10.1515/acsc-2015-0006

Archives of Control Sciences
Volume 25(LXI), 2015
No. 1*, pages 87–107*

The efficiency of detecting the failures and troubleshooting while applying technical diagnostics for multi-computer systems

KYAW ZAW YE, HTIKE AUNG KYAW, EVGENE MIKALOVISH PORTNOV, ALEKSANDAR MIKALOVISH BAIN and PANDIAN VASANT

This paper presents techniques which base on the concept of flows thinning together with the identification techniques. These methods are proposed to determine the expected number of failures to assess the efficiency of technical diagnostics of instruments. Additionally, this research focuses on the improvement of multi-machine troubleshooting systems, based on the 'AND-OR' graphs. Respective algorithms are presented. The majority principle uses the input information to check the correctness of the decision regarding identification of faulty machines. In this paper we base on the complete testing algorithm for elements of multi-computer complexes searching by criteria of failed element.

Key words: technical diagnostics, fault, identification, 'AND-OR' graph, efficiency, control devices, multi-machine, troubleshooting, majority principle.

1. Introduction

The control devices and multi-computing complexes, which are nowadays exploited to improve the overall performance of industrial processes, involve sophisticated digital system design techniques aa well as complex hardware (input-output sensors, actuators, components and processing units). The probability of failure occurrence on such equipment is large thus an automatic supervision control should be used to detect and isolate anomalous working conditions as soon as possible. At present, a number of high-tech industries, research and educational processes use many kinds of instrumentations which significantly improve the efficiency of information processing. Despite the undoubtedly positive effect of the use of such instruments and complex ongoing devices, we observe their lack of effectiveness, due to a number of technical and economic circumstances.

K.Z. Ye, H.A. Kyaw, E.M. Portnov and A.M. Bain are with Department of Informatics and Computer Software System, National Research University of Electronic Technology (MIET), Zelenograd, Moscow, Russia. P. Vasant is with Department of Fundamental and Applied Sciences, Universiti Teknologi PETRONAS, Malaysia. E-mails to corresponding authors: Kyawzawye85@gmail.com, pvasant@gmail.com

Received 10.06.2014. Revised 10.01.2015.

In particular, an acute problem is improving the resiliency and reliability of instrumentations elements to make the their life exceeding the standard. In connection with these issues, one of the most important requirements for the instruments and complexes of them are their high availability and the ability to effective identification of failures [1, 2].

Since the early 1970's, the problem of fault detection and isolation in multi-machine processes has received great attention and a wide variety of model-based approaches has been proposed and developed. Theoretical and practical aspects of technical diagnostics, fault tolerance issues involved in instrument making by the famous scientists such as P.P Parkhomenko, V.V. Caribbean, E.S. Sogomonyan, M.F. Loaf, A.V Lobanov, R. Schlichting, D.A. Rennels,D. Dolev, B. Neilanc and many others. Model-based techniques which have been widely recognized as powerful approaches for the fault diagnosis, require a realistic mathematical model of the monitored system. An effective model-based fault diagnosis system should manage noises and modeled uncertainties in real-world. Mathematical model required, can be expressed either in state space or in input-output structure. A state space description of the system provides general and mathematically rigorous tools for system modeling and residual generation and may be used in fault diagnosis of industrial systems in deterministic and stochastic cases.

The analysis showed that the modern industrial technologies used in various fields, require new approaches to ensure their reliability and effective methods of technical diagnostics. There is observed a steady increase of the number of faults and failures which cause worsening the of production and greater probability of accidents and crashes. Such a situations follow from the unreliability of the devices and can be neutralized by improved skills of the personnel, which should be able to justify decisions undertaken in those negative situations.

2. Methods of determining the expected number of failures in the technical diagnostics of complex instrumentation

As noted in the introduction, one of the most important requirements for instrument complexes is their fault tolerance and the ability to effective identification of the failures [3]. This paper proposes a mathematical description of the physical model of thinning the flows in the context of faulty situations, and then the process of technical diagnostics (TD) can be formalized.

Assume an instrumental complex being in the faulty situations (FS) with the total number of T_{fs} of faults. During the TD they are identified, captured and eliminated. The recovery time during the TD is not taken into account. At the end of the process of recovery TD restarts.

The process of identification of FS refers to core flow of events – failures. Each detected (shown itself) rejection reduces the intensity of the flow on the value of T_{fs} with probability 1, i.e., feed thins.

To develop the methodology for determining the expected number of failures we apply the theory of thinning flows, together with the regression analysis [4,5,6] under the following assumptions: the probability of detecting the FS is equal to 1, the probability of making new FS in the recovery process is equal to 0 and, we assume a Poisson stream of refusals. It is known that thinning flows have characteristics functional which successfully bind parameters statistics TD devices and equipment instrumentation. Among such characteristics are the following.

- Mathematical expectation (ME) number of failures manifested themselves:

$$\bar{T}_{fs}(p_{fs}) = n_{fs}\left[1 - e^{t_{fs}p_{fs}}\right]. \tag{1}$$

- Average value of the stream of refusals

$$\bar{t}_{fs}(p_{fs}) = n_{fs}t_{fs}e^{t_{fs}p_{fs}}. \tag{2}$$

- ME time total exhaustion failure flow

$$\bar{P}(n_{fs}) = \frac{1}{t_{fs}}\sum_{\vartheta=1}^{n_{fs}}\frac{1}{\vartheta}. \tag{3}$$

During the time of $p_{f_{s_i}}$, appearance of the probability is not less than T_{fs}, exactly equal to T_{fs} and n_{fs} failures. They are respectively denoted as $D_{fs}(p_{fs})$, $D_{\overline{fs}}(p_{fs})$. During the time of P_{fs}, would be the last chance of the point for the probability of complete exhaustion failure of flow.

In formulas (1)-(3) the following designations has been used: n_{fs} – total (initial) number of FS embedded in devices and equipment of instrumentation; t_{fs} – the instantaneous failure rate of thinning flow; ϑ – the current variable. Based on the experimental data one can determine the number t_{fs}, n_{fs} and other characteristics thinning flow generated by the technical appliance diagnosis unit (apparatus) or the multi-computer system as a whole.

