Safeness-Enforcing Supervisory Control
for Railway Networks

Fabrizio Diana, Alessandro Giua, Carla Seatzu

Department of Electrical and Electronic Engineering, University of Cagliari
Piazza d'Armi — 09123 Cagliari, Italy
{giua,seatzu}@diee.unica.it
http://www.diee.unica.it/~giua/info.html

Abstract— In this paper we deal with the problem of modeling railway networks with Petri nets so as to apply the theory of supervisory control for discrete event systems to automatically design the system controller. We provide a modular representation of railway networks in terms of stations and tracks including sensors and semaphores. We ensure safeness and local liveness imposing both Generalized Mutual Exclusion Constraints and constraints also involving the firing vector.

I. INTRODUCTION

The specification, analysis and implementation of railway control logic has ever been an important activity since trains and railways were invented centuries ago, and failure of control logic can lead to railway accidents and loss of human life. At present time, this activity is even more important because railway networks are often large, the speed of trains and traffic density is increasing, and activities within networks are taking place concurrently and at geographically different locations. As a result, the overall complexity of railway systems increases, and hence greater demands are placed on the control logic of these systems [10].

The control of a railway network can be divided into two distinct phases. The first one, at a lower level, imposes the satisfaction of a series of safeness constraints (collision avoidance) and liveness constraints (deadlock freeness). The second one, at a higher level, is concerned with the problem of scheduling both the departures and the stops, so as to optimize the efficiency of the net. In this paper the attention is uniquely devoted to the first phase.

We focus our attention on the modeling and control of railway networks with Petri nets [14], that provide a powerful framework for the analysis and control of distributed and concurrent systems. Some of the advantages of Petri nets as models for discrete event control include [8]: graphical representation, solid foundations based in mathematics, the existence of simulation and formal analysis techniques, and the existence of computer tool support for simulation, analysis and control. The literature on modelling and analyzing railway systems using Petri nets is not extensive and a good survey is given by Janczura in [10]. The idea of applying Petri nets theory goes back to Genrich [5], then it was revisited in [1], [11] and in [9] where coloured Petri nets have been used. Significant contributions in this setting are also due to Decknatel and Schnieder [2] and Di

Febbraro *et al.* [4] who used hybrid Petri nets to model transportation systems.

We provide a modular representation of railway networks in terms of stations and tracks including sensors and semaphores. The overall model will be a place/transition (P/T) net whose transitions may be (un)controllable and/or (un)observable, following the paradigm of supervisory control [15]. There exist several techniques for automatically designing controllers for P/T nets with uncontrollable and/or unobservable transitions [8]. In particular, we show how collision avoidance constraints can be expressed as Generalized Mutual Exclusion Constraints (GMECs) [6] and the corresponding controller takes the form of a set of monitor places that can be automatically computed — taking into account uncontrollable and unobservable transitions — using Moody's parametrization [13].

It is well know that a P/T net monitor based solution may not be maximally permissive when there exist uncontrollable or unobservable transitions [6]: the price one has to pay to keep the control structure simple is the fact that the controller may unnecessarily disable some transitions. In the case at hand, this leads to a *local deadlock*, i.e., the automatically designed monitor controller leads to a block when two trains coming from opposite directions cross each other at a station. To solve this problem we modify the controller, writing explicitly a new set of rules that define the admission policy into the stations; we show that the corresponding control structure is still very simple and takes the form of a "monitor with self-loops".

A nice feature of this approach is that the whole control problem can be divided into a certain number of sub-problems, thus making the proposed control procedure suitable even for large-scale networks.

Let us finally observe that in [7] we have also addressed the problem of *global deadlock* avoidance. In fact, when all the modules introduced here are put together and the number of trains in the network increases, it may well be the case that the net enters a blocking state. In [7] a solution to this problem has been provided applying siphon analysis to a simplified net and adding new monitors that, controlling the net siphons to prevent them from becoming empty, ensure global liveness for the system.

II. BACKGROUND

A. Generalities on Petri nets

In this subsection we recall the formalism used in the paper. For more details on Petri nets we address to [14].

