

Complex estimation of status and operability of multilevel distributed systems

Sonkin D.M.

Institute of Cybernetics
National research Tomsk Polytechnic University
Tomsk, Russia
sonkind@tpu.ru

Meita R.V.

Institute of Cybernetics
National research Tomsk Polytechnic University
Tomsk, Russia

Khurul S.A.

Engineering department
Ltd. "STE Kibertsentr"
Tomsk, Russia

Abstract — Methods of evaluation of condition and operability of complex distributed systems are considered in this paper. Author propose a new conceptual model of multi-stage complex performance evaluation based on multi-criteria function including various performance, reliability, stability characteristics indicating their contribution to general system performance. The approach used makes it possible to obtain an integrated assessment of system availability taking into account hardware and software reliability analysis as well as influence of various factors (hardware, software, operator's errors) on overall system condition. Complex Population Notification and Warning System was considered as a subject of the research and was used to test the model developed.

Keywords — methodology of reliability evaluation, state estimation, distributed systems, multi-state connection elements, hardware-software complex in EMERCOM.

I. INTRODUCTION (HEADING 1)

Modern technology makes it possible to increase the number of distributed systems with sophisticated structure which include various software and hardware modules and communication environment providing interaction between these modules. Information about each element's availability is a crucial indicator for assessment of operating state of such systems. Working order of elements determines availability and complete or partial functionality of the whole system.

With further development of information technology human's part in system efficiency and availability assessment decreases [1]. Earlier experts studied data collected from sensors and tested systems manually. Today software-based methods of efficiency and availability assessment of system modules predominate[2]. A module (or a subsystem) in this paper is a functionally complete element of a distributed system.

The most widespread method of availability assessment of system as a whole and each separate subsystem consists in defining a number of states: fault-free, faulty, available, not available and limit state on the basis of some approved criteria [3].

It should be noted that in addition to "Complete availability" and "Complete fault" states for the class of systems considered there are a number of intermediate states with various availability levels of its modules and corresponding performance indicators. All system states are to be clearly determined and logged which in real conditions represents a challenging task for experts in feature identification and criteria determination [4].

In order to determine criteria for the states of system's elements parameters for assessment are to be selected and corresponding coefficients are to be calculated. These parameters and criteria indicate current activity of certain element at any moment of time. Availability factor of each module of the system is to be calculated on an integrated basis taking into account a number of parameters which influence the possibility of continuing full-scale operation [5].

The study of the problem of determination of states and calculation of availability indicators of distributed systems began in 1950s since complex hardware-software devices appeared. In 1980s availability assessment of software component of complex equipment was performed on the basis of hardware reliability assessment methods developed earlier [6]. Research procedure was based on assumption that availability of software and hardware components are inseparably related and software component is just an element of hardware which can be a cause of failures [7]. There were no well-developed methods of reliability assessment of software component of the system. Methods and models of reliability theory [8] customized for hardware were used in the process of software efficiency and availability assessment. Mean time between failures (MTBF) [9] was the major indicator.

A number of approaches to availability assessment of distributed systems were offered in mid-90s including organizational methods of availability assessment [10] third-party software for operational mode calculation on the basis of simulation models [11] and analysis of statistics obtained on important system performance characteristics [12]. General system availability was assessed using the method of summation

of availability of its components regardless of importance or critical role of certain modules.

In 2000s simulation of system structure was performed regarding features of considered system and very often was represented in the form of an oriented graph. Functional degradation (failure of part of system's functions) in the process of system operation was taken into account resulting in model refactoring [13, 14]. A number of availability criteria of system's elements was selected for each developed model on the basis of a set of uncorrelated parameters [15]. A number of assumptions in assessment methods used for system model optimization and simplification of calculation could distort (upwards) data on reliability and availability of elements [16]. These assumptions are given below:

- method of combining elements with high and low fail-safety in order to improve system reliability characteristics [16].
- mutual independence of separate elements and use of fault probability product theorem for evaluation of general system performance [17, 18].
- representation of system model in the form of a graph with singly connected components and without multiple connectivity (cycles, disconnected nodes) [20].

Results of performed analysis indicate that there is a need to develop methods of integrated assessment of availability of multi-level distributed systems taking into account specific features of device operation, sophisticated integration of software and hardware making it possible to use vast theoretical and practical experience of system efficiency and reliability analysis accumulated in this field.

II. PROBLEM STATEMENT

In general, case real distributed systems can have the form of networks with sophisticated configuration [21]. Availability assessment for the majority of such systems cannot be performed on the basis of simple summation of availability values of their elements using multiplication theorem of probability as it can be done for systems with series-parallel connection of elements [17]. In the process of state calculation for such systems, logic and probabilistic calculus (LPC) can be used on the basis of independent registration of failure flow due to various factors influencing hardware and software [22]. One of the most promising approaches in system states identification consists in registration of various information indicators of availability of complex systems at different levels of abstraction and approval of corresponding criteria.

