DANKO KEZIĆ, D.Sc.
E-mail: danko.kezic@pfst.hr
RADOVAN ANTONIĆ, D.Sc.
E-mail: radovan.antonic@pfst.hr
NIKOLA RAČIĆ, M.Sc.
E-mail: nikola.racic@pfst.hr
University of Split, Faculty of Maritime Studies
Zrinsko-Frankopanska 38, HR-21000 Split,
Republic of Croatia

Section: Traffic Safety and Security
Preliminary Communication
Accepted: Dec. 15, 2005
Approved: Feb. 21, 2006

AUTOMATIC SUPERVISORY SYSTEM SYNTHESIS FOR PORT CRANES COLLISION PREVENTION BY USING PETRI NET

ABSTRACT

The article deals with the system of port cranes comprising two independently controlled cranes which are simultaneously engaged in the transhipment of cargo between a vessel and a railway wagon assembly alongside. The crane operator's error can lead to the collision of cranes. Therefore, it is necessary to install the supervisor in an automatic control system with the functions of continuous supervision of the process of crane movement, and blocking of commands that can lead to collision. The article shows the method of crane system modelling as a discrete event system by using P/T and P-timed Petri nets. There is a proposal of a formal mathematic method for calculation of the state supervisor by P-invariant method. The supervisor calculated in this way is maximally permissive supervisor. The efficiency of the supervisor is verified by a computer simulation.

KEY WORDS

collision prevention, motion control of cranes

1. INTRODUCTION

Endless demands for reducing port operations time for loading and unloading of vessels have caused the introduction of large port cranes which are able to load the cargo within the shortest time period. Every extra hour of unexpected extension of port operations incurs added costs, and therefore it is essential to design the system of cranes which can be simultaneously engaged in loading or unloading cargo. The purpose of this crane system is reliable and fast transhipment of cargo. However, apart from the reliability and fast operations, the safety must also be considered. This article considers the formal method of calculation of supervising system for the collision prevention of two cranes which are simultaneously engaged in ship cargo transhipment between the vessel and the railway wagons alongside the vessel.

Port cranes system can be viewed as a discrete events dynamic system (DEDS). The main characteristics of these systems are concurrency or parallelism, asynchronous operations, and event driven operation. In a DEDS many operations take place simultaneously and the evolution of system events is aperiodic. The competition of one operation may initiate more than one new operation. As a result of these dynamic characteristics, the conflict situations can occur when two or more processes require a common resource at the same time [1], [8], [9]. For example, when two port cranes require the same positions, the collision can occur. This state is forbidden or unacceptable and is usually the result of wrong system design [3]. To avoid this situation, the port cranes supervisory system must be applied.

The paper is organized as follows. Section 2 reviews the basics of supervisory control and Petri net properties and describes notations which are used throughout the paper. Section 3 defines formal method for calculating deadlock prevention supervisor by using Petri net. The method described in section 3 is shown in the example of calculation of the port cranes system collision prevention supervisor in section 4. Finally, in section 5, the supervisor is verified by using the computer simulation of cranes movement.

2. BASICS OF SUPERVISORY CONTROL AND PETRI NET

The basics of supervisory control can be briefly described as follows. The system of port cranes considered in this article can be viewed as a DEDS process *G* with the forbidden collision states. To prevent occurrence of collision states, the process must be supervised by the collision prevention supervisor C, which is connected with process G in closed loop (Fig. 1). The process generates a sequence of discrete events s and sends it to a supervisor C. The supervisor C then generates set of allowable events γ that are allowed to happen in the process G in the next step. The set γ depends on the sequence s and does not consist of the events that can lead a process to forbidden collision states in the next step.

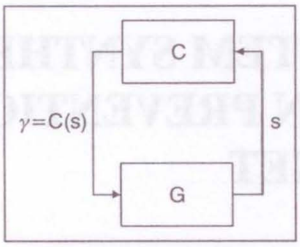

Figure 1 - Closed loop of the process and supervisor

In this article Place/Transition Petri net is used for the calculation of the supervisor C, and P-timed Petri net is used for supervisor verification.