Below is the technique of determining the expected number of failures on the basis of simple probabilistic methods using empirical data for practical purposes instrumentation, on the observed time interval $[0, P_{fs}]$ is T_{fs} failures. The expected value is defined as [6]

$$M(\xi) = \int_{-\infty}^{\infty} X_{fs}dF(X_{fs}) \tag{4}$$

where F is the distribution function of the random variable

$$F(X_{fs}) = \frac{1 - e^{-t_{fs}X_{fs}}}{1 - e^{-t_{fs}p_{fs}}}, \quad X_{fs} \leqslant p_{fs} \tag{5}$$

The function $T(X_{fs})$ takes the value 1 for $X_{fs} = p_{fs}$, i.e. all events take place after P_{fs}. We rewrite the integral (4) with the normalizing factor $\frac{1}{1-e^{-t_{fs}p_{fs}}}$ which rightly discusses as events that occurred time point before:

$$M(\xi) = \int_{-\infty}^{\infty} X_{fs} dF(X_{fs}) = \frac{1}{1-e^{-t_{fs}p_{fs}}} \int_{0}^{p_{fs}} X_{fs} d(1-e^{-t_{fs}X_{fs}}) \tag{6}$$

Expression $\int\limits_{0}^{p_{fs}} X_{fs} d(1-e^{-t_{fs}X_{fs}})$ integrate by parts:

$$\begin{gathered}\int_{0}^{p_{fs}} X_{fs} d(1-e^{-t_{fs}X_{fs}}) = -\int_{0}^{p_{fs}} X_{fs} d(e^{-t_{fs}X_{fs}}) = -X_{fs}e^{-t_{fs}X_{fs}}\Big|_{0}^{p_{fs}} + \int_{0}^{p_{fs}} e^{-t_{fs}X_{fs}} dX_{fs} = \\ -t_{fs}e^{-t_{fs}X_{fs}} - \frac{1}{t_{fs}}e^{-t_{fs}p_{fs}} + \frac{1}{t_{fs}} = \frac{1}{t_{fs}} - \frac{e^{-t_{fs}p_{fs}}}{t_{fs}}(1+t_{fs}p_{fs}).\end{gathered} \tag{7}$$

Thus the equation (7) has the form:

$$\frac{1}{1-e^{-t_{fs}p_{fs}}} \int_{0}^{p_{fs}} X_{fs} d(1-e^{-t_{fs}X_{fs}}) = \frac{\frac{1}{t_{fs}} - \frac{e^{-t_{fs}p_{fs}(1+t_{fs}p_{fs})}}{t_{fs}}}{1-e^{-t_{fs}p_{fs}}}. \tag{8}$$

Rearranging equation (8) one obtains

$$m_{fs} = \frac{1-e^{-t_{fs}p_{fs}}(1+t_{fs}p_{fs})}{t_{fs}(1-e^{-t_{fs}p_{fs}})}. \tag{9}$$

Transcendental equation and provided enough large for the product $t_{fs}p_{fs}$ and it can be neglected by $(1+t_{fs}p_{fs})$ in the numerator of (9) and the expression $e^{-t_{fs}p_{fs}}$ in the denominator. In the case of infinitesimal above expression $t_{fs}p_{fs}$ can also be neglected. Then we have:

$$m_{fs} \approx \frac{1}{t_{fs}}. \tag{10}$$

The value m_{fs} is biased estimate for the entire set of events, but in the interval $[0, p_{fs}]$. It is not biased and is the sum of T_{fs} events (failures) at times p_{fs}. The expected number of failures is defined as follows:

$$m_{fs} = \frac{1}{T_{fs}} \sum_{i=1}^{T_{fs}} p_{fs}, \quad t_{fs} = \frac{T_{fs}}{\sum_{i=1}^{T_{fs}} p_{fs}}. \tag{11}$$

The expected number of failures is defined as follows. Then: $\xi = n_{fs}$ and

$$P_t(\eta = T_{fs}/\xi = n_{fs}) = \begin{pmatrix} n_{fs} \\ T_{fs} \end{pmatrix} \left(1-e^{-t_{fs}p_{fs}}\right)^{T_{fs}} e^{-t_{fs}p_{fs}(n_{fs}-T_{fs})} =$$

$$\left(1-e^{-t_{fs}p_{fs}}\right)^{T_{fs}} e^{t_{fs}p_{fs}T_{fs}} \sum_{n_{fs}=T_{fs}}^{\infty} \left(\begin{array}{c} n_{fs} \\ T_{fs} \end{array}\right) e^{-t_{fs}p_{fs}n_{fs}} = \tag{12}$$

$$= (1-e^{-t_{fs}p_{fs}})^{T_{fs}} e^{t_{fs}p_{fs}T_{fs}} \sum_{n_{fs}=T_{fs}}^{\infty} \frac{n_{fs}(n_{fs}-1)\ldots(n_{fs}-T_{fs}+1)}{T_{fs}} e^{-t_{fs}p_{fs}n_{fs}}.$$

To simplify the calculations, we introduce the following notation: $e^{-t_{fs}p_{fs}} = X_{nw}$. Then

$$\begin{aligned}
&(n_{fs}(n_{fs}-1)\ldots(n_{fs}-T_{fs}+1)X_{nw}^{n_{fsC}}) = X_{nw}^{T_{fs}}\left(\frac{1}{1-X_{nw}}\right)^{K_{fs}} \sum_{n_{fs}=T_{fs}}^{\infty} \frac{X_{nw}^{T_{fs}}T_{fs}!}{(1-X_{nw})^{T_{fs}+1}} = \\
&T_{fs}![X_{nw}^{T_{fs}}(1-X_{nw})^{-T_{fs}-1}] = \\
&T_{fs}![T_{fs}X_{nw}{}^{T_{fs}-1}(1-X_{nw})^{-T_{fs}-1} - (T_{fs}+!)X_{nw}{}^{T_{fs}}(1-X_{nw})^{-T_{fs}-2}] = \\
&T_{fs}!\{X_{nw}{}^{T_{fs}-1}(1-X_{nw})^{-T_{fs}-2}[T_{fs}(1-X_{nw})-(T_{fs}+1)X_{nw}]\} = \frac{1-X_{nw}}{T_{fs}-2T_{fs}X_{nw}.}
\end{aligned} \tag{13}$$

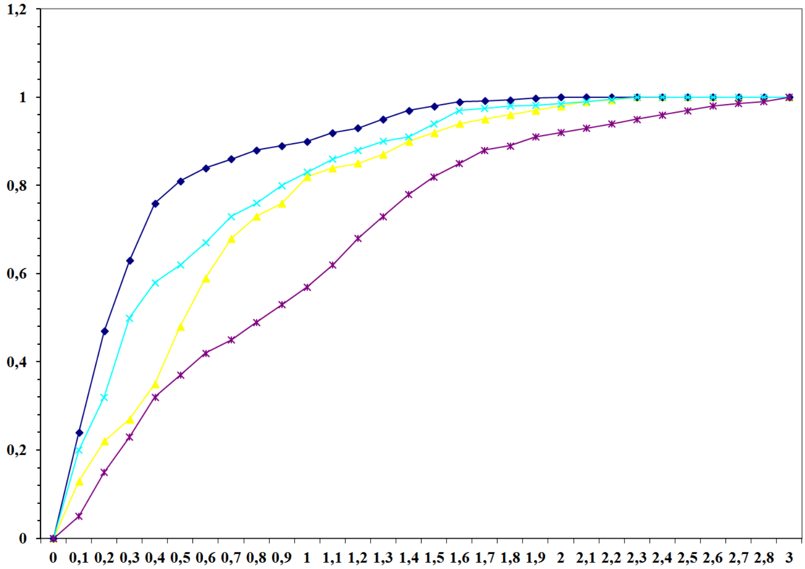

Figure 1: The empirical dependence of the failure distribution $F(X_{fs})$. Horizontal axis is time [h].

