DANKO KEZIĆ, Ph.D.
E-mail: danko.kezic@pfst.hr
IGOR VUJOVIĆ, M.Sc.
E-mail: igor.vujovic@pfst.hr
ANITA GUDELJ, M.Sc.
E-mail: anita.gudelj@pfst.hr
University of Split, Faculty of Maritime Studies
Zrinsko-Frankopanska 38, HR-21000 Split,
Republic of Croatia

Science in Traffic and Transportation
Original Scientific Paper
Accepted: June 16, 2007
Approved: July 12, 2007

PETRI NET APPROACH OF COLLISION PREVENTION SUPERVISOR DESIGN IN PORT TRANSPORT SYSTEM

ABSTRACT

Modern port terminals are equipped with various local transport systems, which have the main task to transport cargo between local storehouses and transport resources (ships, trains, trucks) in the fastest and most efficient way, and at the lowest possible cost. These local transport systems consist of fully automated transport units (AGV – automatic guided vehicle) which are controlled by the computer system. The port computer system controls the fully automated transport units in the way to avoid possible deadlocks and collisions between them. However, beside the fully automated local transport units, there are human operated transport units (fork-lift trucks, cranes etc.) which cross the path of the AGV from time to time. The collision of human operated transport unit and AGV is possible due to human inattention. To solve this problem, it is necessary to design a supervisory control system that coordinates and controls both human driven transport unit and AGV. In other words, the human-machine interactions need to be supervised. The supervising system can be realized in the way that the port terminal is divided into zones. Vehicle movements are supervised by a video system which detects the moving of particular vehicles as a discrete event. Based on detected events, dangerous moving of certain vehicles is blocked by the supervising system. The paper considers the design of collision prevention supervisor by using discrete event dynamic theory. The port terminal is modeled by using ordinary Petri nets. The design of collision prevention supervisor is carried out by using the P-invariant method. The verification of the supervisor is done by computer simulation.

KEYWORDS

Petri nets, AGV, collision prevention supervisor, human-machine interaction

1. INTRODUCTION

Recently, modern and highly sophisticated computer guided local transport vehicles are being used in modern port container terminals. These vehicles, called AGV (automatic guided vehicles), are autonomous and fully controlled by the distributed computer control system. Part of the computer control system which is responsible for lower level of automation is on the AGV, and the other part of the system for supervisory and scheduling tasks is situated in the main port control room. Beside the AGVs, the local transport system of the port terminal could also have human operated vehicles (fork-lift trucks, cranes etc.). The modern human operated vehicles (HOV) are semi-autonomous vehicles because they are partly controlled by humans. Thus, in general, both autonomous and semi-autonomous vehicles may simultaneously exist in a distributed system, where the former are fully controlled by the associated computer controllers, and the latter are partially controlled by the humans. It is possible that HOV cross the path of the AGV from time to time in the specific zones of the port terminals. This may lead to the collision of the vehicles, so it is necessary to coordinate the movement of the vehicles to disable undesired behavior. Vehicle movements are supervised by a video system which detects the moving of particular vehicles as a discrete event. Thus, a human operator at the HOV has to interact with computer control of the whole system through the computer network.

This paper proposes a supervisory framework so as to prevent abnormal operations being carried out either by humans or by computers. Figure 1 shows the proposed supervisory framework of the whole system. The controlled system (controlled process) can be described as a discrete event dynamic system in which movements of AGVs and HOVs generate a set of discrete events [1]. The visual system detects the movement of HOVs. A set of different sensors detects the movement of AGVs. The supervisor acquires these discrete events and advises the HOV human operator and computer system which controls AGV while issuing commands. The supervisor allows or disallows cer-

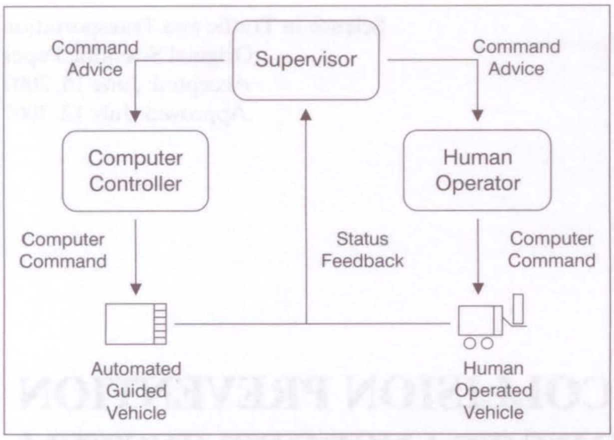

Figure 1 - Proposed supervisory framework for the system

tain human and computer commands so as to meet the requirements. The role of the supervisor is to interact with human operator, computer controller, and controlled system so that system meets the required specifications and to guarantee that undesirable executions do not occur.