The process G and supervisor C are designed by the Place/Transitions Petri net (P/T Petri net), a useful tool for describing discrete event dynamic systems. P/T Petri net is a 6-tuple [12]:

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

where:

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

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

$P \cap T = \emptyset$,

$I: P \times T \rightarrow \{0,1\}$ – 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 5 places and 5 transitions is shown in Fig. 2. Places and transitions are the nodes of Petri net.

A transition $t \in T$ is enabled at a marking $\mathbf{m}$ $iff\_\ \forall p \in \bullet t, m(p) > 0$ ($\bullet t$ is a set of input places to transition t). A transition t that meets the enabled condition is free to fire. When a 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 $\mathbf{m}$ to state $\mathbf{m}'$. This fact will be denoted as $\mathbf{m}[t > \mathbf{m}'$ and is described by:

$$\mathbf{m}' = \mathbf{m} + \mathbf{A}^T \sigma, \tag{2}$$

where:

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

σ – firing vector,

m – number of places in the Petri net,

n – number of transitions in the Petri net.

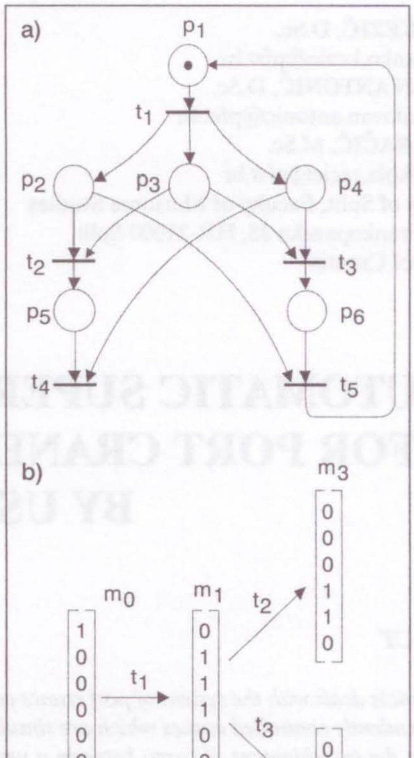

Figure 2 - P/T Petri net a) and its reachability tree b).

A reachability tree displays every possible state that can occur in the Petri net after firing all transitions [10]. 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, liveliness, conflict, deadlock and reversibility. A deadlock is the state when no firing in the Petri net is possible. The state $\mathbf{m_3}$ 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 safe because the maximum number of tokens in the places is one and is partially reversible because it is possible to reach initial state $\mathbf{m_0}$ after firing transitions $\{t_1, t_3, t_5\}$. One of the structural characteristics of P/T Petri net are P invariants. P – invariant corresponds to sets of places whose weighed token count remains a constant for all possible markings.

Timed Petri net [2] is an extension of Place/Transitions net, and is useful for performance evaluation of DEDS. In this article the P-timed Petri net is used. The P-timed Petri net have been defined with timing values which are rational numbers, in order to have periodical functioning. When a token is deposited in place p_i, this token must remain in this place at least for a time d_i. This token is said to be *unavailable* for this time. When the time d_i elapses, the token becomes *available*. Operating of P-timed Petri net can be described as follows.

At any time tm, the present marking $\mathbf{m}$ is the sum of markings $\mathbf{m_a}$ and $\mathbf{m_u}$, such that $\mathbf{m_a}$ is the marking made up of the available tokens and $\mathbf{m_u}$ is the marking made up of the unavailable tokens. A transitions is enabled for the marking $\mathbf{m} = \mathbf{m_a} + \mathbf{m_u}$ if it is enabled for the marking $\mathbf{m_a}$. Firing of transitions is carried out as for Place/Transition Petri net, by only removing available tokens from the input places. This firing has a zero duration. If a token is deposited in a place p_i during a firing carried out at instant tm_1, then this token is unavailable in the interval $[tm_1, tm_1 + d_i]$.

The calculation of the supervisor C for the process G designed by Place/Transition Petri nets poses a complex problem. There are two approaches. The first approach is to restrict the P/T Petri net model of the process G to the appropriate subclass of P/T Petri net for which it is possible to calculate a maximally permissive deadlock prevention supervisor. This approach is described in [4], [10] and [11]. The method proposed in the article is similar to the approach published in [5], [6] and [13] and it is grounded in the concept that the behaviour of the process designed by P/T Petri nets (process P/T Petri net) should be limited by addition of control places. This method also allows the synthesis of the maximally permissive collision prevention supervisors. The synthesis of the controller is simple and appropriate for computer implementation.

