Qatar Univ. Sci. J. (1995), 15 (2) : 307 - 318

MODELLING AND SIMULATION OF FLEXIBLE ASSEMBLY SYSTEMS : A PETRI NET APPROACH

By

Zbigniew Banaszak* and Mowfak H. Abdul-Hussin**

* Department of Robotics and Software Engineering, Technical University of Zielona Gora, Poland.

** Mechanical Engineering University of Technology, Baghdad, Iraq

النموذج والمحاكاة لأنظمة التجميع المرنة : بطريقة شبكة بتري

أربكنيف بناشـك و موفق حسن عبد الحسين

أن هـذا البحـث يوضـح أسـلوباً عنـد تركيب نـوع مـن شبـكة المُقطع الأنتـقالي / الموضـع (Place / Transition net) للنماذج ، والمحـاكـاة (Simulation) - (وذلك لاستحـداث صـورة مشـابهة للحـالـة الحقيقيـة عن سير العمليـات الصنـاعية بـأسلوب شبكـات بتري بهدف بنـاء بـرمجيـات الحاسبـة الألكترونية مستقبـلاً بلغات عالية المستـوى مثل لغة باسكال ، سي ، أو بـرولوج (Pascal , C , Prolog) على ضوء الخوارزمية الجديدة الموجودة في هذا البحث .

وكـذلك السيطرة على عمليـات تحدث في أنظمة التجميع المرنـة . إن وضع الهدف الرئيسي في المشكلـة المصطنعـة من الحالات التي تسمح في تصميم وتـركيب الشبكـة من نماذج سيطـرة ذاتيـة الحركـة وقـابلـة للتطبيق، وهذا يعني ان الشبكـة تحتوي على سيطرات مسلم بها لأنسيـابية تدفق العمليات الصنـاعية . على اعتبار أن عمليـات الانتاج التي تنقسم إلى مجموعة النقل (مثل الـروبوتات الصنـاعية ، والأحزمة النـاقلة) وعمليات التجميع بكل عملية تحتاج إلى موارد (من المواد الأولية ، والمواد النصف مصنعة) ذاتية التجهيز ، مثل ذلك قطع العمل الموجـودة في وحـدات الـخزن المؤقتـة للماكنـة (Buffers) وكـذلك قطـع العمل للروبوتـات الصناعية . هذا الأسلـوب المقترح يطبق في نظام اتخاذ القرار الذي سبق ان تـم تحديد الخطوط العامة لتصميمه وعملياته .

Key Words : Modelling; Petri Nets; Control system synthesis; Control flow; Assembly system; Computer - Aided Process Planning.

ABSTRACT

This paper presents an approach towards constructing a class of Place/Transition nets for modelling, simulation and control of processes occurring in the Flexible Assembly Systems. Its objective lies in the formal statement of the conditions allowing to design the nets consisting of viable discrete control logic models, i.e. nets encompassing admissible control of the processes flow. We consider batch production processes which can be decomposed into a set of transportation and assembly operations. For each operation, required resources are identified like workpiece buffers and industrial robots. The approach proposed is applied to a decision support system whose design and operation is outlined.

1. Introduction

In recent years, the research on control algorithms and the design of strategies and programming languages based on the General Net Theory have had an increasing importance [1, 2, 3, 4].

In this paper, as the processes to be controlled the class of concurrent, pipeline-like flowing processes is considered. Processes are specified by production routes and are performed on a system consisting of a finite set of components. Each asynchronously acting process is described as a partially ordered set of operations running in the time on relevant system components. It is assumed that components can be shared among different, flowing processes and that process operations are executed asynchronously.

The problem studied is concerned with finding out a synchronization mechanism useful for the automatic design of net models reflecting all admissible, i.e. deadlock and buffer overflow free, control. The amount of the components necessary to execute the processes is considered, as a preassumed constraint.

This paper is organized as follows. Section 2 we recall the formalism employed, as well as introduce new definition to aid futher considerations. In section 3, we state the main problem. In section 4, we introduce two standard forms of processes specifications as well as present sufficient conditions for the synthesis of an algorithm aimed at the simulation models design. Besides the presentation of the algorithm an illustrative example of its operation is given. Concluding remarks are finally given in the last section.

2. Basic Notations and Definitions

A class of simple, and safe Place / Transition nets is considered.

Definition 1

We define a simple and safe place / transition net (PT-net, for short) by a quadruple $PN = (P, T, E, M_0)$, where :

P and T are finited sets of places and transitions respectively, such that $P \cup T \neq \phi$, $P \cap T = \phi$,

$E \subset (P \times T) \cup (T \times P)$ is a flow relation such that $\mathrm{dom}(E) \cup \mathrm{cod}(E) = P \cup T$,

$M_0 : P \rightarrow (0,1)$ is an initial marking .

Graphically, the PT-net is represented by a diagram having two types of nodes; places, represented by circles, and transitions, represented by bars. The elements of the flow relation E are pictorially shown as arcs connecting different types of nodes.

In order to describe a net structure, the following representation is considered.

Definition 2.

Let PN = (P,T,E,Mo), ||P|| = m, ||T|| = m be a PT-net. A pair $C = (C^+, C^-)$ is said to be the incidence matrix of PN, where : $C^+ = [c^+_{ij}]_{m \times n}$, $C^- = [c^-_{ij}]_{m \times n}$. We further define the backward incidence matrix : , and the forward incidence matrix :

$$C^+ = c^+_{ij} = \begin{cases} 1 & \text{if } p = t^{\cdot} \\ 0 & \text{otherwise} \end{cases}, \quad C^- = c^-_{ij} = \begin{cases} 1 & \text{if } p = {}^{\cdot}t \\ 0 & \text{otherwise} \end{cases}$$

While $\cdot t = \{ p \mid (p, t) \in E \}$ ($t \cdot = \{ p \mid (t,p) \in E \}$) is called a set of the input (respectively output) places of the transition t .

Moreover, PN will be described either by the pair $PN = (C, M_0)$ or by quadruple $PN = (P, T, E, M_0)$. Also, we write ; $C[i] = (C^+[i], C^-[i])$ for the i-th row of matrix C as well as $C^+[i, j]$ for the j-th column in the i-th row of matrix C^+ (resp. C^-).

The dynamic properties of a PT-net are represented by the position and movement of tokens (indicated by black dots) in the places of the net . The arrangement of tokens in a Petri Net defines the state of the net and is called its marking. Every net is provided with an initial marking M_0. The capacity function K defines the maximum number of tokens in each place, in any permissible marking of the net. The marking may change as a result of the firing of a transition by the following execution rule :

1) A transition is enabled (or able to be fired) if and only if each of its input places has at least one token.
2) Any enabled transition may be chosen to fire.
3) When a transition fires : i) a token is removed from each of its input places, and ii) a token is deposited into each of its output places.

