Available online at www.sciencedirect.com
ScienceDirect
Energy Reports 6 (2020) 1337–1343
www.elsevier.com/locate/egyr

2020 7th International Conference on Power and Energy Systems Engineering (CPESE 2020), 26–29 September 2020, Fukuoka, Japan

The use of Petri computing networks for optimization of the structure of distribution networks to minimize power losses

Murat Asanov (PhD)[a], Sergey Kokin (Doctor of Science)[b], Salima Asanova (PhD)[a], Kalmurza Satarkulov (PhD)[c], Stepan Dmitriev (PhD)[b,*], Murodbek Safaraliev[b]

[a] *Department of the Theoretical Foundations of Electrical Engineering, Kyrgyz State Technical University, Bishkek, Kyrgyzstan*
[b] *Department of Automated Electrical Systems, Ural Federal University, Ekaterinburg, Russian Federation*
[c] *Branch of Academician H.A. Rakhmatulin in Tokmok at the Kyrgyz State Technical University, Tokmok, Kyrgyzstan*

Received 4 November 2020; accepted 6 November 2020

Abstract

The paper suggests a self-organizing multi-component computational algorithm as a solution to the problem of optimizing the structure of distribution electrical networks to minimize the loss of power. The suggested algorithm is consistent with the method of branches and borders and uses the apparatus of the Petri computer networks (PCN) apparatus. The PCN apparatus has a universal computational capability to process symbolic-numeric data, which along with the solution of calculating problems, provides for the structural and logical analysis of the systems and processes under study. The structure of the PCN based algorithm is similar to the studied system, which provides for better visualization and convenience of interpretation, modification, and implementation of this algorithm on one or more computers by paralleling computational processes for better system performance. Computing modules within the general text of the algorithm can be arranged in any given order and solve the problem by organizing themselves in the process of functioning.

Peer-review under responsibility of the scientific committee of the .

Keywords: Distribution electrical networks; Optimization of network structure; Loss of electricity; Petri networks; Algorithm; Graph; Computational modules

1. Introduction

As a rule, distribution electrical networks (DEN) of 6–10 kV comprising of aerial cables or cable lines have several power centers and operate in open mode. Power flow by line and total power losses throughout the network depends on the disposition of the network disjunction points. With this regard, the task of determining the optimal

* Corresponding author.
E-mail addresses: a_ms07@mail.ru (M. Asanov), s.e.kokin@uruf.ru (S. Kokin), a_sm07@mail.ru (S. Asanova), satarkulov46k@mail.ru (K. Satarkulov), sa.dmitriev@urfu.ru (S. Dmitriev), murodbek.safaraliev@urfu.ru (M. Safaraliev).

https://doi.org/10.1016/j.egyr.2020.11.024

Peer-review under responsibility of the scientific committee of the .

points for the network disjunction points, i.e. the choice of the optimal configuration providing for minimal energy loss, is very important [1–4].

The necessity to solve this problem arises both at the stage of design of electrical networks, and, to an even greater extent, during their operation. In the general case, the optimal network configuration can be determined by a simple comparison of all the possible variants of disjunctions with a consecutive calculation of power losses for each variant. However, this is practically impossible due to the huge number of such variants in the analysis of real networks.

Currently, as shown in [5], several methods have been developed to solve the problem: one of them is based on determining the economic current distribution in the analyzed network [6], and the other is based on one of the discrete programming methods, namely the branch and bound method [7,8].

This paper demonstrates the capabilities of Petri computing networks (PCN) [9,10] when developing a self-organizing multicomponent computational algorithm that solves the problem of optimizing the structure of DEN to minimize power losses. The paper defines the task of choosing the optimal network configuration and suggests the scheme for the solution of this task using the branch-and-bound method [5].

PCN apparatus has a universal computing capability for processing alpha-numeric data, which allows, along with solving computational problems, to perform a structural and logical analysis of the systems and processes under study. This paper shows the capabilities of the PCN apparatus by solving the problem of choosing the optimal disjunction points (i.e., choosing the optimal structure) of power distribution networks.