Fig. 1 shows the empirical dependence of the distribution function of the number of failures in time. The value of the failure rate t_{fs} finds from the formula (10) to determine the value of empirically constructed by the table method of curve and normalized distribution function: through time $p_{fs} = t_1$ failures occurred n_1, occurred $p_{fs} = t_2$ failures, etc.

These results lead to the following conclusions. First, we can estimate the efficiency of TD by analyzing the dependence of $n_{fs} = f(n_{fs},t_{fs})$. If the time TD∼ $t(n_{fs})$, the quality (effectiveness) diagnosis can be regarded as acceptable. If $T(n_{fs}) \gg p_{fs}$, on the program of sufficient 'hard', the intensity of the flow of failure is low. In this case, one could review and revise the program of the AP. If a large number of n_{fs}, it is necessary to review and revise the production technology products. The function $T(n_{fs}) = f(n_{fs},b_{fs})$ does not have an unconditional extreme inside the interval of existence. Constrained optimization can be found, if there are known limitations.

Secondly, the model adopted and the results of statistical data processing TD can solve two 'side', but essential to the practice of the problem.

3. Methods of detection of single failures in the automated system of technical diagnosis

Automation system of technical diagnostics (ASTD) belongs to a class of large technical systems consisting of a large number of components and elements with a complex structure. Tasks diagnostics (detection and retrieval failures) of such systems is very difficult, so ASTD should be viewed as an object of technological diagnostics (OTD). Then the system can be represented as a set of n of its constituent elements (set Ω), connected by a functional linkages [4].

The probability of an efficient condition of the ith element is denoted by p_i^{oti}, and the probability of failure by q_i^{oti}, ($q_i^{oti} = 1 - p_i^{oti}$). It is assumed that the failures of individual elements of the system are mutually independent. Monitoring system uses special tests, each of which check the performance of a completely defined sub-set of elements. Checking formula is intend for the following purposes:

- check the system (detecting any failure of existing),
- refusal to find (search all failed elements).

However, sometimes conduct such a test is fundamentally impossible, or inexpedient, since it requires considerable time and (or) means therefore advantageous to use an aggregate of a few simple tests.

To monitor the performance and looking for places bounce OTD has a possibility of testing, p_i^{oti}, $i = 1,\ldots,m^{oti}$, allowing check m^{oti} parameters, the nominal value of each of which is provided by a subset of functional elements Ω_i. The test may consist in supplying the necessary input to measure the response in one or more control points, etc. The results of the application of each test are classified on a 'successful' if all the elements of workable Ω and 'not successful' if refused by at least one member of Ω_i. The application of each test involves some costs c_i^{oti}, which may indicate the time or cost required to verify the parameter value required for this equipment, etc. The set of tests is more convenient to define a matrix, $H^{oti} = \left|h_{ij}^{oti}\right|$, $i = 1,\ldots,m^{oti}$, $j = 1,\ldots,n^{oti}$,lines which swarm match the available tests, and columns - elements of the set. Thus,

$$h_{ij}^{oti} = \begin{cases} 1, & when \quad j \in \Omega_i \\ 0, & when \quad j \in \Omega_i i = 1,..,m^{oti}, 1...,n^{oti} \end{cases} .$$

Column-vector of $c = \{c_i,\ldots,\ldots,c_m\}$ determines the cost associated with the use of each test. We assume that for the task of control ASTD existing tests enough, we now define the procedure on the basis of their current classification.

Control processes are classified by a number of attributes:

A. In depth, fault localization distinguishes control processes working capacity of the system as a whole and to determine the diagnosis of the state of each element.

B. According to the method of the control procedures can be divided into a sequence and combination. In the first case the choice of the next test is performed with a conventional program based on analysis of previous audits. In the second case, the true state of OTD is determined after the application of the entire set of tests.

C. Serial control procedures are usually evaluated in two optimality criteria: minimum average cost of the program and the minimum of the maximum value of this quantity.

D. With an a priori definition of the set of admissible states OTD are applied usually two hypotheses: OTD is likely to fail in not more than one member and various combinations of failed components at the same time.

E. If allowed arbitrary combinations of simultaneous failures, the sequence of monitoring procedures can be combined with the recovery of failed elements as they are discovered or occur without recovery. In the first case is up to full OTD, in the second to - set the status of each element, i.e., to identify the state of OTD.

F. From the combination of elements that remain untested after control, distinguish control with complete and incomplete coverage of elements of OTD.

G. By the degree of reliability of an inspection - instrumentation distinguish reliable and unreliable control.

H. According to the degree of detail of the information about the state of OTI received as a result of monitoring distinguish decision-making problem about the true state of OTI in complete and incomplete information [2].

We formalize the task to identify the defective item. It is known for the existence of a OTD exactly one failure, given a matrix of tests H^{oti} and the probability of failure of each element, q_i^{oti}, $i = 1,\ldots,n^{oti}$. We need a certain group of selected test sufficient for finding the failed element and define the order of sequential application of a conditional test of this group (program) so that the average value of the total cost of the search procedure to a minimum.

The use of any test h_i^{oti} can be regarded as a partition of the elements Ω into two subsets: Ω_i and $\bar{\Omega}_i$. At the end of ‘not successful’ failed element is in the subset Ω, at the end of ‘successful in a subset $\bar{\Omega}_i$. For further localization of failure tests can be used $h_i \in H^{oti}$ (hereinafter referred to as essential), allowing to carry out further divide the subsets Ω_i or $\bar{\Omega}_i$, containing the failed element. Test h_i is essential for a subset Ω_i , if:

$$\begin{cases} \Omega u \cap \Omega_i \neq \emptyset \\ \Omega u \cap \Omega_i \neq \Omega_i \end{cases}$$

Denote by H_i the list of essential tests for Ω_i. If the two tests h_i and h_v and $h_v \Omega_i \cap \Omega_i = \Omega_i \cup v\Omega_i$, then the list H^{oti} leave one, which correspond to lower costs.

List H^{oti} can be a matrix which columns correspond to the elements Ω_i. Each kth row of the matrix corresponds to the test of $h_k^u(i)$, $i = 1\ldots,m_i^{oti}$, monitors the health of subsets $\Omega_{u(i)}^k = \Omega_u \cup \Omega_i$ and at a cost $Z_{u(i)}^k = Z_u$. Subscript without parentheses is used to identify each test in the original matrix H^{oti}. Sometimes, for simplicity it will be omitted. Similarly, it can be formed in a matrix of tests that are essential for $\bar{\Omega}_i$ if further search-failed element that carried out in this subset.

If OTD is only one possible failure, the sign of the adequacy of the matrix H^{oti} for containment of any failure on the level of the element is that all the columns of the matrix must be pair wise different [6].

In the experiments of our method, noted to the following conclusions:

1. Confirm the reliability of the product individual parameter. Indeed, if the statistics are processed TD single (one) product, $p_{fs} = T(n_{fs})$ then the flow of failures can assume virtually stationary and therefore, $t_{fs}[t_{fs} \geqslant T(n_{fs})] = \lambda_{fs}$.