3. DEADLOCK PREVENTION SUPERVISOR SYNTHESIS

Let the process modelled by the P/T Petri net be described by process incidence matrix

$$\mathbf{A_p} = [a_{i,j}]_{m_p \times n}$$

(m_p - the number of P/T Petri net places, n - the number of P/T Petri net transitions). It is presupposed that all enabled transitions in the process P/T Petri net fire. If certain forbidden states M_F can occur in the process, the state supervisor in the form of P/T Petri net needs to be added. 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 P/T Petri net. The supervisor comprises the control places that control the firing of the process. P/T 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 P/T Petri net, a new composite P/T Petri net is generated which cannot reach forbidden states. The composite incidence matrix

$$A = [a_{i,j}]_{(m_p + m_c) \times n}$$

describes a topology of the composite P/T 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 marking of the process P/T Petri net. The constraints to the process P/T Petri net can be expressed in the form of linear inequation:

$$\sum_{i=1}^{m} l_i m(p_i) \geq \beta, \tag{3}$$

where:

$m(p_i)$ – markings of p_i,

l_i, β – the integer constants.

The inequation (3) can be transformed into a linear equation by adding a nonnegative integer slack variable $m(c)$, therefore it can be expressed as follows:

$$\sum_{i=1}^{m} l_i m(p_i) - m(c) = \beta, \tag{4}$$

where:

$m(c)$ – slack variable

In this case, the slack variable represents the number of tokens of a new place c which is added to the process P/T Petri net. The place c is called the control place and it belongs to the supervisor P/T Petri net. The control place c is connected by the arches to the process P/T Petri net, and with places p_i it forms a P-invariant. The places c assure that the sum of tokens in places p_i is always grater or the same as β.

The number of control places depends on the number of inequation that define the set of forbidden states M_F. The set of control places comprises the state supervisor that constrains the set of reachable markings to the process net. The size of the supervisor, as well as the size of the composite P/T Petri net is in proportion with the number of constraints type (4). The connection between control places and process P/T Petri net can be calculated by the application of *P*--invariant condition [12].

$$\mathbf{x^T A = 0}, \tag{5}$$

where $\mathbf{A}$ is a composite incidence matrix, and vector $\mathbf{x}$ is *P*-invariant.

The set of constraints type (3) can be expressed by the matrix inequation:

$$\mathbf{L \cdot m_p \geq b}, \tag{6}$$

where:

$\mathbf{L}$ – constraint matrix $n_c \times m$,

$\mathbf{b}$ – vector $n_c \times l$,

$\mathbf{m_p}$ – marking vector of process P/T Petri net $m \times l$,

n_c – number of constraints.

After the introduction of the slack vector $\mathbf{m_c}$, the inequation (6) can be transformed into equation:

$$\mathbf{L} \cdot \mathbf{m_p} - \mathbf{m_c} = \mathbf{b}, \tag{7}$$

where:

$\mathbf{m_c}$ – marking vector of supervisor P/T Petri net $n_c \times l$.

P – invariant defined by relation (4) must satisfy the requirements of equation (5), so we can write the following:

$$\mathbf{x^T} \cdot \mathbf{A} = [\mathbf{LI}] \cdot \begin{bmatrix} \mathbf{A_p} \\ -\mathbf{A_c} \end{bmatrix} = 0 \Leftrightarrow \mathbf{L} \cdot \mathbf{A_p} - \mathbf{A_c} = 0 \Leftrightarrow$$

$$\Leftrightarrow \mathbf{A_c} = \mathbf{L} \cdot \mathbf{A_p} \tag{8}$$

where:

$\mathbf{I}$ – unity matrix $n_c \times n_c$,

$\mathbf{A} = \begin{bmatrix} \mathbf{A_p} \\ -\mathbf{A_c} \end{bmatrix}$ – composite incident matrix $(m_p + m_c) \times n$.

If the incidence matrix of the process P/T Petri net $\mathbf{A_p}$ as well as the given process constraint $(n_c, \mathbf{L}, \mathbf{b})$ are known, from equation (8) follows the calculation of supervisor incidence matrix $\mathbf{A_c}$. The matrix $\mathbf{A_c}$ defines the arches which connect the control places of the supervisor to the transitions in the process P/T Petri net.