Integrated Population Notification and Warning System (IPNWS) is a typical example of complex-structure systems. Such systems represent a set of modules: reporting objects (computer centers of various levels integrated in one system via data-exchange network) and control objects (sources and recipients of information on threats), considered as finite nodes of the system. Modules consist of software and hardware complexes including various telecommunication equipment and safety devices.

Special elements called Data Transmission Facilities (DTF) are included in IPNWS in order to make use of various data

transmission channels between certain module pairs possible as well as to improve communication channel utilization factor. DTF performance characteristics are determined by the quality of information delivery.

When the states of the systems are studied instead of the real set of objects and products we operate with a certain mathematical model which reflects the most significant properties of a real system [23].

Let us represent IPNWS as a disconnected oriented graph, then: distributed system S can be considered as a set of modules represented by a set of nodes A , communication between nodes is provided by data transmission facilities represented by a set of edges B . Certain level of system hierarchy corresponds to each node of the graph, it indicates centralization degree and importance for the whole system

Thus, formal representation of the system in the form of an oriented graph is as follows:

$$S = \{A, B\} \quad (1)$$

The set of modules can be written as:

$$A = \{a_1, a_2 \dots a_n\}, \quad (2)$$

where a_i – i -th graph node, $i = 1 \dots n$.

A set of inter-modular connection channels in a distributed system:

$$B = \{b_1, b_2 \dots b_m\}, \quad (3)$$

where $b_j = \{a_x, a_z\}$ – j -th DTF, represented as an oriented edge (arc), connecting two nodes with indices $x \in 1..n$, $z \in 1..n$, $j = 1 \dots m$. Each arc represents adjacency relationship, where a_x – initial arc node, a_z – end arc node.

Let us consider formalized model of a geographically-distributed system in mathematical form.

A set of modules distributed in accordance with concentration levels can be written as:

$$A = \{a_1 \dots a_{N_{k1}}, a_{N_{k1}+1} \dots a_{N_{kv-1}}, a_{N_{kv-1}+1} \dots a_{N_{kv}}\} \quad (4),$$

where N_{kp} – total number of modules in the system belonging to levels $[1..p]$,

$$\left\{ \begin{array}{l} N_{k1} = k_1 \\ N_{kp} = k_p + N_{k(p-1)} \\ N_{kv} = n \end{array} \right\}, \quad (5)$$

where n – total number of levels in the system, k_p – number of modules at level p , $p = 1..v$ – geographically-distributed system level number.

E.g. Например: $N_{k4} = k_4 + N_{k3} = k_4 + k_3 + N_{k2} = \dots = k_4 + k_3 + k_2 + k_1$.

In general case each module is a set of objects (software-hardware devices) of various types:

$$a_i = \{e_1 \cdot q_1^{a_i}, e_2 \cdot q_2^{a_i} \dots e_c \cdot q_c^{a_i}\}, \quad (6)$$

where $i = 1..n$, e_r – r -th object type, $r = 1..c$, $q_r^{a_i}$ – number of objects of r -th type in i -th module.

