



# Structural Analysis of the Process Based on Extended Petri Nets with Semantic Relations

O. Kryukov<sup>(✉)</sup> and A. Voloshko

Tula State University, 92, Lenina Avenue, Tula 300012, Russia

**Abstract.** Process analysis and optimization is impossible without simulation. Models make it possible to represent and visualize the main interrelationships between the individual stages of the process, to investigate and predict the course of its development, to find bottlenecks and the ways of modernization. One of the important tasks in process optimization is the analysis of its structure, since various options for its organization impose corresponding restrictions on the possibility of its rebuilding. This paper discusses the issues of structural analysis of processes: the definition of linear structures, branches and cycles. Algorithms for the automatic analysis of the process represented by the extended Petri net with semantic relation are developed. An example of searching for such structures in the process of repairing a truck is considered.

**Keywords:** Petri nets · Manufacturing process · Simulation · Distributed process · Process structure · Analysis · Parallelization

## 1 Introduction

The current state of the industry is characterized by an increasing level of automation. Use of modern systems allows building and analyzing models based on process logs. However, the process of analyzing models remains a rather difficult task, since it depends on the goals and objectives of the analysis, as well as the modeling method.

Processes are a set of interrelated operations and have a complex structure. When analyzing and optimizing processes, it is often important to determine the type of structure to which the considered part of the process belongs. In this regard, it is necessary to develop methods for identifying process structures and, in the future, methods for their optimization.

## 2 Overview of Main Approaches to Structural Analysis

Issues related to the structure of the system can be resolved both at the stage of model creation by building special structural models, and at the stage of models analysis by carrying out structural analysis.

Structural modeling involves modeling the organizational structure of systems and subsystems, such as: informational, organizational, functional, management, i.e. modeling the composition and relations between the elements [1]. A detailed analysis of

possible methods for business processes modeling is presented in [2]. Special aspects of production processes within business processes and possible options for application of some models of production lines are considered in [3]. A large number of researchers suggest using the SADT methodology [4, 5] based on the IDEF [6–8] or BPMN notation and its extensions [9–11] for structural modeling. A comparative analysis of these approaches is presented in [12]. At the same time, methods have been developed to automate the creation of such models [13–15], which allows in the future also to automate the analysis of these models and the synthesis of new solutions.

Structural analysis can be applied for different types of models. Its main purpose is to search for static characteristics of the system related to the interconnections of its elements and subsystems at various levels, as well as to classify structures that are important for decision-making. Depending on the type of the studied models and on the selected classification parameter, the approaches to structural analysis may also differ [16–18]. It is shown in [19] that one of the promising methods for modeling production processes is mathematical apparatus of Petri nets and their extensions. Methods for structural analysis of Petri nets are discussed in [20–22]. However, these analyzes do not consider the possibility of restructuring the net to execute a series of steps in parallel. Therefore, it is necessary to develop your own algorithms that allow you to define the key structures that affect the possibility and the choice of the method of parallel organization of the process.

### 3 Extended Petri Nets with Semantic Relations as an Apparatus for Simulation of Distributed Manufacturing Processes

The simplest extended Petri net with semantic relations (ExpNSR) can be specified by the following set:

$$\Pi = \left\{ A, \left\{ Z^C, \tilde{R}^C, \hat{R}^C \right\}, \left\{ Z^S, \tilde{R}^S, \hat{R}^S \right\} \right\},$$

where  $A = \{a_{1(a)}, \dots, a_{j(a)}, \dots, a_{J(a)}\}$  – a finite set of places;  $Z^C = \{z_{1(z^C)}^C, \dots, z_{j(z^C)}^C, \dots, z_{J(z^C)}^C\}$  – a finite set of transition by control relations;  $\tilde{R}^C$  – an incidence matrix of size  $J(a) \times J(z^C)$  that maps the set of places to a set of transitions by control relations;  $\hat{R}^C$  – an incidence matrix of size  $J(z^C) \times J(a)$  that maps the set of transitions by control relations to a set of places;  $Z^S = \{z_{1(z^S)}^S, \dots, z_{j(z^S)}^S, \dots, z_{J(z^S)}^S\}$  – a finite set of transition by semantic relations;  $\tilde{R}^S$  – an incidence matrix of size  $J(a) \times J(z^S)$  that maps the set of places to a set of transitions by semantic relations for token;  $\hat{R}^S$  – an incidence matrix of size  $J(z^S) \times J(a)$  that maps the set of transitions by semantic relations to a set of places for token.

