DOI: 10.1051/e3sconf/20172502006

Improving reliability of state estimation programming and computing suite based on analyzing a fault tree

Irina Kolosok[1*], *Elena* Korkina[1], and *Alexander* Tikhonov[2]

[1]Melentiev Energy Systems Institute, 130 Lermontov str., Irkutsk, Russia
[2] Joint Stock Company "Irkutsk Electronetwork company"

Abstract. Reliable information on the current state parameters obtained as a result of processing the measurements from systems of the SCADA and WAMS data acquisition and processing through methods of state estimation (SE) is a condition that enables to successfully manage an energy power system (EPS). SCADA and WAMS systems themselves, as any technical systems, are subject to failures and faults that lead to distortion and loss of information. The SE procedure enables to find erroneous measurements, therefore, it is a barrier for the distorted information to penetrate into control problems. At the same time, the programming and computing suite (PCS) implementing the SE functions may itself provide a wrong decision due to imperfection of the software algorithms and errors. In this study, we propose to use a fault tree to analyze consequences of failures and faults in SCADA and WAMS and in the very SE procedure. Based on the analysis of the obtained measurement information and on the SE results, we determine the state estimation PCS fault tolerance level featuring its reliability.

1 Introduction

At a supervisory control of a energy power system (EPS), the procedure of the state estimation (SE) is a basic program tool that provides the tasks of operational and anti-emergency control with reliable and quality data on the EPS current state. This information is based on telemeasurements (TMs), telesignals (TSs), and phasor measurements (PMs). The SE procedure enables to detect wrong measurements, and plays a role of a barrier for the distorted (by various reasons) information to penetrate into the EPS control subsystem. At the same time, the SE software itself is subject to technical failures, and the supervisor, making decisions, should be confident in correctness of an SE-obtained decision. In this report, we propose to use a fault tree to analyze the consequences of technical failures and faults in the data acquisition and data processing systems (SCADA and WAMS) and in the SE software. Based on analyzing the measurement information and on the SE results, we determine the reliability level for the programming and computing suite (PCS) that estimates the state, and propose some measures to increase reliability.

2 Energy Power System (EPS) State Estimation (SE) Programming and Computing Suite (PCS) computing environment

The EPS SE PCS is intended to obtain the EPS current state model from telemetery and telesignals arriving from SCADA and phasor measurements from WAMS.

2.1 Supervisory Control and Data Acquisition (SCADA) fault tolerance

A SCADA system includes: remote telemetry units (RTUs) installed at EPS substations to take telesignals on the switching equipment state and measurements of the state parameters, communication channels, database (DB), systems of on-line display of the state parameters, as well as the software (EMS-application suite) to process the measurement results and to form control commands for dispatching management objects. Figure 1 presents the structure of a SCADA installed at the control center of a regional network company.

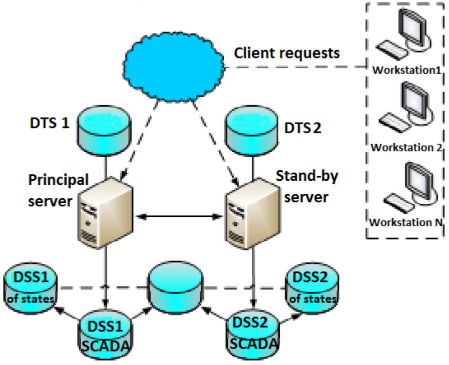

Fig. 1. SCADA structure

The SCADA operates at two independent servers, Principal (1) and Standby (2). The servers constantly

[*] Corresponding author: kolosok@isem.irk.ru

exchange requests between each other. For each server, provided is its own system of data transmission (DTS) and of data storage (DSS). Herewith, DSS 1 and DSS 2 constantly synchronize their data. Once 30 minutes, the data are transmitted to the historic server, at which continuous reservation is provided.

A response absence from Principal Server 1 is considered as a fault, the system switches to Standby Server 2. This occurs at minimal delays not noticeable for the user.

There are two types of failures possible: 1) hardware failures, and 2) software failures. After a failure, there is software restart at the Principal Server (in case, it was a program failure), or switch to the doubling fitments that is in the server hot reserve. Client requests go to the Principal Server, and, in case of its fault, they are redirected to the Standby Server with no request latency change.