Definition 3 :

A transition $t \in T$ is enabled at marking M if $(\forall p \in \cdot t)(M(p) = 1)$ and $(\forall p \in t \cdot \setminus t \cdot \cap \cdot t)$ $(M(p) = 0)$.

The occurrence (firing) of a transition $t \in T$ can fire by removing a token from each input place and putting a token in each output place. This results in a new marking M', and we write : M [t > M'.

Definition 4 :

The next-state function $\delta' \; \{0,1\}^n \times T \rightarrow \{0,1\}^n$ for a Petri net PN = (C, M_0) is defined for each M and t_i such that t_i is enabled at M as follows :

$\delta' (M, t_i) = M + U_{t_i} \; (C^+ - C^-)$ where U_{t_i} is the unit m-vector which is zero everywhere except at the i-th component.

Marking of a place represents the holding of a condition. The firing of a transition represents the occurrence of an event, which ends a set of holdings and begins a set of new holdings. Because of space limitations, the definitions of the extended next-state function δ', of the firing sequence σ, and of the reachability set of marking R (C,M_0) are omitted. They can be found in [5,6].

Now, let us recall the definition of the net's liveness property playing the essential role in further considerations.

Definition 5 :

A PN = (C, M_0) is a live PT-net if ($\forall$ $t \in T$) ($\forall M \in R(C,M)$) ($\exists M' \in R(C,M)$) (t is enabled at M').

It should be noted that the property of Definition 5 implies the absence of deadlocks in the system modelled.

In addtion to the above mentioned well - known definitions, let us introduce two new concepts : concerning the so called open queueing PT-nets (OQPT nets, for short), and a process performance matrix (PPM) .

Definiton 6 :

Consider PN = (P, T, E, M_o) being a PT - net. If the following conditions hold it is said to be a pipeline place/Transition net (PPT-net, for short). The conditions are :

(i) $T = (t_i \mid i = \overline{1,m})$, $P = p^I \cup p^{II}$, $p^I \cap p^{II} = \phi$,

(ii) $P^I = \{ p_i \mid i = \overline{1, m-1} \}$,

$(\forall p_i \in P^I)$ $(\exists ! t_i \in T)$ $(\exists ! t_{i+1} \in T)$ $(\cdot p_i = \{t_i\}$ & $p_i \cdot = \{ t_{i+1} \})$,

(iii) $(\forall p_i \in p^{II})(\exists t_j, t_{j+1} \in T)(t_j \in \cdot p_i$ & $t_{j+1} \in p_i \cdot)$,

$(\forall t_i, t_{i+1} \in T)(\exists i p_j \in p^{II})(p_j \in t_i \cdot$ & $p_j \in \cdot t_{i+1})$,

(iv) M_o is (1 x n) vector, where n = || P || such that ($\forall$ $p \in p$) ($M_o(p) = o$),

Where $P \cdot = \{ t \mid (p,t) \in E \}$ ($\cdot p = \{ t \mid (t,p) \in E \}$) - is a set of output (resp. input) transitions of a place p.

Assumption 1 .

Let PN = (P, T, E, M_O) consisting of a set

$\{ K_{PN} = (K_P, K_T, K_E, K_{M_O}) \mid K = \overline{1, v} \}$ of pipeline Place / Transition net (PPT - nets) , such that the following conditions hold :

(i) $P = \bigsqcup_{i=1,v} i_p$, $P = P^I \cup P^I$, $P^I \cap P^{II} = \phi$, such that

$P^I = \bigsqcup_{i=1,v} i_p{}^I$, $P^{II} = \bigsqcup_{i=1,v} i_p{}^{II}$, and

$(\forall i = \overline{1,v})$ $(\exists j = \overline{1,v})$ $(i \neq j$ & $(i_p{}^{II} \cap j_p{}^{II} \neq \phi)$,

(ii) $T = \bigsqcup_{i=\overline{1,v}} i_T$, $(\forall i,j = \overline{1,v})$ $(i \neq j \rightarrow i_T \cap j_T = \phi)$,

(iii) $E = \bigsqcup_{i=\overline{1,v}} i_E$,

(iv) M_o is an (1 x n) vector such that ($\forall$ $p \in p^I$ ($M_o(p) = 0$ and ($\forall P \in P^{II}$) ($M_o(p) = 1$).

Definition 7.

A PN = (C , M_o) satisfying the assumption 1 is said to be an open queueing PT-net (OQPT-net, for short).

The conditions sufficient for the open queueing PT-net (OQPT-net, for short) liveness as well as an algorithm aimed at the live OQPT - nets design have been developed by Banaszak [7, 8, 9] .

Remark 1. From the above definition it follows that each PN being an element of the class of pipeline place/ transition nets (PPT-nets) contains one source and one sink, and satisfies the following condition : (M_o ,) = M_o , where $M_o = t_1 \, t_2 \, t_3 \, ... \, t_i ... \, t_m$.

Remark 2. For each PN being an element of the class of OQPT - nets such that PN = $\{^k PN \mid k = \overline{1, v}$ & $^k PN$ PPT $\}$ there exists $\sigma = \sigma_1 \sigma_2 \sigma_3 \ldots \sigma_j \ldots \sigma_v$, $\sigma_j = t_{1+1}\, t_{1+2} \ldots t_{1+nj}$, where .

$$1 = \sum_{i=1}^{j-1} r_i, r_i$$ – is a length of firing sequence σ_i,

such that

$$\delta\ (M_o\ , \sigma\ ) = M_o .$$

Remark 3. In the rest of this paper , in order to model the inter-processes cooperation, we use the following interpretation for places, transitions and tokens :

(1) Places represent conditions or resources (machines) or buffers.

(2) If a place represents a condition, a token in the place indicates that the condition is true and no token indicates that the condition is false. If a place represents the resource, a token in it represents the machine readiness to perform an operation, and if a place represents a buffer, a token in it stands for a workpiece.

(3) Transitions represent operations, performed on system components, involved in the course of the workpieces transporatation or machining.

Definitions 8.

A matrix D = (D^+, D^-) such that D^+, D^- are of size g x n, and $(\forall\, i \in \overline{1,g})\ (D^+[i] \in o,1^n$ & $D^-[i] \in \{o,1\}^n)$ is said to be a Process Performance Matrix (PPM, for short), if the following conditions hold :

(i) for each two rows D [k], D [1] , there exists $u \in \overline{1, g}$ and a sequence $D[i_j] \mid j \in \overline{1, u})$ such that $D[k] = D[i_1]$,

$D[1] = D[i_u]$ and $(\forall q \in \overline{1, u\text{-}1})\ (\exists j \in \overline{1, n})$ $(D^+[i_q, j] =$

$D^-[i_{q+1}, j] = 1 \vee (D^+[i_{q+1}, j] = D^-[i_q, j] = 1)$,

(ii) the matrix D consists of submatrices (D_k k $\overline{1, v}$) such that.