In addition, the following transition functions are set:

- input function of transitions by control relations:

$$I_A(Z^C) = \left\{ I_A(z_{1(z^C)}^C), \dots, I_A(z_{j(z^C)}^C), \dots, I_A(z_{J(z^C)}^C) \right\};$$

- output function of transitions by control relations:

$$O_A(Z^C) = \{O_A(z_{1(z^C)}^C), \dots, O_A(z_{j(z^C)}^C), \dots, O_A(z_{J(z^C)}^C)\};$$

- input function of transitions by semantic relations:

$$I_A(Z^S) = \{I_A(z_{1(z^S)}^S), \dots, I_A(z_{j(z^S)}^S), \dots, I_A(z_{J(z^S)}^S)\};$$

- output function of transitions by semantic relations:

$$O_A(Z^S) = \{O_A(z_{1(z^S)}^S), \dots, O_A(z_{j(z^S)}^S), \dots, O_A(z_{J(z^S)}^S)\}.$$

More information about the mathematical apparatus of ExPNSR is presented in [23].

Between the places of the net, the control precedence relation is fulfilled. Place  $a_i$  is considered to be preceding by control relations for  $a_j$  (specified as  $a_i <_C a_j$ ) if there is such a control path from the starting place of the process  $a_S$  to place  $a_j$  that it includes place  $a_i$ .

For the starting place of the process, it is true:

$$\exists z_{i(z^C)}^C (I_A(z_{i(z^C)}^C) = a_S) \wedge \nexists z_{j(z^C)}^C (O_A(z_{j(z^C)}^C) = a_S).$$

### 4 Structures of ExPNSR

In the process model, built in accordance with the definition of the ExPNSR, three types of structures can be distinguished, formed as a result of various combinations of places and control transitions:

- linear section ( $Ln$ );
- cycle ( $C$ );
- branching ( $Br$ ).

The linear section (Fig. 1) is characterized by a sequence of linear control structures, implying the execution of some operation immediately after the completion of the previous one.

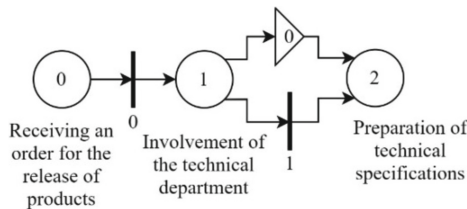
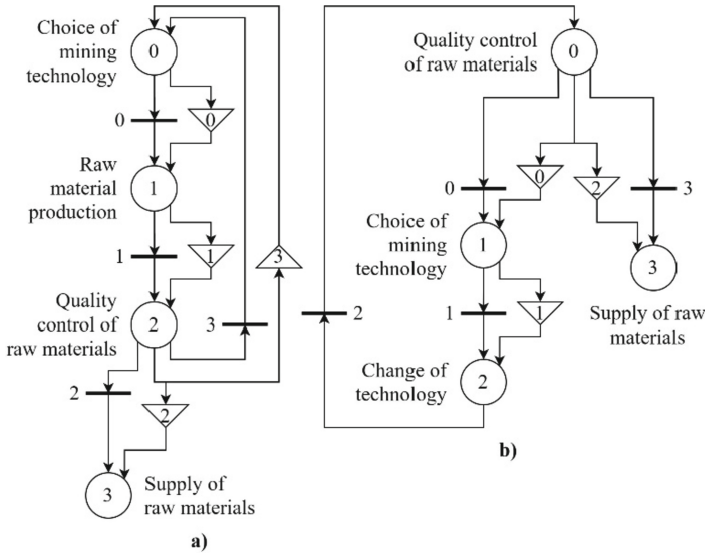


Fig. 1. An example of a linear section.