2. The use of Petri computing networks to optimize the structure of electric power distribution networks

Distribution electric networks (DEN) (6–10 kV) consist of sets of source nodes (SN) $I_{SN} = \{SN_1, SN_2, \ldots, SN_{n_{SN}}\}$, load nodes (LN) $I_{LN} = \{LN_1, LN_2, \ldots, LN_{n_{LN}}\}$, network sections (NS) $I_{\mathrm{NS}} = \{NS_1, NS_2, \ldots, NS_{n_{NS}}\}$, connecting the corresponding nodes from a plurality of nodes $I_N = I_{SP} \cup I_{LC}$, forming a complex closed structure, and, as a rule, they work in open mode. Each of the load nodes $i \in I_{LN}$ can be fed only from one of the various source nodes I_{SN}. Which parts of the network are excluded by the unbuttoning of the respective switchgear equipment (SE) depend on the flow of active P_j^E and reactive capacities Q_j^E at all sections $j \in I_{NS}$ at the given active P_i^E and reactive capacities Q_i^Ein all load nodes $i \in I_{LN}$, and hence also the sum of the net power losses $\Delta P_{\sum}$ are negligible. In this regard, work [7] defines the task of choosing the optimal structure of the open circuit (OSOC) network to minimize the total power loss $\Delta P_{\sum}$ therein:

$$P_{\Sigma} = \frac{1}{{U_{nom}}^2} \sum_{j \in I_{NS}^{OC}} R_j \left[\left(\sum_{i \in I_{LN}^j} P_i^E \right)^2 + \left(\sum_{i \in I_{LN}^j} Q_i^E \right)^2 \right], \tag{1}$$

where U_{nom} is the nominal voltage of the network; R_j - active resistance of the jth section; $I_{NS}^{OC} \subset I_{SN}$ many sections of the open circuit (OC); $I_{LN}^j \subset I_{LN}$ - a plurality of load nodes for which power from the power source passes through section *j*. It also proposed a scheme for its solution using the branch-and-bound method under the assumption that the set of power nodes I_{SP} of the source network is replaced by one generalized equivalent power node SP_0. The essence of this scheme is as follows:

- if for the analyzed subset k the value is greaterΔP_{Σ}^{LL} then the power loss ΔP_{Σ} in at least one previously considered open circuit found by (1), then this subset is excluded from further consideration as unpromising, since there is no optimal option in it, and then follows go to the analysis of another alternate subset of paragraph 1;
- for each obtained subset k using the algorithm for finding the tree of shortest paths with a root in the vertex SP_0 in the equivalent undirected network graph, taking the value as a measure of "length" for each jth section, we calculate the value of the lower limit (LL) of power losses ΔP_{Σ}^{LL} by the formula

$$\Delta P_{\Sigma}^{LL} = \sum_{j \in I_{LN}}^{\Sigma} \left[\left(P_i^E\right)^2 + \left(Q_i^E\right)^2 \right] L_{min.i} L \sum_{j \in I_{NS}^i}^{\sum} l_{imin.i} \tag{2}$$

- if for the analyzed subset k the value is greaterΔP_{Σ}^{LL} then the power loss $\Delta P_{\sum}$ in at least one previously considered open circuit found by (1), then this subset is excluded from further consideration as unpromising, since there is no optimal option in it, and then follows go to the analysis of another alternate subset of paragraph 1;
- otherwise, it analyzes the subset obtained in paragraph 1 as a result of further partitioning of the current kth subset. The optimal structure of an open circuit (OSOC) network, in this case, will be obtained after considering all subsets of options for open circuits (trees).

However, in many complex systems and process modeling tasks, and in particular their management, the problem arises of finding not just arbitrary algorithms to solve the given problems, but structured ones (multi-component and structurally similar) computational algorithms that are efficient in terms of their formation and modification, their visibility in interpretation, convenience in their implementation on a single computer or several computers, to reduce the calculation time by organizing consecutive parallel calculations.

One of the powerful means for producing such algorithms is the Petri computing network apparatus [9,10], which is the further development and generalization of self-modifying [11] and algebraic [12] Petri networks, and has the universal computational ability to process symbolic and numerical information, which makes it possible to perform structural and logical analysis of the systems and processes under investigation, in addition to solving calculation problems.

In this section, therefore, the OSOC is used to solve the PCN selection problem.

