

# Modelling Asynchronous Parallel Process with Petri Net

B.B. Mo'Minov, Khamza Eshankulov

**Abstract:** The asynchronous parallel processes and their structure, the features of the asynchronous parallel process and how to express them in bigram form is the main analysis of this article. Basically, it describes the characteristics and specifications of asynchronous parallel processes and processes in bigram form. The Petri nets are illustrated, the mathematical modeling of asynchronous processes using the Petri net, the marking diagram, the problem of the result and its solutions. Some of the oil and fat production processes were developed and piloted using the Petri net.

**Keywords :** asynchronous parallel process, production, Petri net, time Petri net, marking, status, transition.

## I. INTRODUCTION

When processes occur at the current time interval, these processes are referred to as parallel processes. [1] Pjarikki is classified into species [2]. 1. Asynchronous PJs are areas where the state of the PJ does not depend on any other PJ. 2. Synchronous PJs - PJs are processes that depend on another parallel process. Acynchronous parallel processes can also be performed in parallel [2]. Asynchronous process (AJ) is quadruple, which is a set of non-linear states, the direct relation of the states of the sequence, the set of instances, the set of conclusions.

Mathematical modeling of discrete-state dynamical systems is a relatively young field of control theory. Petri is a simple and convenient tool for modeling multilayered systems and processes. This model was invented in 1939 by German scientist Carl Petri to describe chemical processes. He entered the science in 1962 after defending his thesis "Communication mit Automaten" by Carl Petri [7]. The Petri net is widely used for modeling, and the Petri net is being studied by model.

The model is a representative of the mathematical aspect that is generally considered to be the most studied subject or system. By working with the system model, you can gain new knowledge about it, avoid the high costs, and eliminate the discomfort. Issue: Using the Petri net for modeling and modeling production processes with Petri net for asynchronous PJs.

**Revised Manuscript Received on July 22, 2019.**

B.B. Mo'Minov, A doctor of Tashkent Information Technology University, Uzbekistan

Khamza Eshankulov, A doctorate of Bukhara State University, Uzbekistan

## II. MATERIALS AND METHODS

In production enterprises the terms "state" and "process" are used together. In this case, time does not formally represent the synchronous PJ, but denote the time value in F. At the same time, he always follows his condition. the reaction means that the process stops, leading to the next. Thus, asynchronous causes changes in the state of the process.

## III. DISCUSSION

Initiators are activators of a process, a set of states, and assigned based on the semantic content of the process. Outcome is a set of outcome states.

### 3.1. Modeling Asynchronous PLs via the Petri net.

The Petri net is a directed bigram, with its ends in two types (consisting of states and transitions), crossing points, and crossing positions through arcs and expressed by the initial marking vector [4]. A graph of this type is shown in Figure 2.

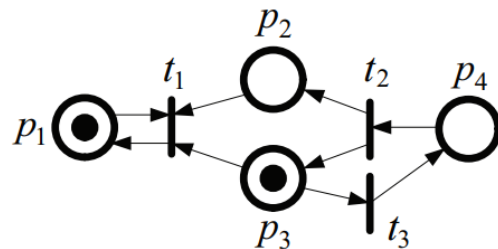


Figure 1: 2-picture.Example of Petri net

As can be seen from Figure 2, the cases are represented by the sign and the transitions are represented by the sign. Through the arcs, the cases are connected only through the transitions, and the transitions are linked through the states, so any condition is the input state or the exit state of one or more transitions. It is represented by a number of cases or a marker, which is indicated by the position of the dots in its position [4,5]. Generally there are several points in the case, and the marking is called marking.

The number of components of the initial marking vector is equal to the number of cases, the value of each component is zero, and if there is no marker in this case, the markers are denoted by bold dots in the condition and are counted as natural numbers. The dynamics of the Petri net is represented by the movement of pointers that represent local characteristics based on network rules.

If each input condition contains at least one pointer, the jump is considered initiated.

After a while, the passage will work. The resulting transition will be indispensable if one marker is removed from each entry point at the same time and one marker is added to the output. The Petri net works by switching processes from one marking to another by working on excited processes.