Cycles imply a return by control to the initial place  $a_{cB}$ , the presence of a condition (branch-place)  $a_{cond}$ , which takes control out of the cycle, and the body of the cycle ( $Cp$ ), which contains a sequence of linear sections, branches and other cycles. Among the cycles, there are cycles with a post-condition (Fig. 2a) and cycles with a pre-condition (Fig. 2b).



**Fig. 2.** Examples of cycles: a - with a post-condition; b - with a pre-condition.

Cycles are characterized by the following:

$$\begin{aligned} &\exists z_{i(z^C)}^C \left( O_A(z_{i(z^C)}^C) = a_{cB} \wedge I_A(z_{i(z^C)}^C) = a_{i(a)} \right) | a_{cB} <_C a_{i(a)}; \\ &\left| Z_{O_i}^C \right| > 1 \forall z_{i(z^C)}^C \in Z_{O_i}^C \left( I_A(z_{i(z^C)}^C) = a_{cond} \right) \wedge \nexists Br_i \left( a_{xB} = a_{cond} \right); \\ &\exists z_{i(z^C)}^C \left( I_A(z_{i(z^C)}^C) = a_{cond} \wedge O_A(z_{i(z^C)}^C) = a_{out} \right) | a_{out} \notin Cp \\ &\wedge a_{out} \neq a_{cB}. \end{aligned}$$

Branch (Fig. 3) is characterized by the presence of a condition describing options for actions according to its result, and the number of such options can vary from 2 to  $n$ , and contains the initial place the final place  $a_{xe}$ , as well as branches  $B_{xi}$ ,  $i = \overline{2, n}$ . Each branch can contain a sequence of linear sections, other branches and cycles. Branches are characterized by the following:

$$\begin{aligned} &\left| Z_{O_i}^C \right| > 1 \forall z_{i(z^C)}^C \in Z_{O_i}^C \left( I_A(z_{i(z^C)}^C) = a_{xB} \right) \wedge . \\ &\nexists C_i \left( a_{cB} = a_{xB} \right) \wedge \nexists z_{j(z^C)}^C \in Z_{O_i}^C \left( O_A(z_{j(z^C)}^C) = a_{i(a)} \right) | a_{i(a)} <_C a_{xB}. \end{aligned}$$

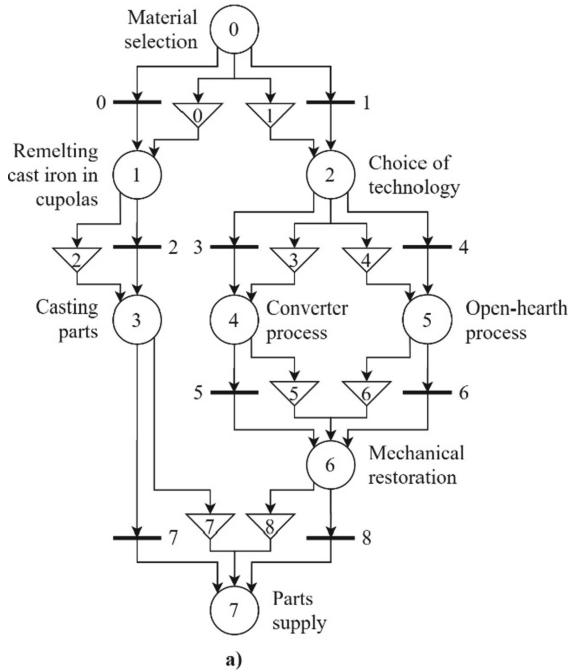


Fig. 3. Example of branch.

### 5 Structural Analysis of ExpNSR

To analyze the structure of the ExpNSR, it is necessary to build a model of the selected process. For this, each stage of the process is transformed into a place of the net, and the sequence of operations is specified by forming control transitions between places. The formation of semantic transitions, that is, the indication of material and informational dependencies between the elements of the process, is not required at this stage.

The constructed net is analyzed in the following order:

- Search for the starting place of the net.
- Search for cycles: determination of initial places.
- Finding branches: determining initial places and forming branches.
- Determination of conditions for cycles and nested cycles with the same initial place.
- Filling the contents of the branches of each of the branches.
- Filling the body of each of the cycles.