3. A structured task of selecting a network for the OSOC to be built by PCN

Representing the network under study as an undirected graph $(I_{SN} \cup I_{LN}, I_{NS})$, and taking into account the fact that the network operating in open mode does not contain circuits, and each of its load nodes $i \in I_{LN}$ can be powered by only one of the many source nodes I_{SN}, the open network circuit can be considered as the spanning subgraph $\left(I_{SD} \cup I_{LN}, I_{NS}^{OC}\right)$ of the original graph $(I_{SN} \cup I_{LN}, I_{NS})$, formed from a set of trees$\left\{\left(\{i\} \cup I_{\mathrm{LN}(i)}, I_{NS(i)}\right) | i \in I_{SN}\right\}$ that do not intersect at the vertices, where each ith tree $\left(\{i\} \cup I_{LN(i)}, I_{NS(i)}\right)$ has a root at the vertex corresponding to a certain node i from the set of nutrition nodes I_{SN} (here) $\left|I_{NS(i)}\right| = \left|I_{LN(i)}\right|$.

Given this definition, the problem of selecting the OSOC can be interpreted as the problem of finding the spanning subgraph $\left(I_{SN} \cup I_{LN}, I_{NS}^{OC}\right)$ of the initial graph $(I_{SN} \cup I_{LN}, I_{NS})$ in the form of a set of trees $\left\{\left(\{i\} \cup I_{LC(i)}, I_{NS(i)}\right) | i \in I_{SN}\right\}$ that do not intersect at the vertices for which the power distributed over all sections $j \in I_{NS}$ of the power P_j^E, Q_j^E at given capacities P_i^E, Q_i^E at all load nodes $i \in I_{LN}$, will lead to a minimum value total power loss in the network $\Delta P_{\sum}$

$$\min_{I_{NS}^{OC} \subset I_{NS}} \left\{ \Delta P_{\sum} = \sum_{j \in I_{NS}^{OC}} \left[(P_j^E)^2 + (Q_j^E)^2 \right] \cdot l_j \text{ at} \right. \tag{3}$$

$$P_j^E = \sum_{i \in I_{LN}^j} P_i^E, \, Q_j^E = \sum_{i \in I_{LN}^j} Q_i^E, \, I_{LN}^j \subset I_{LN}, \forall j \in I_{NS}^{OC}, \tag{4}$$

$$\left(\left(I_{SN} \cup I_{LN}, I_{NS}^{OC} \right) = \bigcup_{i \in I_{SN}} \left(\{i\} \cup I_{LN(i)}, I_{NS(i)} \right) \right) \subset \left(I_{SN} \cup I_{LN}, I_{NS} \right), \tag{5}$$

$$\left|I_{LN(i)}\right| = \left|I_{NS(i)}\right|, \forall i \in I_{SN}, \tag{6}$$

$$\bigcap_{i \in I_{SN}} I_{LN(i)} = \varnothing, \bigcap_{i \in I_{SN}} I_{NS(i)} = \varnothing, \tag{7}$$

$$\left. \left| I_{LC} = \bigcup_{i \in I_{\mathrm{UP}}} I_{LC(i)} \right| = \left| I_{NS}^{OC} = \bigcup_{i \in I_{SP}} I_{NS(i)} \right| \right\} \tag{8}$$

To solve the problem of choosing OSOC, in the above formulation (3)–(8), a structured computational algorithm is proposed that is consistent with the branch-and-bound (BB) method and uses the PCN apparatus [9,10].