In case of the DSS1 failure, the Principal Server switches to DSS2 with minimal delays. In case of the DTS1 fault, the incoming data stream is directed to DTS2.

Like practice shows, failures seldom arise in such systems. Software failures occur once a month, on average, hardware failures occur once a year.

2.2 Wide-Area Measurement System (WAMS) Data Acquisition (DA) Automatic System (AS) fault tolerance [1]

To solve the tasks in acquiring and storing the WAMS information, in 2009-2011, a WAMS Data Acquisition Automatic System (WAMS DA AS), whose structure is shown in Figure 2, was created and put into industrial operation.

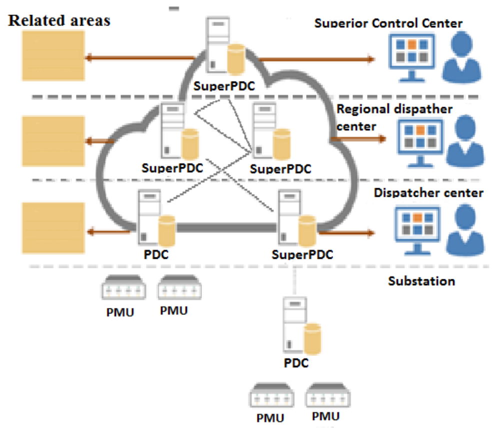

Fig. 2. Structure of WAMS Data Acquisition Automatic System [1]

The lowest hardware, PMU, transmits phasor measurements to the system under Protocol C37.118-2008/2011 into phasor data concentrators (PDCs) for further use in calculation tasks. PMD are relayed onto a higher level of dispatching control, into the super-PDC corresponding to the control hierarchy. This architecture is simple, reliable, and perfectly suitable to solve problems in the absence of restrictions for the computing and telecommunication infrastructure. The measurements are kept in the database of own design [1]. The system servers are united in a cluster operating synchronously: they interact with each other and exchange the information with data sources and clients. The databases included in the cluster are synchronized among themselves. The fault tolerance means within the system allow to create clusters of 2 and more servers. The storage is scaled over almost unlimited number of servers (up to 65535). Failure of any two servers will not lead to information loss. On the one hand, the DA AS architecture is hierarchical, because the servers are at all the levels of the dispatcher control; on the other hand, used is the cloudy technology, when the places of data storage and their traffic routes are not anchored rigidly. This increases the DA AS fault tolerance.

3 Block diagram of SE PCS for the reliability analysis

For a PCS to function necessary are measurements, EPS network, and software including the following algorithms:

to form the current (operational) network based on the basic scheme from telesignals;

to analyze the observability of the network;

to identify and detect gross errors in PMD and telemetry;

· to filter random telemetry errors, i.e., to receive their estimates and to finally calculate non-measured parameters.

For the SE PCS to operate under the WAMS conditions, used is a traditional algorithm of the linear state estimation through the state vector in rectangular coordinates [2]. Also, solving linear SE is successfully realized via the test equation method developed at the Melentiev Energy Systems Institute of the Siberian Branch of the Russian Academy of Sciences [3,4] that has a number of advantages over the traditional non-linear approach [5], the main of them being a possibility to a-priori detect bad data (BD).

In [6], to analyse the software reliability, there are the factors leading to software faults: errors in the program, use of non-optimal and imperfect algorithms (for example, heuristics use), restriction for real-time functioning (the system state changes faster, than the computing cycle lasts). An interaction of several factors and hardware problems in the computing system may also lead to software faults.

The software reliability degree features the probability of its fault-free operation over a certain time interval. In [6], the following is proposed as reliability indicators: the mean number of correctly solved tasks over a certain time interval Δt1, the mean number of errors for that interval, the probability of solving the set number of tasks for the time interval Δt2, the probability of emergence of the set number (probably, limit, being a fault fact) errors for that interval, etc. Such indicators may be used for any software. Conceiving the SE PCS object domain well, we will try to develop other reliability indicators that account for the peculiarities of the solved tasks and feature the fault tolerance of this very software.