A *Place/Transition net* (P/T net) is a structure $N = (P, T, Pre, Post)$, where P is a set of m places; T is a set of n transitions; $Pre : P \times T \to \mathbb{N}$ and $Post : P \times T \to \mathbb{N}$ are the *pre-* and *post-* incidence functions that specify the arcs; $C = Post - Pre$ is the incidence matrix.

A *marking* is a vector $m : P \to \mathbb{N}$ that assigns to each place of a P/T net a non-negative integer number of tokens, represented by black dots. In the following we denote as m_i the marking of place p_i.

A *P/T system* or *net system* $\langle N, m_0 \rangle$ is a net N with an initial marking m_0.

A transition t is enabled at m if $m \geq Pre(\cdot, t)$ and may fire yielding the marking $m' = m + C(\cdot, t)$. The notation $m[t\rangle m'$ means that an enabled transition t may fire at m yielding m'.

A *firing sequence* from m_0 is a (possibly empty) sequence of transitions $\sigma = t_1 \cdots t_k$ such that $m_0[t_1\rangle m_1[t_2\rangle \cdots [t_k\rangle m_k$. A marking m is reachable in $\langle N, m_0 \rangle$ iff there exists a firing sequence σ such that $m_0[\sigma\rangle m$. Given a net system $\langle N, m_0 \rangle$ the set of reachable markings is denoted $R(N, m_0)$. The function $\sigma : T \to \mathbb{N}$, where $\sigma(t)$ represents the number of occurrences of t in σ, is called firing count vector of the fireable sequence σ. If $m_0[\sigma\rangle m$, then we can write in vector form $m = m_0 + C(\cdot, t) \cdot \sigma$. This is known as the *state equation* of the system.

B. Generalized Mutual Exclusion Constraints

Assume we are given a set of legal markings $\mathcal{L} \subseteq \mathbb{N}^m$, and consider the basic control problem of designing a supervisor that restricts the reachability set of the plant in closed loop to $\mathcal{L} \cap R(N, m_0)$. Of particular interest are those PN state-based control problems where the set of legal markings $\mathcal{L}$ is expressed by a set of n_c linear inequality constraints called *Generalized Mutual Exclusion Constraints* (GMECs).

Each GMEC is a couple (w, k) where $w : P \to \mathbb{Z}$ is a $m \times 1$ weight vector and $k \in \mathbb{Z}$. Given the net system $\langle N, m_0 \rangle$, a GMEC defines a set of markings that will be called *legal markings*: $\mathcal{M}(w, k) = \{m \in \mathbb{N}^m \mid w^T m \leq k\}$. The markings that are not legal are called *forbidden markings*. A controlling agent, called supervisor, must ensure that the forbidden markings will be not reached. So the set of legal markings under control is $\mathcal{M}_c(w, k) = \mathcal{M}(w, k) \cap R(N, m_0)$.

In the presence of multiple constraints, all constraints can be grouped and written in matrix form as

$$W^T m \leq k \qquad (1)$$

where $W \in \mathbb{Z}^{m \times n_c}$ and $k \in \mathbb{Z}^{n_c}$. The set of legal markings is $\mathcal{M}(W, k) = \{m \in \mathbb{N}^m \mid W^T m \leq k\}$.

Each constraint requires the introduction of a new place (denoted as *monitor place* or *controller place*). To each
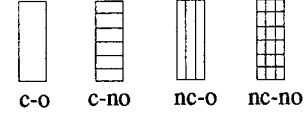

Fig. 1. Convention on transitions: c controllable, o observable, nc uncontrollable, no unobservable.

monitor place, it corresponds an additional row in the incidence matrix of the closed loop system. In particular, let C_c be the matrix that contains the arcs connecting the controller places to the transitions of the plant, and (m_{c0}) m_c the (initial) marking of the controller. The incidence matrix $C \in \mathbb{Z}^{(m+n_c) \times n}$ of the closed loop system is

$$C = \begin{bmatrix} C_p \\ C_c \end{bmatrix} \qquad (2)$$

and the marking vector $m \in \mathbb{Z}^{m+n_c}$ and initial marking m_0 are

$$m = \begin{bmatrix} m_p \\ m_c \end{bmatrix}, \qquad m_0 = \begin{bmatrix} m_{p0} \\ m_{c0} \end{bmatrix}, \qquad (3)$$

where the subscript p has been used to denote the variables of the plant.