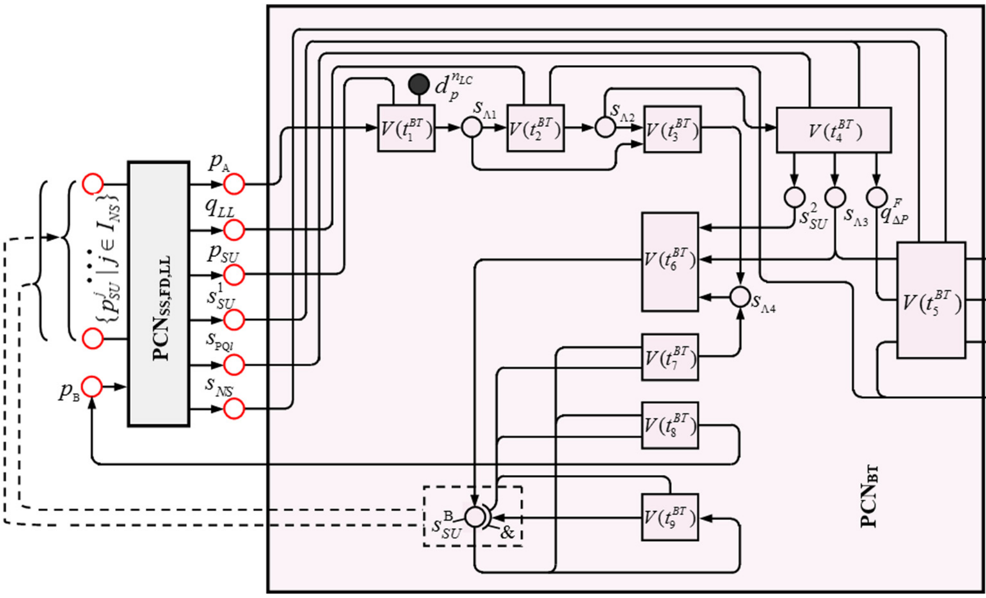

Fig. 1. The functional scheme of the PCN_{OSOC} to solve the problem of choosing the optimal structure of an open circuit (OSOC).

4. Functional diagram of a Petri computational network for solving the problem of choosing the OSOC (PCN_{OSOC})

The functional diagram of the PCN_{OSOC} is shown in Fig. 1. It consists of two functional blocks PCN_{BT}, $PCN_{SS,FD.LL}$, interconnected through intermediate positions $p_B, \{p^j_{SU}|j \in I_{NS}\}$, A detailed description of the $PCN_{SS,FD.LL}$, in the form of interconnected computational modules, is outside the scope of this section. Here, it is considered as a one-piece functional unit with input $p_B, \{p^j_{SU}|j \in I_{NS}\}$ and output $\left\{p_A, q_{LL}, p_{SU}, s^1_{SU}, s_{PQl}, s_{NS}\right\}$ information positions that performs the function assigned to it.

The positions s^0_{SU}, s^0_{NS}, $q^0_{\Delta P}$ are the output for the PCN_{OSOC}, and they are intended for storing intermediate tentative optimal solutions obtained at the stage of searching for the optimal solution to the problem (3)–(8). The state of these positions, at the end of the search procedure, corresponds to the optimal solution to the problem under consideration. Interpretation of all positions used in the PCN_{OSOC} functional diagram is given further when describing a computational algorithm corresponding to this scheme.

The block $PCN_{SS,FD.LL}$ serves to find spanning subgraphs (SS) in the form of a set of trees of the shortest paths satisfying conditions (5)–(8), and analysis of their topologies, to form topological information I^{OC}_{NS}, I^i_{NS}, I^j_{LN} to calculate:

- the length L_{imin} of the shortest path I^i_{NS} to each ith load node from the corresponding power node according to the formulas from (2), i.e.:
  $L \sum_{j \in I^i_{NS}} l_{jimin}$, $l_j = \frac{R_j}{U^2_{nom}}$, and reliability $\rho\left(I^i_{NS}\right)$ of this path: I^i_{NS}: $\rho\left(I^i_{NS}\right) = \prod_{j \in I^i_{NS}} \rho_j$;
- flow distribution (FD), P^E_j, Q^E_j $\forall j \in I^{OC}_{NS}$ according to the formulas from (4);
- estimates of the lower limit (LL) of power losses $\Delta R^{LL}_{\sum}$ according to the formula from (2), i.e.:

$$\Delta P^{LL}_{\sum} = \sum_{i \in I_{LC}}^{\sum} \left[\left(P^E_i\right)^2 + \left(Q^E_i\right)^2\right] \cdot L_{imin}$$

where ρ_j is the reliability of the jth section, i.e. the probability of its existence in the original graph $(I_{SN} \cup I_{LN}, I_{NS})$.