As the supervisor initial marking must satisfy the relation (7), it is possible to write:

$$\mathbf{L} \cdot \mathbf{m_{p0}} - \mathbf{m_{c0}} = \mathbf{b} \Leftrightarrow \mathbf{m_{c0}} = \mathbf{L} \cdot \mathbf{m_{p0}} - \mathbf{b}, \tag{9}$$

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 place c initial marking, as well as the connection between the control place and other places of process P/T Petri net.

4. CASE STUDY: COLLISION PREVENTION SUPERVISOR OF PORT CRANES SYSTEM

The method of supervisor calculation by P-invariant is shown in the example of calculation of collision prevention supervisor of port cranes system [7]. Figure 3 shows the system of two port cranes for the transhipment of cargo between the train and the vessel. The transhipment of cargo between 7 railway wagons and 7 cargo holds on-board ship is shown. Each of the cranes has the grab for cargo transhipment from cargo holds to the railway wagons on the tracks alongside the vessel. The cranes can move to the left towards the lower number holds and to the right towards the higher number holds. It can stand in place above the particular hold and be engaged in transhipment of bulk cargo, all depending on the command of the crane's operator. The command of the crane is performed in the cabin situated below the crane's boom. At the moment when the crane reaches the respective position (there are positions from 1 to 7), the sensor installed on the crane, registers the position. In that way, at any time, each crane "knows" its position, and how many positions it has to the furthest right or left respectively. Apart from that, there are sensors that register the reach of the end positions of cranes 1 and 2. Two cranes work the cargo simultaneously, and they are on the common tracks and if necessary, they can be engaged in the transhipment from more than one cargo hold (mostly 6 in this case), by moving to the left or right along the tracks in discrete steps. Thus, for example, the crane 1 can tranship the cargo from holds 1, 2, 3, 4, 5 and 6 (under condition that crane 2 is transhipping the cargo from hold 7). The similar applies to cranes 2. After all 7 railway wagons have been loaded, the train is moved and new railway wagons are brought to positions from 1 to 7. In order to make the formal supervisor state calculation for prevention of crane collision, it is necessary to design the system of port cranes as the discrete events system by using P/T Petri nets.

The Petri net in Figure 4 shows the model of one crane and describes the crane movement along the tracks to the right and left respectively in discrete steps.

The functions of the nodes in Fig. 4 are the following: p_1 - left control, p_2 - crane available, p_3 - right control, p_4 - crane moving a step to the left, p_5 - crane moving a step to the right, p_6 - the remaining number of positions to left end position, p_7 - the remaining number of positions to right end position, t_1 - the start of the movement to the left, t_2 - the end of the movement to the left, t_3 - the end of the movement to the right, t_4 - the start of the movement to the right.

It can be seen in Figure 4 that the crane is waiting in position No. 3, because $m(p_2) = 1$, and $m(p_4) = m(p_5) = 0$. The crane is two positions away from the left end position, and four positions from the right end

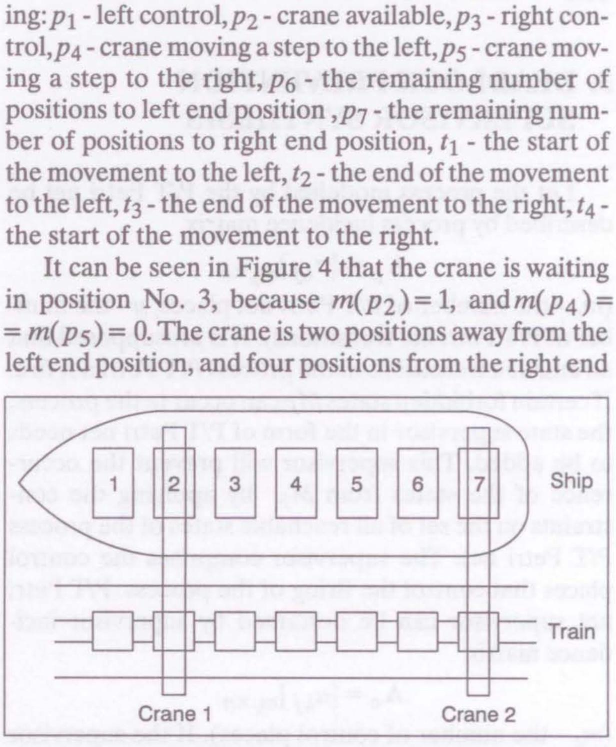

Figure 3 - The system of two port cranes for bulk cargo transhipment

position because $m(p_6) = 2$ and $m(p_7) = 4$. Also it is clear that the operator of the crane gave out the command for moving three steps to the right from the position 3, towards the position 6 because $m(p_3) = 3$.