In the case of controllable and observable transitions, Giua *et al.* provided the following theorem.

Theorem 1 ([6]) *If* $k - W^T m_0 \geq 0$ *then a Petri net controller with incidence matrix* $C_c = -W^T C_p$ *and initial marking* $m_{c0} = k - W^T m_{c0}$ *enforces constraint (1) when included in the closed loop system (2) with marking (3).*

The controller so constructed is maximally permissive, i.e., it prevents only transitions firings that yield forbidden markings. The controller net has n_c control places and no transition is added.

It often occurs that certain transitions can not be disabled by any control action (*uncontrollable transitions*) or can not be directly detected or measured (*unobservable transitions*). The uncontrollability of a transition indicates that we can not draw any arc from the controller places to this transition, so that the controller may never disable it. The unobservability of a transition implies that it must have the same number of input and output arcs to/from each controller place — i.e., its only admissible connection to monitor places is given by self-loops — so that its firing does not modify the controller state. In these cases the previous theorem is no more valid and an appropriate set of transformed constraints needs to be determined so as to construct a Petri net controller.

Theorem 2 ([12]) *Let a Petri net with incidence matrix* C_p *be given with a set of uncontrollable and/or unobservable transitions. Let* C_{uc} *(C_{uo}) be the incidence matrix of the uncontrollable (unobservable) portion of the Petri net. A set of linear constraints on the net marking,* $W^T m \leq k$, *are to be imposed. Assume* $R_1 \in \mathbb{Z}^{n_c \times m}$ *satisfy* $R_1 m \geq 0$ $\forall m \in \mathbb{N}^m$, *and* $R_2 \in \mathbb{Z}^{n_c \times n_c}$ *be a positive definite diagonal matrix, with* $R_1 + R_2 W^T \neq 0$. *Let*

$$\begin{bmatrix} R_1 & R_2 \end{bmatrix} \cdot$$
$$\begin{bmatrix} C_{uc} & C_{uo} & -C_{uo} & m_{p0} \\ W^T C_{uc} & W^T C_{uo} & -W^T C_{uo} & W^T m_{p0} - k - 1 \end{bmatrix}$$
$$\leq \begin{bmatrix} 0 & 0 & 0 & -1 \end{bmatrix}.$$
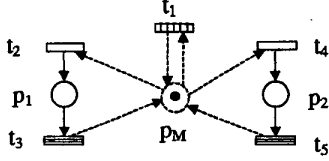
100

Fig. 2. A monitor with self-loop.
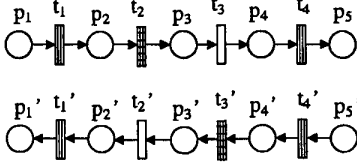

Fig. 3. The Petri net model of a track.

Then the controller

$$C_c = -(R_1 + R_2 W^T)C_p = -WC_p$$
$$m_{c0} = R_2(k+1) - 1 - (R_1 + R_2 W^T)m_{p0} =$$
$$k^T - Wm_{p0}$$

exists and causes all subsequent markings of the closed loop system (2) to satisfy the constraint $W^T m_p \leq k$ without attempting to inhibit uncontrollable transitions and without detecting unobservable transitions.

It is possible to compute matrices R_1 and R_2 following the procedure by Moody by simply performing row operations on a matrix containing the uncontrollable and unobservable columns of the plant incidence matrix [13]. The computational part of this procedure involves little more than the integer triangularization of a matrix.

In the rest of the paper we adopt the convention reported in figure 1 so as to distinguish among controllable and/or uncontrollable, observable and/or unobservable transitions.