The PCN_{SS} block is designed to calculate the actual power loss $\Delta R_{\sum}$ according to formula (3) and to implement branching algorithms provided for by the branch-and-bound method (dividing the set of variants $M_{SS}\left(\left(I_{SP} \cup I_{LC}, I^{OC}_{NS}\right) \subset (I_{SP} \cup I_{LC}, I_{NS})\right)$ of spanning subgraphs satisfying conditions (5)–(8), the initial graph into disjoint subsets), and cutting off unpromising for further consideration of branches (subsets that do not contain the optimal variant of a spanning subgraph).

Note. The principle diagrams of computational modules and their description codes are not considered here. It should be noted only that the set of PCN set of codes for computing modules is both a constructive language for describing an implemented algorithm and at the same time - a computer-based program for its implementation in the programming environment currently being developed based on the PCN.

5. The computational algorithm implemented in the operation of the $\mathrm{PCN_{OSOC}}$

The essence of the computational algorithm laid down in the $\mathrm{PCN_{OSOC}}$ is as follows:

- at the initial stage of the search for OSOC, with $k = 0$, we believe that all SE in the analyzed network are closed, i.e. $\mu\left(r_{SU}^{j}\right) = 1$, $\forall j \in I_{NS}$, which corresponds to the initial graph $(I_{SN} \cup I_{LN}, I_{NS})$ of the network. Control position status $\mathrm{r_V}$: $\mu(r_V) = 1$. State of a constant position $d_r^{n_{LC}}$: $\mu\left(d_r^{n_{LN}}\right) = |I_{LN}| = n_{LN}$. All other positions of the $\mathrm{PCN_{OSOC}}$ (Fig. 1) are empty.
- the algorithm implemented during the operation of the $\mathrm{PCN_{OSOC}}$ at each kth tact of the OSOC ($k = 0, 1, 2, \ldots$) search includes the following computational procedures (paragraphs 1–10):

(1) When $\mu(r_V) = 1$ $\mathrm{PCN_{SS,FD.LL}}$ begins to function for the analysis of the current graph $\left(I_{SP} \cup I_{LN}, I_{NS} \backslash I_{NS}^{off}(k)\right)$ obtained at the kth branch cycle, where $I_{NS}^{off}(k) \subset I_{NS}$ is the set of network sections turned off by opening the corresponding SE, i.e. $\mu\left(r_{SU}^{j}\right) = 0$, $\forall j \in I_{NS}^{off}(k)$; at $k = 0$, $I_{NS}^{off}(k) = \varnothing$. As a result of this, the graph $\left(I_{SN} \cup I_{LC}(k), I_{NS}^{OC}(k)\right)$ is defined as a set of shortest paths disjoint over the treetops, and satisfying conditions (5)–(7), which for this case are described as:

$$\left(\left(I_{SN} \cup I_{LN}(k), I_{NS}^{OC}(k)\right) = \bigcup_{i \in I_{SN}} \left(\{i\} \cup I_{LN(i)}(k), I_{NS(i)}(k)\right)\right)$$
$$\subset \left(I_{SN} \cup I_{LN}, I_{NS} \backslash I_{NS}^{off}(k)\right)$$
$$\left|I_{LN(i)}(k)\right| = \left|I_{NS(i)}(k)\right|, \ \forall i \in I_{SN} \bigcap_{i \in I_{SN}} I_{LN(i)}(k)$$
$$= \varnothing \bigcap_{i \in I_{SN}} I_{NS(i)}(k) = \varnothing,$$

At the same time, with the graph$\left(I_{SN} \cup I_{LN}(k), I_{NS}^{OC}(k)\right)$ search procedure, sets $I_{NS}^{i}(k), \forall i \in I_{LC}$ from (2), $I_{LC}^{j}(k), \forall j \in I_{NS}^{OC}(k)$ from (4) are formed and calculation procedures corresponding to this graph are executed, FD according to formulas from (4) and LL according to formula (2). At the end of the work of $\mathrm{PCN_{SS,FD.LL}}$, 1 is written in the position p_A, i.e. $\mu(p_A) = 1$. At the same time, the results necessary for the $\mathrm{PCN_{SS}}$ function block are stored in positions $q_{LL}, p_{SU}, s_{SU}^{1}, s_{\mathrm{PQ}l}, s_{NS}$, i.e.