Of the described stages, the most difficult stage is the stage of determining the conditions of the cycles, since at this stage the nesting of cycles with the same initial place is also determined. Let's consider this stage in more detail.

Determination of conditions for cycles and nesting with the same initial place is performed as follows for each not yet considered loop  $C_{i(C)}$  using the following notation:

- *Conds* – a set of places that are possible conditions for  $C_{i(C)}$ ;
- $Z_O$  – a set of transitions to which  $a_{cB_{i(C)}}$  leads;
- $eC$  – the set of already considered cycles, the beginning of which coincides with  $C_{i(C)}$ ;
- $parC$  – the set of cycles in which can be nested.

1. Determine the set  $Z_O$ :
2. Determine the set *Conds*:
3. If  $|Conds| = 1$ , then there is no nesting with the same starting place, go to step 3, otherwise go to step 4.
4. If  $|Z_O| > 1$ , then a cycle with a precondition is found:

$$\begin{aligned} a_{cond_{i(C)}} &= a_{cB_{i(C)}}; \\ a_{out_{i(C)}} &= a_{i(a)} | \exists z_i^C(z_i^C) (I_A(z_i^C(z_i^C)) = a_{cB_{i(C)}} \wedge \\ O_A(z_i^C(z_i^C)) &= a_{i(a)} \wedge a_{i(a)} \notin Cp_{i(C)}); \end{aligned}$$

else, a cycle with a postcondition is found:

$$\begin{aligned} a_{cond_{i(C)}} &= Conds_0; \\ a_{out_{i(C)}} &= a_{i(a)} | \exists z_i^C(z_i^C) (I_A(z_i^C(z_i^C)) = a_{cB_{i(C)}} \wedge \\ O_A(z_i^C(z_i^C)) &= a_{i(a)} \wedge a_{i(a)} \neq a_{cB_{i(C)}}. \end{aligned}$$

complete the consideration of the cycle.

5. Determine the set  $eC$ :

$$eC = \{C_{j(C)}\} | a_{cond_{j(C)}} = a_{cB_{i(C)}} \vee a_{cond_{j(C)}} \in Conds.$$

6. If  $|eC| = 0$  and  $|Z_O| > 1$ , then the considered cycle is senior with the precondition:

$$\begin{aligned} a_{cond_{i(C)}} &= a_{cB_{i(C)}}; \\ a_{out_{i(C)}} &= a_{i(a)} | \exists z_i^C(z_i^C) (I_A(z_i^C(z_i^C)) = a_{cB_{i(C)}} \wedge \\ O_A(z_i^C(z_i^C)) &= a_{i(a)} \wedge a_{i(a)} \notin Cp_{i(C)}); \end{aligned}$$

complete the consideration of the cycle, else go to step 6.

7. Removing extra places from *Conds*:

$$\begin{aligned} Conds &= Conds \setminus a_{i(a)} | |Z_{Or}| = 1 \forall z_i^C(z_i^C) \in Z_{Or} (I_A(z_i^C(z_i^C)) = a_{i(a)}) \\ Conds &= Conds \setminus a_{i(a)} | \exists C_{j(C)} (a_{cond_{j(C)}} = a_{i(a)}) \end{aligned}$$

## 8. Completing the formation of the cycle:

$$\begin{aligned}
a_{cond_{i(C)}} &= a_{i(a)} | a_{i(a)} \in Conds \wedge \\
\forall a_{j(a)} \in Conds &(a_{j(a)} <_C a_{i(a)}); \\
a_{out_{i(C)}} &= a_{i(a)} | \exists z_{i(Z^C)}^C I_A(z_{i(Z^C)}^C) = a_{cond_{i(C)}} \wedge \\
O_A(z_{i(Z^C)}^C) &= a_{i(a)} \wedge a_{i(a)} \neq a_{cB_{i(C)}}.
\end{aligned}$$

9. If  $|eC| = 1$ , then this cycle is the parent for the considered one:

$$Cp_{eC_0} = Cp_{eC_0} \cup C_{i(C)};$$

else we search for the parent loop:

$$\begin{aligned}
parC &= \{C_{j(C)}\} | C_{j(C)} \in eC \wedge a_{cond_{j(C)}} \neq a_{cB_{j(C)}}; \\
Cp_{j(C)} &= Cp_{j(C)} \cup C_{i(C)} | C_{j(C)} \in parC \wedge \\
\forall C_{k(C)} \in parC &(a_{cond_{j(C)}} <_C a_{cond_{k(C)}}).
\end{aligned}$$

Complete the consideration of the cycle.