C. Constraints involving the firing vector

Certain control goals may involve the firing vector of a Petri net as well as the tokens content of places [13]. A constraint of this kind takes the form:

$$w^T m + v_j q_j \leq k \qquad (4)$$

where $v_j \in \mathbb{N}$, and $q_j \in \{0,1\}$ is such that $q_j = 1$ if t_j is enabled, otherwise $q_j = 0$. Thus, constraint (4) implies that $w^T m \leq k$ and that transition t_j should be enabled if $k - w^T m \geq v_j$.

The corresponding control structure takes the form of a monitor place with a self-loop. As an example, in figure 2 we have shown the monitor with self-loop p_M that enforces the constraint $m_1 + m_2 + q_1 \leq 1$. Note that transition t_1 must be controllable, transitions t_2 and t_4 must be controllable and observable, transitions t_3 and t_5 must be observable.

III. MODELING RAILWAY NETWORKS WITH PETRI NETS

In this section we show how Petri nets can be efficiently used as a modeling tool for railway networks. In particular,

we show that the whole network can be seen as the composition of a certain number of elementary modules, namely tracks and stations.

A. The track model

An example of Petri net modeling a track is shown in figure 3. It consists of two series of places $(p_1, \cdots, p_5$ and $p'_1, \cdots, p'_5)$ and transitions $(t_1, \cdots, t_4$ and $t'_1, \cdots, t'_4)$, each one representative of the flow of trains in a certain direction. Each couple of places p_i, p'_i represents a segment of the track, i.e., the marking of either p_i or p'_i denotes the presence of a train in the segment. Note that, in the case of a double track, the two lines are independent and places p_i and p'_i correspond to parallel segments and can be marked simultaneously. On the contrary, in the case of a single track two places, p_i and p'_i, are used to represent the same segment of the track that can be crossed in both directions, but places can not be marked at the same time. Note that during simulation a release delay is associated to each transition, to represent the time a train requires to run along that segment.

Transitions may be (un) controllable and/or (un) observable. In this setting, a transition that is both controllable and observable represents a semaphore (see transitions t_3 and t'_2 in figure 3), i.e., in that point of the net the presence of a train can be detected and its transit can be forbidden. In all real situations a semaphore is placed at the exit of a track, or equivalently at the entrance of a station. A transition that is observable but not controllable (see transitions t_1, t_4, t'_1 and t'_4), represents a sensor counting the number of axles of the train, i.e., the number of cars passing through that point.

The number of places used to represent the track depends on the required precision. On one hand, we assume that the Petri net is safe (such a condition will be imposed by the addition of appropriate monitor places), thus the number of places is mainly limited by the required safeness distance, i.e., we assume that the length of each segment is such that no more than one train can be contained within it at any given time instant. On the other hand, we take into account the presence of sensors and semaphores that are modeled by appropriate transitions as discussed above. Note that, even if these elements are only associated to one direction of flow, an equal number of uncontrollable and unobservable transitions should be added in the other direction so as to keep the structure shown in figure 3. Let us finally observe that an arbitrary large number of places may be included in the model of the track, and these places should be connected through uncontrollable and unobservable transitions. Nevertheless they would only change the modeling granularity, while no variation would occur in the controller design.