The Petri net is defined by four  $\langle P, T, E, \mu^0 \rangle$ , where  $P$  - the final set of cases (otherwise, place or conditions);  $T$  - final set of transitions (events);  $E$  - set of arcs,  $E \subseteq P \times T \cup T \times P$ ;  $\mu^0$  initial marking (vector in designation),  $\mu^0 : P \rightarrow N$  where all positive numbers as  $N = \{0, 1, 2, \dots\}$ . Specify the incoming and outgoing transitions. and analog entries represent a set of states of entry and departure.

Specify the incoming and outgoing transitions as  $t_j \in T$  ни  $I(t_j)$  ба  $O(p_j)$  and  $I(t_j)$  and  $O(p_j)$  analogue records represent a set of  $p_j \in P$  inputs and outputs. Petri net gradient representation allows static representation of the Petri net structure. As we have already seen, the dynamics, the marking, the change mechanism, and the initiation of transition (implementation) are usually modeled on the model [10,11].

The graph is marked by a dot inside the circle marking. If we just speak about the principle of the Petri net, it is through the movement of markers in the Petri network and the workings of transitions in their implementation. The switch may work, such a transition as  $t_k \in T$  must have at least one marker in the input state  $I(t_k)$  (for each jump and in each case  $p_s \in I(t_k)$ , equal to  $\mu_s \in \mu(p_s)$  and equal to  $\mu_s \geq 1$  the number of pointers  $P_s$  in the position). Once transitions are triggered, transitions occur in time.

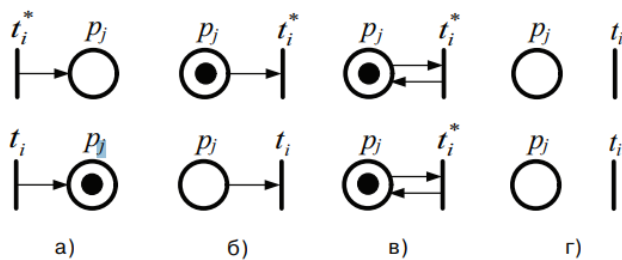


Figure 2: Marking change rule

Figure 3 shows the Petri lattice in four sections showing the rules for changing the labeling for each case. These are the local rules.

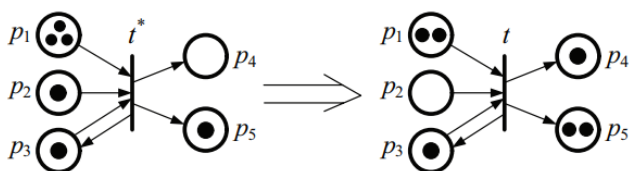


Figure 3: Results of training switch.

$$I(t) = \{p_1, p_2, p_3\}, \quad O(t) = \{p_4, p_5, p_6\}$$

$$i(p_1) = 3, i(p_2) = 1, i(p_3) = 1, i(p_4) = 0, i(p_5) = 1$$

are given.

At this point  $t$  is triggered by  $t^*$ , there is only one marker in all input states, and after the transition, the current marking is positioned to the right as shown in the figure.

$$i'(p_1) = 3 - 1 = 2, \quad i'(p_2) = 1 - 1 = 0$$

$$i'(p_3) = 1 - 1 + 1 = 1, \quad i'(p_4) = 0 + 1 = 1$$

$$i'(p_5) = 1 + 1 = 2$$

At this, the switch  $t$  will not work the second time because one of its inputs has no markup and  $P_2$  is empty. The excitement of an arbitrary transition will result in the replacement of the marking on the Petri net. In the current case, the Petri netting marking represents the state of the net, which is considered a temporary condition. The Petri net makes the transition from one marking to another by triggering the transition. In general, the dynamics of the Petri net can be described by the following three  $\langle \mu^0, \rightarrow, M \rangle$ .  $\mu^0$  - Initial marking,  $\rightarrow$  - tracking of marking  $\mu^u \rightarrow \mu^v$  (as you can see from marking  $\mu^u$  to marking  $\mu^v$ ).  $M$  - a set of possible markings.

The Petri net is also represented in vector form  $(\mu = \mu_1, \mu_2, \dots, \mu_n)$ . The number of components  $i$  is equal to the number of cases in the Petri net.  $1 \leq i \leq n$ . The value of the component,  $\mu_i = \mu(p_i)$   $i$  - the number of marking of the component.