This paper deals with the calculating of supervisor based on Petri nets. Most existing methods for supervisory design are based on automata models. However, this method often involve exhaustive search of overall system behavior and results in a state space explosion problem. PN approach is better, because PN modelling has more compact syntactical representation than the automata approach. The proposed supervisor is applicable to DES, but it is also applicable to the continuous-variable dynamic systems which can be viewed as DES at a higher level of abstraction.

The organization of this paper is as follows: Section 2 reviews basic concepts of PN, describes notations which are used throughout the paper and also describes PN-based system modelling; Section 3 describes PN based supervisor design using P-invariant method; A Case study - Collision prevention supervisor design in port system is given in the Section 4; Finally, Section 5 gives the conclusion.

2. PN – BASED SYSTEM MODELLING

A PN is identified as particular kind of bipartite directed graph populated by three types of objects: places, transitions and directed arcs connecting places and transitions. P-T Petri net is a 6-tuple [7]:

$$Q=(P,T,I,O,\Phi,\mathbf{m_0}) \quad (1)$$

where:

$P=\{P_1,P_2,\dots,P_m\}$ – set of places,

$T=\{t_1,t_2,\dots,t_n\}$ – set of transitions,

$P \cap T = \varnothing$,

$I: P\times T \rightarrow \{0,1\}$ – input function,

$O: T\times P \rightarrow \{0,1\}$ – output function,

$\Phi: (I,O) \rightarrow \{1,2,3,\dots\}$ – weight function,

$\mathbf{m_0}: P \rightarrow \{0,1,2,\dots\}$ – initial marking vector.

A simple P-T Petri net with 2 places and 2 transitions and its reachability tree are shown in Fig. 2. Places and transitions are the nodes of Petri net.

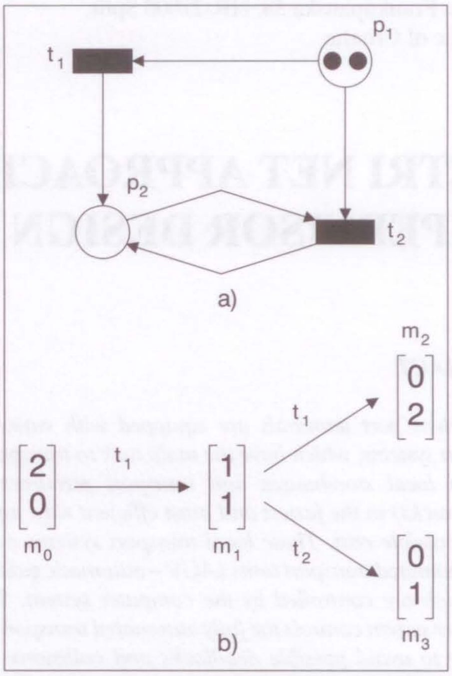

Figure 2 - P-T Petri net a) and its reachability tree b)

Transition $t \in T$ is enabled at marking $\mathbf{m}_{(mx1)}$ if and only if every input place connected to the transition t has at least one token, which can be denoted as « $t \in T$ is enabled if and only if $\forall p \in \bullet t, m(p)>0$ ($\bullet t$ is a set of input places to transition t, $m(p)$ denotes number of tokens in place p)». Transition t that meets the enabled condition is free to fire. When transition t fires, all of its input places lose a number of tokens, and all of its output places gain a number of tokens. The state of Petri net changes from state $\mathbf{m}_{(mx1)}$ to state $\mathbf{m}_{(mx1)}$ after firing transition t. This fact will be denoted as $\mathbf{m}[t \rightarrow \mathbf{m}'$ which reads «state $\mathbf{m}'$ can be reached from the state $\mathbf{m}$ after firing transition t». The state change equation can be expressed in following way:

$$\mathbf{m}' = \mathbf{m} + \mathbf{A}\mathbf{q}, \quad (2)$$

where:

$\mathbf{A} = [a_{i,j}]_{m\times n}$ – incidence matrix,

$\mathbf{q} = [q_{i,j}]_{m\times 1}$ – firing vector,

m – number of places in the Petri net,

n – number of transitions in the Petri net.