2. Reasonably plan warranty products. Knowing the expected number of fault situation, ‘embedded’ in the product, the number of detected faults, and the probability of failures at certain intervals of time, can determine the amount of the stock of tools and accessories (spare parts), the number and size of repair crews, timelines (stochastic) of their operations, financing activities, etc.

3. The methods of detection of single failures during the technical diagnostics aimed at a variety of tests, sufficient for finding the failed component, and the definition of a conditional order of the consistent application of tests on the criterion of minimizing the average total cost of the search.

4. Complete testing algorithm for elements of multi-computer complexes searching by criteria failed element

Optimize the process of diagnosing for multi-computer complexes (MCC) using overlapping tests with complete coverage of elements. The theoretical aspects of the problem are the following.

Let the beginning of a Mth step of the verification process carried out by a sequence of tests $H^{(M-1)^*} = \{h(1), h(M-1)\}$ to reduce the problem to finding a subset of the failed component of $\Omega^{(M-1)}$ (before the start of the system checked $\Omega^{(0)} = \Omega$; $\Omega^{(0)}$ includes all elements of the system, and does not include any test). The algorithm searches only failed component is as follows [4].

1. Define values $\widetilde{q}_j^{(0)}$ – conditional probability of failure is the j-th element, if the tested set exactly one element failed:

   $$\widetilde{q}_j^{(0)} = \hat{q}_j \left(\sum_{i \in \Omega(0)} \hat{q}_i \right)^{-1},$$

   where $\hat{q}_i = q_i p_i^{-1}$.

2. For each significant test calculate the probability of unsuccessful outcome of the tested subset:

   $$Q_i^{(0)} = \sum_{j \in \Omega_j \cap \Omega^{(0)}} \widetilde{q}_j^{(0)}$$

3. For each material test h_i there are associated costs $Z_i^{(0)}$ in view of the fact that a test sequence is performed $\sigma^{(0)}$. In general, the costs of conducting the test h_i can decrease or increase, subject to other tests. For example, it can be connected by previous inspections necessary for the test devices, or vice versa, holding previous audits may hinder access to the right parts of the system. For each test t_i determine values

   $$g_i^{(0)} = Z_i^{(0)} / Q_i^{(0)}$$

4. Selected test h_k, for which a minimum is reached.

5. Apply test h_k:

   - If the test h_k succeeds, the problem reduces to finding a subset of the failed component $\Omega^{(1)} = \Omega^{(0)} / \Omega_k$.
   - If the test h_k fails, the problem is reduced to finding a subset of the failed component $\Omega^{(1)} = \Omega^{(0)} \cup \Omega_k$.

   If in these cases, the subset $\Omega^{(1)}$ consists of a single element, then the searching for failed element ends here.

6. Apply a new fixed sequence of tests $H^{(1)}$ which contains the previous sequence $H^{(0)}$ and the last test h_k: $H^{(1)} = \{H^{(0)}, h_k\}$.

7. Continue the verification procedure as long as formed at some step k of the point 6th subset $\Omega^{(1)}$ consists of a single element.

Procedure described in the application to multi-computer complexes will implement consistent with the development of the verification process. For current calculations and selection of another test used computer with the necessary software and advance the memorized array of source data (probability of failure, duration of inspections, test specifications).

The same procedure can be proceeded in advance and make a statement by the order of application of tests according from the previous results, for example: ‘if the test h_k is successful, then the next test h_i; if the test h_k is unsuccessful, then to test h_j’ (Fig. 2).

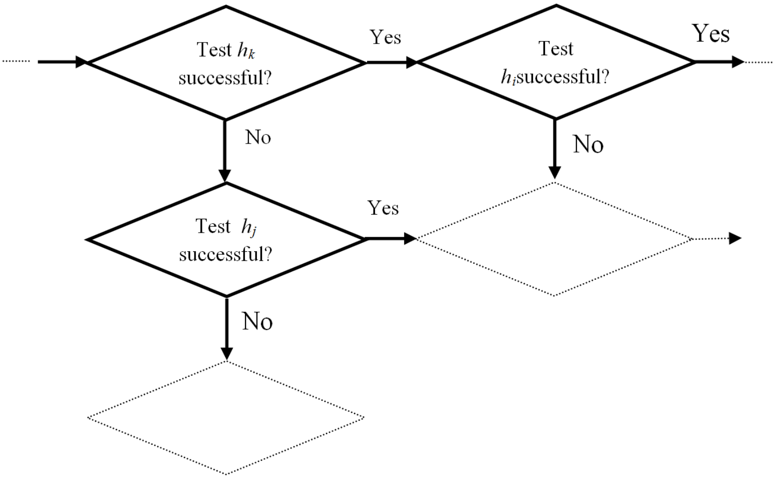

Figure 2: Fragment algorithm of complete testing for multi-computer complexness.

Let concretize the task. When testing multi-computer systems, choose 8 devices (main modules). No. 1 – LA (line adapter), No. 2 – controller for internal line devices MCC, No. 3 – first PC, No. 4 – second PC, No. 5 – linear controller for interfacing of PC, No. 6 – analog input module, No. 7 – third PC, No. 8 - - output of the module of control commands which can be tested within six tests which matrix is shown in Tab. 1. Time costs (in relative terms) to conduct each test is known: $Z_1 = 2.5$, $Z_2 = 2$, $Z_3 = 1$, $Z_4 = 1.2$, $Z_5 = 1.5$, $Z_6 = 1.3$, and the values of z_i do not depend on the order of tests. Empirically established a priori probability of failure of the above-listed items are as follows:

$$q_1 = 0.04 \quad q_2 = 0.03 \quad q_3 = 0.01 \quad q_4 = 0.01$$
$$q_5 = 0.03 \quad q_6 = 0.02 \quad q_7 = 0.01 \quad q_8 = 0.02$$

Table 6: Matrix of testing

Number	Numbers of elements							
s of test	1	2	3	4	5	6	7	8
1	1	1			1	1		
2		1		1			1	
3			1		1	1		
4	1					1		1
5	1		1	1	1			
6					1		1	1

The conditional probability of failure for each element are as follows:

$$\begin{array}{llll}
\widetilde{q}_1 = 0.028 & \widetilde{q}_2 = 0.018 & \widetilde{q}_3 = 0.005 & \widetilde{q}_4 = 0.005 \\
\widetilde{q}_5 = 0.018 & \widetilde{q}_6 = 0.011 & \widetilde{q}_7 = 0.005 & \widetilde{q}_8 = 0.011
\end{array}$$

Calculation the magnitude of the probability of unsuccessful outcome $Q_i^{(0)}$ for each test give the results:

$$\begin{array}{ll}
Q_1^{(0)} = \widetilde{q}_1 + \widetilde{q}_2 + \widetilde{q}_5 + \widetilde{q}_6 = 0.075 & Q_2^{(0)} = \widetilde{q}_2 + \widetilde{q}_4 + \widetilde{q}_7 = 0.028 \\
Q_3^{(0)} = \widetilde{q}_3 + \widetilde{q}_5 + \widetilde{q}_6 = 0.034 & Q_4^{(0)} = \widetilde{q}_1 + \widetilde{q}_6 + \widetilde{q}_8 = 0.049 \\
Q_5^{(0)} = \widetilde{q}_1 + \widetilde{q}_3 + \widetilde{q}_4 + \widetilde{q}_5 = 0.055 & Q_6^{(0)} = \widetilde{q}_5 + \widetilde{q}_7 + \widetilde{q}_8 = 0.034
\end{array}$$