$$\mu(q_{LL}) = \Delta P_{\Sigma}^{LL}, \ \mu(r_{SU}) = \left|I_{NS}^{OC}(k)\right|,$$
$$\mu\left(s_{SU}^{1}\right) \rightarrow \left\{p_{SU}^{j} | j \in I_{\mathrm{US}}^{OC}(k) \cup I_{NS}^{off}(k)\right\},$$
$$\mu\left(s_{\mathrm{PQ}l}\right) \rightarrow \left\{\left(q_j^P, q_j^Q, d_{q_j^l}\right) | j \in I_{NS}^{OC}(k)\right\},$$
$$\mu\left(q_j^P\right) = P_j^E, \ \mu\left(q_j^Q\right) = Q_j^E,$$
$$\mu\left(d_{q_j^l}\right) = l_j, \ \mu(s_{NS}) \rightarrow \left\{s_j | j \in I_{NS}^{OC}(k)\right\},$$
$$\mu(s_j) \rightarrow \left\{p_i | i \in I_{LC}^{j}(k)\right\},$$
$$\mu(p_i) \in \{0, 1\}, \ \forall i \in I_{LC}^{j}(k), \tag{9}$$

where $q_j^P, q_j^Q, q_j, s_j, p_i$ are the internal positions of $\mathrm{PCN_{SS,FD.LL}}$.

(2) When $\mu(p_A) = 1$, the transition t_1^{BT} is active. $V\left(t_1^{BT}\right)$ is triggered, designed to verify whether the $\left(I_{SN} \cup I_{LN}(k), I_{NS}^{OC}(k)\right)$ SS graph obtained in paragraph 1 is the current graph $\left(I_{SN} \cup I_{LN}(k), I_{NS}^{OC}(k)\right)$, i.e.

whether the condition is satisfied $\left(|I_{LN}(k)| = \left|I_{NS}^{OC}(k)\right|\right) = |I_{LN}|$ (see (8)). For this, the predicate function $\mu(s_{\Lambda 1}) =$ {"1" is calculated $\mu(p_{SU}) = \mu\left(d_p^{nLC}\right)$;, otherwise} ("1"- SS, "0"- not SS). At the same time, the state of the position is modified p_A: $\mu(p_A) = 0$.

(3) When $\mu(s_{\Lambda 1}) =$ "1", the transition t_2^{BT} is active. $V\left(t_2^{BT}\right)$, is triggered to check whether the current subset $M\left(\left(I_{SN} \cup I_{LN}, I_{NS}^{OC}\right), \left(I_{SN} \cup I_{LN}, I_{NS} \backslash I_{NS}^{off}(k)\right)\right)$ of the SS $\left(I_{SN} \cup I_{LC}, I_{NS}^{OC}\right)$ options of the current graph $\left(I_{SN} \cup I_{LN}, I_{NS} \backslash I_{NS}^{off}(k)\right)$ is promising for further splitting into subsets (branches). To do this, the predicate function $\mu(s_{\Lambda 2}) =$ {"1", $\left(\mu\left(q_{\Delta P}^0\right) = 0\right) \vee \left(\mu(q_{LL}) < \mu\left(q_{\Delta P}^0\right)\right)$; "0", is calculated, otherwise} ("1"- continue branching, "0" - cut off the current version). At the same time, the state of the position is modified $s_{\Lambda 1}$: $\mu(s_{\Lambda 1}) =$ "".

(4) When $(\mu(s_{\Lambda 1}) = \text{"0"}) \vee (\mu(s_{\Lambda 2}) = \text{"0"})$, the transition t_3^{BT} is active. $V\left(t_3^{BT}\right)$ is triggered, preassigned to develop a command to cut off the current option and consider another subset. To do this, the position status is modified $s_{\Lambda 4}$: $\mu(s_{\Lambda 4}) =$ "1". Simultaneously -$\mu(s_{\Lambda 1}) =$ "", $\mu(s_{\Lambda 2}) =$ "".