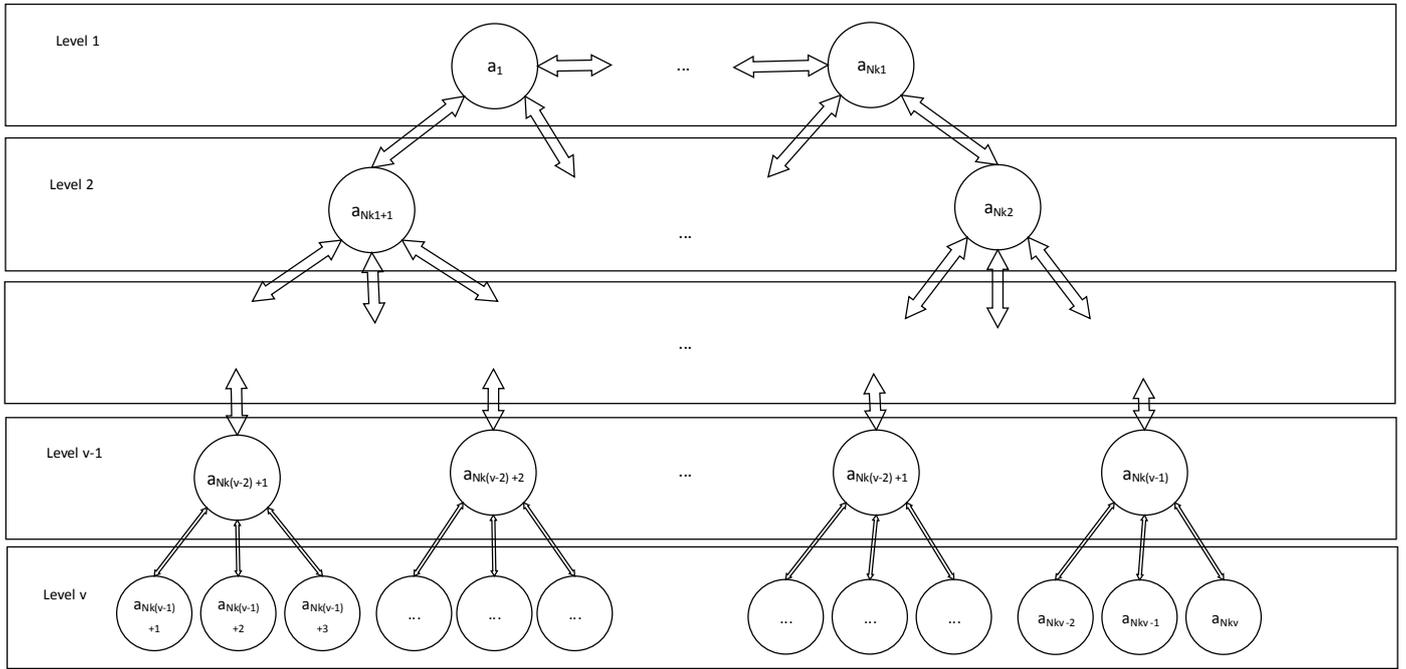


Fig. 1. Graphic representation of structure chart of IPNWS in the form of an oriented graph

Let us denote a set of objects of various types by E , then

$$E = \{e_1, e_2 \dots e_c\}.$$

The number of r -th type objects in the system considered $Q_r^e = \sum_{i=1}^n q_r^{a_i}$, then the set of sizes of objects can be represented as $Q^e = \{Q_1^e, Q_2^e \dots Q_c^e\}$. It should be noted that DTF may include several communication channels with individual technical characteristics.

$$b_j = \{ch_1 * q_1^{b_j}, ch_2 * q_2^{b_j} \dots ch_d * q_d^{b_j}\}, \quad (8)$$

where $j = 1..m$, ch_d – d -th channel type, $z = 1..d$, $q_z^{b_j}$ – number of z -type communication channels in j -th data transmission complex.

Let us denote the set of all communication channels of various types by Ch , then $Ch = \{ch_1, ch_2 \dots ch_d\}$.

The number of z -type channels in the considered system is $Q_z^{ch} = \sum_{j=1}^m q_d^{b_j}$, then the set of combined dimensions $Q^{ch} = \{Q_1^{ch}, Q_2^{ch} \dots Q_d^{ch}\}$.

Let us decompose the system down to low-level elements regardless of hierarchy level. Element in this case is an indivisible unit of the system performing specified functions. The term “Element” includes communication channels and hardware unit including system software functions.

A set of modules in decomposed model of a distributed system can be expressed as

$$A = \{E, Q^e\}, \text{ and a set of DTF } - B = \{Ch, Q^{ch}\};$$

Thus the number of elements F of a distributed system can be represented as:

$$F = \{f_1, f_2 \dots f_y, f_{y+1} \dots f_{yx}\}, \quad (9)$$

where $\{f_1, f_2 \dots f_y\} \in E$, $\{f_{y+1}, f_{y+2} \dots f_{yx}\} \in Ch$,

$$x = \sum_{z=1}^d Q_z^{ch}, y = \sum_{r=1}^c Q_r^e, yx = y + x.$$

Given model of the system makes it possible to combine connected nodes into a sub-graph of the basic graph which will remain a disconnected oriented graph (e.g. if we select one of the elements of level $v - 2$ as a root and all connected nodes of level v will be selected as leaf elements). Such division of basic model into sub-graphs makes it possible to assess availability of a subsystem – from objects (leaves) up to top-level structures. In case of an Integrated Population Notification and Warning System such division will make it possible to obtain availability assessment for interaction between reporting objects in any given district of the city, threat detection facilities on a potentially hazardous facility and regional dispatch center responsible for response coordination.

III. MATHEMATICAL MODEL OF AVAILABILITY ASSESSMENT

One of the most efficient methods of description of a distributed system in the process of availability assessment is to use simulation models and their mathematical abstraction, formulated on the basis of the term “logical element of the system” which performs certain functions instead of taking the set of software and hardware objects as basis. At the first stage of development these models make it possible to select and calculate efficiency and reliability parameters for each separate logical element. Then assessment can become more sophisticated as we replace abstract modules containing a number of elements with actual functional subsystems combining modules according to certain features and then to a general system model.