Some important PN properties include boundness (no capacity overflow), liveness, conservativeness (conservation of no consumable resources), and reversibility (cyclic behavior). A reachability tree displays every possible state that can occur in the Petri

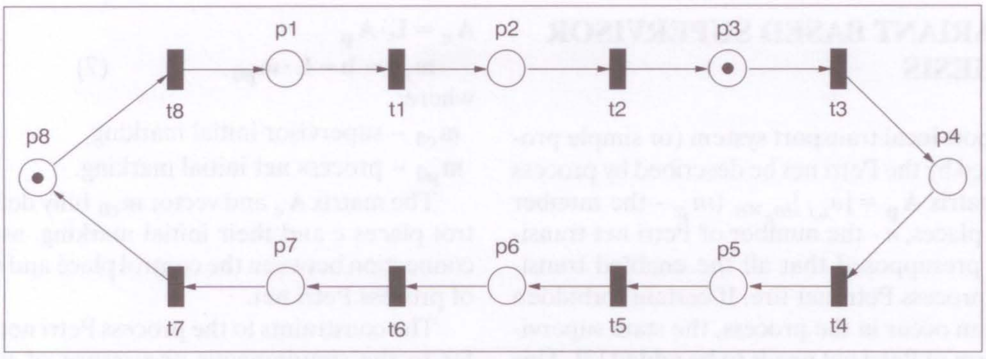

Figure 3 - Petri net model of cyclic loop of AGV path

net after firing all transitions [7]. The reachability tree of Petri net in Fig. 2a) is shown in Fig. 2b). From the reachability tree it is possible to see the main characteristics of Petri net, such as reachability, boundness, liveness, conflict, deadlock and reversibility. The concept of liveness is closely related to the complete absence of deadlock [4]. A PN is said to be live if, no matter what marking has been reached from the initial marking, it is possible to ultimately fire any transition of the net by progressing through some further firing sequences. A deadlock is the state when no firing in the Petri net is possible.

One of the structural properties of Petri nets i. e. properties that depend only on the topological structure of the Petri net and not on the net's initial marking, are the net invariants. Here we are interested in place invariants. Place invariants are sets of places whose token count remains constant for all possible markings. A single invariant is represented by an m--column vector $\mathbf{x}_{(m\times1)}$, where m is the number of places of the Petri net, whose non-zero entries correspond to the places that belong to the particular invariant and zero everywhere else. A place invariant (P-invariant) is defined as integer vector $\mathbf{x}$ that satisfies

$$\mathbf{x}^T\mathbf{m} = \mathbf{x}^T\mathbf{m_0} \tag{3}$$

where:

$\mathbf{m_0}$ – initial marking of Petri net

$\mathbf{m}$ – any subsequent marking of Petri net

The state $\mathbf{m_3}$ and $\mathbf{m_2}$ in Fig 2b) is the deadlock state, because there are no arcs from this state to the other states. The P-T Petri net is not safe because the maximum number of tokens in the places is more than one and it is not reversible because it is not possible to reach initial state $\mathbf{m_0}$ after firing transitions. The net in Fig 2 is not live. There are no place invariants in the net.

2.1 Modelling of AGV and HOV movements by using Petri net

It is possible to model the process of moving AGV vehicle by using cyclic Petri net. The predefined AGV path is divided into certain sections. Figure 3 shows a cyclic loop of the AGV path divided in 8 sections modelled by the P-T Petri net. Every section of the path is associated with one of the places $p_1 - p_8$. The place becomes marked if AGV enters the section. There are 2 AGVs in the sections 3 and 8 (places p_3 and p_8 are marked). The capacity of the places is 1 because only 1 AGV can exist in the particular section, so the net is safe. Transitions $t_1 - t_8$ denote AGV movement from one section to another. The existence of the AGV in the particular section can be observed by the sensor system. This information is observed by the supervisor. Then, the supervisor (which can be modelled by added control place c to the Petri net) can allow or disallow firing of certain transitions $t_1 - t_8$ and thus control the movements of the AGV. All transitions in the model are controllable, so that the supervisor can control every transition.