Let us start with presenting the SE PCS, a program tool, as a technical system that is to operate safely and qualitatively. SE PCS will operate correctly, as long as all three elements are operable: measurements, the network, and SE algorithms. For the initial analysis, we make the block diagram (Fig. 3).

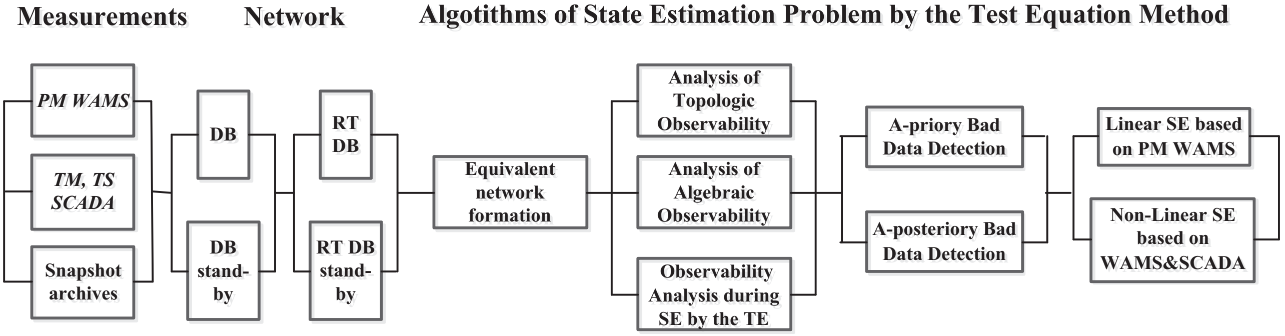
Measurements
Network
Algotithms of State Estimation Problem by the Test Equation Method
PM WAMS
TM, TS SCADA
Snapshot archives
DB
DB stand-by
RT DB
RT DB stand-by
Equivalent network formation
Analysis of Topologic Observability
Analysis of Algebraic Observability
Observability Analysis during SE by the TE
A-priory Bad Data Detection
A-posteriory Bad Data Detection
Linear SE based on PM WAMS
Non-Linear SE based on WAMS&SCADA

Fig. 3. SE PCS block diagram

From Figure 3, one can see that the bulk of the PCS components is reserved:

Measurements: when no PMs, then the SCADA telemetry is used; if there are no both PM frame and SCADA snapshot, PCS can operate with archival snapshot; besides, the incoming measurements are recorded in the real-time DB (RT DB) that is also reserved;

Network: the data on the diagram are stored at the DB server of constant information; at computer centers of major power facilities, a standby server and a standby DB are provided;

Algorithms: algorithms for the observability analysis, for the measurement validation, and for the state estimation are reserved by alternate blocks. The SE PCS produces the fastest solution, when the PMU LSE algorithm functions. Therefore, the Observability Analysis (OA) program should reply, whether the scheme plan is PMU-observable. If the answer is negative and LSE is impossible to run, OA is performed by a set of SCADA measurements (here is the example of the OA algorithm redundancy) for non-linear SE (here is the example of the SE redundancy). For the algorithm of the a-priori validation by control equations, there is a redundancy in the form of a-posteriori validation by the state estimation remainders.

However, when analysing the reliability of complex systems, it is necessary to find, which of the elements are critical, whose serious faults affect the system operability to a greater extent, in general. Typical criticality indicators [6] are the fault probability, the severity of consequences, the element tolerance to malicious activities, the risk value due to a fault, the possibility of fault localization, the controllability of the element state during the operation, reserving, etc. Ranging the elements by the criticality degree is possible at different levels of structuring the system objects. Critical elements may be visually provided by the Fault Tree technology.

4 Fault tree technology

For the first time, the term "fault tree" in Russian literature was mentioned in Yu. Guk's book [7]. Known since 1960s, the Fault Tree Analysis technology applied by expert systems in military aviation, then in nuclear power, and in some other industries [8], appeared a convenient means to analyse the operation capacity (fault tolerance) of any technical system or its separate complex nods.

The fault tree is presented in the form of a hierarchical structure:

top level - tree root - is the addressed technical system;

second level is the system indicators featuring this system;

third level - system elements - is the details of system indicators;

fourth level - tree leaves - is the events leading to a fault of the system operability (technological problems);

lowest level is the measures to suppress the fault causes.

