

Model of reliability of the software with Coxian distribution of length of intervals between the moments of detection of errors

Vladimir P. Bubnov
Petersburg State Transport
University, assoc. professor
Saint-Petersburg, Russia
email: petr1923@yandex.ru

Alexey V. Tyrva
Petersburg State Transport
University, post-graduate
Saint-Petersburg, Russia
email: altyr@mail.ru

Anatoly D. Khomonenko
Petersburg State Transport
University, professor
Saint-Petersburg, Russia
email: khomon@mail.ru

Abstract—The generalized software reliability model on the basis of nonstationary Markovian system of service is proposed. Approximation by distribution of Cox allows investigating growth of software reliability for any kinds of distribution of time between the moments of detection of errors and exponential distributions of time of their correction. The model allows receiving the forecast of important characteristics: the number of the corrected and not corrected errors, required time of debugging, etc. The diagram of transitions between states of the generalized model and system of the differential equations are presented. The example of calculation with use of the offered model is considered, research of influence of variation coefficient of Cox distribution of duration of intervals between the error detection moments on values of look-ahead characteristics is executed.

Keywords—software reliability; approximation by Coxian distribution

I. INTRODUCTION

A great attention to a question of software reliability is paid now. The considerable quantity of software quality models has been developed. These models allow predicting software reliability at various stages of life cycle. The majority of existing models, for example, Jelinski-Moranda, Musa, Littlewood [1, 2, 3], use the assumption about exponential distribution of length of intervals between the moments of detection of program errors. The sequence of intervals of time between detection of errors or sequence of number of detected errors per time periods are used as initial data for these models.

The model developed by Jelinski and Moranda [4] assumes, that time intervals between the moments of program errors detection follow to exponential distribution with rate proportional to number of remaining errors. Musa model [5] belongs to the exponential class of models as well, with assumption that failure (error detection) rate exponentially decreases along the execution time. The similar assumption is used in Littlewood-Verrall model [6], but time intervals between the moments of detection of errors are assumed to follow gamma distribution.

Beside mentioned above, other exponential models are widely known, as well as the models that use general distribution laws: Weibull, gamma and others (see [1]).

The assumption about the exponential (or other) law of distribution of length of intervals between the moments of detection of errors, in our opinion, is serious restriction of models, because practical researches show, that variation coefficient of error detection flow is less than one ($v < 1$).

Use in model of any nonexponential laws of the distributions (gamma distribution, for example) is also unwanted from the computing point of view. In particular, such assumptions complicate the decision of corresponding systems of the differential equations.

In paper the nonstationary model of reliability of the software with Coxian distribution of length of intervals between the moments of detection of errors is proposed. The model is not limited by assumption about exponential distribution of time of error detection. Proposed model can be used for modeling of error detection and correction processes with general distribution of intervals between detection of errors on two or three ordinary moments.

II. APPROXIMATION OF INTERVALS BETWEEN THE MOMENTS OF DETECTION OF ERRORS BY COXIAN DISTRIBUTION

Cox has shown [7], that it is possible to present any distribution of random variable as a mix of exponential phases, or as phase-type distribution (hyper-exponential, Erlang or Coxian). Such representation has an important advantage: it is convenient to turn random process into Markovian one and easy to write and solve the system of the equations which describes behavior of corresponding model. This approach is widely used for non-Markovian systems of service analysis (see for example [8]). The approximating distribution parameters are either real or complex conjugate [7, 8], but probabilities of investigated system states are real.

In this paper we offer to approximate any general distribution of length of intervals between the moments of detection of errors with Coxian distribution for software reliability modeling.

The two-phase Coxian distribution is presented on Fig. 1. It is a mix of two exponential phases with rates λ_1 , λ_2 and probability of transition to the second exponential phase after the first one p .

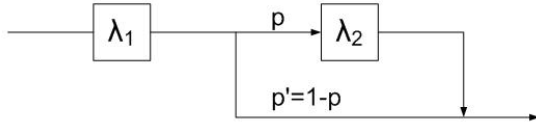


Figure 1. Two-phase Coxian distribution

Two-phase Coxian distribution parameters can be found from the system of the equations obtained by making equal first three ordinary moments of both source and approximating distributions:

$$(1) \quad \begin{cases} \frac{1}{\lambda_1} + \frac{p}{\lambda_2} = f_1, \\ \frac{1}{\lambda_1^2} + \frac{p}{\lambda_1 \lambda_2} + \frac{p}{\lambda_2^2} = f_2, \\ \frac{1}{\lambda_1^3} + \frac{p}{\lambda_1^2 \lambda_2} + \frac{p}{\lambda_1 \lambda_2^2} + \frac{p}{\lambda_2^3} = f_3, \end{cases}$$

where $f_i = g_i / i!$, $i = \overline{1, 3}$;

g_i – i th ordinary moment of source distribution, $i = \overline{1, 3}$.