B. The railway station model

In this subsection we present the Petri net model of a three–tracks railway station that is sketched in figure 4.a (ignore place $p_{M,i}$ and all connected arcs), where double arrows have been used to denote self–loops. Note that we
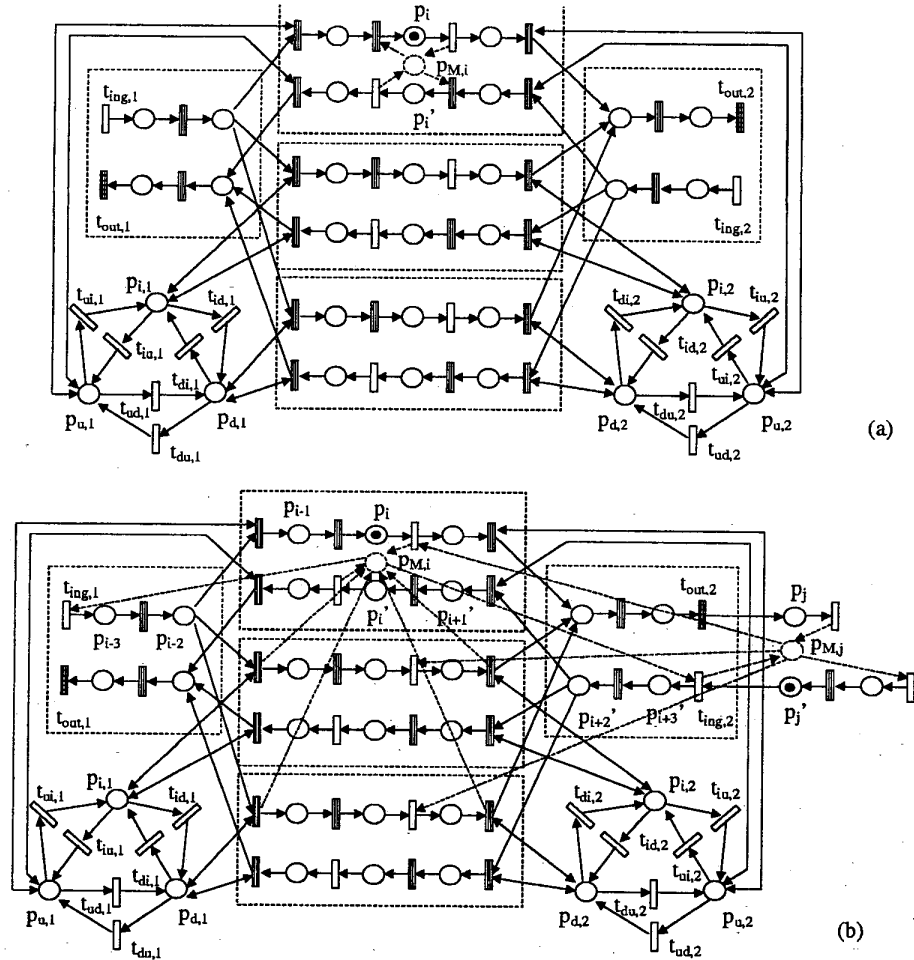
101

Fig. 4. The Petri net model of a three-tracks railway station with the monitor place relative to the constraint $m_i + m'_i \leq 1$ assuming that all transitions are controllable and observable (a) and taking into account the uncontrollability and unobservability of transitions (b).

can easily extend this model to a railway station with an arbitrary number of tracks.

As shown in figure 4.a, the station is composed of five different stretches shown within dashed boxes: three parallel stretches in the station and input tracks on both sides. The models of the stretches are similar to those already presented in the previous subsection.

The firing of controllable and observable transitions $t_{ing,1}$ and $t_{ing,2}$ represent the input of a train in the station, while the firing of uncontrollable and unobservable transitions $t_{out,1}$ and $t_{out,2}$ represent the output of a train from the station. Note that, as in the case of the track model, a controllable and observable transition is used to model a semaphore, while an observable but uncontrollable transition is used to model an axles counter.

The two subnets containing places $p_{u,1}, p_{d,1}, p_{i,1}$ and $p_{u,2}, p_{d,2}, p_{i,2}$, and transitions $t_{ui,1}, t_{iu,1}, t_{di,1}, t_{id,1}, t_{ud,1}, t_{du,1}$ and $t_{ui,2}, t_{iu,2}, t_{di,2}, t_{id,2}, t_{ud,2}, t_{du,2}$, respectively, model the points, i.e., when places $p_{u,1}$ and $p_{u,2}$ are

marked, trains may be directed to the up–track or may leave the up–track; on the contrary, when places $p_{d,1}$ $(p_{i,1})$ and $p_{d,2}$ $(p_{i,2})$ are marked, trains may be directed to the down (intermediate)–track or may leave the down (intermediate)–track.

Let us finally observe that as in the case of the track model, during simulation we associate a time delay to each transition so as to simulate time intervals required to get across a given segment of the station.

IV. THE CONTROLLER DESIGN FOR TRACKS AND STATIONS

As already discussed in the introduction, in this paper we shall deal with the problem of designing a Petri net supervisor for a railway network so as to ensure safeness and local liveness. In other words, the goal of the supervisory controller is that of guaranteeing that two trains may flow through the net in opposite directions without colliding, while prohibiting that blocking conditions may occur.