Figure 4 presents the SE PCS fault tree to analyse the reliability of its operation.

At the top level, there is PSC itself. The system indicators (measurements, network, algorithms) is the second level. Those indicators contain basic elements (measurements types, databases, procedures etc.). Basic elements are exposed to these or those failures or technical faults which are on the forth level of Fault Tree. At the tree lowest level, there are counter-measures written in italics. The set of counter-measures enables to calculate the indicators for the EPS SE PCS algorithm operation efficiency and fault tolerance.

The figure 4 shows the block analyser, the program determining the most vulnerable PCS spots in terms of fault tolerance. This determination is based on the statistic block stored by the calculated indicators for a certain period of time (both blocks in bold).

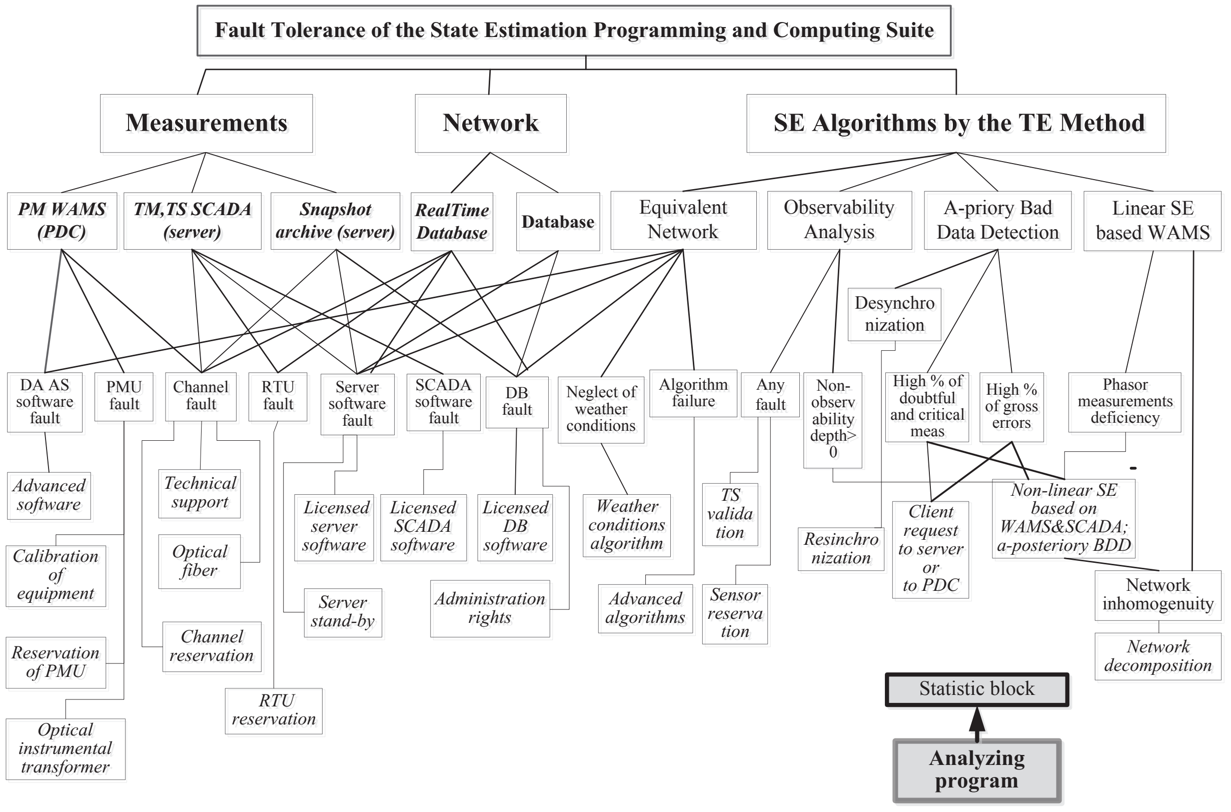

Fig. 4. EPS SE PCS Fault Tree

As compared with the known term "decision tree" [9], in a fault tree, the total of possible options to solve a problem decreases up to the number of options obviously threatening the operability of the addressed technical system. In fact, the fault tree is a diagnostic method offering a way out of a specific problem situation.

