$m_1 + m_4 + m_6 \leq 3$, $m_2 + m_4 + m_6 \leq 3$, $m_2 + m_4 + m_5 \leq 3$, $m_2 + m_3 + m_5 \leq 3$, $m_2 + m_4 \leq 1$, $2m_4 + m_6 \leq 2$.

In this model there is an invariant as $m_1 + m_2 = 1$. So according to the property 2, it is possible to reduce the two constraints $m_1 + m_4 + m_6 \leq 3$ and $m_2 + m_4 + m_6 \leq 3$ to one constraint as follows:

$$m_4 + m_6 \leq 2 \quad (21)$$

But the inequality (17) covers the inequality (21). So, it is possible to apply (17) instead of (21). Thus, (17) is a constraint for the forbidden states $P_1P_4P_6^2$, $P_2P_4P_6^2$ and $P_1P_4P_5P_6$.

Also, in this example there is an invariant as $m_3 + m_4 = 1$. So, according to the property 2, the constraints $m_2 + m_4 + m_5 \leq 3$ and $m_2 + m_3 + m_5 \leq 3$ can be converted to one constraint as follows:

$$m_2 + m_5 \leq 2 \quad (22)$$

Also the inequalities (17) and (22) cover the inequality (8) since $m_2 \leq 1$, $m_4 \leq 1$ and $m_5 + m_6 = 2$. So, 6 constraints are reduced to two constraints as follows:

$$2m_4 + m_6 \leq 2, \quad m_2 + m_5 \leq 2 \quad (23)$$

Applying these two constraints on the system prevents system from reaching to all the forbidden states. These two constraints can be applied on the system by the method presented in [8].

Calculation of control places for applying constraints on the system

After reducing the number of constraints in example 1, the final constraints must be applied on the system. For this reason the control places according to the idea in [8] must be calculated. In the following, we review this method.

To calculate control places, suppose that the set of constraints is shown as follows:

$$L.M_P \leq b \quad (24)$$

Where M_P is the marking vector, L is a $n_c \times n$ matrix, b is a $n_c \times 1$ vector, n_c is the number of constraints and n is the number of places. In this method instead of each constraint, a control place is added to the system. These control places are calculated by converting inequalities in (24) to equalities. So, when the control places are added to the system, its incidence matrix changes and instead of each constraint (control place), a row is added to this incidence matrix. Now, suppose that the incidence matrix of system before applying control places is shown by W_P . Also suppose that the rows that must be added to this matrix are shown by W_c , so this matrix is calculated as follows:

$$W_c = -L.W_P \quad (25)$$

This matrix must be added to the system, and the incidence matrix for controlled model is obtained as follows:

$$W = \begin{bmatrix} W_P \\ W_c \end{bmatrix} \quad (26)$$

After this calculation, the initial marking of these control places must be calculated. For this reason suppose that the initial marking of system is shown by M_{P_0} , then the initial markings of the control places are calculated as follows:

$$M_{s_0} = b - L.M_{P_0} \quad (27)$$

So the initial marking of controlled model is as follows:

$$M_0 = \begin{bmatrix} M_{P_0} \\ M_{s_0} \end{bmatrix} \quad (28)$$

Now, we must calculate the control places for the constraints in example 1. The two constraints for this example were:

$$2m_4 + m_6 \leq 2, \quad m_2 + m_5 \leq 2$$

So, for this example we have:

$$L = \begin{bmatrix} 0 & 0 & 0 & 2 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix} \quad \text{and} \quad b = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$$

So, W_c is calculated as follows:

$$W_c = \begin{bmatrix} 0 & -2 & 1 & 0 \\ -1 & 0 & 0 & 2 \end{bmatrix}$$

And the initial marking of the control places is as follows:

$$M_{s_0} = \begin{bmatrix} 0 \\ 2 \end{bmatrix}$$

Following this calculation, the controlled model of this example is illustrated in Fig 3.