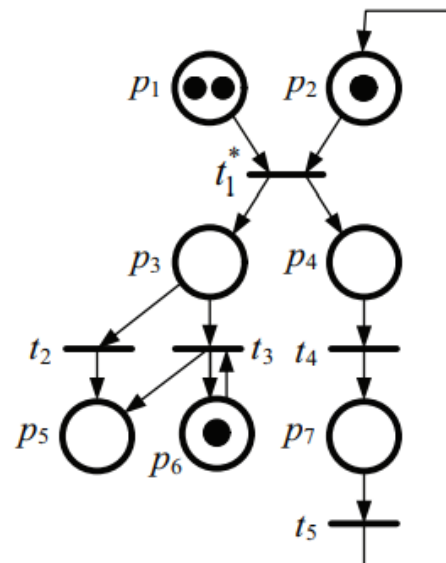


Figure 4. Petri net of oil separation process of oil and fat enterprise

Petri nets have seven stages of oil separation process  $P = \{p_1, p_2, \dots, p_7\}$  and five transitions  $T = \{t_1, t_2, t_3, t_4, t_5\}$  in oil and fat factories.

In the vector there is a starting marker  $\mu^0 = \{\mu_1^0, \mu_2^0, \dots, \mu_7^0\}$  with three non-zero components, so  $\mu_1^0 = 2, \mu_2^0 = \mu_6^0 = 1$  the initial marking is visible and there is a switch that can trigger one marking on the initial marking  $\mu^0 = \{2100010\}$ . Such marking as  $t_4, t_2, t_3$  is done through excitement and transitions  $\mu^1 = \{1011010\}$   $\mu^0 \rightarrow \mu^1$

During the initiation of the model, the following will be detected:  $t^2$  and  $t^3$  at the same time, and the behavior of the Petri net is determined by the marking  $\mu^2 = (1001110)$ .  $t^4$  works out, it is marked with  $\mu^3 = (1010011)$

Simultaneously  $t_2$  and  $t_4, t_3$  and  $t_4$ , the behavior of the Petri net  $\mu^4 = (1000111)$  is determined by marking.

### 3.2. Diagram of marking

The Petri net is a labeled diagram, which is called the case diagram (margins), which is derived from the movement dynamics of the net. Marking diagrams can be represented as a diagram in the direction chart. At the ends of the graph are a set of markings that can be reached, if there is a direct switch  $\mu^a, \mu^b \in M$  from  $\mu^a$  to  $\mu^b$  ( $\mu^a \rightarrow \mu^b$ ). We illustrate the Petri lattice in Figure 5 as a diagram of markings (Figure 6).

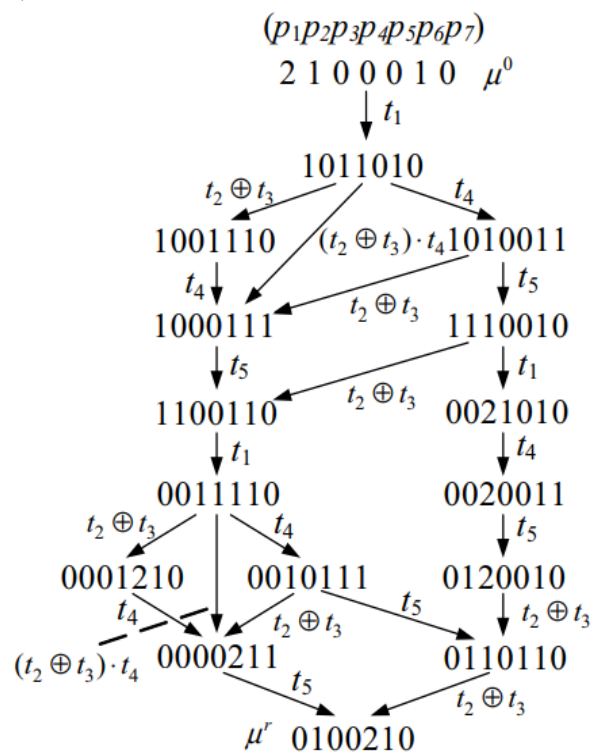


Figure 5. Marking diagram for Petri net Generally, a startup can result in multiple output markings from one marking. The model does not specify when the asynchronous process transitions are performed.