5 Fault probabilities for the Fault Tree elements

The analysing program is run several times per day after a certain time, in which N runs of the SE PCS are executed. The probability for the PCS fault tolerance is calculated like

$$P_{PCS} = P_{meas} * P_{network} * P_{a\lg},$$

in any other case, the PCS is non-operative, i.e.,

$$\mathrm{Q}_{PCS} = 1 - P_{PCS}$$

The PCS operation fault is non-entry of measurements, or the absence of the network, or the software crash, i.e.,

$$Q_{PCS} = Q_{meas} \cup Q_{\mathrm{network}} \cup Q_{a\lg}$$

where Q_{meas} is the fault probability of the suite measuring part, Q_{network} is the fault of all components that are responsible for the assembly of the current network, $Q_{a\lg}$ is the fault of the suite computing part, i.e., non-entry of the calculation results. In turn,

$$Q_{meas} = 1 - (1 - Q_{PM})(1 - Q_{\mathrm{TM}})(1 - Q_{archive})$$

where $Q_{изм} = 1$ means the total absence of measurements at the PCS input.

Equivalent scheme formation is switching of the scheme elements, which is imposing the telesignals on the basic network (scheme), combining the adjacent nods into one nod at the switched-on bus-tie switch, creating the common equivalent line for several parallel lines. Failure of forming the network at the PCS run readiness is possible at the errors downput in the algorithms for equalizing the network elements. The procedure provides an emergency shutdown, $Q_{network} = 1$ is the indicator of non-operability of forming the network:

Procedure for the observability analysis (OA) initially was in checking the correspondence of the number and structure of measurements in the SCADA-created snapshots to the network graph. To improve the network observability quality, a special algorithm for telemetry allocation enabled to indicate the spots poorly equipped with sensors on the network, provided recommendations on reserving sensors and measuring channels [10]. At present, along with telemetry, PMs are also used in EPS; therefore, now, developed are the algorithms for the WAMS sensor allocation accounting for possible shutdowns of individual connections, failures of separate sensors, loss of separate measurements [11].

OA enables to detect the observable and unobservable fragments in the network. It is known that, by the tension vector measurements in a node, and by the current vector measurements in the offline, one can obtain a PM calculated value in a node on other end of this line. Such sensor location on the network (area), when all the absent nodal measurements may be calculated by the available PMs, is called "non-observability depth equal to 0" [12]. If the available real and calculated PMs do not provide a completely PM-observable network(area), we consider the occurrence of the network(area), "unobservability depth" equal to 1 and more (which implies a measurement insufficiency for the linear SE operation) as an OA procedure failure for the linear SE (from the WAMS data).

$$Q_{OA} = \frac{n_{\mathrm{stop}\_unobserv=1}}{\mathrm{N}_{\mathrm{run}\_PCS}}$$

If there is no possibility to perform the linear SE, the procedure of nonlinear SE is run from the SCADA and WAMS data. An OA procedure failure for the nonlinear SE is practically impossible.

The procedure of the a-priori validation based on the test equation method is performed before the SE procedure run, and shows: the number of reliable and non-reliable measurements that arrived at the PCS input, the number of critical measurements, whose exclusion leads to presence of non-observable parameters whose errors cannot be detected; and the groups of doubtful measurements, where it is impossible to detect, which among them are erroneous. The measurements with the detected gross errors (bad data) are replaced with specified ones during the algorithm operation; thus, their dispersion values increase, showing a trust reduction concerning such measurements. It is much worse, if, among the measurements, there are critical ones that are not a part of test equations, and the doubtful ones, whose quality, at the given test equation set, is impossible to check. Therefore, we will consider a high percent of the doubtful and critical measurements present in the incoming snapshot a failure of the a-priori validation procedure:

$$Q_{apriri\_BDD} = \frac{n_{\mathrm{run}\_PCS\_with\_high\%\_doubt\&critical\_meas}}{\mathrm{N}_{\mathrm{run}\_PCS}}$$

Herewith, the PCS operation does not stop, because, further, the algorithms for the robust state estimation and for the a-posteriori validation are started. A correspondence of the SCADA snapshot time tags to those of WAMS is a strict requirement for the validation procedure.

