UOT:519.95
DOI: https://doi.org/10.30546/09090.2025.01.2005

DEVELOPMENT OF CONTROL ALGORITHM FOR FLEXIBLE MANUFACTURING SYSTEM MODULE USING FİNİTE AUTOMATA WİTH PETRİ NET

SHAHLA HUSEYNZADE[1*], ELMİRA NASİROVA[2]

[1]Sumgayit State University,[2]Sumgayit State University
Sumgayit, Azerbaijan
shahla.huseynzada@sdu.edu.az,elmira.nasirova@sdu.edu.az

ARTICLE INFO

Article history:
Received: 2025-03-06
Received in revised form:2025-03-06
Accepted: 2025-04-04
Available online

Keywords:
flexible manufacturing system; control system; machine-building industry; finite automata; Petri net

ABSTRACT

On the basis of machine-building industry area analysis there were defined actuality and a goal of this material. As the applied object of this study, flexible manufacturing system is chosen and also it was given a task of designing an algorithm for investigation of its control system. When creating a control model for a flexible manufacturing system a finite automata was used to describe the operation of a crane manipulator. The capabilities of the finite automata are not enough for a general system analysis of the control object. For a detailed functional analysis with the definition of transfers and productions of this control object, it is necessary to use a more powerful mathematical apparatus, such as Petri net. The study of the control algorithm was held on the basis of the round composed flexible manufacturing module scheme of flexible manufacturing system. The resulting algorithm with using the transformation of a finite automata of Petri net allows for the creation of a more correct, efficient and reliable control system for the production module.

INTRODUCTION

In modern times, one of the areas that ensure the economic development of the leading Eurasian countries is considered to be industrial enterprises producing high-quality mechanical products of mechanical engineering. In flexible manufacturing system (FMS), which combine mechanical assembly, processing and production modules of various purposes, automation of the process of manufacturing various electrical, automatic, mechatronic, etc. parts for technological equipment, machine tools and production lines is carried out [1,2]. However, since many complex mechanical technological operations are performed both by working personnel and by control units controlled by digital programs, the design of both mechanical machines and a special manipulator that automatically serves them in the modules of mechanical FMSs of mechanical engineering enterprises and the creation of a control system for automating its operations is considered a scientifically relevant issue.

APPLİCATİON AREA ANALYSİS

The layout of the mechanical production module of the applied FMS is based on a circular structure (Fig. 1).

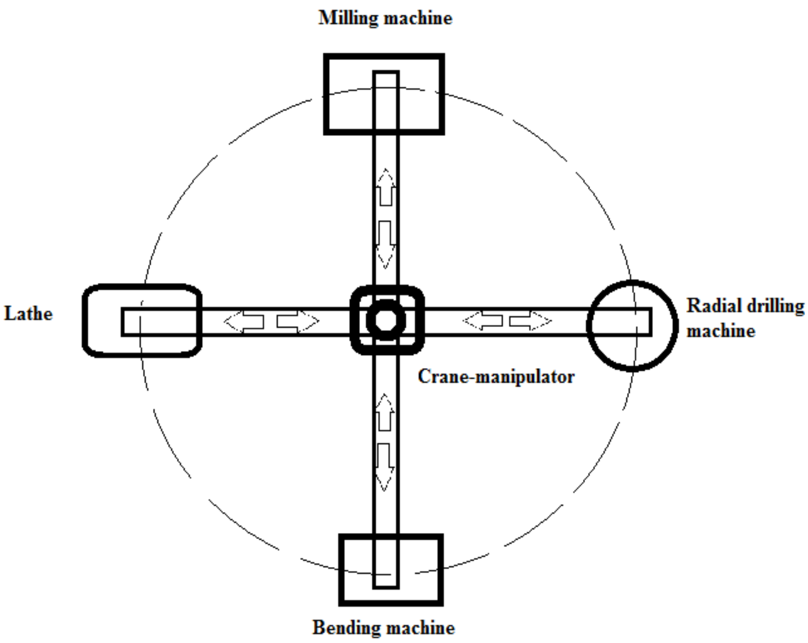

Fig. 1 Layout diagram of the mechanical production module of the FMS

In order to reliably control the service operations of lathes, milling machines, radial drills and bending machines located in the positions of the circular scheme, the issue of conducting a functional analysis of the technological process and interpreting the control functions with knowledge is posed. The analysis of this problem shows that it is more expedient to apply a graphical representation method to more effectively describe the control process [3].