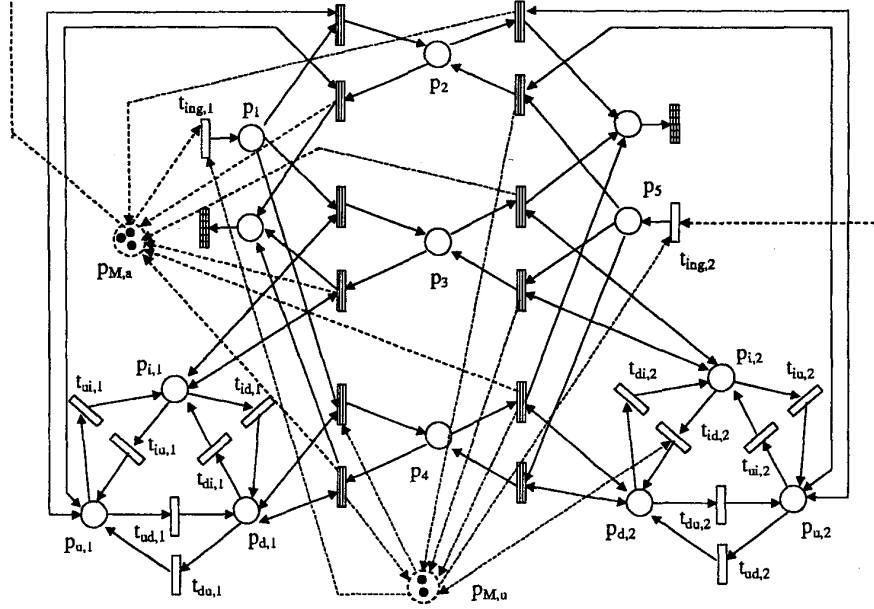Note that as the number of trains in the system increases,

Fig. 5. The reduced Petri net model of a three–tracks railway station and the monitor places relative to constraint 7.a and 7.u.

additional blocking conditions may occur. This issue has been discussed in [7].

To do this we first consider single modules and derive a controller for tracks and stations separately. In particular, we observe that GMECs may be satisfactorily applied when controlling tracks. On the contrary, this kind of constraints are too restrictive when controlling stations. We show in detail that in this latter case, safeness may be ensured by imposing appropriate logical constraints that also ensure local liveness.

GMECs have been firstly imposed so as to ensure safeness, i.e., to ensure that each couple of places corresponding to the same segment of a single–track (that may also belong to a station) are not marked simultaneously, and each place never contains more than one token at a time.

In accordance to the supervisory control theory briefly summarized in subsection II-B each constraint requires the introduction of a monitor place. Moreover, in the case of uncontrollable and/or unobservable transitions, constraints need to be appropriately transformed. As an example, a constraint of the form

$$m_i + m_i' \leq 1 \qquad (5)$$

relative to a given segment of a track within a station (see figure 4.a) ensures that places p_i and p_i' are not marked at the same time and each place never contains more than one token. If all transitions were controllable and observable, the monitor place $p_{M,i}$ enforcing (5) would have been that in figure 4.a. The presence of uncontrollable and unobservable transitions requires the transformation of (5) into

a more restrictive constraint

$$m_{i-3} + m_{i-2} + m_{i-1} + m_i + m_i' \\ + m_{i+1}' + m_{i+2}' + m_{i+3}' \leq 1. \qquad (6)$$

The corresponding monitor place is $p_{M,i}$ in figure 4.b. This result can be formally obtained with Moody's procedure [13] as discussed in subsection II-B.

Constraints of this kind ensure safeness. Nevertheless, as we now show, they are too restrictive when applied to places relative to tracks within the stations, while they ensure a satisfactory behaviour of the net when imposed to places modeling the intermediate tracks.