(5) When $\mu(s_{\Lambda 2}) =$ "1", the transition is active. It works $V\left(t_4^{BT}\right)$. At the same time, to continue splitting the current subset (see clause 3), the previous state of the position s_{SU}^2 is replaced by a new one from s_{SU}^1 (see (9)): $\mu\left(s_{SU}^2\right) = \mu\left(s_{SU}^1\right)$. Based on the information $\mu\left(s_{\mathrm{PQ}l}\right)$ (see (9)), the actual value of the power loss is calculated by the formula (3) $\mu\left(q_{\Delta P}^F\right) = \sum_{j \in I_{NS}^{OC}} \left[(\mu\left(q_j^P\right))^2 + \left(\mu\left(q_j^Q\right)\right)^2\right] \cdot \mu\left(d_{q_j^l}\right)$. Simultaneously - $\mu(s_{\Lambda 2}) =$ "⇄⇄", $\mu(s_{\Lambda 3}) =$ "1".

(6) When $(\mu(s_{\Lambda 3}) = \text{"1"})\, \&\left(\left(\mu\left(q_{\Delta P}^0\right) = 0\right) \vee \left(\mu\left(q_{\Delta P}^F\right) < \mu\left(q_{\Delta P}^0\right)\right)\right)$, the transition t_5^{BT} is active. It works $V\left(t_5^{BT}\right)$. At the same time, SS, the current organized crime group (see paragraph 1) is taken as the conditionally optimal: $\mu\left(s_{SU}^0\right) = \mu\left(s_{SU}^1\right)$, $\mu\left(s_{NS}^0\right) = \mu(s_{NS})$, $\mu\left(q_{\Delta P}^0\right) = \mu\left(q_{\Delta P}^F\right)$.

Computational modules $V\left(t_6^{BT}\right)$, $V\left(t_7^{BT}\right)$, $V\left(t_8^{BT}\right)$, $V\left(t_9^{BT}\right)$, are designed to implement the procedure of sequentially breaking the set of OPG variants into disjoint subsets by inverting the state of the next position with the name from $p_{SU}^j \in \mu\left(s_{SU}^2\right)$ the list of SE position names $\mu\left(s_{SU}^2\right)$ that are used when sequentially breaking the set of variants into disjoint subsets.

(7) When $(\mu(s_{\Lambda 3}) = \text{"1"}) \vee (\mu(s_{\Lambda 4}) = \text{"1"})$, the transition t_6^{BT} is active. $V\left(t_6^{BT}\right)$ is triggered. The following operations are performed: the next name p_{SU}^j is read from the list of names of the SE positions $\mu\left(s_{SU}^2\right)$; the name p_{SU}^j is written to the position s_{SU}^B and removed from the list $\mu\left(s_{SU}^2\right)$. As a result of this $\mu\left(s_{SU}^B\right) = p_{SU}^j$, $\mu'\left(s_{SU}^2\right) = \mu\left(s_{SU}^2\right) \backslash p_{SU}^j$, where $p_{SU}^j \in \mu\left(s_{SU}^2\right)$. Simultaneously - $\mu(s_{\Lambda 3}) =$ "", $\mu(s_{\Lambda 4}) =$ "".

(8) When $\left(\mu\left(s_{SU}^B\right) \neq \text{""}\right) \& \left(\mu\left(\&s_{SU}^B\right) = \text{"0"}\right)$, the transition t_7^{BT} is active. $V\left(t_8^{BT}\right)$ is triggered. Where in $\mu(s_{\Lambda 4}) =$ "1". Here $\&s_{SU}^B$: - a pointer to a position with a name p_{SU}^j stored in the memory of this pointer.

(9) When $\left(\mu\left(s_{SU}^B\right) \neq \text{""}\right) \& \left(\mu\left(\&s_{SU}^B\right) = 1\right)$, the transition t_8^{BT} is active. $V\left(t_8^{BT}\right)$ is triggered. Where in $\mu(p_B) = 1$.