The linear SE (LSE) procedure is solved non-iteratively, an indispensable condition for its start is the observability of the entire equivalent scheme through PMs. Therefore, the linear SE algorithm fault is:

$$Q_{LSE} = Q_{OA} = \frac{n_{\mathrm{stop}\_AO=1}}{\mathrm{N}_{\mathrm{run\_PCS}}}$$

In our Fault Tree, the algorithms for non-linear SE and for the a-posteriori validation are presented as the counter-measures against the PCS operability faults (see Figure 4). Therefore, we do not address their fault probability, but we surely account for the quality of their operation.

6 Case study

It is needed to calculate the SE PCS fault tree basic element fault probability values:

$$Q_{изм} = 1-(1-Q_{СВИ})(1-Q_{\mathrm{ТИ}})(1-Q_{архив})$$

$$Q_{расч\_сх} = 1-(1-Q_{ТИ})(1-Q_{архив})(1-Q_{БД})$$

$$Q_{алг} = 1-(1-Q_{форм\_сх})(1-Q_{АН})(1-\mathrm{Q}_{\mathrm{дост}})(1-Q_{ОС})\ 1.$$

1.The analyser is run 4 times per day, 1 time every 6 hours.

2. Software failures occur, on average, once a month, hardware failures occur ones a year

$$\mathrm{Q}_{\mathrm{server}} = \frac{1}{365*6*60} = 7.6*10^{-6}\ \mathrm{min}^{-1};$$

$$\mathrm{Q}_{\mathrm{SCADA\_software}} = \frac{1}{30*6*60} = 9.3*10^{-5}\ \mathrm{min}^{-1}$$

3. Let us take DA AS software failures (as newer, undergoing debugging) 3 times a month

$$\mathrm{Q}_{\mathrm{software\_DA\_AS}} = \frac{1}{10*6*60} = 28*10^{-5}\ \mathrm{min}^{-1};$$

$$\mathrm{Q}_{\mathrm{Database\_software}} = \mathrm{Q}_{\mathrm{SCADA\_software}}$$

4. PCS is run 3 times every 2 minutes, i.e., 540 runs in 6 hours.

5. In WAMS, there are 6000 frames over 2 minutes, but we also take 3 minutes (like in SCADA); i.e., over this interval, 540*6000=3240000 frames will arrive at PDC, of them 10 being faulty ones

$$Q_{PDC} = \frac{\mathrm{n}_{\mathrm{fault\_frames}}}{\mathrm{N}_{\mathrm{total\_PM\_frames}}} = \frac{10}{3240000} = 3*10^{-6}$$

6. Technical indicators for the PCS hardware reliability

$$Q_{PMU} = 0.0879\ [13]; \qquad Q_{channel} = 0.001\ [14];$$

$$Q_{RTU} = 0.0001\ ;$$

7. The number of PCS runs with a high % of gross errors

$$\mathrm{Q}_{\mathrm{gr\_err}} = \frac{33}{540} = 0{,}061$$

8. The number of PCS runs with a high % of critical measurements $\mathrm{Q}_{\mathrm{critical\_meas}} = \frac{15}{540} = 0{,}028\cdot$

$$Q_{meas} = 1-(1-Q_{PM})(1-Q_{\mathrm{TM}})(1-Q_{archive})$$

$$Q_{network} = 1-(1-Q_{TM})(1-Q_{archive})(1-Q_{DB})$$

$$Q_{a\,\mathrm{lg}} = 1-(1-Q_{form\_net})(1-Q_{OA})(1-\mathrm{Q}_{\mathrm{BDD}})(1-Q_{SE})$$

The elements, whose operability may be improved through reservation, are presented with the n_i indicator:

$$Q_{PM} = 1-(1-Q_{PDC}^{\mathrm{n}_1})(1-Q_{channel}^{n_2})(1-Q_{PMU}^{n_3})(1-\mathrm{Q}_{\mathrm{DA\_AS}})$$

$$Q_{TM} = 1-(1-Q_{SCADA})(1-Q_{channel}^{n_4})(1-Q_{RTU}^{n_5})(1-\mathrm{Q}_{\mathrm{server}}^{\mathrm{n}_6})$$

$$Q_{a\,\mathrm{lg}} = 1-(1-Q_{network})(1-Q_{OA})(1-Q_{BDD}^2)(1-Q_{\mathrm{SE}}^2)$$

In Table 1, compared are the SE PCS fault tree basic element fault probability values and the PCS operability probability summary values. The calculation results corroborate that reserving individual elements demonstrates the PCS fault tolerance increase.