Next, for each test we find:

$$q_1^{(0)} = \frac{Z_1}{Q_1^{(0)}} = 33.2 \quad q_2^{(0)} = \frac{Z_2}{Q_2^{(0)}} = 72.4 \quad q_3^{(0)} = \frac{Z_3}{Q_3^{(0)}} = 29.8$$

$$q_4^{(0)} = \frac{Z_4}{Q_4^{(0)}} = 24 \quad q_5^{(0)} = \frac{Z_5}{Q_5^{(0)}} = 26.8 \quad q_6^{(0)} = \frac{Z_6}{Q_6^{(0)}} = 38.8$$

It is seen that the first test should use h_4, and the value $g_4^{(0)}$ is smallest. The test h_4 can be successful or unsuccessful. Consider first the outcome, i.e. the failed element is among those which were not covered by the test h_4. We have the set $\Omega^{(1)} = \{2,3,4,5,7\}$. For each of the remaining tests we calculate $Q_i^{(1)}$:

$$\begin{array}{lll}
Q_1^{(1)} = \widetilde{q}_2 + \widetilde{q}_5 = 0.036 & Q_2^{(1)} = \widetilde{q}_2 + \widetilde{q}_4 + \widetilde{q}_7 = 0.028 & Q_3^{(1)} = \widetilde{q}_3 + \widetilde{q}_5 = 0.023 \\
Q_5^{(0)} = \widetilde{q}_3 + \widetilde{q}_4 = 0.028 & Q_6^{(0)} = \widetilde{q}_5 + \widetilde{q}_7 = 0.023. &
\end{array}$$

And then $g_i^{(1)}$:

$$g_1^{(1)} = 68.8, \quad g_2^{(1)} = 72.4, \quad g_3^{(1)} = 43.7, \quad g_5^{(1)} = 54.3, \quad g_6^{(1)} = 56.8.$$

Thus, after a successful test h_4, test h_3 should be carried out. This test may be, in turn, successful or non-successful. Consider the second possibility: the failed element is in the subset, which is verified by the test h_3, of the elements $\Omega^{(2)} = \{2, 4, \text{and } 7\}$. Then:

$$\begin{array}{ll} Q_1^{(1)} = \widetilde{q}_2 = 0.036 & Q_2^{(1)} = \widetilde{q}_2 + \widetilde{q}_4 + \widetilde{q}_7 = 0.028 \\ Q_5^{(0)} = \widetilde{q}_4 = 0.028 & Q_6^{(0)} = \widetilde{q}_7 = 0.023 \\ g_1^{(1)} = 68.8 & g_2^{(1)} = 72.4 \\ g_5^{(1)} = 54.3 & g_6^{(1)} = 56.8. \end{array}$$

Value $g_5^{(1)}$ is lowest, but successful test $h5^{oti}$ does not give useful information, as does not share many elements into two subsets. We now consider the other branch, i.e. the test h_3^{oti} which is unsuccessful, and for a subset of elements $\Omega^{(2)} = \{3, 5\}$ we need to compute $Q_i^{(1)}$.

The next step, the test h_4 is unsuccessful. We need to repeat the procedure for the subset $\Omega^{(2)} = \{1, 6, \text{ and } 8\}$.

$$\begin{array}{ll} Q_1^{(1)} = \widetilde{q}_1 + \widetilde{q}_6 = 0.039 & Q_3^{(1)} = \widetilde{q}_6 = 0.011 \\ Q_5^{(0)} = \widetilde{q}_1 = 0.028 & Q_6^{(0)} = \widetilde{q}_8 = 0.011 \end{array}$$

and then $g_i^{(1)}$:

$$\begin{array}{ll} g_1^{(1)} = 64.3 & g_3^{(1)} = 94.3 \\ g_5^{(1)} = 53 & g_6^{(1)} = 122.5. \end{array}$$

Hence we conclude that if h_5 is unsuccessful, then the first element is eliminated. If successful, it will causes exploration $\Omega^{(2)} = \{6, 8\}$.

$$Q_1^{(1)} = \widetilde{q}_6 = 0.011, \quad Q_2^{(1)} = \widetilde{q}_6 = 0.011, \quad Q_6^{(0)} = \widetilde{q}_8 = 0.011$$

and then $g_i^{(1)}$:

$$g_1^{(1)} = 235.7, \quad g_3^{(1)} = 94.3, \quad g_6^{(1)} = 122.5.$$

Obviously, minimal cost has the test h_3, and if it is successful, then the failed element is 8th, otherwise, if not succeed then the failed element is 6th. If we want to make a guide with a description of the sequence of inspections, we should fix the resulting sequence only (Fig. 3) and return to the stage when the test h_4 was performed. However, now we assume that the test was unsuccessful, i.e. search for the failed element of the subset $\Omega^{(2)} = \{1, 6, 8\}$ is to be performed.

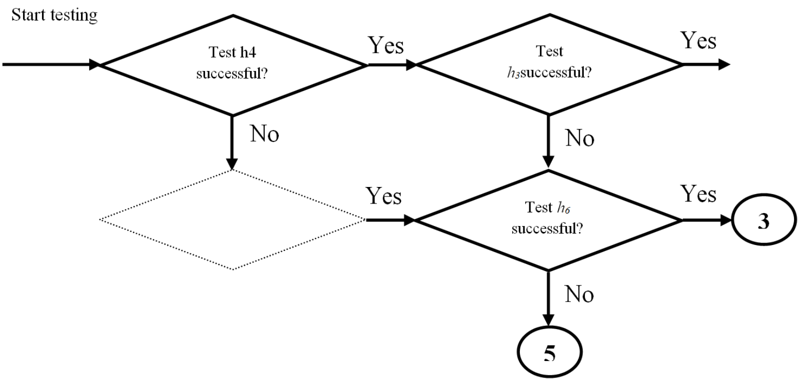

Figure 3: The first fragment of the algorithm for complete testing of multi-computer complexness (in the circles – numbers of failed elements).