The process P/T Petri net showing the behaviour of two cranes consists of 2 P/T Petri net presented in Figure 4. In such case, the process P/T Petri net of two cranes has the total of 14 places and 8 transitions. The Petri net topology of each crane is exactly the same, and there is a difference only in the initial marking of certain places. It should be considered that the number of tokens in the places of respective cranes is not equal, as the cranes are not at the same positions (one crane is waiting while the other is moving). This model does not take into account that the cranes have the common track, and so it is possible to give out the order, for example to move the crane 1 which is in position 2 right to position 5, and at the same time to move crane 2 from the position 5 to the left to position 2. In this case, the collision of cranes is inevitable. Therefore, the calculation of maximally permissive collision prevention state supervisor must be done. The supervisor consists of one or more control places which have to be added to the existing process P/T Petri net. The control places are connected by arches to the transitions of the process P/T Petri net of two cranes. The function of the supervisor is to block the transitions firing selectively i. e. only those transitions that may cause the forbidden state of crane collision.

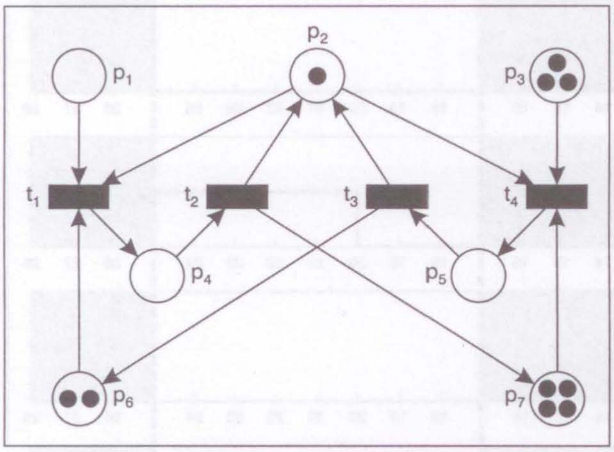

Figure 4 - Petri net model of one crane

Figure 5 shows a process P/T Petri net comprising crane 1 and crane 2, (nodes p_1-p_{14}, t_1-t_8). The calculation of the control place c_1 for collision prevention of crane 1 and crane 2 starts with the constraint $m(p_7)+m(p_{13}) \geq 7$ which can be expressed in the form of (4) as $m(p_7)+m(p_{13})-m(c_1) = 7$. Supervisor incidence matrix $\mathbf{A_c}$ which defines the connection of control place c_1 with a process P/T Petri net, can be calculated from (8):

$$A_c = L \cdot A_p = [0\,1\,0 -1 -1\,0\,1\,0] \tag{10}$$

where:

$$\mathbf{A_p} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 \end{bmatrix}$$, process P/T Petri net incidence matrix,

$\mathbf{L} = [0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,1\,0]$, a constraint matrix.

By matrices $\mathbf{A_p}$ and $\mathbf{A_c}$ it is possible to calculate the composite P/T Petri net incidence matrix comprising of a process P/T Petri net with an added control place c_1. Control place initial marking can be calculated from (9):