(10) When $\mu\left(s_{SU}^B\right) \neq$ "", the transition t_9^{BT} is active. $V\left(t_9^{BT}\right)$ is triggered. At the same time $V\left(t_9^{BT}\right)$, through the pointer $\&s_{SU}^B$, inverts the state of the position with the name p_{SU}^j stored in the memory of this pointer s_{SU}^B, i.e. $\mu'\left(\&s_{SU}^B\right) = !\mu\left(\&s_{SU}^B\right)$ (this expression is equivalent to the expression $\mu'\left(p_{SU}^j\right) = !\mu\left(p_{SU}^j\right)$, where ! is the sign of the invert operation, & is the sign of the pointer). This means that the current graph $\left(I_{SP} \cup I_{LC}, I_{NS} \backslash I_{NS}^{off}(k)\right)$ is converted to $\left(I_{SP} \cup I_{LC}, I_{NS} \backslash I_{NS}^{off}(k+1)\right)$ for subsequent analysis $(k+1)$ on the next branching step (see point 1), where $I_{NS}^{off}(k+1) = \left\{I_{NS}^{off}(k) \cup \{j\} \text{ at } !\mu\left(\&s_{SU}^B\right) = !\mu\left(p_{SU}^j\right) = 0;\ I_{NS}^{off}(k) \setminus \{j\} \text{ at } !\mu\left(\&s_{SU}^B\right) = !\mu\left(p_{SU}^j\right) = 1\right\}$. Simultaneously - $\mu\left(s_{SU}^B\right) =$ "".

6. Conclusion

This paper shows the capabilities of Petri computing networks (PCN) for solving the problem of choosing the optimal disjunction structure of the network [9,10]. Based on the Petri computing networks (PCN) apparatus, the paper builds a structured computational algorithm in the form of a set of interconnected computational modules.

It should be noted that the specifics of the functioning of Petri computer networks allow arranging computational modules within the texts of computational algorithms in any given order. The aggregate of codes describing the principal schemes of computational modules is both a constructive language for describing an algorithm and at the same time - a program for its implementation on a computer in a programming environment currently being developed based on the PCN apparatus.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The reported research was partly funded by Russian Foundation for Basic Research and the government of the Yamal region of the Russian Federation, grant No. 19-48-890001.

References

[1] Bathaii M, Chahardoly T. Optimum distributed generation allocation and capacity for reducing power loss. 2003.
[2] Nazari MH, Parniani M. Determining and optimizing power loss reduction in distribution feeders due to distributed generation. In: 2006 IEEE PES power systems conference and exposition, Atlanta, GA. 2006, p. 1914–8. http://dx.doi.org/10.1109/PSCE.2006.296206.
[3] Yang Lin, Guo Zhizhong. Comprehensive optimization for energy loss reduction in distribution networks. In: IEEE power and energy society general meeting - Conversion and delivery of electrical energy in the 21st century, Pittsburgh, PA. 2008, p. 1–8. http://dx.doi.org/10.1109/PES.2008.4595980.
[4] Reconfiguration of electric distribution networks for energy losses reduction. In: Third international conference on electric utility deregulation and restructuring and power technologies, Nanjing, 2008. 2008, p. 662–7. http://dx.doi.org/10.1109/DRPT.2008.4523488.
[5] Ponomarenko IS. Automation of operation management processes in urban electricity supply systems: A textbook for universities. Moscow: Publishing House MPEI; 1990, p. 70.
[6] Filippova Yu. Sidorkin T, A. Rusina A. Optimization of regimes of power plants and power systems: A textbook for universities. Novosibirsk: Publishing house of the NSTU; 2016, p. 356.
[7] Hao Nguyen Dinh. The method of choosing the optimal places for opening electrical networks. Izv AN USSR Energy and transport 1986;(5):38–44.
[8] Seagal IKh, Ivanova AP. Introduction To applied discrete programming: models and computational algorithms. Moscow: Fizmatlit; 2007, p. 304.
[9] Asanov MS, Asanova SM, Satarkulov KA. Structural model of computing Petri nets. News of KSTU - Bishkek 2008;(13):78–85.
[10] Asanov MS, Asanova SM, Satarkulov KA. Computing components, description language and rules for the functioning of Petri computing networks. News of KSTU - Bishkek 2008;(13):85–95.
[11] Peterson JL. Petri net theory and the modeling of systems. Englewood Cliffs: Prentice-Hall; 1981.
[12] Hrúz Branislav, Zhou MengChu. Modeling and control of discrete-event dynamic systems: with Petri nets and other tools. In: Advanced textbooks in control and signal processing. Springer Science & Business Media; 2007, p. 342.