Modelling of HOV is more complicated. The human behavior can be modelled using the command - response concept [6]. As shown in Fig 4, each human operation is modelled as a task with a start transitions, end transitions, progressive place and completed place. Transitions drawn with dark symbols are events that are controllable by the human operator. Note that the start transition is a controllable event as "command" input, while the end transition is an uncontrollable event as "response" output. The supervisor can enable or disable only controllable transition.

To model the same cyclic path of the HOV as shown in Figure 3, every place in Figure 3 must be replaced by the places and transitions shown in Figure 4. Thus, the total number of places and transitions is 32 (16 places + 16 transitions). Half of the transitions are controllable by the supervisor.

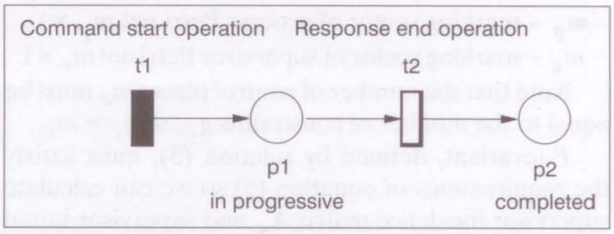

Figure 4 - Modelling of human behavior using the command-response concept

3. P-INVARIANT BASED SUPERVISOR SYNTHESIS

Let the port local transport system (or simple process) designed by the Petri net be described by process incidence matrix $\mathbf{A_p} = [a_{i,j}]_{m_p \times n}$ (m_p - the number of Petri net places, n - the number of Petri net transitions). It is presupposed that all the enabled transitions in the process Petri net fire. If certain forbidden states M_F can occur in the process, the state supervisor in the form of Petri net needs to be added [2]. This supervisor will prevent the occurrence of the states from M_F by applying the constraints on the set of all reachable states of the process Petri net. The supervisor comprises the control places that control the firing of the process. Petri net supervisor can be described by supervisor incidence matrix $\mathbf{A_c} = [a_{i,j}]_{m_c \times n}$ (m_c - the number of control places). If the supervisor control places are added to the process Petri net, a new composite Petri net is generated which cannot reach forbidden states [3]. The composite incidence matrix $\mathbf{A} = [a_{i,j}]_{(m_p+m_c)\times n}$ describes a topology of the composite Petri net, and it can be obtained if the rows of supervisor incidence matrix $\mathbf{A_c}$ are added to the process incident matrix $\mathbf{A_p}$. It can be presented as follows:

$$\mathbf{A} = \begin{bmatrix} \mathbf{A_p} \\ \mathbf{A_c} \end{bmatrix}.$$

Each supervisor control place defines the constraints to the set of reachable states of the process Petri net. The constraints to the process Petri net can be expressed in the form of linear non-equation [8]:

$$\sum_{i=1}^{m} l_i m(p_i) \le \beta, \qquad (4)$$

where:

$m(p_i)$ – number of tokens in place p_i,

l_i, β – integer constants.

The set of non-equation (4) can be transformed into matrix equation:

$$\mathbf{L} \cdot \mathbf{m_p} + \mathbf{m_c} = \mathbf{b}, \qquad (5)$$

where:

g_c – number of constraints

$\mathbf{L}$ – constraints matrix $g_c \times m_p$,

$\mathbf{b}$ – vector $m_c \times 1$,

$\mathbf{m_p}$ – marking vector of process Petri net $m_p \times 1$,

$\mathbf{m_c}$ – marking vector of supervisor Petri net $m_c \times 1$.

Note that the number of control places m_c must be equal to the number of constraints g_c, so $g_c = m_c$.

P-invariant, defined by relation (3), must satisfy the requirements of equation (5) so we can calculate supervisor incidence matrix $\mathbf{A_c}$ and supervisor initial marking $\mathbf{m_{c0}}$. The whole mathematical procedure for the equations (6) and (7) is explained in [5].

$$\mathbf{A_c} = \mathbf{L} \cdot \mathbf{A_p} \qquad (6)$$

$$\mathbf{m_{c0}} = \mathbf{b} - \mathbf{L} \cdot \mathbf{m_{p0}}, \qquad (7)$$

where:

$\mathbf{m_{c0}}$ – supervisor initial marking,

$\mathbf{m_{p0}}$ – process net initial marking.

The matrix $\mathbf{A_c}$ and vector $\mathbf{m_{c0}}$ fully define the control places c and their initial marking, as well as the connection between the control place and other places of process Petri net.