### 3.3. Interactions between events

In Petri's net, one has to obey one of the following fundamental relationships.

- the transition  $t_j$  follows a transition  $t_i$  (written in Figure 7a), which represents a sequence of events;
- $t_i$  or  $t_j$  one of the transitions may work, but not together ( $t_i \rightarrow t_j$  as seen in Fig. 7b), which represents the value of free choice (nodeterminism).
- Both transitions can work,  $t_i$  and  $t_j$  both transitions are optional, or both are displayed together ( $t_i || t_j$  written as Fig. 7v), and this event does not exist. These transitions can work sequentially, parallel, or simultaneously.

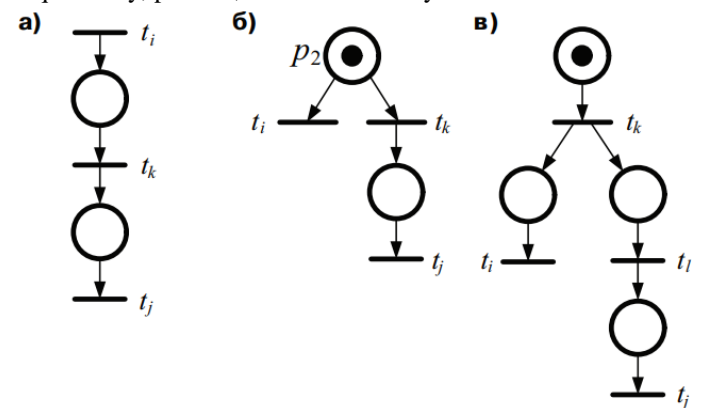


Figure 6: Different relationships between transitions in the Petri net

### 3.4. Petri net in interpreting an asynchronous PJ model.

The set of M markers on the net of Petri contains a model of the asynchronous PJs  $\mu_0$ . F- is the suitability of mark markers. M is the sum of the equivalents between the markings in the set, and the equivalence of the equivalent marking classes is established. places will be unavailable. Real objects are not represented by a single Petri net, but  $\langle P, T, E \rangle$  are represented by a network of several identical structures. It is represented by a triad and initial labels are different. An example is given in Figure 8 for understanding such Petri nets.

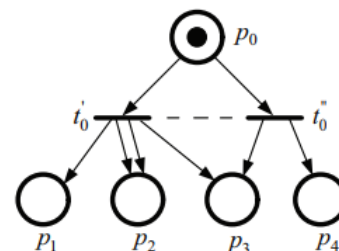


Figure 7: Primary Petri net

If one of the additional transitions runs or sets one  $t_0'$  or  $t_0''$  the markings on it. This is evident from the fact that its output  $p_1, 2p_2, p_3$  is connected by two arcs  $p_1, 2p_2, p_3$ , which may not be the Petri net, but given the use of multigraph  $t_0'$  instead of the graph  $P_2$ , it is also a Petri net.

IV. CLASSIFICATION OF PETRI NET

Petri net classification is based on existing configurations (via static constraints) and possible marking (based on dynamic constraints).

4.1 Tasks and algorithms of analysis

Analysis of a particular Petri net can be understood as a set of methods, algorithms, and techniques used to study its static (structural) and dynamic (behavior) characteristics.

4.2 Structural constraints and viability analysis

Unmarked Petri net will be structurally limited  $\mu^0$  if the optional Petri net  $\langle N, \mu^0 \rangle$  is hoisted and labeled by the Petri net  $\mu^0$ . Also, if the unmarked N Petri net is structurally viable, if there is  $\langle N, \mu^0 \rangle$  at least one marker  $\mu^0$

The analysis of such features is usually done by limiting the structure and  $I(t) \cap O(t) \neq \emptyset$  is called the Petri net. We define the event matrix  $C = \parallel C_{ij} \parallel_{n \times m}$  of the net-labeled Petri net.

$$c_{ij} = \begin{cases} \beta(t_j, p_i), a_{\beta} \alpha \beta(t_j, p_i) \neq 0, \\ -\alpha(p_i, t_j), a_{\alpha} \alpha(p_i, t_j) \neq 0, \\ \text{акс холда } 0, \end{cases}$$