(ii-1) each D_k consists of a set of rows

$\Delta = \{D[i_{k-1} + 1']\ \mid\ 1' \in \overline{1, i_k - i_{k-1}}\}$, where $D[i_{k-1} + 1]$ and $D[i_k]$ stand for the first and the last row of submatrix D_k, and $i_0 = 0$,

(ii–2) for each $\overline{1 \in i, i_k - i_{k-1}}$, ther exists $W = \{k_1, \mid 1' \in \overline{1,s}\}$, $W \subset \overline{1, i_k - 1_{k-1}}$, such that $1 \notin W$, and there exists $q \in W$ such that $^{\bullet}D[i_{k+1} + 1] \subset D^{\bullet}[i_{k-1} + q]$ and

$$\sum_{r \in W_j} D^+[i_{k-1} + r] - D^-[i_{k-1} + r]) + (D^+[i_{k-1} + 1] - D^-[i_{k-1} + 1]) + X' \{0,1\}^{n'},$$ as well as for each

$W_j = \{K_1, \mid 1' \in \overline{i, j}\}, j \in \overline{1, s'}$,

$$\sum_{r \in W_j} D^+[i_{k-1} + r] - D^-[i_{k-1} + r]) + X' \in \{0,1\}^{n'},$$ hold , where .

$^{\bullet}D[k] = \{j \mid D^-[k, j] = 1$ & $D^+[i,j] \neq i\}$, $D^{\bullet}[k] = \{j \mid D^+[k,j = 1$ & $D^-[i, j] \neq 1\}$, moreover, there exists an ordered set.

$\overline{W} = \{k_1 \mid \overline{1 \in 1, s'}\}$ such that $\overline{W} \subset \overline{1, i_k - i_{k-1}} \setminus \overline{W}$, $1 \in \overline{W}$, and for each $W_j = \{k_1, \mid 1' \in \overline{1, j}\}$ j $\in \overline{1, s'}$ there hold

$$\sum_{r \in W \in W_j} D^+[i_{k-1} + r] - D^-[i_{k-1} + r]) + X' \in \{0,1\}^n,$$ and

$$\sum_{r \in W \in W} D^+[i_{k-1} + r] - D^-[i_{k-1} + r]) = O,$$

where dim x' = n', and

$$n' = \| \bigsqcup_{1, \in \overline{1, i_k - i_{k-1}}} \{D^*[i_{k-1} + 1'] \cup {}^*D[i_{k-1} + 1^*]\} \| ,$$

$$X'(j) = \begin{cases} 1 \text{ if } D^+[i_{k-1} + 1', j] = D^-[i_{k-1} + 1', j] = 1 \text{ and} \\ (\forall\ r \in \overline{1, 1'-1})\,(D^+[i_{k-1} + r, j] = D^-[i_{k-1} + r, j] = O, \\ O \text{ otherwise} \end{cases}$$

(iii) for each $W = \{k_1, \mid \overline{1' \in 1,s}\}$, $W \subset \overline{1, g}$, such that for each $W_j = \{k_1, \mid 1' \in \overline{1, j}\}$, $j \in \overline{1, s''}$.

$$\sum_{r \in W_j} (D^+[r] - D^-[r] + X \in \{O,1\}^n,$$ there exists $\overline{W} = \{K_1, \mid 1' \in \overline{1,s''}\}$, $\overline{W} \subset \overline{1, g} \setminus W$ such that

for each

$\overline{W}_j = \{k_1, \mid 1' \in \overline{1, j}\}$, $j \in \overline{1, s''' - 1}$,

$$\sum_{r \in W \in W_j} D^+[r] - D^-[r]) + X \in \{o,1\}^n,$$ and for

$$\overline{W}_j = \overline{W}, \sum_{r \in W \in \overline{W}} D^+[r] - D^-[r]) = O,$$

where dim $X = n$, and

$$X'(j) = \begin{cases} 1 \text{ if } D^+[i,j] = D^-[i,j] = 1 \text{ and} \\ (\forall k \in \overline{1,1'-1})(D^+[k-j] = D^-[k,j] = O), \\ O \text{ otherwise} \end{cases}$$

The main idea of the Process Performance Matrix (PPM concept introduced is elucidated by the following proposition.

Proposition 1

If a matrix $D = (D^+, D^-)$ is a PPM then there exists a $PN = (C, M_0)$ being live for Mo PT-net, where the incidence matrix $C = (C^+, C^-)$ of size m x n and the initial marking M_0 are defined as follows :

(i) $(\forall i \in \overline{1,m})(\exists ! 1 \in \overline{1,g})(\forall k \in \overline{1,1-1})$
$(C[i] = D[1] \; \& \; i \leq 1 \; \& \; D[k] = D[1])$,

$$\text{(ii) } M_0(j) = \begin{cases} 1 \text{ if } D^+[i,j] = D^-[i,j] = 1 \text{ and} \\ (\forall k \in \overline{1,1'-1})(D^+[k-j] = D^-[k,j] = O), \\ O \text{ otherwise} \end{cases}$$

3. Statement of the Problem

Consider an assembly system consisting of a finite number of components e. g. robots, conveyors, assembly robots and assembly stations, which may operate asynchronously. The system is aimed to perform a finite set of concurrently flowing assembly processes. Along of each process the preassumed batch size of products is completed.

Some of the system components may be shared among different processes. Each process realizes the pipeline-like flow of partially ordered operations. It is assumed that the processes are specified by the precedence digraphs of operations. It is also assumed that to eah operation there may correspond a subset of "alternative" system components . Clearly, the operation can be processed on the only one, e.g. actually admissible, component from among the relevant system components.

Variant - specific components and standard parts together with the assembly objects are provided from input stores or conveyors to the appropriate buffers of assembly stations by the robots. Then, the partially assembled objects are displaced among the buffers of the relevant assembly stations. Our task is then to design an algorithm transforming the given process specification, e.g. a set of production routes and buffers capacity constraints, into the related net model of its control flow.

The above observation unerlines the importance of automatic design of control procedures encompassing modelled processes performance which takes into account the asynchronous operation of system components (robots and machine tools) and is responsible for the interacting processes cooperation. Solution of the problem mentioned plays a key role in the design of computer aided process and production planning systems aimed at automatic synthesis of correct simulation programmes guaranteeing deadlock - free execution of the processes modelled.

4. Modelling of Cooperative Processes

The main purpose of this paper lies in the formal invesigation of the conditions sufficient for the design of a class of a priori regular net models reflecting the required behaviour of concurrently flowing processes. The results delivered, based on petri net concept, allow us to design an algorithm transforming any set of the process specifications into the PT-net model of the control flow ensuring a pipeline-like and deadlock - free execution of asynohronously flowing processes.