As an algorithm for filling these structures, consider the filling of branches. Filling the body of cycles is done in a similar way.

To fill in the branching  $Br_{i(B_r)}$ , each of its branches  $Bx_{i(B_r)}$  is analyzed according to the following algorithm with the notation:

- $p$  – considered place;
- $Ln$  – formed linear section;
- $Z_I$  - set of transitions leading to  $p$ ;
- $np$  – the next place to be considered;
- $Z_{In}$  - set of transitions leading to  $np$ .

## 1. Find the first place of the branch:

$$p = O_A(z_{i(Z^C)}^C) | I_A(z_{i(Z^C)}^C) = a_{xB_{i(B_r)}}.$$

## 2. Form the current linear section:

$$Ln = \emptyset$$

## 3. Form the current linear section:

$$Z_I = \{z_{i(Z^C)}^C\} | O_A(z_{i(Z^C)}^C) = p \wedge \forall a_{i(a)} \in I_A(z_{i(Z^C)}^C) (a_{i(a)} <_C p).$$

4. If  $|Z_l| > 1$ , then branch completion is detected:

$$\begin{aligned} a_{xe_i(Br)} &= p; \\ Bx_{i_i(Br)} &= Bx_{i_i(Br)} \cup Ln|Ln \neq \emptyset; \end{aligned}$$

complete the consideration of the branch.

5. If  $\exists Br_j(Br) (a_{xB_j(Br)} = p)$ , then a nested branch is detected:

$$Bx_{i_i(Br)} = Bx_{i_i(Br)} \cup Br_j(Br);$$

go to step 6, else go to step 11. If  $a_{out_j(Br)} = \emptyset$ , then fill the branching  $Br_j(Br)$  in accordance with the algorithm.

6. Set the following place for consideration:

$$np = a_{xe_j(Br)}.$$

7. Save the linear section:

$$Bx_{i_i(Br)} = Bx_{i_i(Br)} \cup Ln|Ln \neq \emptyset.$$

8. Form a set  $Z_{In}$ :

$$Z_{In} = \left\{ z_{i(ZC)}^C \right\} | O_A \left( z_{i(ZC)}^C \right) = np \wedge \forall a_{i(a)} \in I_A \left( z_{i(ZC)}^C \right) (a_{i(a)} <_C np).$$

9. If  $|Z_{In}| > |Bx_j(Br)|$ , then complete the consideration of the branch, else  $p = np$ ; continue the consideration of the branch from step 2.

10. If  $\exists C_i(C) (a_{cB_i(C)} = p)$ , then nested cycle detected:

$$\begin{aligned} Bx_{i_i(Br)} &= Bx_{i_i(Br)} \cup C_i(C); \\ p &= a_{out_i(C)}; \\ Bx_{i_i(Br)} &= Bx_{i_i(Br)} \cup Ln|Ln \neq \emptyset; \end{aligned}$$

continue the consideration of the branch from step 2.

11. Fill in the linear section:

$$Ln = Ln \cup p.$$

12. Set the next place for consideration:

$$p = a_{i(a)} | \exists z_{i(ZC)}^C \left( I_A \left( z_{i(ZC)}^C \right) = p \wedge O_A \left( z_{i(ZC)}^C \right) = a_{i(a)} \right);$$

continue the consideration of the branch from step 3.

The described methodology will make it possible to form data structures that allow to unambiguously determine the presence and place of nonlinear structures in the process model, which, at the transformation stage, will facilitate the introduction of the hierarchy into the network by replacing the whole structure with one place.