As an example, let us consider the monitor places $p_{M,i}$ and $p_{M,j}$ in figure 4.b, that have been introduced so as to enforce the GMECs $m_i + m_i' \leq 1$ and $m_j + m_j' \leq 1$, respectively, and applying Moody's transformation to make them controllable and observable. By looking at figure 4.b we can immediately observe that whenever a train is in the upper track (place p_i or p_i' is marked) no train can enter the station (transition $t_{ing,2}$ is not enabled because place $p_{M,i}$ is empty). Consider now the case in which a train is in the upper track going right (place p_i marked) and another one is arriving from the right (place p_j' marked) as shown in figure 4.b: a deadlock occurs. Such a case, as well as other analogous conditions for trains in the intermediate and lower tracks, demonstrate that GMECs do not guarantee a satisfactory behaviour of the net, resulting to be too restrictive.

A better solution to this problem consists in the introduction of a new set of constraints, some of whom may also involve the firing vector, that regulate the input of trains in the stations, and the points within them. In fact, whenever there is at least one empty stretch a reasonable

admission policy should let the incoming train enter the station, while the points should direct the incoming train towards the empty stretch.

To do this, let us consider the simplified Petri net model of the three–tracks railway station reported in figure 5, that is obtained from the previous one by simply grouping together some places. The set of constraints that have been proved to satisfy the desired requirement of safeness and local liveness is [3]:

$$
\left\{
\begin{array}{ll}
m_1 + m_2 + m_3 + m_4 + m_5 \leq 3 & (a) \\
q_{ing,1} + m_2 + m_{u,1} \leq 2 & (b) \\
q_{ing,1} + m_3 + m_{i,1} \leq 2 & (c) \\
q_{ing,1} + m_4 + m_{d,1} \leq 2 & (d) \\
q_{iu,1} + m_1 + m_2 \leq 2 & (e) \\
q_{du,1} + m_1 + m_2 \leq 2 & (f) \\
q_{ui,1} + m_1 + m_3 \leq 2 & (g) \\
q_{di,1} + m_1 + m_3 \leq 2 & (h) \\
q_{ud,1} + m_1 + m_4 \leq 2 & (i) \\
q_{id,1} + m_1 + m_4 \leq 2 & (l) \\
q_{ing,2} + m_2 + m_{u,2} \leq 2 & (m) \\
q_{ing,2} + m_3 + m_{i,2} \leq 2 & (n) \\
q_{ing,2} + m_4 + m_{d,2} \leq 2 & (o) \\
q_{iu,2} + m_2 + m_5 \leq 2 & (p) \\
q_{du,2} + m_2 + m_5 \leq 2 & (q) \\
q_{ui,2} + m_3 + m_5 \leq 2 & (r) \\
q_{di,2} + m_3 + m_5 \leq 2 & (s) \\
q_{ud,2} + m_4 + m_5 \leq 2 & (t) \\
q_{id,2} + m_4 + m_5 \leq 2 & (u)
\end{array}
\right.
\tag{7}
$$

where constraints (a)–(d) and (m)–(o) regulate the input of trains in the station, while the others regulate the points. Note that, apart from constraint (a) that is a GMEC, all the other constraints also involve the firing vector.

The inequality (a) implies that transitions $t_{ing,1}$ and $t_{ing,2}$ may only fire if no more than two trains are already contained in p_1, p_2, p_3, p_4 and p_5. In such a case we can be sure that there is always a free track in the station, where the last entered train may flow.

The inequalities (b)–(d) imply that transition $t_{ing,1}$ may fire whenever the following conditions hold. (b) Places p_2 and $p_{u,1}$ are not marked at the same time: in fact, if both these places are marked, and a new train enters the station, then the last entered train may only go to p_2, thus violating safeness constraint. (c) Places p_3 and $p_{i,1}$ are not marked contemporary (similar to the previous case). (d) Places p_4 and $p_{d,1}$ are not marked at the same time (similar to the previous cases).

Inequality (e) implies that transition $t_{iu,1}$ cannot fire if both places p_1 and p_2 are marked: the firing of $t_{in,1}$ would allow the train in p_1 to enter the up–track thus colliding with that one in p_2.

Analogous considerations lead to the formulation of the other constraints in (7).