$$\mathbf{m_{c0}} = \mathbf{L} \cdot \mathbf{m_{p0}} - \mathbf{b} = 1 \tag{11}$$

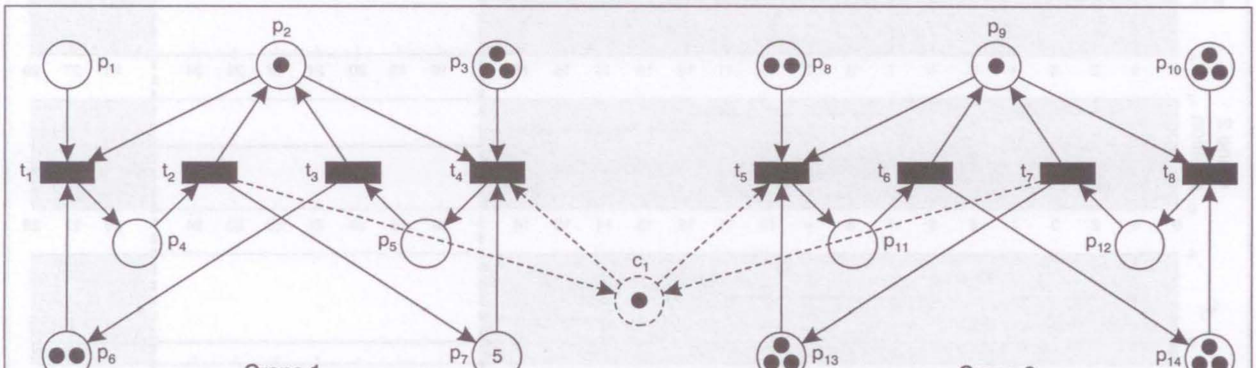

Figure 5 - Two cranes - P/T Petri net with a control place c_1

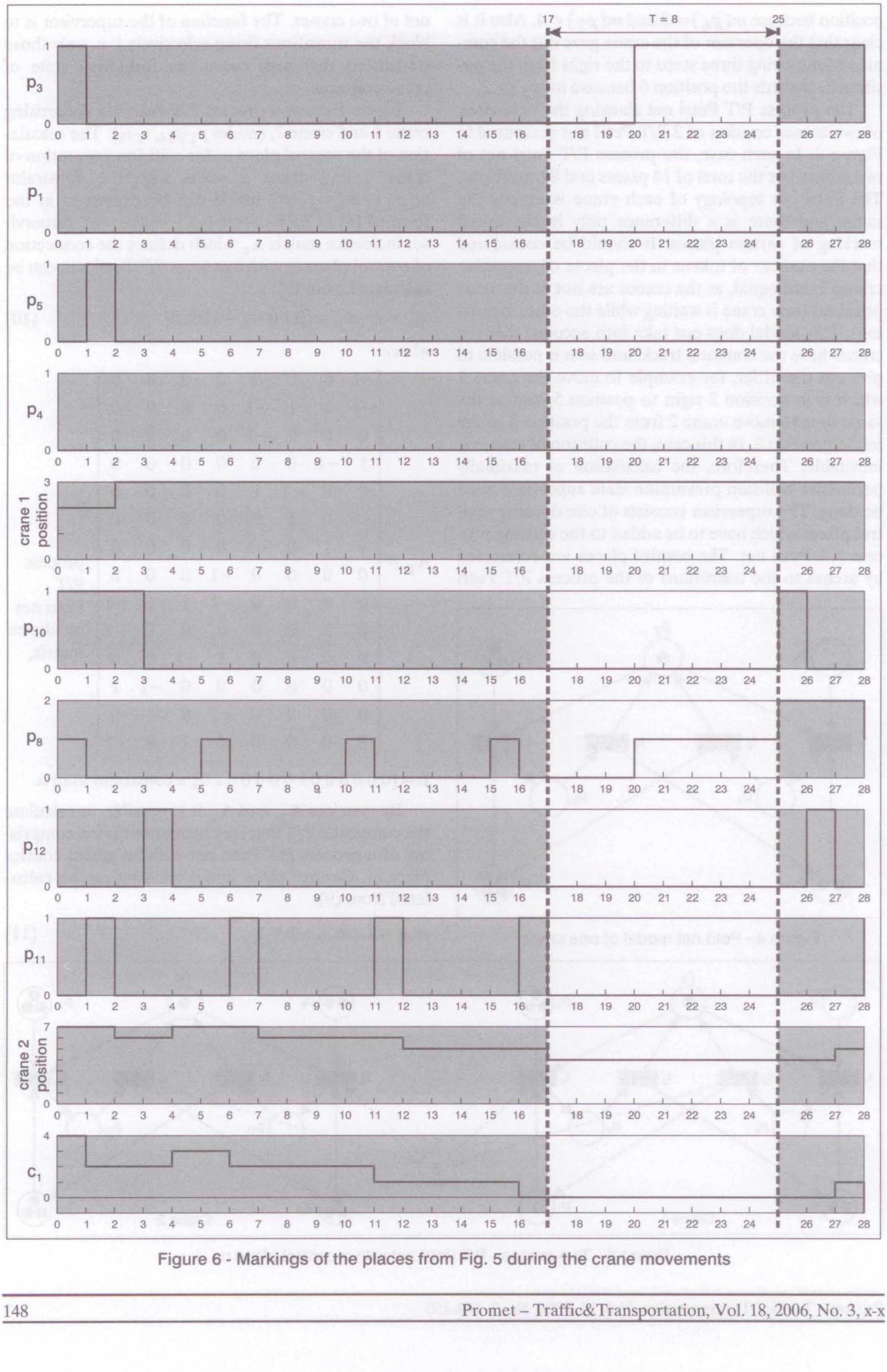

Figure 6 - Markings of the places from Fig. 5 during the crane movements

where:

$m_{p0} = [0\,1\,3\,0\,0\,0\,5\,2\,1\,0\,0\,0\,3\,3]$, initial marking of process P/T Petri net

b = 7, constraint vector

Figure 5 shows a composite P/T Petri net. It can be seen that a control place c_1 has the function of securing that the total sum of tokens in places p_7, p_{13} and c_1 remains always the same or larger than 7, which is the condition for the collision prevention of two cranes. The number of tokens at control place c_1 shows the current number of free positions between the two cranes. If the cranes happen to be next to each other, there is no free position between them, the control place c_1 does not have a token, and the transitions t_4 and t_5 are blocked, which prevents further movement of the crane 1 to he right and crane 2 to the left, regardless of commands that are given by the crane's operator. All the other operator's commands that cannot lead to a collision are not blocked by the control place. Therefore this supervisor is the maximally permissive. Real automatic control system for collision prevention can be easily done from the composite P/T Petri net in Fig. 5.

5. VERIFICATION OF SUPERVISOR USING P-TIMED PETRI NET COMPUTER SIMULATIONS

Verification of collision prevention supervisor is done by the "Petri . Net Simulator" software for simulation P/T, P-timed or T- timed Petri nets. This application is used for drawing and simulation Petri net in Fig. 5. Every place in the net is defined by the timing value of 1 minute, so duration of every operation is exactly 1 minute. Duration of the whole simulation is 28 minutes. Figure 6 shows dynamics evolution of the places marking of the net in Fig. 5 during the crane movements. The functions of the graphs from top down in Fig. 6 are the following: p_3 - crane 1 right control, p_1 - crane 1 left control, p_5 - crane 1 is moving a step to the right, p_4 - crane 1 is moving a step to the left, crane 1 position, p_{10} - crane 2 right control, p_8 - crane 2 left control, p_{12} - crane 1 is moving a step to the right, p_{11} - crane 2 is moving a step to the left, crane 2 position, c - marking of the control places.