The constraints to the process Petri net can also refer to the simultaneous occurrence of two or more events. Assume that the constraint is of type:

$$\sum_{j=1}^{r} q_j \le r - 1 \qquad (8)$$

which indicates that not all transitions $q_1, q_2, \ldots, q_r$ can fire simultaneously. The enabling condition of Petri net theory suggests that the constraints can be transformed into an equivalent constraint by replacing each transition with the sum of its input places in the left-hand side of inequality (8) and by modifying the right-hand side appropriately. The constraint then becomes:

$$\sum_{j=1}^{r} m_j(p_1) + m_j(p_2) + \ldots + m_j(p_c) \le \sum_{j=1}^{r} c_j - 1 \qquad (9)$$

where $m_j(p_1) + m_j(p_2) + \ldots + m_j(p_c)$ are the input places of the transition t_j, while c_j denotes their number. The transformed constraint does not allow all of the input places of all transitions to be marked simultaneously, so it ensures that not all transitions can fire together and it contains only elements of marking vector.

There are several steps for the design of Petri net supervisor:

1. Design a process Petri net. From the net it is possible to define process incidence matrix
   $$A_p = [a_{i,j}]_{m_p \times n}$$
   and $\mathbf{m_{p0}}$ - process net initial marking
2. Define set of constraints of type (4) or (8) to the set of all reachable markings
3. Calculate supervisor incidence matrix $\mathbf{A_c}$ and $\mathbf{m_{c0}}$ from equations (6) and (7)
4. Design a composite Petri net from the composite incidence matrix
   $$\mathbf{A} = \begin{bmatrix} \mathbf{A_p} \\ \mathbf{A_c} \end{bmatrix}$$

4. COLLISION PREVENTION SUPERVISOR DESIGN – CASE STUDY

The method of designing collision prevention supervisor using Petri net, which is presented in sections 2 and 3 is shown in the case study example of port local transport system. The Figure 5 shows port transhipment system which contains 4 lanes, 2 for the automated guided vehicles and 2 for human operated vehicles. Lane $AGV1$ contains 2 automated guided vehicles and connects rail crane $Cr1$ with storehouse S_1. Lane $AGV2$ also contains 2 automated guided vehicles and connects rail crane with container ship crane $Cr3$. Lane $MAN1$ contains 2 human operated vehicles and connects storehouse S_2 with another container ship crane $Cr2$. Lane $MAN2$ contains 2 human operated vehicles and connects storehouse S_2 with the storehouse S_4. Cranes are responsible for loading and unloading containers from the rail and container ship to the vehicles. The crane $Cr1$ serves lanes $AGV1$ and $AGV2$, and these two lanes have to share this crane. Crane $Cr2$ serves lane $MAN1$, and crane $Cr3$ serves lane $AGV2$. All storehouses are equipped with automatic loading and unloading systems.

The port system described in Figure 5 contains 4 dangerous zones:

Zone 1) – Lanes $AGV1$ and $AGV2$ are very close to each other, and it is forbidden to have 2 automated guided vehicles simultaneously.

Zone 2, 3, 4) – Lanes $AGV2$, $MAN1$ and $MAN2$ intersect zones 2, 3, 4 and there is a possibility for the vehicles collision. To avoid this undesirable event, only one vehicle may exist in every particular zone at a time.

As mentioned above, vehicle movements are supervised by a video system which detects the movements of particular vehicles in zones 1-4. A video system sends status feedback as discrete events to the supervisor. Based on detected events, dangerous movements of certain vehicles are blocked by the supervising system (see Figure 1).

Design of Petri net supervisor begins with designing of the process Petri net (Petri net in Figure 6 without places $c1-c5$), and defining process incidence matrix $A_p=[a_{i,j}]_{m_p \times n}$ and process net initial marking $\mathbf{m_{p0}}$.