In the mechanical flexible manufacturing module of the FMS, technological operations are planned as follows: the crane-manipulator takes the workpiece from the central plate and moves it linearly upwards. Then, moving linearly on the crane in the left direction, it positions itself in the working zone of the lathe, places the workpiece on the machine, fixes it and moves upwards, waiting for the preparation of the part [4]. After the product is ready, the manipulator moves linearly downwards and takes the finished part from the lathe and moves linearly upwards. After linear movement in the central direction, it moves the part in the direction of the milling or radial drilling machine and the bending machine. The sequence of these operations depends on the type of product being prepared [5].

PROBLEM STATEMENT AND DESCRİPTİON

The One of the more accurate and reliable methods for describing the planning and functions of a complex machine-building technological process control system is considered to be a Petri net (PN).

Modeling of complex dynamic systems (DS) can be considered at several levels, and PNs can model each of these levels. But in practice, it is often the case that individual subtasks of a complex system are modeled using finite automata (FA). Due to the simplicity of application at a low level, system modules are often described with FA. The FA apparatus is intended primarily for modeling individual processes. When modeling complex DS, it is necessary to take into account not only individual processes, but also their interaction. In particular cases, such problems are solved by the capabilities of the FA themselves. This is accomplished by connecting the outputs of some machines to the inputs of others.

At a high level, models based on PN are more convenient for providing a general system analysis of the control object. For formal verification of the system when modeling complex dynamic systems, it is necessary to use a sufficiently powerful formal apparatus such as PN. In this case, there is often a need to solve the emerging problem of transforming the FA into PN.

To describe the control process using a PN, it is possible to effectively describe the operation of the crane manipulator (CM) serving the application object (mechanical assembly flexible manufacturing module) with production rules and, having transformed it into a PN, study the control algorithm using computer experiments as a result of the analysis of the main properties of the latter. This issue was considered in the previous subsection [6].

The issues of developing an algorithm for transforming the finite automata representation of an automatic transport manipulator transporting raw materials to the working area of a crane manipulator into a PN are considered.

Finite automata (FA) is defined by a set of six objects:

$$A = (X, U, Y, x_0, \varphi, \psi)$$

Where $U = (u_1, u_2, \dots, u_m)$ – a finite set of input signals, called the input alphabet of the automata; $Y = (y_1, y_2, \dots, y_g)$ – a finite set of output signals, called the output alphabet of the automata; $X = (x_1, x_2, \dots, x_z)$ – an arbitrary set, called the set of internal states of the automata; x_0 – an element from the set X called the initial state of the automata; $\varphi\colon (x, u)$ and $\psi\colon (x, u)$– two functions that define singular-valued maps of the set of pairs (x, u), where $u \in U$ and $x \in X$, to the sets X and Y. The function $\varphi\colon (x, u)$ is called the transition function of the automata, and the function $\psi\colon (x, u)$ is called the output function [7]:

$$\varphi\colon (X \times U) \to X;\ \psi\colon (X \times U) \to Y.$$

FA can be represented graphically, in the form of transition and output tables, and analytically. $\varphi\colon (x, u)$ and $\psi\colon (x, u)$the transition and output functions of the automata can be represented in the form of a table.