Parameters of approximating two-phase Coxian distribution are calculated analytically:

$$(2) \quad \begin{cases} p = \frac{\lambda_2 (f_1 \lambda_1 - 1)}{\lambda_1}, \\ \lambda_1 = \frac{\lambda_2 f_1 - 1}{\lambda_2 f_2 - f_1}, \\ \lambda_2 = \frac{f_1 f_2 - f_3 \pm \sqrt{D}}{2(f_2^2 - f_1 f_3)}, \end{cases}$$

where $D = (f_1 f_2 - f_3)^2 - 4(f_2^2 - f_1 f_3)(f_1^2 - f_2)$.

In the elementary case when $p=1$ the system of the equations (1) for definition of parameters of approximating two-phase Coxian distribution looks like:

$$(3) \quad \begin{cases} \frac{1}{\lambda_1} + \frac{1}{\lambda_2} = f_1, \\ \frac{1}{\lambda_1^2} + \frac{1}{\lambda_1 \lambda_2} + \frac{1}{\lambda_2^2} = f_2, \end{cases}$$

From (3) it follows:

$$(4) \quad \lambda_{1,2} = \frac{f_1 \pm \sqrt{4f_2 - 3f_1^2}}{2(f_1^2 - f_2)}.$$

From (4) it follows that the parameters λ_1, λ_2 are real when approximating source distribution with variation coefficient $\frac{1}{\sqrt{2}} \leq \eta < 1$.

To approximate the distribution with variation coefficient $0 \leq \eta < \frac{1}{\sqrt{2}}$ it is reasonable to use complex conjugate parameters $\lambda_1 = \alpha + j\beta, \lambda_2 = \alpha - j\beta$. From (4) it follows:

$$(5) \quad \alpha = \frac{f_1}{2(f_1^2 - f_2)}, \quad \beta = \frac{\sqrt{3f_1^2 - 4f_2}}{2(f_1^2 - f_2)}.$$

Expectation-Maximization algorithms for calculation of phase-type distribution parameters in software reliability models are shown in [9, 10] as well as comparison with quasi-Newton method.

For comparison, it should be mentioned that gamma distribution enables to approximate any distribution as well as Coxian one. But while using gamma distribution with means of linear processes software reliability growth model is described by integro-differential equations [11] and solution becomes more complex.

III. THE CALCULATION ALGORITHM

Let's present the process of detection and correction of program errors by imbedded chain of Markov with discrete set of states and continuous time [12, 13]. The graph of transitions between states of software reliability model, designated as $C_2(i)/M(j)/N$, is presented on Fig. 2.

It is assumed, that the program contains N errors. Error detection and correction rates depend on its number. The distribution of i th error detection time is approximated by a special case of Coxian distribution – by two-phase distribution with parameters $p_i=1$ ($p_i'=0$) and exponential phases rates λ_i, λ_i' ($i=\overline{1, N}$) which are calculated under (4). Time of fixing of the found errors follows exponential distribution with rate μ_j which depends on error number j ($j=\overline{1, N}$).

The states of $C_2(i)/M(j)/N$ system at any moment are described by (i, k, j) vector, where i is the number of detected (but not corrected yet) program errors ($i=\overline{0, N}$), j is the number of corrected program errors ($j=\overline{0, N}$) and k is a phase number of Coxian distribution of length of intervals between the moments of detection of errors ($k=\overline{0, 1}$). The transition from a state $(i, 0, j)$ to $(i+1, 0, j)$ means that $(i+1)$ th error has been detected while testing. Transition from a state (i, k, j) to $(i-1, k, j+1)$ means that $(j+1)$ th error has been fixed. The general number of states N_c is calculated under the formula:

$$(6) \quad N_c = (N+1)^2.$$

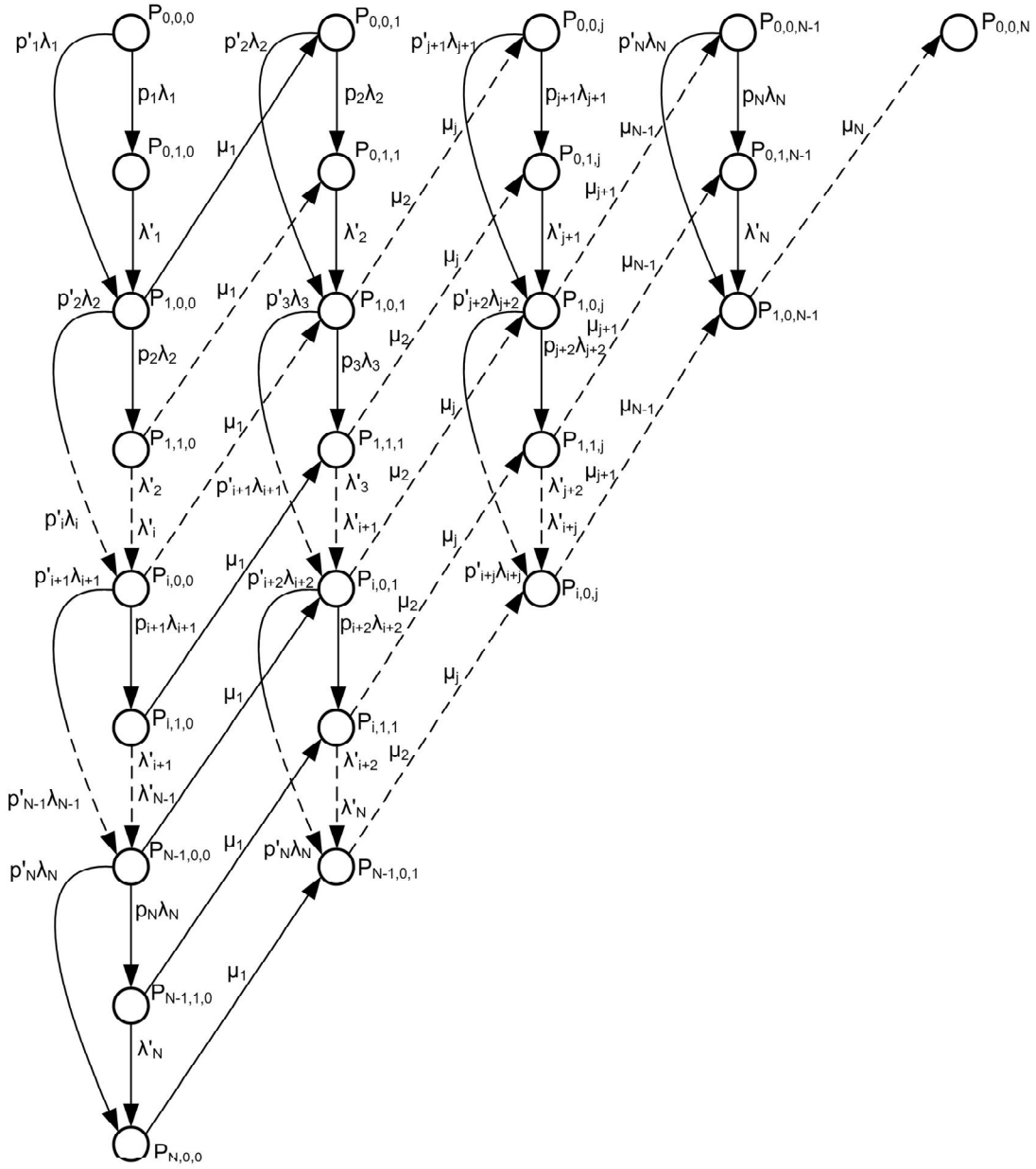


Figure 2. The graph of transitions between states of software reliability C2(i)/M(j)/N model

The graph shown above is described by the following system of the differential equations in total size N_C :

$$\begin{aligned} \frac{dP_{i,k,j}(t)}{dt} = & \delta(i)\delta(1-k)P_{i-1,k+1,j}(t)\lambda'_{i+j} + \\ & + \delta(k)P_{i,k-1,j}(t)\lambda_{i+j+1} + \delta(j)P_{i+1,k,j-1}(t)\mu_j - \\ & - \delta(N-i-j)\delta(1-k)P_{i,k,j}(t)\lambda_{i+j+1} - \\ & - \delta(k)P_{i,k,j}(t)\lambda'_{i+j+1} - \delta(i)P_{i,k,j}(t)\mu_{j+1}, \end{aligned}$$

where $i=0, N$, $j=0, N-i$, $k=0, \delta(N-i-j)$.