As an example, in figure 5 we have reported the monitor places relative to constraints 7.a and 7.u ($p_{M,a}$ and $p_{M,u}$, respectively).

Let us finally observe that, even if constraints (b)–(d) and (m)–(o) involve the firing vector, no self–loop exists

between transitions $t_{ing,1}$ and $t_{ing,2}$ and the corresponding monitor places. This is due to the uncontrollability of all transitions entering places p_2, p_3 and p_4, and can be immediately observed by simply transforming the above constraints in accordance to the Moody's procedure.

V. CONCLUSIONS

In this paper we provided a modular representation of railway networks in terms of stations and tracks including sensors and semaphores. The resulting model is a P/T net whose transitions may be (un)controllable and/or (un)observable.

We have shown how collision avoidance constraints can be expressed as GMECs and the corresponding controller takes the form of a set of monitor places that can be computed using Moody's parametrization. This solution may lead to local deadlock: a better solution that guarantees safeness and local liveness, can be written in the form of constraints also involving the firing vector and the corresponding control structure takes the form of a monitor with self–loops.

REFERENCES

[1] J. Billington, "Many–sorted high–level nets," *Proc. of the Third Int. Workshop on Petri Nets and Performance Models*, IEEE Computer Society Press, 1989.

[2] G. Decknatel, and E. Schnieder, "Modelling railway systems with hybrid Petri nets," *Proc. 3rd Int. Conf. on Automation of Mixed Processes*, Reims, France, March 1998.

[3] F. Diana, "Supervisory control of railway networks using Petri nets," *Laurea Thesis*, Dep. of Electrical and Electronic Engeneering, University of Cagliari, Italy, 2000 (in italian).

[4] A. Di Febbraro, and A. Ferrara, "A new two–level model for multiclass freeway traffic," *IEEE Trans. on Vehicular Technology*, pp. 189–200, 1996.

[5] H.J. Genrich, "Predicate/transition nets," In W. Brauer, W. Reisig, and G. Rozenberg, editors, *Advances in Petri nets*, Lecture Notes in Computer Science, Vols. 254 and 255, Springer Verlag, 1987.

[6] A. Giua, F. DiCesare, M. Silva, "Generalized mutual exclusion constraints for nets with uncontrollable transitions", *Proc. IEEE Int. Conf. on Systems, Man & Cybernetics*, Chicago, USA, pp. 974–979, October 1992.

[7] A. Giua, C. Seatzu, "Supervisory Control of railway networks with Petri nets", *40th IEEE Conf. on Decision and Control*, Orlando, Florida, (submitted).

[8] L. E. Holloway, B. H. Krogh, A. Giua, "A survey of Petri net methods for controlled discrete event systems", *Discrete Event Systems: Theory and Applications*, vol. 7, pp. 151-190, 1997.

[9] K. Jensen, *Coloured Petri nets. Basic concepts, analysis methods and practical use. Volume 2: Analysis methods*, EATCS Monographs on Theoretical Computer Science, Springer Verlag, 1994.

[10] C.W. Janczura, "Modelling and analysis of railway network control logic using coloured Petri nets," *Ph.D. Thesis*, University of South Australia, August 1998.

[11] N.G. Levson and J.L. Stolzy, "Safety analysis using Petri nets," *IEEE Trans. on Software Engineering*, Vol. 13, N. 3, pp. 386–397, 1987.

[12] J.O. Moody, K. Yamalidou, M.D. Lemmon and P.J. Antsaklis, "Feedback control of Petri nets based on place invariants," *Automatica*, Vol. 32, N. 1, pp. 15–28, January 1996.

[13] J.O. Moody, P.J. Antsaklis, *Supervisory control of discrete event systems using Petri nets*, Kluwer Academic Publishers, 1998.

[14] T. Murata, "Petri nets: properties, analysis and applications," *Proc. IEEE*, Vol. Proc. 77, N. 4, pp. 541–580, April 1989.

[15] P.J. Ramadge and W.M. Wonham, "Control of discrete-event systems", *Proc. IEEE*, vol. 77, no. 1, pp. 81-98, January 1989.