At the start of simulation, crane 1 is at position No. 2, and crane 2 is at position No. 7. The crane 1 operator gives command to move crane 1 a step to the right at the start and in 20th minute. The crane 2 operator gives command to move crane 2 a step to the left at the start and in the 5th, 10th, 15th, 20th, 25th minute, and a step to the right in 3rd and 25th minute. The collision prevention supervisor, which controls the crane movements, allows executions of all commands from operators, except the commands from 17th to 25th minute, because in this period the cranes are too close and the wrong commands from each of the operator may lead to crane collision. Note that marking of control place in this dangerous period of 8 minutes is zero, which means that the firing of transitions t_4 and t_5 are blocked and the commands from operators cannot be executed.

5. CONCLUSION

This paper presents one of the methods of calculating collision prevention supervisor in the crane system for transhipment of cargo between a ship and a railway wagon assembly alongside. A collision in the port cranes systems can occur during the crane movement as the consequence of the improper control of port cranes, while they are trying to reach the same position for cargo transhipment. To prevent collision, the process of moving must be supervised by the collision prevention supervisor. In this paper P/T Petri nets have been used as a tool for supervisor calculation, and a P-timed Petri net for supervisor verification. The paper proposes the method for creating a suitable P/T Petri net model of port cranes system, as well as the algorithm for generating collision–free P/T Petri net. To calculate the supervisor, it is necessary to define the set of inequality which defines the P/T Petri net forbidden states. After that, it is possible to calculate control places using the P-invariant method. The control places are connected with the P/T Petri net model of port cranes system, and form collision-free P/T Petri net without the set of forbidden states. The proposed formal method generates the maximally permissive supervisor, which blocks a crane 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 discrete event system.

