Control model of parallel functioning production modules as fuzzy Petri nets

V A Mustafayev¹, I S Zeynalabdiyeva¹ and O Ja Kravets²

¹ Sumqait State University, 43rd block, Sumgait, AZ5008, Azerbaijan
² Voronezh State Technical University, 14 Moscow ave., Voronezh, 394026, Russian Federation

E-mail: Irada0907@mail.ru

Abstract. The article presents the results of research of Data Mining methods with Microsoft SQL Server. Microsoft Clustering algorithm was used for improving the effectiveness of medical prevention and treatment in a cohort of patients with arterial hypertension. There are rationales for monitoring of cardiovascular risk and desire to correct the risk with Data Mining at medical decision support systems. Authors used medical and sociological monitoring data from regional clinical hospital. The segmentation of arterial hypertension patients was performed using Microsoft Clustering algorithm. As a result, a quantitative assessment of the population profile for patients with arterial hypertension was obtained. The authors presented diagrams and profiles of clusters. They were compared. The developed approach is applied for decision support at regional health information management system for reduce of cardiovascular risk.

1. Introduction

While modeling and managing complex systems of various subject areas, operational accounting of many often conflicting factors is required. These primarily include [1, 2]: complex parallel-sequential interaction of the elements of the object; the fuzzy nature of interacting dynamic processes and the space of their states; significant complexity of the tasks; a significant proportion of the human factor, which largely determines the quality and level of modern solutions.

Classical approaches to the construction of decision-making systems were largely oriented towards deterministic or stochastic processes, which fundamentally does not solve the problems of their use in conditions of essentially unclear state space [3, 4]. Mathematical models based on the apparatus of fuzzy sets do not allow explicitly taking into account parallelism and dynamics in their interaction, as well as many parameters and features of the subject area. An effective solution to these problems requires the need for an integrated systematic approach based on the development of models, methods and algorithms using modern mathematical methods, modeling devices and computational intelligence technologies [5, 6]. A promising direction in this case is the application of the mathematical apparatus of the theory of Petri nets (PNs) and their various expansion. In this regard, the presented work developed a fuzzy control model for parallel-functioning production modules in the machining system.

2. Rules for triggering and structural calculation algorithm elements of fuzzy joint ventures

The control model of parallel functioning flexible production modules in a machining system is presented in the form of fuzzy Petri nets.
Fuzzy joint venture is called five [4] \( N = (P, T, I, O, \mu, B) \), where \( P \) and \( T \) - fuzzy sets of positions and transitions; and \( I : P \times T \rightarrow (0,1,...) \) and \( O : T \times P \rightarrow (0,1,...) \) - the functions of the input and output incidents, respectively. The mapping \( \mu : P \rightarrow [0,1] \) assigns to each position a distribution \( p_i \) vector of the degrees of belonging of the chips to the position \( \mu(p_i) \), \( B \) - a base of production rules that determine the process of starting and functioning of fuzzy PNs, which include the conditions for the activity and triggering of transitions, the availability of markers in the position of fuzzy PNs, changing the initial and subsequent markings.

If the distribution vector of the degrees of belonging of each input position \( p_i \in P \) has a component that is not equal to zero, with a number equal to or greater than the number of arcs connecting this position with the transition, \( t_j \in T \) the transition \( t_j \) is triggered; after the transition is triggered, there is a process of redistributing chips in positions. The number of chips in the positions determines the state of the network.

The triggering of transitions and changing states of fuzzy PNs are determined by the rules:

- if the distribution vector of the degrees of belonging of each input position \( p_i \in P \) has a component non-zero with a number equal to or greater than the number of arcs connecting this position with the transition \( t_j \in T \), then the transition \( t_j \) is triggered;
- after the transition is triggered, there is a process of redistributing chips in positions;
- the number of chips in the positions determines the state of the network.

Transition \( t_j \) with marking \( \mu \) is allowed, under the conditions:

- all \( f_{ij} \neq 0 \) are chosen, for \( i = 1, n \);
- for each fixed \( i \), \( \exists \mu_{ik} \neq 0 \), for \( k = f_{ij}, k_i \), where \( k_i \) is the length of the distribution vector of degrees of membership of the \( i \)-th position.

After the transition \( t_j \) is triggered, the dynamics of the network state, the process of redistributing chips by positions and a new marking are determined by the algorithm:

A1. Form the distribution vector of the degrees of membership of each input position after the transition \( t_j \) has been triggered:

- A1.1. calculate the zero component of the distribution vector of membership degrees:
  \[ \mu_{i0} = \bigvee_{\alpha=0}^{f_{ij}} \mu_{i\alpha} \]
  where \( \bigvee \) is the operation of the logical maximum;
- A1.2. determine the remaining components of the distribution vector of degrees of membership:
  \[ \mu_{i\beta} = \mu_{i\beta} + f_{ij}, \beta = 1,2,...,k_j - f_{ij} \];
- A1.3. when the transition \( t_j \) starts, the dimension of the distribution vector of the degrees of belonging of each input position decreases by the number of input arcs:
  \[ k_i = k_i - f_{ij} \];
A2. Form a vector of the distribution of degrees of membership of each output position after the transition is triggered. This vector is equal to the diagonal convolution vector of the Gram matrix of the original output vector and the intermediate vector $r = (r_0, r_1, ..., r_{h_j})$.