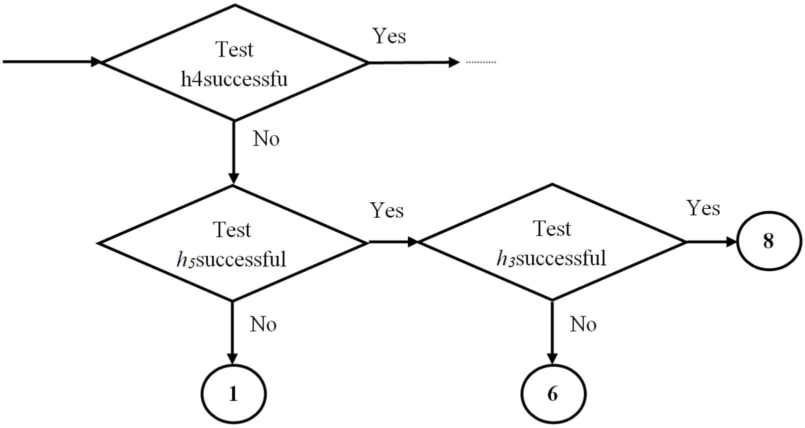

Figure 4: The second fragment of the algorithm for complete testing the multi-computer complexness.

The result is a second part of test instructions is shown in Fig. 4. Such a procedure continues until all the pieces are constructed and algorithm completes the testing of system to localize the failure to the room a single element. Chart of a complete test for this case is shown in Fig. 5.

It should be noted that in the case of testing one element after another we can get a simple rule for numbering the tests for finding procedures which minimizes searching costs of the failed element. A trick is that from any arbitrary numbering pair permutation tests can only check the neighbor finite number of steps to go to any predetermined sequence of them conducted, including the optimal.

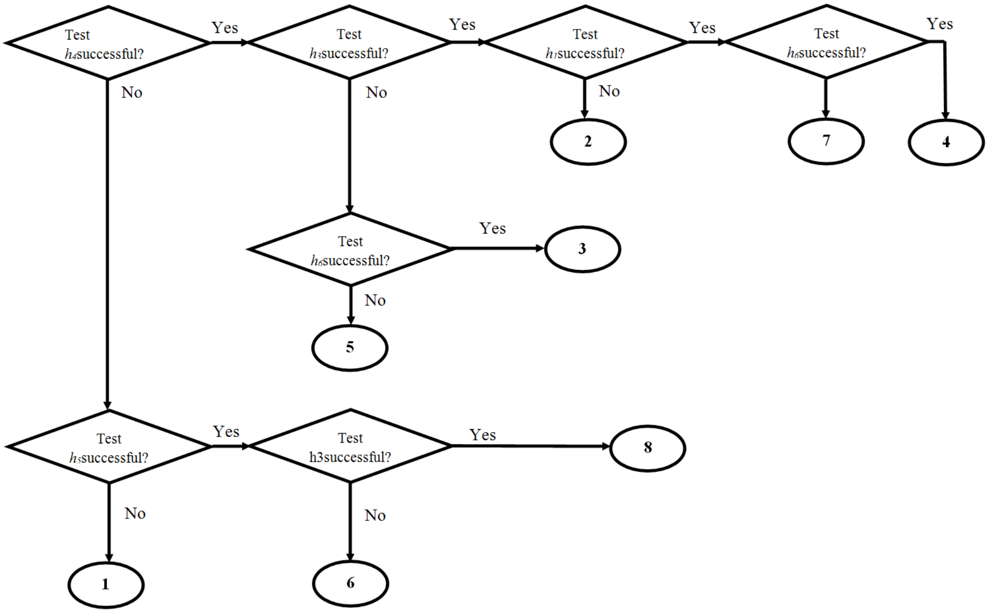

Figure 5: Diagram complete testing MCC.

If we would find a useful criterion for comparing two different tests with respect to the effect of their applications on the target functional result of the average search time of a failed component under certain conditions then it would be possible to calculate the criterion for each test and then enumerate all the tests in accordance with a monotonic variation of this criterion [3, 5, 6, 9].

For an arbitrary numbering objective functional tests

$$C[\sigma(\Omega_i)] = z^k_{(i)} + Q^k_{(i)} C[\sigma(\Omega^k_{(i)})] + (1 - Q^k_{(i)}) C[\sigma(\bar{\Omega}^k_{(i)})] \tag{14}$$

has the form:

$$C[\sigma(\Omega)] = z_1 + \widetilde{q}_1 C[\sigma(e_1)] + \widetilde{P}_1 C[\sigma(\Omega \backslash e_1)] \tag{15}$$

where e_1 is a singleton, which is no longer need to check and

$$C[\sigma(\Omega \backslash e_1)] = c_2 + C[\sigma(\Omega \backslash e_1 \vee e_2))].$$

Finally we have:

$$C = z_1 + \widetilde{p}_1(z_2 + \widetilde{p}_2(z_3 + \widetilde{p}_3(z_4 + \ldots))). \tag{16}$$

A similar expression for the case when the item numbers k and $k+1$ changed the procedure for checking and comparing the values of the total cost for both of these cases can be derived. We find the optimal order if possible check piecemeal responsible

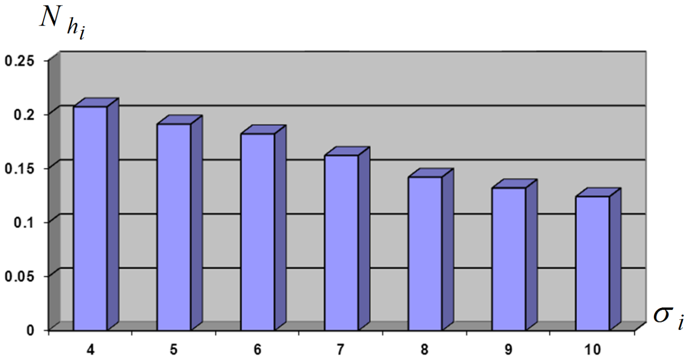

Figure 6: Graphical relationship $\sigma_i(N_{h_i})$.

numbering elements appears in accordance with the condition:

$$\frac{z_1}{q_1} \leqslant \frac{z_2}{q_2} \leqslant \ldots \leqslant \frac{z_n}{q_n}.$$

To confirm the benefits of the proposed method we carried out the software simulation for different numbers N_{h_i} of test, providing complete testing complexity. During the simulation of testing MCC we compared the relative time spent on the search for a single failed component: the traditional approach $Z^{mp}_{cym_i}$ and the proposed methodology $Z^{np}_{cym_i}$. We also defined improved performance testing $\sigma_i = \frac{Z^{mp}_{cym_i} - Z^{np}_{cym_i}}{Z^{mp}_{cym_i}}$. Simulation results are presented in Tab. 2 and Fig. 6 which shows a graphical representation of $\sigma_i\,(N_{h_i})$.

Table 7: Results of simulation by complete algorithm testing MCC

N_{h_i}	$Z^{mp}_{cym_i}$	$Z^{np}_{cym_i}$	σ_i
4	6.32	5.01	0.207
5	7.96	6.44	0.191
6	9.22	7.54	0.182
7	11.11	9.31	0.162
8	12.55	10.77	0.142
9	13.72	11.91	0.132
10	16.2	14.19	0.124

From the data presented in Tab. 2 and Fig. 6, it is clear that depending on the number of tests provided speedup of complete testing reduces from 12% to 20%. The results of simulation confirmed this advantage of the proposed method, which helps to test the full MCC searching by single failure criteria.