4.1. Processes Specification

At first, let us introduce a class of processes specification dedicated to the open queueing PT-net (OQPT-nets). Consider a set of processes where each process is a decomposition of partially ordered transportation and assembly operations. Processes are conncurrently performed in an assembly system consisting of a finite set of robots and assembly stations. It is assumed that each assembly operation is directly preceded and directly succeeded by the relevant sets of transportation operations. The transportation and assembly operations are performed by robots and assembly stations respectively. Each operation can be processed by precisely one preassumed system component. However, some of the system components can perform different operations. The assembly stations are equipped with buffers where the pallets loaded with assembly objects and/or standard parts as well as partially assembled objects are stored.

In order to specify the processes in question, let us consider the set PR = (PR_i | $i \in \overline{1, v}$), where

PR_i := B (i, j, k_g) [B (i,1,k_h) , ... , B (i,m,k_p)] , ...
...., B (i , n , k_r) [B (i , v , k_s) , ... , B (i , q , k_r)] is the list description of the partially ordered set of pairs (operation - relevant system component), necessary to complete the products along the i-th assembly process. The interpretation of the notations introduced is as follows :

B (i , j , k_1) – denotes the j-th operation performed for the i – th process on the k_1 - th system component,

B (i, j, k_1) [B (i , m , k_p) , ... , B (i , 1 , k_s)] – denotes that in the i-th process , the j - th operation precedes the set of operations (m, ... , 1) performed on the set of corresponding components (K_p , ... , k_s) .

Note that to each list description PR_1 associated is a set of routes { R_i^1 | $1 \in \overline{1, L_i}$ } where each R_i^1 is such that for $r \in 1, \overline{| R_i^1 | -1}$, | R_i^1 | stands for the length of R_i^1, the operation included in $crd^r R_i^1$ directly precedes the operation included in $crd^{r+1} R_i^1$. It is assumed that for each $i \in \overline{1, v}$, and for each $1 \in \overline{1, L_i}$ | R_i^1 | is an odd number. Moreover, each odd component of R_i^1 denotes a transportation operation, e.g. pallet displacement between preassumed buffers, performed by the industrial robots, while the even components of each R_i^1 denote the assembly operations performed at relevant assembly stations. In order to illustrate the usefulness of the introduced form of processes specification in the course of OQPT–net model designing, let us consider the following example.

Example 1 : let

PR_1 : = B (1,1,1) [B(1,2,2)] , B (1,2,1) [B (1,2,2)] , B (1,2,2) [B (1,3,3)],
, B (1,3,3) [B (1,4,4)] , B (1,5,5) [(1,4,4)],
, B (1,4,4) [B (1,6,5) , B (1 , 7, 6)],
PR_2 : = B (2,8,6) [B (2,9,4)] , B (2,10,7) [B (2,9,4)] ,
, B (2,9,4) [B (2,11,3)] , B (2,11,3) [B (2,12,2)],
, B (2,13,7) [B (2,12,2)] , B (2,12,2) [B (2,14,7)],
, B (2,14,7) [B (2,15,8)] , B (2,16,7) [B (2,15,8)],
, B (2,15,8) [B (2,17,7)]

be the specification of the processes performed on the assembly system shown in Fig. 1.

The system consists of five industrial robots M_1, M_3, M_5, M_6, M_7, and three assewlty stations M_2, M_4, and M_8 equipped with the input S_i^j and output $\bar{S}_i^j$ store places, where S_i^j , $\bar{S}_i^j$ – denotes the j - th input (output) store place in the i-th machine's buffer.

Note that the buffers of assembly stations M_2 and M_4 can store pallets flowing along both assembly processes.

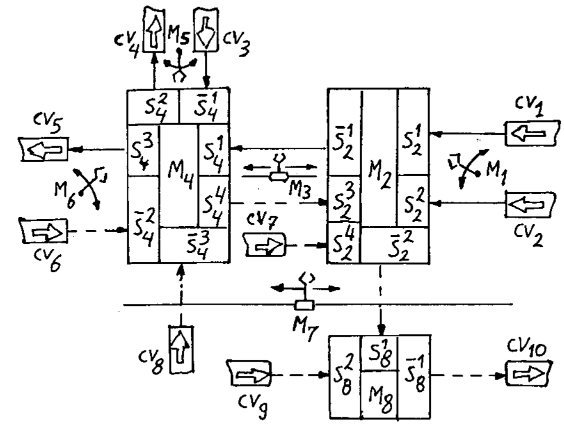

Fig. 1. Flexible assembly cell.

Legend : M_i – is the i-th system component, i.e. the industrial robot or the assembly station, S_i^j, ($\bar{S}_i^j$) – is the j-th input (output) store place in the buffer of the i-th station, CV_i – is the i-th conveyor or pallets store .

The OQPT – net model corresponding to the given processes specification is shown in Fig. 2. The elements of the net are interpreted as follows : Transitions $t_1 \rightarrow t_8$ correspond to the operations constituting process PR_1 while transitions $t_9 \rightarrow t_{18}$ correspond to the operations constituting process PR_2. Transitions t_3 , t_{14} and t_6, t_{11} and t_{17} correspond to the operations performed on assembly stations M_2, M_4 and M_8 , respectively. Remaining transitions correspond to the operations processed on robots. Places $P_1 \rightarrow P_{16}$ are interpreted as store places of the assembly station buffers. For instances, places P_{14} and P_{15} correspond to the input store places $\bar{S}_8^1$ and S_8^2 and place P_{16} corresponds to the output store place S_8^1 of the stations M_8 buffer. Places $P_{17} \rightarrow P_{20}$ and $P_{21} \rightarrow P_{23}$ correspond to the robots and assembly stations respectively.

Note that our approach to the net model designing implies that the capacity of each station buffer is equal to the number of the station occurrences, in the processes specification.

In order to specify the so-called "redundant processes", i. e. the processes where some of operations can be performed on one, actually available, system component from among given system components, let us consider the set :

$PR = (PR \mid i \in \overline{1, v})$, where

$PR_i := \{ (B(i, j, k_a), \dots B(i, j, k_d) \} \{(B(i, 1, k_c), \dots, B(i, 1, k_h)\}, \dots,$
$\{ B(i, n, k_1), \dots, B(i, n, k_f)\}, \{B(i, v, k_l), \dots \}$
$[\{B(i, s, k_w), \dots \}, \dots, \{ B(i, w, k_q \}, \dots \}]$ is the list description of the partially ordered set of pairs (operation - relevant system component) employed in the course of assembly processes completion. This processes sepecification differs from the former one by the introduction of a new kind of list : $\{B(i, j, k_a), \dots, B(i, j, k_d) \}$ denoting that in the i-th process, the j-th operation can be performed on only one of the actually available system components from the set $\{ k_a, \dots, k_d \}$.
For this purpose of illustration, let us consider an assembly process specification as follows :