A2.1. Select all $h_j \neq 0$, for $z = 1, n$;

A2.2. Calculate the last component of the vector $r$ for all fixed $i$ for $f_{ij} = 0$;

$$r_{h} = \bigwedge_{i = j_0}^{k-1} \mu_{i}$$

where $\bigwedge$ is the operation of the logical minimum;

A2.3. Calculate the zero component: $r_0 = 1 - r_j$;

A2.4 Determine the remaining components: $r_i = 0, i = 1, h_j - 1$;

A3. Form the Gram matrix of the vectors $\mu_z$ and $r^T$ ( $T$ is the transpose sign):

$$G(\mu_z, r^T) = (\mu_{z0} \mu_{z1} ... \mu_{zk}) \times$$

$$\begin{bmatrix}
   r_0 \\
   r_1 \\
   \vdots \\
   r_{h_j}
\end{bmatrix} = \begin{bmatrix}
   \mu_{z0} \land r_{0} \\
   \mu_{z1} \land r_{1} \\
   \vdots \\
   \mu_{zk} \land r_{h_j}
\end{bmatrix}$$

where $g_{ik} = \mu_{z} \land r_{k}$, $i = 0, k_z, k = 0, h_j$.

A4. The vector of diagonal convolution of the Gram matrix $C(G(\mu_z, r^T))$ is calculated:

$$C(G(\mu_z, r^T)) = \begin{bmatrix}
   (\mu_{z0} \land r_{0}) \\
   (\mu_{z1} \land r_{1}) \lor (\mu_{z0} \land r_{1}) \\
   \vdots \\
   (\mu_{z} \land r_{h_j}) \lor (\mu_{z-1} \land r_{1}) \lor ... \lor (\mu_{z-h_j} \land r_{h_j})
\end{bmatrix}$$

Elements of this vector are calculated by the formula:

$$c_i = \bigvee_{k+i = l} (\mu_{z} \land r_{l})$$

for $k = 0, k_z, i = 1, h_j$;

the dimension of the distribution vector of membership degrees increases by the number $h_j$:

$$k_z = k_z + h_j$$

A5. Form the vector $\mu_{z} = (\mu_{z0}, \mu_{z1}, ..., \mu_{zk})$ of the distribution of the degrees of membership of the output position $p_z: \mu_{zk} = c_k, k = 0, k_z$.

3. Fuzzy model of parallel operating production modules in the machining system

Consider a model of parallel-functioning flexible production modules (FPM). In a flexible production system for machining, each FPM consists of one industrial robot (IR), one personal input drive, the same
A processing device for performing the same operations on different workpieces of the same type, and one personal output drive. Each module processes parts of the same type. The workpieces arrive at the personal input drive and await processing. A free device captures the workpiece from the input drive. After the operation is completed, the processed part enters the output personal drive.

The composition of parallel-functioning processing devices includes: 1 – stock of the workpiece; 2, 3, 4 – respectively, the input drives of device 1, device 2, device 3; 5, 6, 7 – respectively, device 1, device 2, device 3; 8, 9, 10 – respectively IR1, IR2, IR3; 11, 12, 13 – respectively, the output drives of the device 1, device 2, device 3; 14 – warehouse of products.

The structure of parallel processing devices is shown in figure 1.

![Figure 1. The structure of parallel FPMs functioning.](image)

In the graph model of parallel functioning FPMs in a flexible production system of machining (figure 2), their states are described by the following positions:

- $p_1$ – stock of the workpiece;
- $p_2, p_3, p_4$ – respectively, the input drives of device 1, device 2, device 3;
- $p_5, p_6, p_7$ – respectively labels excluding downloads of unloaded devices 1, devices 2, devices 3;
- $p_8, p_9, p_{10}$ – respectively, an industrial robot (IR) 1, IR2, IR3 performing loading and unloading of device 1, device 2, device 3;
- $p_{11}, p_{12}, p_{13}$ – respectively, device 1, device 2, device 3 performing operations on the workpiece;
- $p_{14}, p_{15}, p_{16}$ – respectively, the output drives of device 1, device 2, device 3;
- $p_{17}$ – warehouse of products.