5. Method of searching fault using 'AND-OR' graphs

One of the main aspects of increasing multi-computer systems resiliency is the task of fault diagnosis of network computers where location and cause are identified [1, 13]. It is well known that this problem is difficult to solve, especially for large, including cluster computing systems, in which the number of computers can reach several thousands. The reason for this is the difficulty of formalizing the fault information in MCC. It requires hard and formal description of such characteristics as the intensity of problems, features of the machines operation in the process of solving the problems before and during the fault occurrence, etc. One of the way to solve the problem of determination the cause and fault location in these systems is to perform other actions when there is information on ways of problem motion in the computer network. While solving this problem the following aspects should be taken into account:

- Each problem is solved on several nodes (computers) at the same time, using different methods. Result is not known beforehand. When the results coincide on all computers then all computers are considered serviceable. If the results do not coincide, one computer is faulty.
- Each problem is solved in one of the n nodes (computers). The computer, which solves the problem is unknown, that is, it is randomly selected by the system itself. The result of the solutions is known in advance and the faulty computer is determined by incorrect respond.
- Each problem can be solved on several nodes (computers) of the randomly selected systems, using different methods. The faulty computer is determined by comparing the responses and taking into account the errors. The users determine the appropriateness of solutions made by the user task.

The structures that look like graphs are called 'AND / OR' graphs and are used to partition the problem image into alternative sets of the resulting problems [1, 13]. Let there be given the task A_k, which can be solved either by solving the problems A_{k1} and A_{k2}, or by solving the problems A_{k3} and A_{k4}, or by solving the problem A_{k5}. This relationship is represented by the structure in Fig. 7.

Under the 'AND-OR' graph is often understood as the graph for which the first property holds, and for the output arcs functions 'and' always holds. The model of 'AND-OR' graph for solving the problem A on three computers K_1, K_3 and K_5 is presented in Fig. 8.

The conclusion whether the computer is faulty or not is made based on the output results, presenting the combination of solving the problems K_1, K_3 and K_5 on the computers [1, 13]. In general, when a sufficiently large number is used the question whether the problem is solved correctly on not is taken using the majority principle, which means that the right result is considered if it is obtained on more than half of the computers, if their number is not less than $2m+1$ in case of friendly fault and is not less $3m+1$ in case of hostile malfunction. In the sequel the faults considered are assumed to be friendly only.

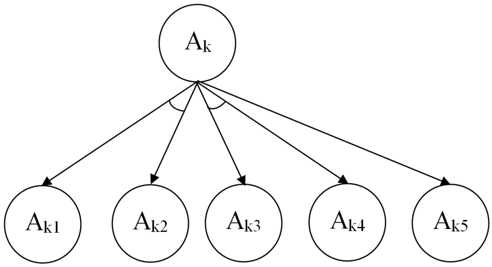

Figure 7: General Scheme of the ‘AND-OR’ graph.

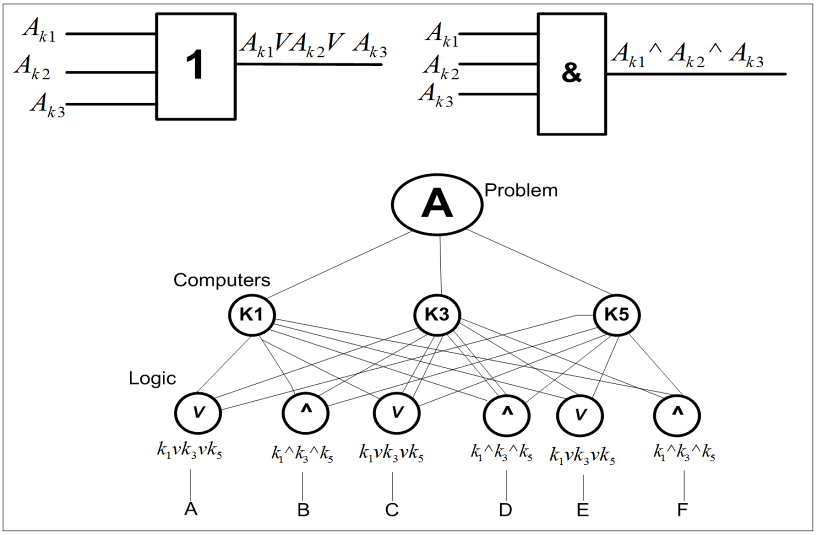

Figure 8: Model ‘AND-OR’ graph for the problem solving of fault searching in third computer.

Table 8: Matrix Z.

Problem	Computers						
A	+			+		+	
B	+	+			+		
C	+	+	+			+	
D	+	+	+				+
E			+	+	+		+

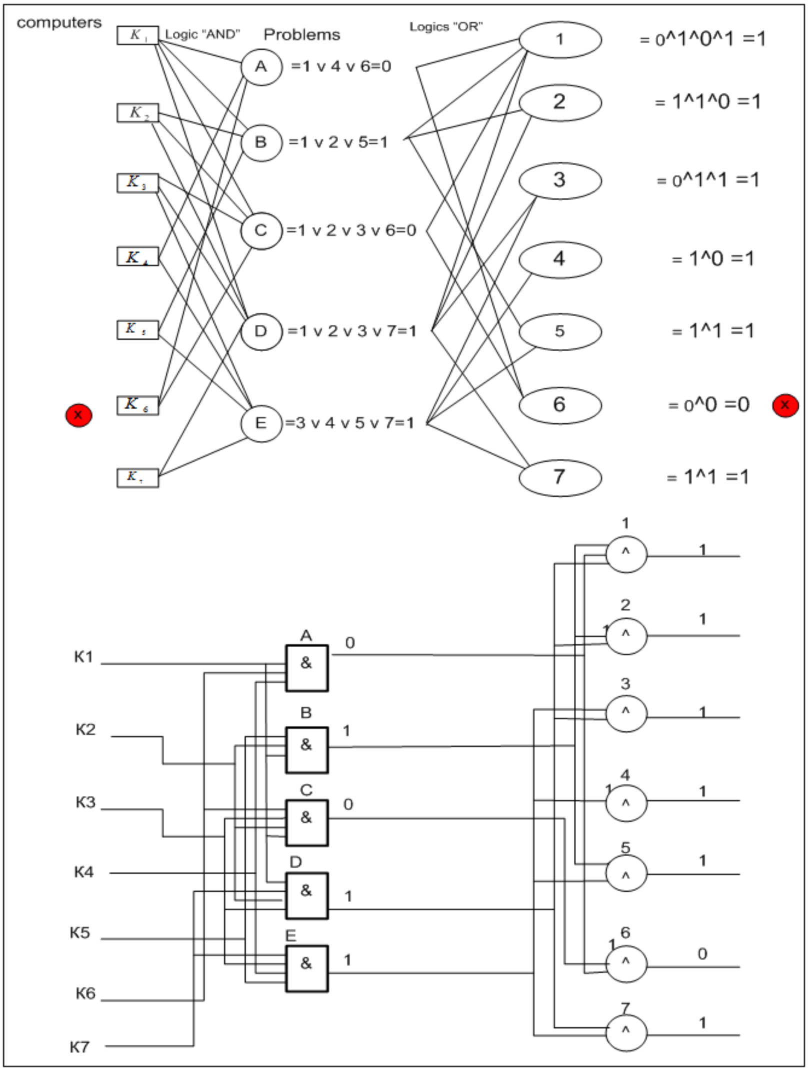

Figure 9: Model diagram for the problem solving in the multi-computer computing system.