$PR := \{ t_1, t_2 \} [\{ t_3, t_4 \}], \{ t_3, t_4 \} [t_5, t_6 \}, \{t_7\}], \{t_7\} [\{ t_9, t_{10} \}], \{ t_8 \} [\{t_9, t_{10}\}], \{t_9, t_{10}\} [\{t_{11}, t_{12}\}, \{ t_{13} \}]$, where

$t_1 = B(1,1,1)$, $t_2 = B(1,2,3)$, $t_3 = B(1,3,2)$, $t_4 = B(1,4,4)$,
$t_5 = B(1,5,5)$, $t_6 = B(1,6,6)$, $t_7 = B(1,7,7)$, $t_8 = B(1,8,8)$,
$t_9 = B(1,9,9)$, $t_{10} = B(1,10,10)$, $t_{11} = B(1,11,11)$, $t_{12} = B(1,12,12)$
$t_{13} = B(1,13,11)$. According to special interpretative rules, the above specification is converted into the following one :

$PR' := \{t_1\} [\{t_3\}], \{t_2\} [\{t_4\}], \{t_3\} [\{t_5, t_6\} \{t_7, t'_7\}], \{t_4\} [\{t'_5, t'_6\}, \{t''_7, t'''_7 \}], \{ t_7, t''_7 \} [\{ t_9 \}]. \{t'_7, t'''_7 \} [\{ t_{10}\}], \{t_8\} [\{ t_9 \}], \{ t'_8\} [\{t_{10}\}], \{t_9\} [\{t_{11}, t_{12}\}, \{t_{13}\}], \{t_{10}\} [\{t'_{11}, t'_{12}\}, \{t'_{13}\}]$,

where :

$t'_5 = B(1,51,5)$, $t'_6 = B(1,61,6)$, $t'_8 = B(1,81,6)$,
$t'_{11} = B(1, 111, 11)$, $t'_{12} = B(1,121,12)$, $t'_{13} = B(1,131,11)$.

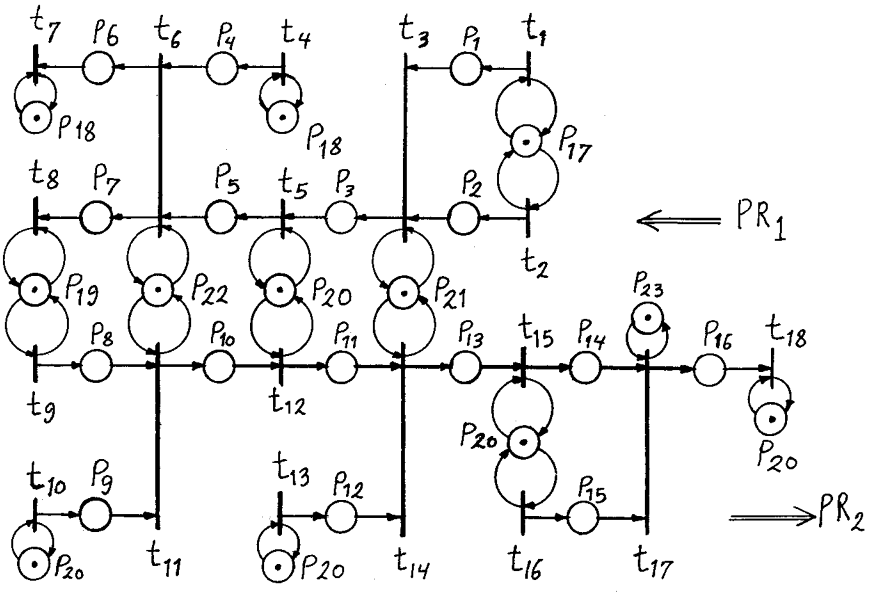

Fig. 2 OQPT-net for Example 1.

It should be noted that according to a given standard form of processes specification, different net-modelling building algorithms can be proposed. Of course, only some of them which lead to the models encompassing appropriate aspects of systems behaviour may find a practical implementation.

5. Net Model Designing

It can be readily seen from Proposition 1, that live PT-net, i.e. net models encompassing deadlock-free processes realizations. can be easily reconstructed from the matrices belonging to the PPMs class. Thus, our task lies in finding out the conditions sufficient for the design of matrices being PPMs of OQPT-nets. The conditions in question are set up in the following theorem.

Theorem 1 :

If matrix $D = (D^+, D^-)$, where D^+, D^- are of size g x n, satisfies the following conditions then there exists a PPM such that $PN = (C, M_o)$, associated with matrix D, is a live OQPT-net.

(i) matrix D consists of rows $\Delta_k = \{D[i_{k-1}+1] \mid 1 \in 1, i_k - j_{k-1}\}$, where $D[i_{k-1}+1]$ and $D[i_k]$ stand for the first and the last row of the submatrix D_k, and $D^+[i_{k-1}+1] \in \{0,1\}^n$, $D^-[i_{k-1}+1] \in \{0,1\}^n$, $1 \in \overline{1, i_k - i_{k-1}}$,

(ii) for each D_k, $k \in \overline{1,v}$, the following conditions hold :

(ii-i) $(\forall D[1] \in \Delta_k)(\forall D[i] \in \Delta_k)(D[1] = D[i])$,

(ii-ii) $(\exists J_k \subset \overline{1,g})(\exists \Psi_k \subset \overline{1,g})(J_k \cap \Psi_k = O)$,

$\| \bigcup_{D[1] \in \Delta_k} \{D[1] \cup D[1]\} \| = \| J_k \cup \Psi_k \|$,

(ii–iii) $(\forall j \in \Psi_k)(\exists ! D[1] \in \Delta_k)(\exists ! D[i] \in \Delta_k)(D[1] = D[i] \ \& \ j \in D^\bullet[1] \cap {}^\bullet D[i] \ \& \ {}^\bullet D[1] \cap {}^\bullet D[i] = 1)$,

$(\forall D[i] \in \Delta_k)(\exists ! j \in J_k)(D^+[i,j] = D^-[i,j] = 1)$,

$(\forall j \in J_k)(\exists D[i] \in \Delta_k)(D^+[i,j] = D^-[i,j] = 1)$,

(ii-iv) for each two rows D [i] , D [1] Œ Δk the condition (i) of Definition 8 holds ,

(ii-v) there does not exist the following sequence of rows

$(D[k_i] \mid i \in \overline{1,r})$ where $D[k_1] = D[k_r]$, such that $D^+[k_1, j_1] = D^-[K_2, j_1] = D^+[k_2, j_2] = \dots\dots = D^+[k_r - 1' j_r - 1] = D^-[k_r, j_r - 1] = 1$, for $j_i \in \Psi k$,

(ii-vi) each D_k consists of non-empty disjoint subsets $X_k \subset D_k$, Y_k, $\subset D_k$, $Z_k \subset D_k$ defined as follows:

$X_k = \{D[i] \mid D[i] \in \Delta_k \ \& \ (\forall j \in \Psi_k)(D^-[i,j] = 0) \ \& \ (\exists j \in \Psi_k)(D^+[i,j] = 1)\}$,

$Z_k = \{D[i] \mid D[i] \in \Delta_k \ \& \ (\exists j \in \Psi_k)(D^-[i,j] = 1) \ \& \ (\exists j \in \Psi_k)(D^+[i,j] = 1)\}$,

$Y_k = \{D[i] \mid D[i] \in \Delta_k \ \& \ (\forall j \in \Psi_k)(D^+[i,j] = 0 \ \& \ (\exists j \in \Psi_k)(D^-[i,j] = 1)\}$

(ii-vii) for each sequence of rows $(D[k_j] \mid i \ \overline{1,r})$ such that $D[k_1] \in X_k$, $D[k_r] \in Y_k$, $\forall i \in \overline{2,r-1}$ $(D[k_j] \in Y_k)$ and $(\forall i \in \overline{1,r-1})$ $(D^\bullet[k_i] \cap D[k_{i+1}] \neq \phi)$ the following conditions holds : i' ∈ r is an odd number, ii

$\| D''[K_1] \| = 2$ and $\| {}^\bullet D[k_r] \| = 2$ and for each $D[K_i]$ such that $i \in 3$, r-i is an odd number , $\| D[k_j] \| = \| D[k_i] \| 2$,

(iii) $(\forall k, 1 \in \overline{i,v})(\Delta_k \cap \Delta_1 = \phi)$, $(\forall k \in \overline{1,v})(\exists 1 \in \overline{1,v})\ i_k = 1 \ \& \ J_k \cap J_1 \neq \phi)$, $(\forall K, 1 \in \overline{1,v})(K \neq 1 \rightarrow \Psi_k \cap \Psi_1 = \phi \ \& \ J_k \cap \Psi_i = \phi \ \& \ \Psi_k \cap J_1 = \phi)$.

The conditions sufficient for matrix D to be a PPM of a live OQPT-net can be formulated analogously. Note that the conditions provided in Theorem 1 can be employed in the designing of the control flow net models. The relevant algorithm consists of three steps :

step 1 : For each element of $\{PR_i \mid i \in \overline{1,v}\}$, according to conditions (i), (ii) of Theorem 1, set up matrix D_k of size $g_k \times B_k$ as follows .

Convert PRk into a set of <u>terminal paths</u> $\{TP_k^j \mid j \in 1, q\}$ "Covering" PR_k For TP_k^1 set up submatrix D_k^1 of size $g_k^1 \times n_k^1$ taking into account that to each $B(i, j, k_1)$ in the terminal path, there corresponds a unique row $D[1_{k-1} + q] \in \Delta_k$. Then for TP_k^2 set up submatrix D_k^2 of size $g_k^2 \times n_k^2$, for TP_k^3 set up D_k^3 and so on. The resultant matrix D_k is of size $g_k \times n_k$, where

$$g_k = \max\{g_k^i \mid j \in \overline{1,g}\}, n_k = \sum_{i=1}^{r} n_k^i.$$

Step 2 : According to condition (iii) of Theorem1, set up matrix D′ of size g x n′ such that $g = \sum_{k=1}^{v} g_k$, $n' = \sum_{k=1}^{v}$. Then, set up matrix D″ which extends matrix D′ to matrix D of size g x n, where $n = n' + \| M \|$ and each column $1 \in \overline{n'+1, \| M \|}$ corresponds to the unique system component $M_1 \in M$.

Step 3 : According to conditions (i) , (ii) of Proposition 1 set up $PN = (C, M_o)$ being a OQPT-net model of the control flow .

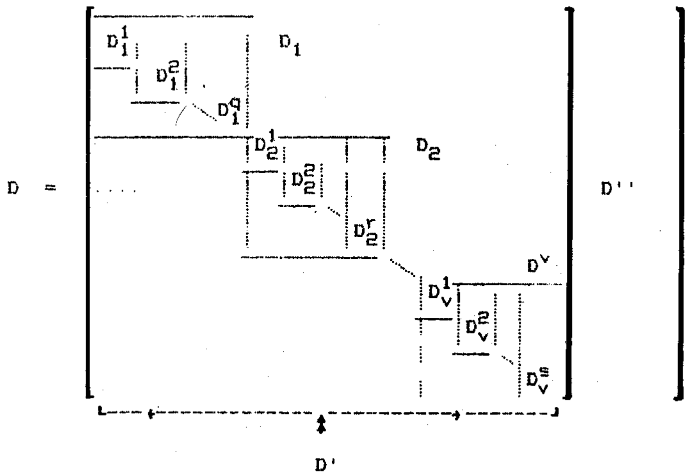

Fig. 3. The Structure of matrix D.

For illustration. let us consider the following example,
Example 2 : Let PR_1 : = B (1,1,8) [B (1,3,10)] , B (1,2,9) [B (1,3,10)], B (1,3,10) B (1,4,1) [B (1,5,11)],

PR_2 : = B (2,6,11) [B (2,8,10)], B (2,7,12) [B (2,8,10)], ,B (2,8,10) [B (2,9,9)], be a given processes specification.

Matrix D produced with the help of the algorithm is shown in Fig. 4 Blanked entries in matrix D are assumed to be zero .

D^+ =

	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}	p_{11}	p_{12}
t_1	1							1				
t_3		1		1						1		
t_4								1				
t_2			1						1			
t_5											1	
t_6					1				1			
t_8						1				1		
t_9											1	
t_7							1					1

, and

D^- =

	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}	p_{11}	p_{12}
t_1								1				
t_3	1		1							1		
t_4		1						1				
t_2									1			
t_5				1							1	
t_6									1			
t_8					1		1			1		
t_9						1					1	
t_7												1

Fig. 4 Incidence matrix of OQPT - net from Fig. 5

6. Planning of Robots Cooperation

The algorithm just introduced provides net models of control flows encompassing all admissible, time ordered activations of the system components. However, in models of this type it is assumed that firing of a transition takes no time. Such an assumption, as mentioned by Naranari and Viswandaham[10], restricts the models usefulness only to the qualitative analysis of systems with regard to deadlock, buffer overflows, invariance of jobs number etc. (Note that our case the automatically generated net models are a priori deadlock-free).

To overcome this disadvantage we consider modified models where to each transition (interpreted as an assembly or transportation preassumed operation) associated is a firing time (proportional to the preassumed operation time). Such a model specification, amounting to timed petri nets introduced by Ramachandan [ii], can be used to construct a simulation program. Clearly on the basis of a given processes specification and operation times, a simulation model for the performance system evaluation can be created automatically, In general, the simulation model can be extended to encompass other input data such as batch size of products, costs of resources utilization, buffers capacity and so on.