Next, it is possible to define a set of constraints of type (4) for the crossing zones 1-4 to calculate control places $c1-c5$. These constraints define that the maximal number of vehicles within zones 1-4 must not exceed one vehicle. These constraints are as follows:

Zone 1: $m(p_3)+m(p_4)\leq 1$ (10)

Zone 2: $m(p_5)+m(p_6)+m(p_7)+m(p_8)\leq 1$ (11)

Zone 3: $m(p_9)+m(p_{10})+m(p_{11})+m(p_{12})\leq 1$ (12)

Zone 4: $m(p_{13})+m(p_{14})+m(p_{15})+m(p_{16})\leq 1$ (13)

There is another constraint of type (8) which defines that it is possible to fire only one of the transitions t_1 or t_2 simultaneously (rail crane $Cr1$ can serve only one of the lanes $AGV1$ and $AGV2$ simultaneously). This fact can be written as constraint (14):

$$q_1+q_2\leq 1 \quad (14)$$

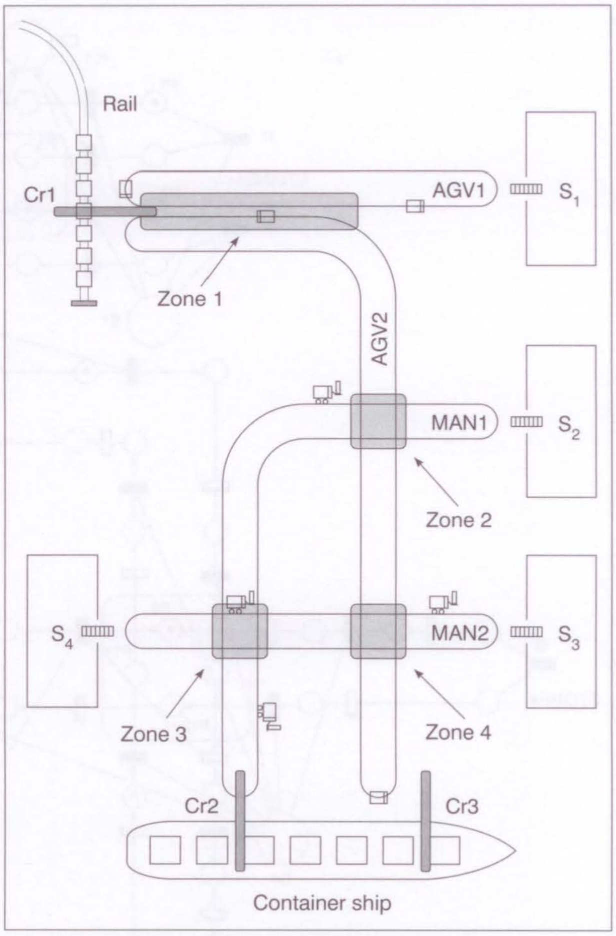

Figure 5 - Port transshipment transport system

Using (9) it is possible to convert (14) to:

$$m(p_1)+m(p_2)\leq 1 \quad (15)$$

From the non-equations (10), (11), (12), (13), (15) it is possible to calculate control places c_1-c_5. Figure 6 shows process Petri net (all transitions and small places) with 5 supervisor control places (big places c_1-c_5), which together form a fully controlled composite Petri net. As shown in Figure 6, all control places can block only controllable transitions. Marking of the places indicates position of the vehicles in Figure 5. Table 1 describes certain places and transitions. All other places p and transitions t describe the process of movements of vehicles in the port system.

5. CONCLUSION

This paper considers the design of collision prevention supervisor by using the discrete event dynamic theory and Petri net. The port local transport system which contains automatic guided and human operated vehicles is analyzed. To prevent a collision, the process of vehicle movements must be monitored by a video system and supervised by the collision prevention supervisor. The article proposes a suitable Petri net