Dr. sc. **DANKO KEZIĆ**
E-mail: danko.kezic@pfst.hr
Dr. sc. **RADOVAN ANTONIĆ**
E-mail: radovan.antonic@pfst.hr
Mr. sc. **NIKOLA RAČIĆ**
E-mail: nikola.racic@pfst.hr
Sveučilište u Splitu, Pomorski fakultet
Zrinsko-Frankopanska 38, HR-21000 Split,
Republika Hrvatska

SAŽETAK

SINTEZA AUTOMATSKOG NADZORNOG SUSTAVA ZA SPRJEČAVANJE SUDARA LUČKIH DIZALICA KORIŠTENJEM PETRIJEVIH MREŽA

U članku se razmatra sustav lučkih dizalica koji se sastoji od dvije zasebno upravljane dizalice koje istovremeno obavlja-

ju prekrcaj tereta između broda i kompozicije vlaka pozicionirane pored broda. Uslijed greške operatera koji upravljaju dizalicama, moguće doći do sudara dizalica. Radi toga je potrebno ugraditi nadzornik u sustav automatskog upravljanja koji ima zadatak kontinuirano nadzirati proces pomicanja dizalica, te blokirati komande koje mogu dovesti do sudara. U članku je prikazan način modeliranja sustava dizalica kao sustava s diskretnim događajima pomoću P/T i P-vremenskih Petrijevih mreža. Predložena je formalna matematička metoda za proračun nadzornika stanja metodom P-invarijanti. Nadzornik koji je proračunat na ovaj način je najviše dozvoljavajući. Djelotvornost nadzornika verificirana je kompjuterskom simulacijom.

KLJUČNE RIJEČI

sprječavanje sudara, kontrola pomicanja dizalica

LITERATURE

[1] **Barkaoui, K., Abdallah, J.**: *Deadlock Avoidance in FMS Based on Structural theory of Petri Nets*. IEEE Symposium on Emerging Technologies and Factory Automation, 1995, pp 499-510.

[2] **David, R., Alla, H.**: *Discrete, Continuous, and Hybrid Petri Nets*, Springer, 2005.

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

[4] **Giua, A., DiCesare, F., Silva, M.**: *Petri Net Supervisors for Generalized Mutual Exclusion Constraints*. Proc. IFAC 12th Triennial World Congress, Vol. 1, Sydney 1993, pp. 267-270.

[5] **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.

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

[7] **Kezić D., Antonić R., Munitić A.**: *Calculation of Collision Prevention Supervisor of Port Cranes System by using Petri nets*, Proc. of 13th International Conference on Electrical Drives and Power Electronics, Dubrovnik, Croatia, 26-28 September 2005, CD Proceedings EDPE 2005, ISBN 953-6037-43-2, UDK 621.31 (063) (048), 621.38.022 (063) (048), pp. E05-46-1, E05-46-10.

[8] **Kezić, D., Perić, N., Petrović, I.**: *A Petri Net Approach of Deadlock Prevention in Batch Processes in Case of Failure*. in Proc. of Int. Symposium of Industrial Electronics, L'Aquila, Italia, 2002, pp. 205-210.

[9] **Kezić D., Perić N., Petrović I.**: *Deadlock Prevention in Manufacturing Systems Based on Petri Nets*. Proc. of 11th International Power Electronics and Motion Control Conference, Vol. 4, Riga, Latvia, 2-4 September 2004, pp. 320-327.

[10] **Li, Y., Wonham, W. M.**: *Control of vector discrete systems I – the base model*. IEEE Trans. on Automatic Control, Vol. 38, No. 8, 1993, pp. 1214-1227.

[11] **Li, Y., Wonham, W. M.**: *Control of vector discrete systems II – controller synthesis*. IEEE Trans. on Automatic Control, Vol. 39, No. 3, 1994, pp. 512-531.

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

[13] **Yamalidou, K., Moody, J., Lemmon, M., Antsaklis, P.**: *Feedback Control of Petri Nets Based on Place Invariants*. Automatica, Vol. 32, No. 1, 1996, pp. 15-28.