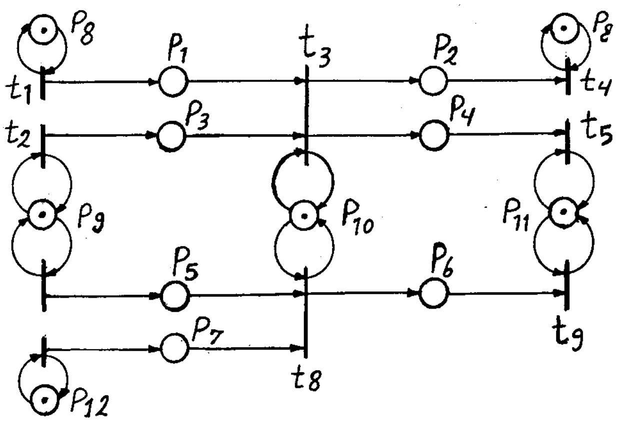

Fig. 5. The PPT-net for Example 2.

In the sequel, the simulation model serves as the conceptual framework in a process of a simulation program design. The appropriate simulation algorithm, based on concept similar to that introduced by Smigielski and coauthors [12], has been implemented on AMSTRAD CPC personal computer. The aim of the developed package, working in the menu-driven mode, is to estimate the time needed by an assembly system to perform the production task, i.e. balancing, i.e. the idle time reduction. The results of the simulation are depicted in the form of the Gantt's chart. Besides the Gantt's chart, the output statistics concerning the cycle time, throughput and idle time, as well as measures of system components utilization are provided.

Because the assembly stations and robots can perform their tasks asynchronously, some conflicts may occur. In order to control such activities, some priority rules have been introduced to the system. The rules are concerned with priorities of the robots operation as well as of concurrently flowing processes. Another task one could consider concerns the robots breakdowns occurrence. This task lies in finding out the best arrangement of robots and/or stations in order to meet pre-assumed faulty and flexibility requirements.

It should be pointed out, however, that out considerations are only related to the systems consisting of the assembly stations equipped with pallet-loading buffers. It is assumed that each buffer's capacity depends on the actually considered processes specification. Clearly, the capacity of each buffer is equal to the number of the station occurrences in the process specification. In order to take into account the arbitrarily given constraints imposed on buffers capacity, we need to introduce to the net models the relevant "synchronization mechanism".

As an illustration, let us consider the assembly cell considered in Example 2. According to the interpretation introduced in Example 1, the assmbly station M_{10} performs two operations t_3 ad t_8 occurring in the processes PR1 and PR_2, respectively. Station M_{10} is equipped with buffer consisting of four input and three output store places. Input store places $S_1^{\ 1}$ and $S_1^{\ 2}$ correspond to P_1 and P_3' output store places $S_1^{\ 1}$ and $S_1^{\ 2}$ correspond to P_2 and P_4 , the store places are utilized in the course of process PR_1 execution. In PR_2, input store places $S_1^{\ 3}$ and $S_1^{\ 4}$ correspond to P_5 and P_7 while $S_1^{\ 3}$ corresponds to P_6 .

Let us assume that two different store places $S_1^{\ 2}$ and $S_1^{\ 3}$ are replaced by one input / output store place A and places $S_1^{\ 3}$ and $S_1^{\ 2}$ are replaced by one store place 3. The graphical interpretation of the above assumption, drawn by *dashed lines*, is shown in Fig. 6. Comparing with the net shown in Fig. 5, the introduced places eliminate the considerations all processes realizations corresponding to the marking M R (C,M_o) such that M (P_3) = M (P_6) = 1 and M (P_4) = M (P_5) = 1. All the available displacements of the pallets into the store places A and B are shown in Fig. 7 .

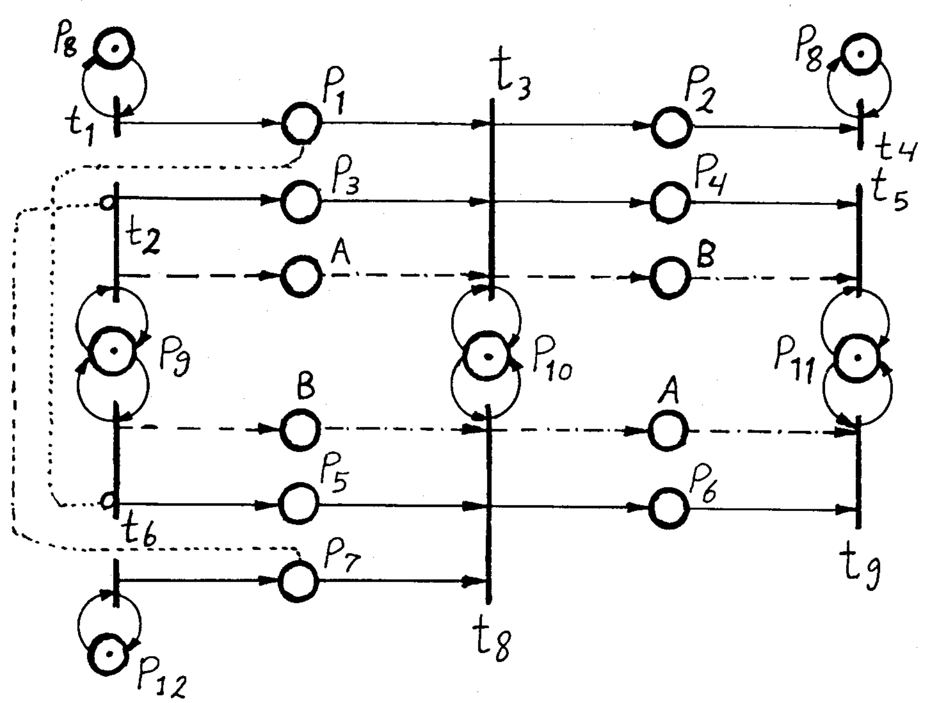

Fig. 6. Modified PPT - net from Example 2.