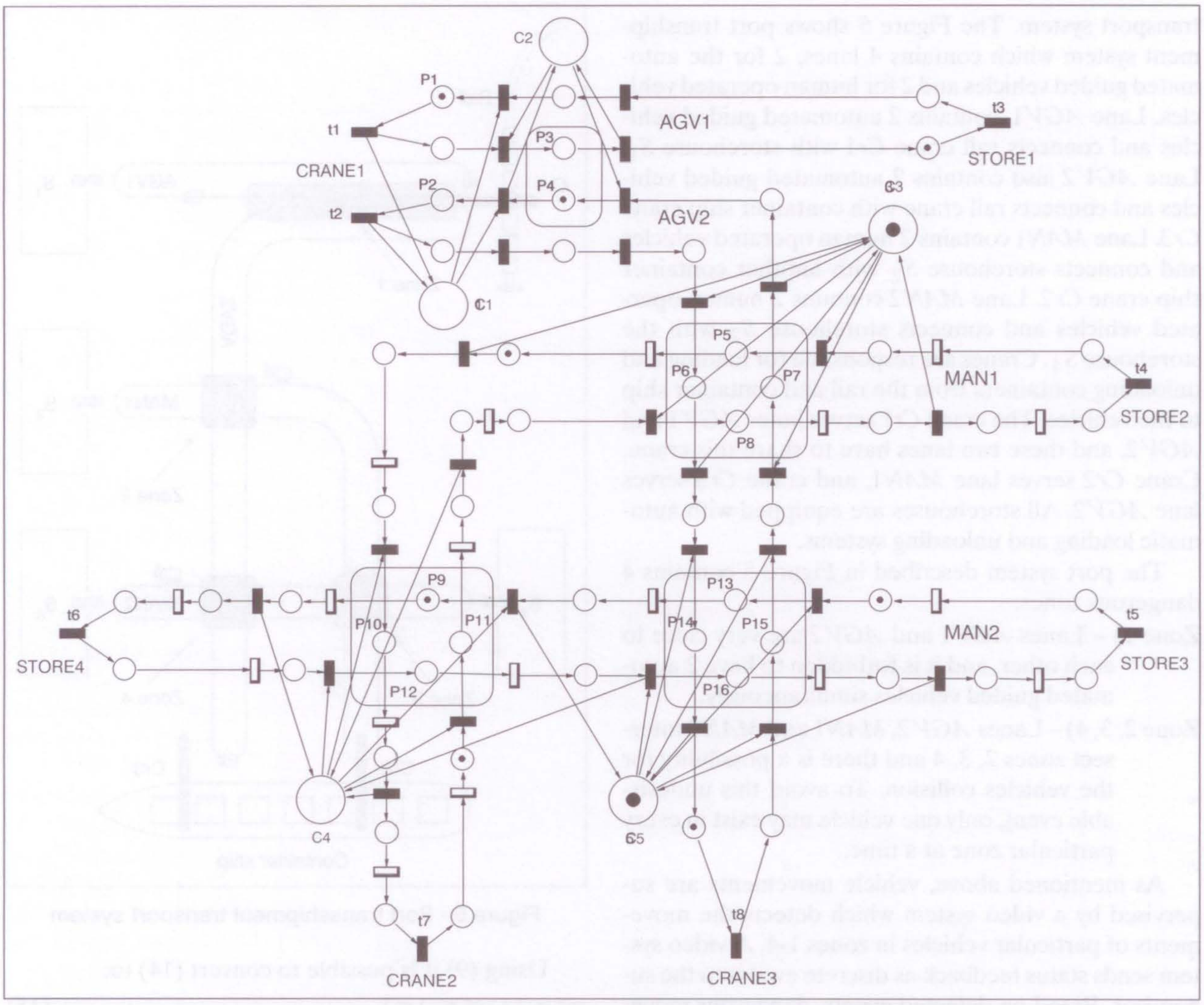

Figure 6 - Composite Petri net of port transshipment transport system.

Table 1 - Description of places and transitions

Place&Transition	Description	Place&Transition	Description
t_1 / t_2	Crane 1 loading or unloading AGV1/AGV2	p_3, p_4	States in zone 1
t_3, t_4, t_5, t_6	Loading or unloading vehicles in store 1, 2, 3, 4	p_5, p_6, p_7, p_8	States in zone 2
t_7, t_8	Crane 2, 3 loading or unloading MAN1, AGV2	$p_9, p_{10}, p_{11}, p_{12}$	States in zone 3
p_1, p_2	AGV1, AGV2 are waiting for crane 1	$p_{13}, p_{14}, p_{15}, p_{16}$	States in zone 3

model of automatic guided vehicles as well as human behavior Petri net model for controlling human operated vehicles which uses command – response concept. The mathematical framework and the necessary steps for generating collision–free Petri net is also given. To calculate the supervisor, it is necessary to define the set of constraints for all reachable markings of Petri net in the form of inequality. These constraints define the set of Petri net forbidden states, which enables the calculation of control places using the P-invariant method. The calculated control places are connected with the Petri net model of port system, and form the collision-free Petri net without the set of forbidden states. The proposed formal method generates the maximally permissive supervisor, which blocks a vehicles' movement only in the situation of immediate danger of collision. The proposed method is simple and appropriate for computer implementation and can be applied to the complex port transhipment system.