A PN is formally represented as a set of the form [8]:

$$N = (P, T, F, H, \mu^0)$$

Where:

$P = \{p_1, p_2, \dots, p_n\}$, $n > 0$ – a finite non-empty set of positions (otherwise states or locations);

$T = \{t_1, t_2, \dots, t_m\}$, $m > 0$ – a finite non-empty set of transitions (events);

$F\colon P \times T \to \{0,1,2, \dots\}$ and $H\colon P \times T \to \{0,1,2, \dots\}$ - the initial marking (marking of each position) of the functions and mapping of the input and output states;

$\mu^0\colon P \to \{0,1,2, \dots\}$ – initial distribution of markers by position.

The graphical representation of a PN is a bidirectional directed graph with two types of vertices. The vertices p ∈ P are represented by circles, and the vertices t ∈ T are represented by with rectangles. The arcs correspond to the drop functions of positions and transitions .

The FA is defined for the PN as follows:

$$P = U \cup X \cup Y, T = \{t_x, x / x \in X \text{ и } u \in U\},$$

$$I(t_x, u) = \{x, u\}, O(t_x, u) = \{\varphi(x, u), \psi(x, u)\ \}.$$

The resulting PN is a model of a FA.

Based on the given topology, the definition of the structural elements of the PN is presented - transitions and positions according to the following rules [8]: for each pair (state and input symbol), a transition is defined; the combination of sets of input signals $U = (u_1, u_2, \dots, u_m)$, output signals $Y = (y_1, y_2, \dots, y_g)$, internal states $X = (x_1, x_2, \dots, x_z)$ is a set of positions of the automata. The transitions and positions of the PN are defined as $T = \{t_1, t_2, \dots, t_m\}$, , $P = \{p_1, p_2, \dots, p_n\}$, where $n = m + z + g$.

Let's consider the model of the functioning CM of the FMS. The CM can be in one of two states X=(x_1, x_2): x_1– open, - x_2– closed (number of internal states z=2). The transition from one state to another $U = (u_1, u_2)$ is carried out by control actions: u_1– closed, u_2– switching on the PN (number of input signals m=2). The transition and output functions are given by the following transformations [7]:

$$\begin{cases} \varphi: (x_2, u_1) \to x_1; \\ \varphi: (x_1, u_2) \to x_2; \end{cases} \quad (1)$$

$$\begin{cases} \psi: (x_2, u_1) \to y_1; \\ \psi: (x_1, u_2) \to y_2. \end{cases} \quad (2)$$

The external states of the CM are determined by the output alphabet: $Y = (y_1, y_2)$. $y_1$is off CM, y_2 is on CM (the number of output signals g=2). Therefore, to control the CM, it is necessary to determine the internal states X based on the information U from the corresponding sensors and generate control signals Y in accordance with the transition function.

Now, to define transitions from one state to another, we can represent the network transitions as follows. For each pair (state and input symbol), we define a transition whose input positions correspond to the state and input symbol, and whose output positions correspond to the next state and output.

Thus, $T = \{t_1, t_2\}$ the PN transitions are determined (number of transitions m=2). The PN positions (number of positions $n = m + z + g = 2 + 2 - 2 = 6$) are determined according to the following rule [8]:

$$P = U \cup X \cup Y = \{u_1, u_2, x_1, x_2, y_1, y_2\}$$

$$P = \{p_1, p_2, p_3, p_4, p_5, p_6\}.$$

CONSTRUCT A PN SIMULATING A FA

Creating a transition matrix:

$$C(m, z) = C(2,2) = \{c_{j,i}\}, i = \overline{1,2};\ k = \overline{1,2};\ j = \overline{1,2};$$

According to the (1):

$$c_{1,1} = 0, c_{1,2} = 1,\ c_{2,1} = 2, c_{2,2} = 0.$$

As a result, obtaining matrix transitions:

$$C(2,2) = \begin{vmatrix} 0 & 1 \\ 2 & 0 \end{vmatrix}.$$

Creating the output matrix:

$$B(m,z)=B(2,2)=\{b_{j,i}\},\ i=\overline{1,2};\ k=\overline{1,2};\ j=\overline{1,2};$$