In this case  $1 \leq i \leq n, 1 \leq j \leq m$ , where  $n = |P|, m = |T|$ . For example, for the non-labeled Petri net  $\langle N, \mu^0 \rangle$ , shown in Figure 9, the matrix of incidences (events) is represented in the table. We represent it as a vector  $\bar{\sigma} = (\sigma_1, \dots, \sigma_m)$ . In this case, each component  $\bar{\sigma}_j (1 \leq j \leq m)$   $t_j$  is the same expressed by  $\sigma$  components. The actions that lead to the marking  $\mu$  through excessive transitions  $t_j$  are determined by the equation  $\mu' = \mu + C_j(1)$

Where  $C_j$  - j-vector,  $C$  the string of the Incidences (events) matrix, corresponds to the  $t_j$  transition. The (1)-formula summarizes the feasibility of transitions  $\sigma$ .  $\mu^k = \mu^0 + \bar{\sigma} \cdot C_j$ . (2)

Equation (2) is called the fundamental equation of the Petri net.

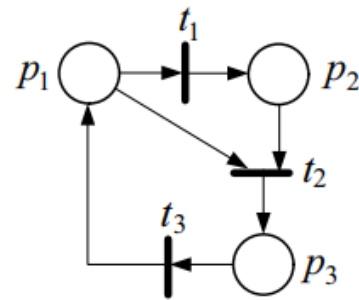


Figure 8: Unmarked Petri net

Table 1. Event Matrix

	p1	p2	p3
t1	-1	1	0
t2	-1	-1	1
t3	1	0	-1

It has a pure Petri net and a net unmarked Petri net.  $\langle C, \mu^0 \rangle$  or  $\langle C, \rightarrow \mu^k$  a sequence which executes a number of transitions, leading to marking.

4.3 The issue of viability of Petri net

The question of viability of the Petri net is that theoretically the probability of a voluntary passage of the net is analyzed. If the voluntary marking in M is optically excited, the Petri net is called life (active). The vitality issue has two types of modifications.

- for this  $\langle P, T, E, \mu_0 \rangle$  Petri net, whether all of its transitions are life-span  $t_i \in T$  - is a matter of general vitality;
- The life of some transitions in Petri's net  $t_i \in T$  is a matter of viability.

3.4. Issue of Appointment

The main purpose of the Petri nets analysis is to achieve the solution of the existing question marking, whether it is penetrated by the initial marking, whether the marking or marking scheme is present. The following standard is developed.  $\langle P, T, E, \mu^0 \rangle$  and some (target)  $\langle \mu^* \rangle$

Petri nets that are labeled.  $\mu^*$  whether marking sets belong to M, whether the initial labeling results, in other words, some predicates need to be installed.

$$\alpha = \begin{cases} 1, a_{\beta} \mu^* \in M, \\ 0, a_{\beta} \mu^* \notin M. \end{cases}$$

4.4.1. Modification of the solution problem

- It is possible to explain in more detail by adding additional requirements to the above formula.



- If there is a  $\mu^*$  marking set, then these labels are called  $\mu^* = (00...0)$  a blank marking set. This modification is known as the issue of free labeling.
- Make a mark diagram (graph) with the ends of the M markings and mark the transitions through the arcs, which will work when the current markup changes, so this will be a complete achievement.
- Generation (detection) of the achieved marking package M.

**Solution of a dispute**

The first example is the Petri net, shown in Figure 9

$$\mu^k = (1010) + \bar{\sigma} \cdot \begin{pmatrix} 0-1-1 & 0 \\ 0+2+1-1 \\ 0 & 0-1+1 \end{pmatrix}$$

$$\mu^k = (1801)$$

It is necessary to determine whether marking is achieved, put  $\mu^k$  on the left side of the equation and find the set of transitions  $\bar{\sigma}$  made. This is equal to the equation  $\bar{\sigma} = (045)$  and follows in the following sequence.  $\sigma = (t_3 t_2 t_3 t_2 t_3 t_2 t_3)$ . This approach is not universal. The first reason is that  $\bar{\sigma}$  the recovery of vector transitions  $\bar{\sigma}$  in the general case is unknown. The second reason is that the existence of the solution of this equation is only necessary, but not sufficient for the existence.