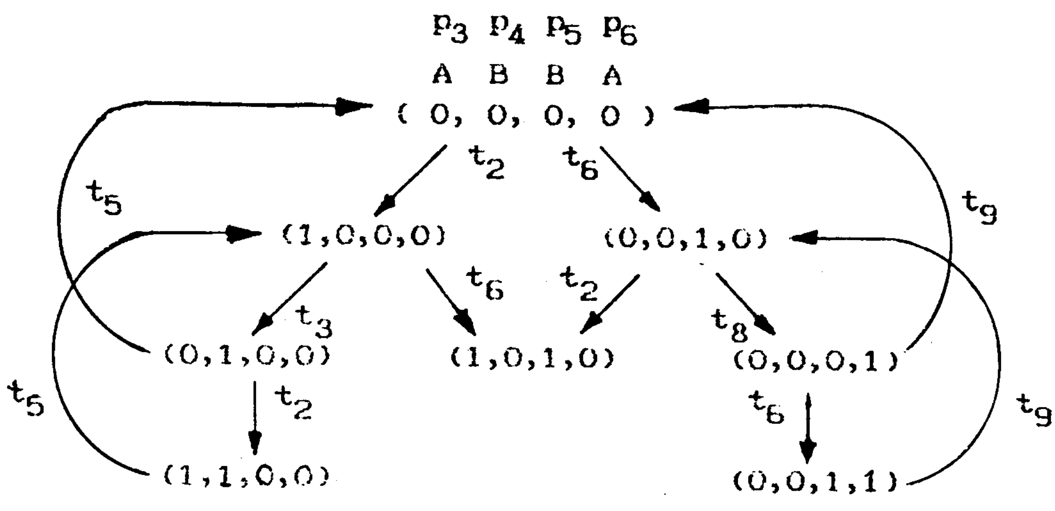

Fig. 7. Reachability set of pallets displacements.

Note that among all readhable states there exists a state M such that M (P3) = M (P5) = 1 , corresponding to a deadlock situation, i.e. the pallet stored in A cannot be displaced in B while the pallet stored in B cannot be displaced in A. In order to avoid the occurrence of the above deadlocks, a synchronization mechanism modelled with help of *inhibitor arcs*, see Peterson [5], can be introduced. Its graphical illustration, drawn by doted lines, is shown in Fig. 6.

In order to present an idea of the synchronization mechanism construction, let us set up the following definition.

Definition 11 :

Let PN = (P, T, E, M_0) be a PT - net. A set of the Inhibitor arcs linking t ∈T with a given set of places p'⊂ P is said to be *a conjunction set of the Inhibitor arcs* (CI-arcs, for short) if for each M ∈ R (C,M_0) such that (∀ P ∈ P') (M (p) = 1), transition t is not enabled. Let PN' = (P', T', E' , M'_0) be a PPT - net, and let.

{ CC_i | i ∈ $\overline{1,w}$), where CC_i = { P_j | J ∈ $\overline{1,u}$ } such that (∀ i, k ∈ $\overline{1,w}$)

(CC_i ∩ CC_k = ϕ) and (∀ i ∈ $\overline{1,w}$) (∀ p ∈ CC_i) (•p p• = ϕ) be a set of the buffer's capacity constraints.

Consider PN = (P, T', E, M_0) such that p = p' U p" and p' ∩ p" = ϕ and (∀ p ∈ p") (∃ ! i ∈ $\overline{1,w}$) (∀ p ∈ CC_i) (•p = •p' & P• = p'•).

Theorem 2 : Let M ⊆ R (C, M_0) be the reachability set of $\overline{\text{PN} = (\text{P}, \text{T}'}$, E, M_0) such that there exist t, t' ∈ T where t ≠ t', and (∀ M' ∈ R (C, δ (M,t))) (M_0 ∉ R (C,M')), (∀ M' ∈ R (C, δ (M, t'))) (M_0 ∈ R (C,M')).

If a CI - arc is introduced to PN = (P ,T' , E ,M_0) in such a way that each arc links some p ∈ p' , such that (∀ t ∈ T) (p ∉ •t t•) and M (p) = 1, with t where p ∈ •t, then from reachability set R (C , M_0) the set R (C, δ M , t)) is removed.

The above theorem provides the synchronization mechanism aimed at the deadlock-free process execution in the case when some arbitrarily given constraints are imposed on buffers capacities.

7. Conclusions :

The approach presented is a certain contribution to the automatic control and the performance evaluation of FASs. Its objective is to reduce time and effort involved in the development of the control programs oriented toward the promotion of real - time robots cooperation.

The modelling of FASs by means of Petri nets has allowed us to determine an algorithm transforming a given processes specification into their simulation program. The program, reflecting the structure of admissible controls involved in the processes accomplishment, can serve as a control program for a system controller as well as a task – oriented package for the computer - assisted process planning.

A prospective implementation of the results obtained is related to the statistical analysis of processes performances in the preassumed types of assembly systems in order to establish the knowledge base for an expert system. Further research devoted to this problem constitutes our most actual, primary task.

REFERENCES

[1] **Bruno, G.**, and **G. Marchette**, 1986. "Process-Translatable Petri Nets for the Rapid Prototyping of Process Control Systems", IEEE Trans. Software Engi., SE-12, 346-357.

[2] **Sathaye, A.S.**, and **B.H. Krogh**, 1993." Synthesis of Real- Time Supervisors for Controlled Time Petri Nets", Proceedings of 32nd IEEE Conf. on Decision and Control, San Antonio, Texas, USA.

[3] **Ravichandran R.**, and **A. K. Chakravarty**, 1986. "Decision Support in Flexible Manufacturing Systems Using Timed Petri Nets", Journal of Manufacturing Systems, 5, 89–98,

[4] **Lee, D.Y.**, and **F. DiCesare**, 1993 . " Scheduling Flexible Manufacturing Systems with the Consideration of Setup Times", Proceedings of 32nd IEEE Conf. on Decision and Control, San Antonio, Texas, USA.

[5] **Peterson, J. L.**, 1981. Petri Net Theory and the Modelling of Systems, Englewood Cliffs, NJ : Prentice – Hall.

[6] Reisig W., Petrinetze, Springer – Verlag, Berlin, 1982.

[7] **Banaszak, Z.**, 1986. "Coordination of Concurrent Processes : Automatic Program Synthesis", In R. Trappl (Ed.), Cybernetics and Systems 86, D. Reidel Publishing Company, Dordrecht, Holland, 709 - 716.

[8] **Banaszak, Z.**, 1986." Automatic Modelling of Pipeline Concurrent Processes", The 4th IFAC/IFORS Symposium, Large Scale Systems : Theory and Applications, Zurich, Switzerland, 89–94.

[9] **Banaszak, Z.**, and **M. Abdul-Hussin**, 1988." Petri Net Approach to Automatic Real - Time Program Synthesis," Control and Cybernetics, 17, 361-375.

[10] **Narahari, Y.**, and **N. Viswanadham**, 1985."A Petri Net Approach to the Modelling and Analysis of Flexible Manufacturing Systems", Annals of Operations Research, 449 - 472.

[11] **Ramchandani, C.**, 1974. " Analysis of Asynchronous Concurrent Systems by Timed Petri Nets", Ph. D. dissertation, Department of Elect. Engi., MIT, Cambridge MA.

[12] **Smigielski, T.**, **T. Murata**, and **M. Sowa**, 1985. A Timed Petri Net Model and Smulation of a Dataflow Computer, Proceeding of the Intern. Workshop on Timed Nets, Torino, Italy, 56-63 .