Obviously the quality of system availability assessment depends on degree of decomposition and how detailed the system model is. At the top level of decomposition the state of

the system is determined by operation mode of the set of modules as well as functional condition of corresponding data transmission facilities. At the lower level system availability depends on availability of each element as well as technical parameters of each communication channel between modules.

System approach to hardware reliability assessment for distributed systems includes selection of a functional mathematical model describing possible operational modes of the system as well as system regime calculation methods including various parameters (MTBF, availability remaining safe operation life, recovery time (uptime), availability assessment, performance and quality factors) [7].

A set of states of system's element f_g can be written as follows:

$$st_g(t) = \{st_g^1(t), st_g^2(t) \dots st_g^h(t)\}, \quad (10)$$

where $st_g^l(t)$ – l -th state of g -th element at the moment of time t , $l = 1..h$, $g = 1..y + x$.

As failures are events which occur accidentally due to unfavorable development of some events that took place, the process of error occurrence and subsequent process of availability recovery can be regarded as a queuing system (QS). Mathematical model of system availability assessment from this point of view must generally be stochastic, i.e. reflecting operational conditions and characteristics of occurrence of failures of simulated object by means of random values distributed in accordance with corresponding distribution law.

Thus, in the process of availability assessment of QS elements we are to use a set of parameters reflecting equipment structure with sufficient accuracy. Various incoherent data of both objective (instrumental) and subjective control (results of technical inspections, researches in the framework of preventive maintenance, expert estimates) can be used to calculate these parameters. On the basis of this information a common database can be formed serving as a platform for integrated assessment of current state taking into account influence of various parameters and structural units on the whole system.

Each of the states is characterized by a set of parameter values, describing object's state and qualitative features for which quantitative assessments are inapplicable. As a rule the list of these parameters and features as well as tolerable limits of their values is determined in standards, technical documentation and design documentation.

Status in this paper is considered as a complex property – a number of parameters correspond to each status of the module. The following can be regarded as availability parameters for each element st_g^l :

$$st_g^l(t) = \left\{ V_g(t), P_g(t), T_g^{MTBF}(t), K_g^{uptime}, \right. \\ \left. K_g^{O.A.H}(t), Pr_g(t), T_g^{fail}, Rq_g \right\}^l, \quad (11)$$

where $V_g(t)$ – assessment of available functions of g -th element (element's functional status), $P_g(t)$ – probability of failure g -th element at moment of time t , $T_g^{fail}(t)$ – Mean time between failures (MTBF) of g -th element, $Pr_g(t)$ –

performance of g -th element, T_g^{uptime} – uptime of g -th element, K_g^{uptime} – availability factor of j -th element during uptime period, $K_g^{O.A.H}(t)$ – operational availability factor during time period t , Rq_g – quantity of spare parts, tools and accessories (SPTA).

More detailed information on the impact of each parameter on availability of elements and list of operations necessary for calculation is given in corresponding sections.

The set of all states of system elements can be written as:

$$\bar{S}(t) = \{st_1(t), st_2(t) \dots st_l(t)\} \quad (12)$$

A. Mathematical Model Of Hardware Reliability Assessment

Operation of system's hardware and communication channels can be determined in accordance with [13] as cycle operation for intended purpose. Then regarding selected reliability indicators, the following can be considered as basic ones:

1. Functional status of element $V_g(t)$

Checking results of functional failure tests is the most informative assessment of element's availability regarding requirements of design and engineering documentation. This check is to be performed regularly; current functional availability assessment can be obtained on the basis of results of this check.

The importance of functions performed by certain element for system's priority tasks depends on the place of this element in hierarchy and its importance regarding availability of the whole system. Weight factor for separate functions is to be introduced in order to assess general functional status of an element.

$$V_g(t) = \sum_{l=1}^k I_g^l w_g^l, \quad (13)$$

where I_g^l – l -type function availability indicator in g -th element $I_g^l = \begin{cases} 0, & \text{if fail} \\ 1, & \text{if work} \end{cases}$, w_l^u – l -type function weight factor in g -th element, the following condition is met $\sum_{l=1}^k w_l^u = 1$.