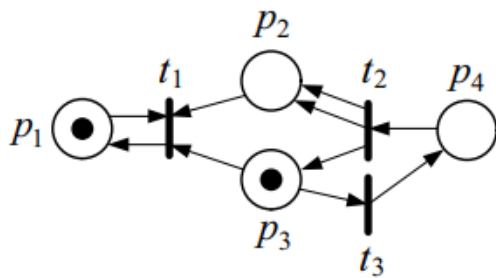


Figure 9: Petri net 4.

$\langle PN, f \rangle$  Timely Petri net the pair is called the interim Petri net, where  $PN = \langle P, T, E, \mu^0 \rangle$  the ordinal Petri net  $T$  is a delayed  $f$  function  $f : T \rightarrow N^+$ ,  $N^+$  - a set of non-negative integers. The transition takes place in the unit of time  $\tau_i = f(t_i)$  after excitation. If several transitions in this label conflict, then the smallest delay will work. If two or more delays of these delays  $\tau_i$  are activated and the delays are the same, then all of these delays occur simultaneously.

An example of this is the process of measuring the intake of raw materials by electronic scales at oil and fat enterprises. At

the same time, raw materials are delivered to the oil and gas factory, and the load is taken in turn by the weighing machine. If the point N enters the hold position, then N must be present in the active mode and leave. The sequence of machines is

shown in Figure 11: means weighing machines  $x_2$  will start only if two conditions are met - someone is waiting for the queue (status metric) and the balance is active due to the arcs  $x_2 | x_1$ .

$x_2$  machine will run when the machine has enough time, i.e. when two are upright. After the machine has been measured, it activates the arc.

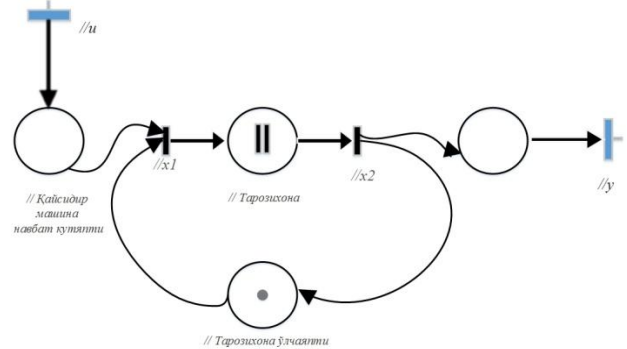


Figure 10. Model of the process of measurement of seeds in the oil-and-oil enterprise through the time Petri net

There are several new features of the Petri net. An analysis of these is presented in Table 2.

Table-II. Advantages and disadvantages of modifications of Petri net

	Advantages	Disadvantages
Ordinator Petri net	The ease of use and analysis is simple	It is used for narrow and limited issues.
The Hierarchical Petri net	The Petri net avoids large volumes of records	The scope of application is limited
Color Petri net	Ordinar is more commonly used than the Petri net, modeling different types of octopus	Compared to a simple Petri net, analysis is more complex, and it is not possible to model and analyze time-dependent processes.
Time Petri net	It is possible to model and analyze time-dependent processes	The color Petri is used less than the net
Petri net for Uzbeks	It is convenient for modeling of Uzbek-oriented programming systems.	It is not possible to model and analyze time-dependent processes.



**Modeling of the basic elements of the production process on the Petri net (for example in the oil and fat enterprise)**

1. Technological operations (processes of cleaning, smelting, grinding) of the oil and gas industry can be described in several different ways. In this case, when the number of inputs and outputs is the same, the model of Petri shown in Figure 1 is modeled.

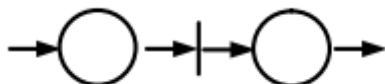


Figure 10: Modeling of technological operations

2. If there is a possibility of simultaneous production of a resource K during production, its model will be as shown in Figure 12. An example is a press shop. In this case, if the amount of crushing is equal to K, then the calibration

$$S = \frac{M}{K}$$

value is calculated M in times.