According to the (2):

$$b_{1,1}=0,\ b_{1,2}=1, b_{2,1}=2, b_{2,2}=0.$$

Finally, the results matrix is obtained:

$$B(2,2)=\begin{vmatrix}0 & 1\\ 2 & 0\end{vmatrix}.$$

Determining the input incidence matrix of the Petri net:

$$F(n,m)=F(6,2)=\{f_{i,j}\},\ \text{here } i=\overline{1,6};\ j=\overline{1,2};$$

$n=m+z+g=6$ (n-number of positions, m-number of transitions) .

Determining the output incidence matrix of the Petri net:

$$H(m,n)=H(2,6)=\{h_{j,i}\},\ \text{here } i=\overline{1,6},\ j=\overline{1,2}.$$

Petri net entrance incidence matrix elements formation:

$$F\ (6,2)=\{f_{i,j}\}:$$

Matrix all elements resetting

$$f_{i,j}=0,\ \text{here } i=\overline{1,6};\ j=\overline{1,2};$$

$$\text{if } c_{j,i}\neq 0,\ \text{then } f_{j,j}=1 \text{ and } f_{2+i,j}=1,$$

here $i=\overline{1,2};\ j=\overline{1,2}$:

Because $c_{1,2}\neq 0$, it turns out $f_{j,j}=f_{1,1}=1$,

$$f_{2+i,j}=f_{2+2,1}=f_{4,1}=1;$$

Because $c_{2,1}\neq 0$, it turns out $f_{j,j}=f_{2,2}=1$,

$$f_{2+i,j}=f_{2+1,2}=f_{3,2}=1;$$

As a result, it is obtained that:

$$F(6,2)=\begin{vmatrix}1 & 0\\ 0 & 1\\ 0 & 1\\ 1 & 0\\ 0 & 0\\ 0 & 0\end{vmatrix}$$

Formation of the elements of the output events matrix $H(2,6)=\{h_{j,i}\}$:

Resetting all matrix elements to zero $h_{j,i}=0$, here $j=\overline{1,2};\ i=\overline{1,6};$

if $c_{j,i}\neq 0$, then $h_{j,m+l}=h_{j,2+l}=1$, here $j=\overline{1,2};\ i=\overline{1,2};\ l=c_{j,i};$

Because $c_{1,2}\neq 0$, it turns out $h_{j,m+l}=h_{1,2+1}=h_{1,3}=1$, $(l=c_{j,i}=c_{1,2}=1)$;

Because $c_{2,1}\neq 0$, it turns out $h_{j,m+l}=h_{2,2+2}=h_{2,4}=1$, $(l=c_{j,i}=c_{2,1}=1)$;

Because $b_{j,i}\neq 0$, then $h_{j,m+z+l}=h_{j,2+2+l}=1$, here $j=\overline{1,2};\ i=\overline{1,2};\ l=b_{j,i}$:

Because $b_{1,2} \neq 0$, it turns out that

$h_{j,m+z+l} = h_{1,2+2+1} = h_{1,5} = 1, (l = b_{j,i} = b_{1,2} = 1);$

Because $b_{2,1} \neq 0$, it turns out that

$h_{j,m+z+l} = h_{2,2+2+2} = h_{2,6} = 1, (l = b_{j,i} = b_{2,1} = 2);$

As a result, it is obtained that

$$H(2,6) = \begin{vmatrix} 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{vmatrix} \quad (6)$$

Incident matrix elements of PN formation:

$$D(\mathrm{n,m}) = \mathrm{D}(6,2) = \{\mathrm{d_{i,j}}\}:$$

Because $D = F - H^T$, then $d_{i,j} = f_{i,j} - h_{j,i}$, here $j = \overline{1,2};\ i = \overline{1,6};$

$$D(6,2) = F(6,2) - H^T(2,6) = \begin{vmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \end{vmatrix} - \begin{vmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{vmatrix}.$$

$$D(6,2) = \begin{vmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 1 \\ 1 & -1 \\ -1 & 0 \\ 0 & -1 \end{vmatrix}.$$

Using the obtained results, a graph of a given finite automaton simulating Petri net was created in the CPN Tools system [8] (Fig. 2).