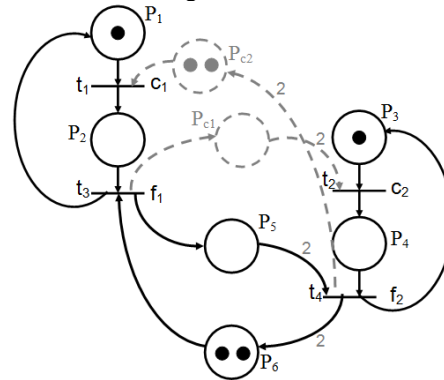


Fig. 3. The controlled model of the system in example 1

The control places and their related arcs are shown in gray color and dashed lines.

V. CONCLUSION

In this paper, the problem of assigning constraints to forbidden states in non safe Petri net is considered. So a systematic method is proposed to assign a linear constraint to a forbidden state. This constraint can be constructed by having authorized states and the considered forbidden state. At first the method which was presented for safe PN is generalized for non safe PN. But the constructed constraint by this method may forbid some authorized states. To solve this problem, a basic constraint is considered. From this constraint, some inequalities are extracted. These inequalities are the ones which one of them is deduced from the violation of basic constraint by the considered forbidden state and the others are deduced from verifying the basic constraint by the authorized

states. So, these inequalities must be solved to obtain an inequality which verifies all of them. But when the number of these constraints is large, simplification of constraints for reducing the number of them must be performed. In this paper, this simplification is accomplished by generalization of the method which was presented for safe Petri net and uses the invariant and partial invariant property.

REFERENCES

- [1] P. Ramadge, W. Wonham, "Modular feedback logic for discrete event systems," *SIAM Journal of Control and Optimization*, 1987, 25(5), 1202–1218.
- [2] P. Ramadge, W. Wonham, "The control of discrete event systems," *Dynamics of discrete event systems [Special issue]. Proceedings of the IEEE*, 1989, 77(1), 81–98.
- [3] A. Giua, Petri Nets as Discrete Event Models for Supervisory Control. Ph.D. Thesis, 1992.
- [4] J. Moody, P. Antsaklis, "Petri net supervisor for DES with uncontrollable and unobservable transition," *IEEE Trans. Automatic Control*, 2000, 45(3): 462-476.
- [5] B. Krogh, L. Holloway, "Synthesis of feedback control logic for discrete manufacturing systems," *Automatica*, 1991, vol. 27, no. 4, pp. 641–651.
- [6] L. Holloway, X. Guan, and L. Zhang, "A Generalisation of state avoidance Policies for Controlled Petri Nets," *IEEE Trans. Autom. Control*, 1996, AC-41, 6, 804- 816.
- [7] A. Ghaffari, N. Rezg, and X. Xie, "Design of live and maximally permissive petri net controller using the theory of regions," *IEEE Transactions on Robotics and Automation*, 2003, 19(1).
- [8] K. Yamalidou, J. Moody, M. Lemmon, and P. Antsaklis, "Feedback control of petri nets based on place invariants," *Automatica*, 1996, 32(1), 15–28.
- [9] A. Giua, F. DiCesare, M. Silva, Generalized Mutual Exclusion Constraints on Nets with Uncontrollable Transitions. In *Proc. IEEE int. conf. on systems, man, and cybernetics*, 1992, pp. 974–799.
- [10] A. Dideban, and H. Alla, "From forbidden state to linear constraints for the optimal supervisory control," *Control Engineering and applied Informatics (CEAI)*, 2005, 7(3), 48-55.
- [11] A. Dideban, and H. Alla, "Reduction of Constraints for Controller Synthesis based on Safe Petri Nets," *Automatica*, 2008, 44(7): 1697-1706.
- [12] A. Dideban, M. Zareiee, and H. Alla, Controller synthesis with very simplified linear constraints in PN model. *The 2nd IFAC workshop on Dependable Control of Discrete Systems*, June 10-12, Bari, Italy, 2009, 265-270.
- [13] R. Kumar, L. Holloway, "Supervisory control of deterministic Petri nets with regular specification languages," *IEEE Trans. Automatic Control*, 1996 41(2):245-249.