## 6 The Example of Structural Analysis of the Process

Let's consider the process of structural analysis using the example of the technological process of repairing a truck. Figure 4 shows an EXPNSR that simulates this process. Structural analysis does not need the study of information and material relations, therefore the presented net includes transitions by only control relation in order to make the reading of the graph easier.

In accordance with the generalized structural analysis algorithm, the starting place of the network is the first determined. Since the place 0 fulfills the condition  $\exists z_i^C \left( I_A \left( z_i^C \right) = a_S \right) \wedge \nexists z_j^C \left( O_A \left( z_j^C \right) = a_S \right)$ , then  $a_S = 0$ .

At the next stage, the initial places of the cycles are determined, as a result of which structures of the following type are formed:

$$C_0 = \{a_{cB} = 3, a_{cond} = \emptyset, a_{out} = \emptyset, Cp = \emptyset\};$$

$$C_1 = \{a_{cB} = 10, a_{cond} = \emptyset, a_{out} = \emptyset, Cp = \emptyset\};$$

$$C_2 = \{a_{cB} = 20, a_{cond} = \emptyset, a_{out} = \emptyset, Cp = \emptyset\};$$

$$C_3 = \{a_{cB} = 32, a_{cond} = \emptyset, a_{out} = \emptyset, Cp = \emptyset\}.$$

Next, the initial places that allows to choose the further development of the process are determined and the branches are formed:

$$Br_0 = \{a_{xB} = 4, \{B_0 = \emptyset, B_1 = \emptyset\}, a_{xe} = \emptyset\};$$

$$Br_1 = \{a_{xB} = 11, \{B_0 = \emptyset, B_1 = \emptyset\}, a_{xe} = \emptyset\};$$

$$Br_2 = \{a_{xB} = 21, \{B_0 = \emptyset, B_1 = \emptyset\}, a_{xe} = \emptyset\}.$$

As a result, the definition of cycle conditions, previously formed structures are supplemented to the following form:

$$C_0 = \{a_{cB} = 3, a_{cond} = 3, a_{out} = 31, Cp = \emptyset\};$$

$$C_1 = \{a_{cB} = 10, a_{cond} = 10, a_{out} = 30, Cp = \emptyset\};$$

$$C_2 = \{a_{cB} = 20, a_{cond} = 20, a_{out} = 25, Cp = \emptyset\};$$

$$C_3 = \{a_{cB} = 32, a_{cond} = 32, a_{out} = 34, Cp = \emptyset\}.$$

When filling out the branches, linear sections are formed that belong to the every variant of the process development, and the nested cycles and branches are determined:

$$Br_0 = \{a_{xB} = 4, \{B_0 = \{Ln_0\}, B_1 = \{Ln_1, C_1\}\}, a_{xe} = 30\};$$

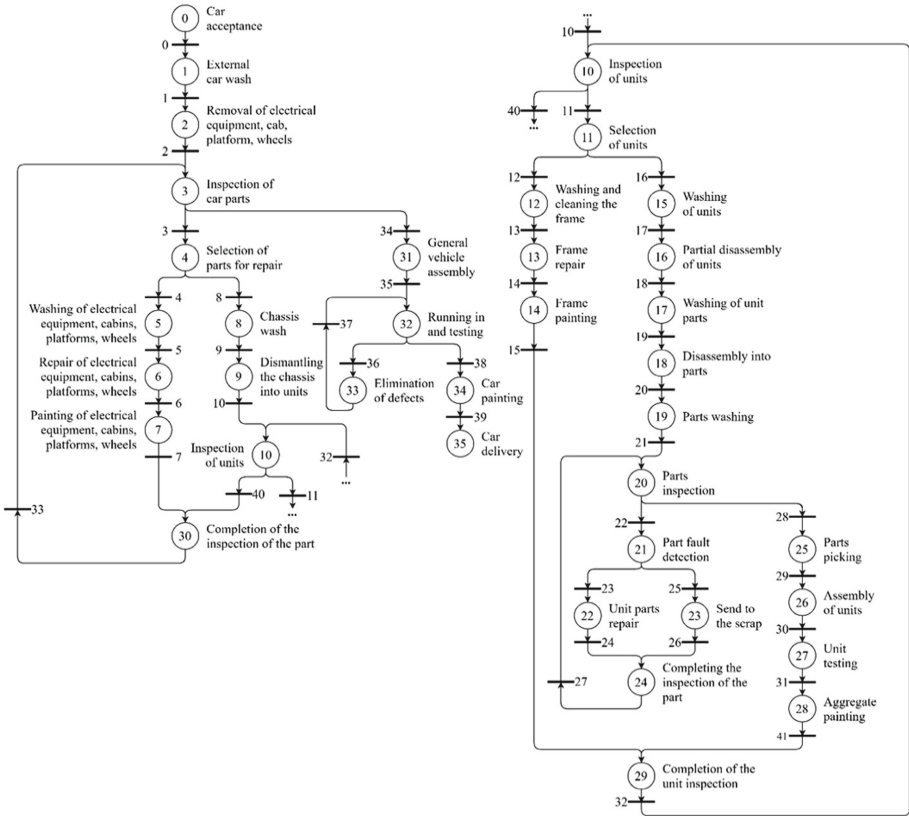


Fig. 4. Technological process diagram of a truck repair.

$$Br_1 = \{a_{xB} = 11, \{B_0 = \{Ln_2\}, B_1 = \{Ln_3, C_2, Ln_4\}\}, a_{xe} = 29\};$$

$$Ln_0 = \{5, 6, 7\};$$

$$Ln_1 = \{8, 9\};$$

$$Ln_2 = \{12, 13, 14\};$$

$$Ln_3 = \{15, 16, 17, 18, 19\};$$

$$Ln_4 = \{25, 26, 27, 28\};$$

$$Ln_5 = \{22\};$$

$$Ln_6 = \{23\}.$$

The same actions are performed when filling the body of the cycles. As a result the following structures are formed:

$$C_0 = \{a_{cB} = 3, a_{cond} = 3, a_{out} = 31, Cp = \{Br_0, Ln_7\}\};$$

$$C_1 = \{a_{cB} = 10, a_{cond} = 10, a_{out} = 30, Cp = \{Br_1, Ln_8\}\};$$

$$C_2 = \{a_{cB} = 20, a_{cond} = 20, a_{out} = 25, Cp = \{Br_2, Ln_9\}\};$$

$$C_3 = \{a_{cB} = 32, a_{cond} = 32, a_{out} = 34, Cp = \{Ln_{10}\}\};$$

$$Ln_7 = \{30\};$$

$$Ln_8 = \{29\};$$

$$Ln_9 = \{24\};$$

$$Ln_{10} = \{33\}.$$

The sets of positions  $\{0, 1, 2\}$ ,  $\{31\}$  and  $\{34, 35\}$  are linear sections that do not belong to cycles or branches, and are not separated into separate structures. Thus, the general structure of the algorithm is as follows:

$$Alg = \{\{0, 1, 2\}, C_0, \{31\}, C_3, \{34, 35\}\}.$$

## 7 Conclusion

The paper describes the main nonlinear structures found in the model of a process based on ExPNSR. A method for analyzing the ExPNSR for the presence of nonlinear structures is proposed, which can be used during the optimization of production processes. The information obtained as a result of its application will make it possible to more efficiently distribute the resources available to the enterprise at the stage of planning the modernization of production or when developing new projects.

**Acknowledgments.** The research was partially supported by Grants of the President of the Russian Federation for state support of young Russian scientists – Ph.D. (project No. MK-1160.2020.9).