Dr. sc. **DANKO KEZIĆ**
E-mail: danko.kezic@pfst.hr
Mr. sc. **IGOR VUJOVIĆ**
E-mail: igor.vujovic@pfst.hr
Mr. sc. **ANITA GUDELJ**
E-mail: anita.gudelj@pfst.hr
Sveučilište u Splitu, Pomorski fakultet
Zrinsko-Frankopanska 38, 21000 Split, Republika Hrvatska

SAŽETAK

PRORAČUN NADZORNIKA ZA SPRJEČAVANJE SUDARA U LUČKOM TRANSPORTNOM SUSTAVU POMOĆU PETRIJEVIH MREŽA

Suvremeni lučki terminali opremljeni su različitim lokalnim transportnim sustavima kojima je osnovni zadatak izvršiti prebacivanje tereta između lokalnih skladišta i prijevoznih sredstava (brodova, vlakova ili kamiona) na što brži i efikasniji način uz što manje troškova. Takvi lokalni transportni sustavi se često sastoje od potpuno automatiziranih transportnih jedinica (primjerice AGV – automatic guided vehicle) kojima upravlja računalni sustav. Računalni sustav lučkog terminala brine se o upravljanju takvih potpuno automatiziranih jedinica na način da ne može doći do zastoja ili sudara između njih. Međutim, pored potpuno automatiziranih lokalnih transportnih jedinica, postoje i lokalne prijevozne transportne jedinice kojima upravlja čovjek (viličari, dizalice) i koji se ponekad križaju s putanjom kretanja AGV-a. Uslijed nepažnje čovjeka koji upravlja takvim lokalnim transportnim sustavom može doći do sudara. Da bi se riješio ovaj problem, potrebno je projektirati automatski nadzorni sustav koji koordinira rad čovjeka koji upravlja lokalnim transportnim vozilom i AGV-a. Drugim riječima, potrebno je nadzirati interakciju čovjek-stroj (human–robot interaction). Nadzorni sustav se može realizirati na način da se lučki terminal podijeli na određene zone. Kretanje svih vozila motri se video sustavom koji kretanje određenih vozila detektira kao diskretne događaje. Nadzorni sustav na temelju snimljenih događaja odlučuje blokirati određena opasna kretanja vozila. U članku se razmatra način proračuna nadzornika za sprječavanje sudara primjenom teorije sustava diskretnih događaja. Lučki terminal je modeliran pomoću običnih Petrijevih mreža. Proračun nadzornog sustava izvršen je primjenom P-invarijant metode. Verifikacija nadzornog sustava izvršit će se računalnom simulacijom.

KLJUČNE RIJEČI

Petrijeve mreže, AGV, nadzornik za sprječavanje sudara, interakcija čovjek-stroj

REFERENCES

[1] **Desrochers, A. A., Al-Jaar, R. Y.**: *Applications of Petri Nets in Manufacturing Systems*. IEEE Control Systems Society, IEEE press, 1995, New York.

[2] **Haoxun, C.**: *Control synthesis of Petri nets based on S-decreases*. Discrete Event Dynamic Systems, Kluwer Academic Publishers, vol. 10, no. 3, Netherlands 2000, pp. 233-249.

[3] **Harustak, M., Hruz, B.**: *Supervisory control of Discrete event systems and its solutions with the Petri net P-invariants*, IFAC Conf. on Control Systems Design (CSD' /2000), Bratislava, Slovak Republic 2000, pp. 390-394.

[4] **Kezić D., Perić N., Petrović I.**: *An Iterative Siphon Control Algorithm for Deadlock Prevention Based on Petri net*, Automatika, vol 47, no. 1-2/2006, pp. 19-30.

[5] **Kezić D., Antonić R., Račić N.**: *Automatic Supervisory System Synthesis for Port Cranes Collision Prevention by Using Petri Net*, Promet, vol 18, no. 3/2006, pp. 143-150.

[6] **Lee, J. S., Yhou, M. C.**: *An Application of Petri Nets to Supervisory Control for Human-Computer Interactive Systems*, Automatica, IEEE Transactions on Industrial electronics, vol. 22, no. 5, 2005, pp. 12201226.

[7] **Murata, T.**: *Petri Nets: Properties, Analysis and Applications*, in Proc. of the IEEE, 77(4), 1989, pp 541-580.

[8] **Yamalidou, K., Moody, J., Lemmon, M., Antsaklis, P.**: *Feedback Control of Petri Nets Based on Place Invarijants*. Automatica, vol. 32, no. 1, 1996, pp. 15-28.