Table 1. Fault probabilities for the Fault Tree elements

	With no reserving the elements	With reserving the elements
Q_{PM}	0,0891	0,00802
Q_{TM}	0,0012	0,0000094
$Q_{archive}$	0,0001006	0,000093
Q_{meas}	0,0903	0,0081
$Q_{network}$	0,0781	0,0781
$Q_{a\lg}$	0,2245	0,0908
Q_{PCS}	0,3496	0,1686
P_{PCS}	0,65	0,83

One should note that the system element redundancy is not infinite, it should be guided by common sense: reservation of a system technical element is more preferable to implement through a functionally similar one, but made by another manufacturer to avoid possible recurrings of the discovered or (which is much more dangerous) undiscovered defects. Algorithm redundancy should be performed by producing new software written in such programming languages that do not enable to introduce malicious distortions into the program code during a cyberattack when operating the SE PCS.

7 Conclusions

There are universal indicators including those created based on the waiting theory for software reliability that enable to reveal the level of its working capacity in terms of functionality and delivery of result. Application packages, standard algorithm libraries, operating systems belong to this software type. SE PCS is specialized software, and it is intended to provide quality results (based on a deep analysis of the input data) affecting the control of a technological process. It is extremely important to know, what the SE PCS behaviour under real conditions is. Therefore, it is extremely important to develop such indicators for software fault tolerance that would enable to protect weak spots of program blocks, and, thereby, to increase the PCS operability in any hostile environment.

References

1. F. Gaidamakin, A.Kislovskiy. *Proc. Of the 5th Intern. Scientific and Technical Conf.”Actual trends in development of power system relay protection and automation”*, Sochi, (2015).
2. Kolosok I., Korkina E., Buchinsky E. *Proc.of the 5th Intern. Conf. «Liberalization and Modernization of Power Systems: Smart Technologies for Joint Operation of Power Grid»*, Irkutsk, Russia, (2012).
3. A. Gamm, I. Kolosok. *Power and Electrical Engineering: Scientific Proc. of Riga Technical University*. Riga: RTU, (2002).
4. Kolosok I., Korkina E., Buchinsky E. *Proc. of the 18th Intern.Power System Computation Conference (PSCC-2014)*, Poland, Wroclaw, (2014).
5. Gamm A.Z. *Observability of power systems (*Moscow: Nauka. - 1990. - 220 p).
6. Polovko A.M., and S.V. Gurov, *Reliability theory grounds*. (SPb: BKhV-Petersburg. 2006. 704).
7. Guk Yu.B., *Computing the energy facility reliability.* (Leningrad.: EnergoAtomIzdat, 1988 – 224).
8. Ericson C.A. Fault tree analysis http://www.eecs.ucf.edu/~hlugo/cop4331/ericson-fta-tutorial.pdf
9. Quinlan J. R. Simplifying decision trees // *Intern. Journal of Man-Machine Studies*, V.**27** (3): 221 (1987). doi:10.1016/S0020-7373(87)80053-6.
10. Gamm A.Z., I.I. Golub *Observability of power systems*. (Moscow: Nauka (Science). - 1990. - 220 p).
11. Ya. Kuzkina. *Proc. of Intern. Conf. ”Relay protection and automation for electric power system”*, 2017. S-Pb, S.1-8.
12. R. F. Nuqui. *State Estimation and Voltage Security Monitoring Using Synchronized Phasor Measurement, Ph.D.dissertation*, Virginia Polytechnic Institute & State University, Blacksburg, USA, 2001.
13. O. Gomez, C. Portilla, M.-A Rios. *IEEE Trans.on.Power Systems* ,**30 (**2), (2015).
14. Aminifar F. *IEEE Transactions on power delivery*, V. **26** (2), (2011).
15. G.Kovalev. *Electrichestvo* **№4**. Pp.14-21. (2017).