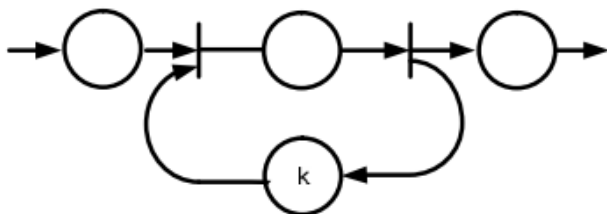


Figure 11. Inbound modeling of K production ability during technological operations

The production process may be when a resource is recycled into two or more parts, then we model the Petri net as shown in Figure 13. the substance goes to the next production process.

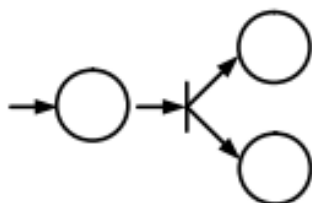


Figure 12. Modeling the process of dividing resources into several parts

The reverse process is the process of pooling resources, that is, the two cases form one state after a process (Figure 13). An example of this is the process of purifying the extracted oil by gasoline at the oil and fat factories. At the same time, the state of unrefined oil and gasoline-based products are converted to pure oil during the cleaning process.

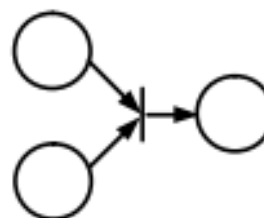


Figure 13. Modeling of merging processes

**V. CONCLUSION**

This article describes the asynchronous parallel process, bigraf expression, and the construction of an asynchronous process model by Petri's net. The methods of application of the Petri net in modeling synchronous processes are shown. The diagrams of Petri's net markings and their dynamics are analyzed.

**REFERENCES**

- [1] Мараховский В. Б., Розенблюм Л. Я., Яковлев А. В. Моделирование параллельных процессов. Сети Петри. Курс для системных архитекторов, программистов, системных аналитиков, проектировщиков сложных систем управления. - Санкт-Петербург: Профессиональная литература, АйТи-Подготовка, 2014. - 400 с.
- [2] Ачасова С.М., Бандман О.Л. Корректность параллельных вычислительных процессов. – Н.: Наука, 1990. – 253 с.
- [3] S. Achasova, O. Bandman, V. Markova, et al., Parallel Substitution Algorithm. Theory and Application., Singapore: World Scientific, 1994.
- [4] Котов, В.Е. Сети Петри [Текст] / В.Е. Котов. – М.: Наука, 1984.
- [5] A.Gruler, M. Leucker, and K. Scheidemann, "Calculating and modeling common parts of software product lines," in International Software Product Line Conference, September 2008, pp. 203–212.
- [6] Kurkovsky, S., Loganantharaj, R.: Extension of petri nets for representing and reasoning with tasks with imprecise durations. Appl. Intell. 23(2), 97–108 (2005). <https://doi.org/10.1007/s10489-005-3415-8>
- [7] PhD „Kommunikation mit Automaten“; University of Technology Darmstadt, 1962 [http://edoc.sub.uni-hamburg.de/informatik/volltexte/2011/160/pdf/dis\\_s\\_petri.pdf](http://edoc.sub.uni-hamburg.de/informatik/volltexte/2011/160/pdf/dis_s_petri.pdf)
- [8] Васильев В.В., Кузьмук В.В. Сети Петри, параллельные алгоритмы и модели мультипроцессорных систем. К.: Наукова Думка, 1990. 216 с.
- [9] Ratzner, A., Wells, L., Lassen, H., Laursen, M., Qvortrup, J., Stissing, M., Westergaard, M., Christensen, S., Jensen, K., 2003, CPN Tools for Editing, Simulating, and Analysing Coloured Petri Nets. Proceedings of the 24th ICATPN 2003, Netherlands, 450-462.
- [10] K. Jensen, S. Donatelli, M. Koutny. Transactions on Petri Nets and Other Models of Concurrency. — Springer, 2011.
- [11] W. Reisig. Elements of Distributed Algorithms: Modeling and Analysis with Petri Nets . — Springer, 2010.
- [12] W. Penczek, A. Pólrola. Advances in Verification of Time Petri Nets and Timed Automata: A Temporal Logic Approach. — Springer, 2010.