Possible events in parallel functioning FPMs are described by the following transitions:

- $t_1$ – the delivery of the workpiece from the warehouse to the position of device 1;
- $t_2, t_3, t_4$ – respectively, loading devices 2, device 3, device 4;
- $t_5, t_6, t_7$ – respectively, the discharge of the device 1, device 2, device 3;
- $t_8, t_9, t_{10}$ – respectively transporting the processed parts from the output of device 1, device 2 and device 3 to the product warehouse.

As a result of a computer experiment, a sequence of triggering transitions $\delta = (t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9, t_{10})$ from the initial marking $\mu^0$ was obtained.
One of the advantages of fuzzy PNs is their use for visualization of the rules of fuzzy products and the implementation on their basis of the conclusion of fuzzy conclusions [7]. In this case, the following interpretation of the position and transitions of fuzzy PNs is used. The rule of fuzzy production of the form “RULE: IF A THEN B,” is presented as a transition of fuzzy joint ventures $t_j \in T$, while condition A of this rule corresponds to the input position $p_j \in P$ of this transition $t_j$, and to the conclusion, the output position $p_j \in P$ of this transition $t_j$ [8].

Based on fuzzy linguistic statements, the model of the rule base of fuzzy products can be written in the following form:

Form 1.
RULE <#>: IF " $b_1$ is $a$" and " $b_2$ is $a$" then " $b_3$ is $c$"

Here the fuzzy saying " $b_1$ is $a$" and " $b_2$ is $a$" represents the terms of the fuzzy products, and the fuzzy saying " $b_3$ is $c$" concludes the rules.

The triggering of transitions and changing states of fuzzy PNs is determined by the following rules:

Rule 1: IF the stock has blanks and input drives of device 1 and devices 2 and devices 3 are missing blanks, THEN transport the blank from the warehouse to the input drives of devices 1, devices 2, devices 3;

Rule 2: IF the input drive of devices 1 has a blank, and devices 1 are free and IR1 is free, THEN devices 1 should be booted;

Rule 3: IF the input drive of devices 2 has a blank, and devices 2 are free, and IR2 is free, THEN devices 2 should be booted;

Rule 4: IF the input drive of devices 3 has a blank, and devices 3 are free, and IR3 is free, THEN devices 3 should be booted;

Rule 5: IF the device 1 has finished the operation on the workpiece, and IR1 is free, THEN the unloading of the devices 1 should be performed;

Rule 6: IF device 2 has finished the operation on the workpiece, and IR2 is free, THEN unloading device 2 should be performed;

Rule 7: IF devices 3 finished operations on the work piece, and IR3 is free, THEN unload devices 3 should be performed;

Rule 8: IF the output drive of devices 1 has a machined part, THEN transport the processed part from the output of devices 1 to the product warehouse;

Rule 9: IF the output drive of device 2 has a machined part, THEN transport the processed part from the output of devices 2 to the product warehouse;

Rule 10: IF the output drive of device 3 has a machined part, THEN transport the processed part from the output of device 3 to the product warehouse.
4. Conclusion
An algorithm has been developed for calculating the structural elements of modified fuzzy Petri nets. Based on the initial data indicated above, experiments were carried out to simulate the network and the results were obtained in the form of a state space. Possible service paths for parallel-functioning FMM have been identified in a flexible manufacturing system for machine processing. An optimal trajectory has been chosen which ignores random delays during simulation of the model. It is shown that the accepted rules for triggering transitions completely describe the process of functioning of fuzzy PNs.

References
[1] Akhmedov M A and Mustafayev V A 2013 Modeling of dynamic interacting processes using stochastic and fuzzy Petri nets Electronic Modeling 4(35) 109-21
[2] Bodyansky E B, Kucherenco E I and Mikhalev A I 2005 Neuro-fuzzy Petri nets in the problems of modeling complex systems (Dnepropetrovsk: System Technology) 311
[3] Peterson J 1984 Theory of Petri Nets and Systems Modeling (Moscow: Mir) 264
[4] Leskin A A, Maltsev P A and Spiridonov A M 1989 Petri nets in modeling and control (Leningrad: Nauka) 133
[5] Ding Z, Zhou Y and Zhou M 2015 A Polynomial Algorithm to Performance Analysis of Concurrent Systems via Petri Nets and Ordinary Differential Equations IEEE Transactions on Automation Science and Engineering 1(12) 295-308
[6] Lopez-Mellado E 2002 Analysis of discrete event systems by simulation of timed Petri net models Mathematics and Computers in Simulation 1(61) 53-9
[7] Borisov V V, Kruglov V V and Fedulov A S 2012 Fuzzy models and networks (Moscow: Telecom) 717
[8] Ahmedov M A, Mustafayev V S and Huseynzade Sh S 2015 Presentation and analysis of fuzzy rules productions using a modified fuzzy Petry nets 5th International Conference on Control and Optimization with Industrial Applications (Baku, Azerbaijan) 184-6