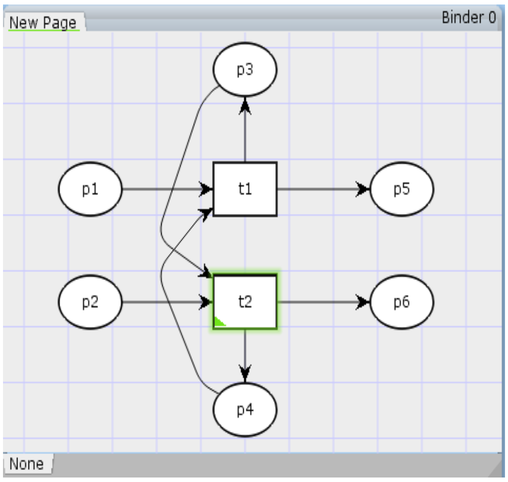

Fig. 2 Graph model of a PN simulating a FA given by transition and output functions.

CONCLUSION

The structure of the equipment of the FMS has been developed and descriptions of its control in the form of production rules and with FA have been created. The PN properties were used for a comprehensive analysis of the control of the computer model for FMS module described by a FA. The algorithm for transforming a FA into a PN was used for this. The model was studied as a result of simulation with computer experiments. The model of the control system of the agile manufacturing module obtained using FA is analyzed using PN. This approach can be considered important in terms of creating a more correct, effective and reliable control system.

REFERENCES

1. J. Mammadov, I. Aliyev, G. Huseynova & G. Orujova. (2021) Algorithmic Support for the Management of the Computer-Aided Design of Flexible Manufacturing System and its Equipment, Cybernetics and System Analyses, v. 57, p. 950-958.
2. I.R. Aliyev, J.F. Mammadov, S.R. Rakhimov. (2022) Selection of information-measuring components on the basis of layout diagram of flexible manufacturing cell, 11th International Conference on Theory and Application of Soft Computing, Computing with Words and Perceptions and Artificial Intelligence-ICSCCW, p.124-131.
3. Mustafaev V.A., Huseynzade Sh.S. (2018) Development of a control model of a processing center using colored networks Petri, Bulletin of computer and information technologies, Moscow, 3(165), p. 36-44. (In Russian).
4. Kumar B., Mahesh V. (2015) Modeling and Analysis of Flexible Manufacturing System with FlexSim. International Journal of Computational Engineering Research (IJCER). 2250-3005, v. 05, Issue 10, p. 1-6.
5. Kolker A.B., Livenets D.A., Kosheleva A.I. (2012) Study of options for creating intelligent robotics systems based on single-board computers and free operating systems. Automation and Software Engineering, Novosibirsk. №.1, p. 84-98.
6. Mukhopad Y.F. (2014) Analysis and synthesis of control automata of complex technical systems, XII All-Russian meeting on control problems, Moscow, June 16-19, p. 7295-7306.
7. Huseynzade Sh.S. (2020) Transformation of finite automata tables into incident matrices of Petri net. Proceedings of the 7th International Conference on Control and Optimization with Industrial Applications (COIA-2020). Baku, Azerbaijan, p. 173-176.
8. Akhmedov M.A., Huseynzade Sh.S., Nasirova E.A. (2019) Development of an algorithm for automating the transformation of a finite state machine into a Petri net". Automation. Modern technologies. Monthly inter-industry scientific and technological journal. Moscow, 2019T. 73, №3, p. 108-112. (In Russian)