Thus, "functional status" parameter can be represented as an analytical availability of element to perform its functions, values are expressed as percentage. It is obvious that a criterion of critical value is to be introduced; if a certain value is reached the element is no longer considered available:

$$V_g^{crit}(t) = \sum_{l=1}^k (q_l^j - qs_l^j(t)) \cdot w_l^u, \quad (14)$$

where $qs_l^j(t)$ – number of failed functions of l -th type element in j -th module, w_l^u – weight factor of l -th element type in j -th module, $\sum_{l=1}^k w_l^u = 1$

2. Hardware operational availability factor $K^{O.A.H}$

$$K^{O.A.H} = K^{A.A.H} \cdot P(t), \quad (15)$$

where availability factor (achieved availability in hardware) $K^{A.A.H}$ – probability of the fact that hardware will be available at the moment of time t ; $P(t)$ – probability of fail-safe operation during the time period t ;

3. Availability factor is determined as:

$$K^{A.A.H} = \frac{T^{fail}}{T^{fail} + T^{uptime}}, (16)$$

4. Mean time to recovery (uptime) (for all types of failures) T^{uptime}

Faultless operation is the ability of hardware to perform its functions even in case of failure (complete or recoverable) of its elements.

Hardware fault tolerance and uptime characteristics are determined as follows:

$$T^{fail} = \sum_i^N \frac{t_i}{N}; T^{uptime} = \sum_i^N \frac{t_i^{dtime}}{N}, (17)$$

where t_i – i -th system continuous operation period;

t_i^{dtime} – system down time, caused by i -th fault;

N – number of faults.

$K^{A.A.H}$ availability factor for a top-level element is calculated as a product of incorporated modules:

$$K^{A.A.H} = \prod_g K_g^{A.A} \prod_g (1 - (1 - K_g^{A.A})^{Q_g^H}), (18)$$

where Q_g^e , – number of elements of one type.

5. Probability of faultless system operation $P(t)$

For top-level element $P^H(t)$ reliability is calculated as a product of single and duplicated hardware units:

$$P^H(t) = \prod_g P_g(t) \prod_g (1 - (1 - P_g(t))^{Q_g^H}) \quad (19)$$

$$P_g(t) = e^{-\lambda_g t}$$

where $P_g(t)$ probability of faultless operation of a single unrepairable element,

$\lambda_g = \frac{-1}{T_g^{fail}}$ – fault occurrence rate for $P_g(t)$ of a single hardware unit.

6. Probability of successful transmission of information $P_g^\phi(t)$

Value P_g^ϕ must be calculated on the basis of DTF parameters of current element, operate characteristics and quick-action parameters of used communication channels

$$P_g^\phi(t_0) = \prod_{i=1}^m e^{-cn\mu/m}, (20)$$

where μ – rate error (measured at the statistics of failures at each link side when the information broadcasts), $c = \frac{t_0}{(t_0 + \tau)m}$ – count of message transmission repetitions, t_0 – sending time (examine time point), τ – average weighted time of transmission of one cluster), m – count of clusters in the message, n – message length in bits.

7. Quantity of SPTA Q_g^{SPTA}

It needs to determine values of criteria such as: minimum total cost of spare parts; ensuring the requirements for uptime of system operation; space, that occupied by equipment from SPTA, demands a method of recruitment; other indicators of

system reliability to evaluate the sufficient quantity and optimum composition of spare parts. Evaluation of SPTA is to resolve following system of equations:

$$\begin{cases} K_g^{SPTA} = 1 - \Upsilon(A_g, Q_g^{SPTA}) + \frac{Q_g^{SPTA+1}}{A_g} \Upsilon(A_g, Q_g^{SPTA} + 1), \\ \Delta t_g^{SPTA} = -\ln(K_g^{SPTA} (Q_g^{SPTA}, T^{ren})) / (I_g \lambda_g) \end{cases}, (21)$$

where $K_g^{SPTA} = K_g^{A.A.H} + \Delta K^{SPTA}$ – is the availability factor with SPTA using, ΔK^{SPTA} – is the effect of spare parts use, is reflected in the increase/decrease of readiness of module/system, Δt_g^{SPTA} – replenishment time of spare parts, T^{ren} – the replenishment period of spare parts, $A_g = \frac{\lambda_g T^{ren}}{yx}$ – the average consumption of g -th element in a time T^{ren} , λ_g – failure rate g -th element, $\Upsilon(A_g, Q_g^{SPTA}) = 1 - \sum_{s=0}^{Q_g^{SPTA}} \frac{A_g^s}{s!} e^{-A_g}$ – the probability that during the operating time T^{ren} refused ($Q_g^{SPTA} + 1$) elements, I_g – vector of operability functions of g -th element.

B. Mathematical Model of Software Reliability Assessment

Criteria and parameters of software availability assessment are different from the ones used for hardware. First of all, this can be explained by the difference between their operational characteristics. Based on some assumptions (about the size of computational resources and data storage capacities) we can say that software unlike hardware is not susceptible to wear. In general case software reliability increases with time as updates and new versions are released. Step by step it approaches to a steady period as the error occurrence rate reduces. Software may contain errors (defects), however, if they do not come out, software can successfully perform its functions for a long period of time.