Consider example of the diagnosis problem MCC consisting of 7 computers that solve 5 similar problems. Initial allocation problem is represented by the matrix Z (see Tab. 3). Rows A , B, C, D of the matrix Z correspond to the current tasks, and columns correspond to computers K_1–K_7, on which they are solved. After analyzing the results of solving the problem they are compared and processed using the 'AND-OR' graphs. The result of solving the problem is not known beforehand. Each problem is solved on three computers. For example, the problem A is solved on computers K_1, K_4 and K_6, the problem B on computers K_1, K_2, K_5, etc. Thus, the computer K_1 solves problems A, B, C and D, the computer K_2 solves problems B, C and D, etc.

Each wrong response while solving the problem in the proposed model corresponds to 0, and the correct corresponds to 1. If each problem solutions coincide with its own response, the output result $k_i = 1$ otherwise $k_i = 0$.

Fig. 9 presents a model which is a diagram for solving these problems in a multi-computer computing system. The left side of Fig. 9 is the solution of problems based on the ‘AND’ graph, which shows that the problem A and B are solved correctly (output result is 0). Thus, the preliminary conclusion is that there is a fault in one of the computers K_1, K_2, K_3, K_4 and K_6. The right side of the figure shows the solution of problems using the ‘OR’ graph. This scheme is an inspection and is a mirror image of the ‘AND’ graph. Using the output, results of ‘OR’ graph can come to final conclusions about the computer malfunctions K_6 (output value is 0). Thus, the model presented in the form of ‘AND-OR’ graph enables to pick up a message about the potentially faulty computer using the input information on the correctness of the problem solution.

6. Conclusion

Based on the findings of this research the following conclusion can be stated:

- This paper is based on the concept of thinning pulse flows, together with the theory of regression analysis which is the techniques for determining the expected number of failures in order to assess the efficiency the technical diagnostics for multi-computer systems.

- The method for detecting a single failure while conducting technical diagnostics is aimed at selecting a group of tests sufficient for finding the failed component and determining the order of sequential use of conditional tests by minimizing the average value of total cost of the search.

- An algorithm for complete testing the elements MCC criterion at the minimum search time of the failed component, in the course of simulating shows that depending on the number of tests provided speedup of completing the tests of multi-computer complex from 12% to 20%.

- The technique of multi-computer troubleshooting complexes bases on the ‘AND-OR’ graphs and allows to identify the faulty computer problems on the basis of the input information about the correctness of the decision.

References

[1] K.Z. YE, E.S. KUKUSHKIN and O.I. LISOV: Fault finding in the multi-computer computer systems based on the 'AND-OR' graph. *J. of the Defense Complex - the Scientific and Technical Progress of Russia*, Federal State Unitary Enterprise VIMI, **1** (2012), 71-75.

[2] K.Z. YE: Method for determining a single fault with the technical diagnosis of multiple computer systems. *Scientific and Practical J. Industry Aspects of Engineering*, INGN, **1** (2012), 18-19.

[3] K.Z. YE: Evaluation model of CPU performance of distributed fault-tolerant automation systems. *Int. Scientific and Technical Conf. Modern information technology*, Penza, (2013), 124-127.

[4] D.S. KAZAK: The model for the evaluation of fault tolerance of automated control devices and diagnostics. *Proc. of the Universities. Electronic.*, MIET, **1** (2009), 89-92.

[5] D.S. KAZAK: A mathematical model of CPU performance evaluation of distributed automation systems for process testing. *Scientific and Technical J. Engineering and Technology.*, Moscow, Publishing House of the Company Sputnik, **5** (2008), 37-38.

[6] D.S. KAZAK: The checks performance of automated systems for technological tests in instrument. *Scientific and Technical J. Engineering and Technology*, Moscow, Publishing House of the Company Sputnik, **5** (2008), 39-40.

[7] K.Z. YE, A.M. BAIN: Methods to improve the detection of failures and troubleshooting for technical diagnostics in instrument. *Manufacturing Science and Technology*, **1**(2), (2013), DOI: 10.13189/mst.2013.010202, Horizon research publishing cooperation. USA, http://www.hrpub.org/download/20131107/MST2-15401707.pdf

[8] A. PHILIPPOT, P. MARANGÉ, F. GELLOT and R. RIERA: .Decentralized diagnosis and diagnosability by model checking. *Universal J. of Control and Automation*, **1** (2013), 28-33, DOI: 10.13189/ujca.2013.010202. http://www.hrpub.org/journals/article_info.php?aid=379

[9] T. WÓJCICKI: Supporting the diagnostics and the maintenance of technical devices with augmented reality. *Diagnostyka*, **15**(1), (2014), 43-47.

[10] M. LUKASIEWICZ, T. KALACZYNSKI, M. LISS and T. KASPROWICZ: Diagnostics investigations of dmg-1a transmission with operational modal analysis methods application. *Diagnostyka*, **14**(3), (2013), 23-28.

[11] I.N. Kovalenko: Asymptotic methods of evaluating the reliability of the complex systems. O Nadezhnosti Slozhnykh Tekhnicheskikh System (On the Reliability of the Complex Technical Systems), Moscow, Sovetskoe Radio, 1966.

[12] Y. Genis: On reliability of renewal systems with fast repair. *Reliability: Theory and Applications*, **1**(3), (2006), 31-47.

[13] D.S. Kazak: Selection of hardware and software platform for distributed automated information systems. *Proc. of the Second All-Russian Inter-University Scientific Conf. on Actual problems of information. The development of information technology infrastructure and systems.*, MIET, (2008).

[14] K.Z. Ye and O. Lisov: Intelligent training complex for learning processes of technical diagnostics in multi-computer computer systems. *Drive Higher Education Institutions Chernozem*, **1**(23), (2011), 73-75.

[15] K.Z. Ye, H.A. Kyaw, A.M. Bain, R.A. Kassimov: Technique for improve efficiency intermodal information exchanges in automated control systems in the energy complex. *Drive Higher Education Institutions Chernozem*, **1**(31), (2013), 49-53.

[16] Ya.G. Genis: Failure rate of a renewable system with arbitrary distributions of the duration of failure-free operation of its elements. *Soviet Journal of Computer and Systems Sciences*, **23**(5), (1985), 126-131.

[17] Ya.G. Genis; On reliability of systems with periodic maintenance under rare failures of its elements. *Automation and Remote Control*, **71**(7), (2010), 1337-1345

[18] K.Z. Ye, A.M. Bain, R.A. Kassimov and E.A. Hrinchenko: Methodology of cycles during information exchanges and control for distributed energy units. *J. Defense Complex – Scientific and Technical Progress of Russia*, FSUE 'VIMI', **4** (2013), 18- 23.