## References

1. Danelyan, T.Ya.: Structural modeling. *Stat. Econ.* 6:166–169 (2014)
2. Trifonov, P.V.: Comparative analysis of models for describing business processes in an organization. *Manag. Sci. Modern World* **1**, 307 (2016)
3. Pignasty, O.M.: Review of models of controlled production processes of production lines production systems. *Econ. Comput. Sci.* **34**(7), 204 (2015)
4. Fedorova, O.V., Mamaeva, A.A., Yakunina, E.A.: Application of SADT and ARIS methodologies for modeling and management of business processes of information systems. *Bull. Voronezh State Univ. Eng. Technol.* **80**(1), 75 (2018)
5. Minnullina, A., et al.: Production strategy development for an energy company based on SADT. *E3S Web Conf. EDP Sci.* **217**, 07015 (2020)
6. Alontseva, E.N., Anokhin, A.N., Saakyan, S.P.: *Structural Modeling of Processes and Systems*. IATE NIYAU MIFI, Obninsk (2015)
7. Tsebreiko, K.N.: Functional-structural modeling of the system of automation of accounting of commodity-material values. *J. Sci.* **8**, 23 (2019)
8. Baghbani, M., Iranzadeh, S., Bagherzadeh Khajeh, M.: Extracting manufacturing process map in the form of the IDEF model prerequisite for the Implementation of PFMEA in the Sugar Industry. *J. Mod. Process. Manuf. Prod.* **7**(2), 79–93 (2018)
9. Aspidou, M.: *Extending BPMN for modeling manufacturing processes*. diss., Master thesis, Business Information Systems, TU/e (2017)
10. Erasmus, J., et al.: Using business process models for the specification of manufacturing operations. *Comput. Ind.* **123**, 103297(2020)
11. Yakimov, I.M., et al.: Structural modeling of business processes in BPMN EDITOR, ELMA, RUNAWFE systems. *Bull. Kazan Technol. Univ.* **17**(10) (2014)
12. Kelani, K.T.: *Modelling techniques: comparison between BPMN2.0 and IDEF (0&3)*. Doctoral dissertation (2018)
13. Serysheva I.A., Chirkova, E.Yu.: Automation of analysis of technological schemes of production processes. *Bull. Irkutsk State Tech. Univ.* **12**(83) (2013)
14. Bocharov, E.P., Aleksentseva, O.N., Ermoshin, D.V.: Simulation model of the production process as an element of the management system of an industrial enterprise. *Appl. Inform.* **3** (2007)
15. Khan, R., et al.: A Framework for automated reengineering of BPMN models by excluding inefficient activities. In: *Proceedings of the 2020 9th International Conference on Software and Computer Applications*, pp. 147–151 (2020)
16. Honti, G., Dörgő, G., Abonyi, J.: Review and structural analysis of system dynamics models in sustainability science. *J. Clean. Prod.* **240**, 118015 (2019)
17. Jadhav, J.R., Mantha, S.S., Rane, S.B.: Analysis of interactions among the barriers to JIT production: interpretive structural modelling approach. *J. Ind. Eng. Int.* **11**(3), 331–352 (2014). <https://doi.org/10.1007/s40092-014-0092-4>
18. Lager, T.: A structural analysis of process development in process industry: a new classification system for strategic project selection and portfolio balancing. *R&D Manag.* **32**(1), 87–95 (2002)
19. Stepanenko, V.E., Frolov, D.N., Maryin, B.N.: Method of simulation modeling of the organization of production processes using extended Petri nets. *Sci. Notes Komsomolsk-on-Amur State Tech. Univ.* **1**(7), 71–78 (2011)
20. Abdul-Hussin, M.H.: A Structural analysis of petri nets-based siphons supervisors of flexible manufacturing systems. In: *Proceedings of IEEE-UKSim-AMSS, 17th International Conference on Computer Modelling and Simulation*, pp. 235–241 (2015)

21. Cabasino, M.P., Giua, A., Seatzu, C.: Structural analysis of Petri Nets. *Control of Discrete-Event Systems*, pp. 213–233 Springer, London (2013). <https://doi.org/10.1007/978-1-4471-4276-8>
22. Park, J.: *Structural Analysis and control of Resource Allocation Systems using Petri Nets*, Georgia Institute of Technology (2000)
23. Voloshko, A., Ivutin, A., Kryukov, O.: Petri nets based digital adaptive models for simulation of manufacturing processes. In: *Advances in Automation II: Proceedings of the International Russian Automation Conference, RusAutoConf2020*, September 6–12, 2020, pp. 223–234. Springer International Publishing, Sochi (2021)