Software availability assessment includes the following terms: *durability* (program does not deteriorate due to external actions of user), *faultlessness* (errors do not influence on correctness of operation), *correctness* (either there are no errors, or their number decreases with time), *correctness of input data* (assumption on correct set of input data).

In order to obtain software availability assessment for the systems of considered class a simulation model based on Jelinski-Moranda and Schick-Wolverton software reliability models [24] which according to researches [25-26] are the most suitable for availability assessment of large-scale programs with a long debugging period. Assessment model assumes that existing errors can be corrected in the process of software operation as well as new ones can be made. This approach makes it possible to calculate analytically the number of defects left in software.

The basic assumption in the model is the fact that error detection rate is proportional to the number of defects left after $i - 1$ time period, total debugging time up to the beginning of current period, average error finding time in current i -th debugging time period t_i , and makes it possible to estimate the probability of faultless system operation and software availability factor. It should be noted that the model we use would provide us with low (pessimistic) values of reliability assessment.

1. Probability of faultless operation is determined as:

$$P(t) = 1 - \frac{Q}{N} = \exp\left(-\frac{K_{JM}(E_0 - M)t}{2}\right), \quad (22)$$

where Q – count of registered failures, N – experiments count, K_{JM} – proportionality factor; E_0 – number of errors at the beginning of debugging stage; M – total number of time periods in which at least one error was detected. In our case, it is equal to total number of errors detected, as we made an assumption that only one error takes place in each time period.

In order to assess model parameters K_{JM} and E_0 a set of equations is to be solved (iterative solution, then E_0 is rounded and E_0 is obtained):

$$\begin{cases} K_{JM} = M / \left[\sum_{i=1}^M t_i \cdot \left(E_0 + 1 - \frac{\sum_{i=1}^M it_i}{\sum_{i=1}^M t_i} \right) \right] \\ E_0 = \frac{M}{\sum_{i=1}^M (1/(E_0 - i + 1))} + \frac{\sum_{i=1}^M it_i}{\sum_{i=1}^M t_i - 1} \end{cases}$$

2. Mean time between errors detected:

$$T = \sqrt{\pi/2K_{JM}(E_0 - M)}, \quad (23)$$

3. Availability factor of software is determined as:

$$K_g^{A.A.S} = \frac{T^{S.fail}}{T^{S.fail} + T^{S.uptime}}, \quad (24)$$

where $T^{S.fail}$ – mean time between failures for software, $T^{S.uptime}$ – mean time to recovery (uptime) after failure for software, $K_g^{O.A.S}$ – software availability factor (evaluated regarding subsystems) и determinates with term of steady-state operation (when $t \rightarrow \infty$).

$$T^{S.fail} = \frac{\sum_{i=1}^M t_i}{M}, \quad T^{S.uptime} = \frac{\sum_{i=1}^M t_i^{dtime}}{M}$$

t_i – i -th period of continuous software operation; t_i^{dtime} – system down time, caused by i -th error.

Achieved availability ratio for application software of the entire module can be calculated as achieved availability of functions, that module performed:

$$K^{A.A.S} = \prod_{i=1}^{\Omega} K_i^{A.A.S} \prod_{i=1}^{\Omega} (1 - (1 - K_i^{A.A.S})Q_i^S), \quad (25)$$

where $K_i^{A.A.S}$ – i -th functions performed by module ($i = 1 \dots \Omega$), Q_i^S – quantity of same-type functions in module.

C. Model of Integrated Assessment of System Operability

The main difficulty in the assessment of the system's state consist in the formation of a single indicator of operability. Structuring the performance and reliability of a collection of different software and hardware systems that makes up the distributed systems is complicated by the dynamic model elements correlation to different modes of functioning of the system. It should be noted that the definition of clear criteria for the state of the entire system, including the concept of a complete failure, as is hard solve problem.

One possible approach is the following: determinate a set of elements that perform simple (standard) function. This set is based on graph S , which present a functional structure of a distributed system IPNWS. The process of assessing performance is the recursion from the last (terminal equipment, performing the role of information) to the basic elements

(control centers, address coordination messages and informing managers). In this approach, the entire distributed system is splited into its constituent elements, which are presented as consistent connection graph nodes in structural scheme of state's assessment model. The arcs of the graph that act as channels of communication between system modules, presented in the form of elements with only one function - data transfer. The relationship between the graph nodes and arcs will be presented in the form of a matrix effects (adjacency matrix) $M_{[y \times x]}^{adj} = m_{i,j}^{adj}$, which reflects the priority of specific communication channel use.

The weighting factor enters to calculate the effect of a specific element in the overall state of the entire system. That factor equals to the weighted quantity of functions implemented by them, comparatively on all systems running functions. This approach takes into account the degree of influence of a single element failure level p ($p = 1..v$) on overall system reliability.

Research of operability will be carried out with the assumption that the stationary mode is set. In this mode, system can change the state, but the transition probability does not change: $P_g = \lim_{t \rightarrow \infty} P_g(t)$

Assuming the notion of equivalence complete system failure and total failure of all its elements, we can be formulate overall probability at time t , expressing it, according to a consistent scheme of including all indicators:

$$\begin{aligned} P^{sys}(t) = & \sum_{g=1}^{yx} \prod_g P_g(t) \prod_g (1 - (1 - P_g(t))^{Q_g^H}) + \\ & \sum_{g=1}^y \sum_{j=1}^x m^{adj}(g,j) P_g(t) (1 - P_j(t)) + \\ & \sum_{j=1}^x \sum_{g=1}^y m^{adj}(g,j) P_j(t) (1 - P_g(t)) + \exp\left(-\frac{K_{JM}(E_0 - M)t}{2}\right), \end{aligned} \quad (24)$$

Since the elements of the system at different levels are almost typical (i.e. represented a monotonous structure in terms of set theory) and do not overlap with each other (including application software), we can get a final assessment of the availability coefficient as a weighted sum of the components, taking into account the importance of each element in the system:

$$K_{sys}^{A.A} = \sum_{g=1}^{yx} [K_g^{A.A.H} w_g^H \cdot \prod_j^{yx} (1 - K_g^{A.A.H} Q_g^H)] + \sum_{i=1}^{\Omega} [K_i^{A.A.S} \cdot w_i^S \cdot \prod_j^{\Omega} (1 - K_j^{A.A.S} Q_j^S)] \quad (25)$$

where w_j^e – meaningful factor of j -th module in system operability, besides $\sum_{j=1}^n w_j^e = 1$.

Choose the meaningful factors should take into account that the failure of the entire system, from the point of view of the possibility of transmission through communication channels, will come after the failure of the DTF of the lower level of the system. The failure of one channel will be considered a partial failure (or refusal of one DTF), but will not be considered as failure of the entire system.

The resulting formula evaluation hardware and software elements of system state can analytically calculate the efficiency of the considered IPNWS system, consisting of a large number of subsystems with different characteristics of efficiency and

reliability, recovered during the operation and associated multi-channel data transfer subsystem.

CONCLUSION

An approach to integrated assessment of availability of multi-level distributed systems on the basis of a multi-criteria function taking into account various technical impact factors.

A method of integrated assessment of availability of software and hardware systems taking into account specific features of use of system and application software.

Availability parameters of elements obtained in the process of simulation of a distributed system were used for development of an integrated availability model of Integrated Population Notification and Warning Systems.

Distributed system model description is given in this paper using the example of IPNWS and developed availability assessment model for software and hardware components of the system consisting of devices, communication channels, system and application software.

The purpose of development of simulation model of a distributed system is to obtain qualitative availability assessment in order to ensure faultless operation by means of various measures improving efficiency (backup and sparing, preventive maintenance, modernization of equipment and communication channels etc.)

The developed research procedure for availability assessment of a distributed system can be used for:

- development of requirements to reliability, efficiency and availability of geographically-distributed systems
- problem statement for development of structural diagrams of complex systems including reliability hardware and software;
- development of mathematical models (logical, analytical and statistical) for simulation of failures and other emergency situations;
- calculation of necessary reserves and SPTA to provide system operation in accordance with approved reliability and safety requirements;
- use of obtained results for development and feasibility analysis of research, design, operational and other solutions.

This article accomplished in the context of government tasks by Russian Federation Ministry of Education and Science of the higher education institutions in terms of R&D on the theme "Research and development of new technologies, algorithms and software for navigation and telecommunication systems of new generation using the GLONASS navigation system and unmanned aircraft." № 4.1313.2014 - 1201459777.

REFERENCES

- [1] D. Cheriton. The V Kernel: A Software Base for Distributed Systems. IEEE Software, Vol. 1(2):19–42, 1984.
- [2] Jorrit N. Herder, Herbert Bos, Andrew S. Tanenbaum. A Lightweight Method for Building Reliable Operating Systems Despite Unreliable Device Drivers. Technical Report IR-CS-018, 2006.
- [3] Kulagin V.A., Tsarev R.Y., Capulina D.V., Pupkov A.N., Kukartsev V.V., Conceptual model of a multi-stage comprehensive evaluation of reliability automated ERP systems. Fundamental Research, Penza, "Academy of Natural Sciences" Publ., 2015, Vol. 7-2. – pp. 323-327. (in Russian)
- [4] Drozdov I. A., Konstantinov S.V. Assessment of probability-time characteristics of the process controls failover multiservice network communication. Technical sciences - From Theory to Practice, Novosibirsk, "SibAK" Publ., Vol. 5 (30), 2014, pp.29 – 32. (in Russian)
- [5] Belyaev S.M., Vasiliev V.S. Reliability analysis of the technical systems of the Russian Emergencies Ministry, having structural and functional redundancy. Fires and emergencies: Prevention, elimination, Moscow, 2015, Vol. 3, pp. 52-55. (in Russian)
- [6] Glenford J. M. Reliable Software through Composite Design. Van Nostrand Reinhold, 1979.
- [7] Turski V. Methodology of programming. Moscow, Mir Publ., 1981. (in Russian)
- [8] Feller V. Introduction to Probability Theory and its applications. Moscow, Mir Publ., 1967. (in Russian)
- [9] Robert L. G. Software Reliability Guidebook. Prentice Hall PTR, Upper Saddle River, NJ, 1979.
- [10] Romanyuk S.G. Evaluation of software reliability. Otkritie Sistemi, Moscow, Vol. 4, 1994. (in Russian)
- [11] Strelnikov V.P. New technology of reliability studies. Mathematical Machines and Systems, Vol. 2 , 1997, pp. 78-83.
- [12] The reliability and efficiency in technique. [Reference data], edited by Kuznetsov V.D., Moscow, Mechanical Engineering, 1990, 330 p. (in Russian)
- [13] Akimova G.P., Soloviev A.V., Janiszewski I.M. Methodology for evaluating the effectiveness of hierarchical information systems. System approach to information management. [Proceedings of ISA RAS Management], Moscow, KomKniga Publ., Vol.23, 2006. pp 48-66. (in Russian)
- [14] Jurkov N.K., Almametov V.B. Instrumental environment reliability of complex systems. Reliability and quality: Proceedings of the International Symposium, Penza, Penz.gos.univ. Publ., 2004, p.347-349. (in Russian)
- [15] Alexandrovich A.E., Borodakiy U.V., Chukanov V.O. Designing high-reliable, information and computing systems. Moscow, Radio and Communications Publ., 2004, 144 p.
- [16] Korolev, Yu.N., Losev, V.N., Yaskin, Yu.S. Increasing the reliability and quality of information control systems by utilizing the latest scientific and technical achievements in satellite navigation technologies. Measurement Techniques Publ., 2002, Vol. 45 (2), pp. 122-125.
- [17] Watanabe, Y., Sato, S., Ishida, Y. An approach for self-repair in distributed system using immunity-based diagnostic mobile agents. Lecture Notes in Computer Science, Springer Verlag, 2004, Vol. 3214, pp. 504-510.
- [18] Adler B. Using HART to increase field device reliability. Technical Papers of ISA: Safety Instrumented Systems for the Process Industry, 2002, pp. 93-106.
- [19] Birman K. P. Guide to Reliable Distributed Systems-Building High-Assurance Applications and Cloud-Hosted Services. Springer, 2012, 730 p.
- [20] Waseem Ahmed, Yong Wei Wu. A survey on reliability in distributed systems. Journal of Computer and System Sciences, v.79, n.8, 2013, pp.1243-1255.
- [21] Puzis R., Shirtz D., Elovici Y. A particle swarm model for estimating reliability and scheduling system maintenance. Enterprise Information Systems, Taylor & Francis Publ., Vol. 10(4), 2016, pp. 349-377.
- [22] Sinyutin A. Yu., Sinyutina D.S. Methodical approach to assessing the reliability of aircraft equipment control system based on fault flow formation using recursive algorithm. Young scientist, Vol.8, 2015, pp. 301-305
- [23] Gazizov T.T., Mytnik A.A., Butakov A.N. Generic Model of Security Threats for Personal Data in regard of Information Systems Dedicated to

Academic Planning. Reports of Tomsk state university of control systems and radioelectronics, Vol. 2 (32), 2014, pp. 47-50.

- [24] Skrypnikov A.V., Kondrashova E.V., Burmistrova O.N., Yakovlev K.A. Reliability complex estimation of logging machines. Modern problems of science and education, Vol. 2, 2013. Available at: <http://www.science-education.ru/ru/article/view?id=8537>

- [25] Gulati S., Bhatia K., Yadav P.K. Reliability optimization for distributed systems through task clustering. International Conference on Advanced Computing and Communication Technologies, ACCT, 2015, art. no. 7079075, pp. 176-182