Here and next along the paper:

$$\delta(m) = \begin{cases} 1, m > 0 \\ 0, m \leq 0 \end{cases}. \quad (7)$$

(6) The following normalization condition should be met at any time moment t :

$$\sum_{i=0}^N \sum_{j=0}^{N-i} \sum_{k=0}^{\delta(N-i-j)} P_{i,k,j}(t) = 1. \quad (8)$$

It is possible to get the numerical solution of corresponding Cauchy problem for any value t , having set the following initial conditions to the system:

$$P_{i,k,j}(0) = \begin{cases} 1, i+k+j=0 \\ 0, i+k+j \neq 0 \end{cases} \quad (9)$$

The solution of the system (6) can be used to receive a number of software reliability characteristics, such as the probability of detection and correction of a given number of errors for a given time; the mathematical expectation of detected, fixed and remaining number of errors and many others.

For example, the probability $R_i(t)$ of detection of exactly j program errors can be calculated under the following formula:

$$R_i(t) = \sum_{j=0}^i \sum_{k=0}^{\delta(N-i-j)} P_{j-i,k,i}(t) \quad (10)$$

The probability $P_j(t)$ of fixing of exactly j program errors can be calculated under the following formula:

$$P_j(t) = \sum_{i=0}^{N-j} \sum_{k=0}^{\delta(N-i-j)} P_{i,k,j}(t) \quad (11)$$

The time $T_{n,p}$ required to fix n errors with probability p looks like:

$$T_{n,p} = t : F_n(t) \geq P_n, \quad (12)$$

where $F_n(t)$ – distribution function of correction time of n errors:

$$F_n(t) = \sum_{l=n}^N P_l(t), n = \overline{0, N}. \quad (13)$$

The no-failure operation probability $P_{t,\tau}$ during time τ after debugging during time t looks like:

$$\begin{aligned} P_{t,\tau} = & 1 - \sum_{j=0}^{N-1} \int_0^\tau P_{0,0,j}(t) (1 - e^{-P_{j+1} \lambda_{j+1} x}) \times \\ & \times (1 - e^{-\lambda'_{j+1}(\tau-x)}) dx - \\ & - \sum_{j=0}^{N-1} P_{0,0,j}(t) (1 - e^{-P'_{j+1} \lambda_{j+1} \tau}) - \\ & - \sum_{j=0}^{N-1} P_{0,1,j}(t) (1 - e^{-\lambda'_{j+1} \tau}). \end{aligned} \quad (14)$$

The proposed model $C_2(i)/M(j)/N$ is the general one of $M(i)/M(j)/N$ models (Jelinski-Moranda, Musa, Littlewood [1, 2, 3]) in the sense that it enables to analyze the systems

with nonexponential distribution of duration of intervals between the error detection moments. In case of exponential distribution of duration of intervals between the error detection moments proposed model shows the same results as $M(i)/M(j)/N$ models.

The model is programmed as Matlab application. Program functions are calculation of parameters of two-phase Coxian distribution, building of coefficient matrix of differential equation system, which is solved by standard ode45 function. The results are presented as figures and tables of dependence of reliability characteristics from testing time. These results can be saved as MS Excel file.

IV. THE CALCULATION EXPERIMENT

This section contains the results of the calculation experiment which illustrates the use of proposed software reliability model with two-phase Coxian distribution of length of intervals between the moments of detection of errors. The results of calculations received with using of the assumption about exponential distribution of intervals between the moments of detection of errors (as described in papers [12, 13]) are also given for comparison.

Fig. 3 illustrates the dependence of probability that there are no detected (and not corrected) program errors from time. It is assumed that the program contains $N=20$ errors, which are detected and corrected with mean lengths of time intervals equal 12.5 hour ($\lambda_1 = \dots = \lambda_{20} = \mu_1 = \dots = \mu_{20} = 0.08$ hour⁻¹). The first plot on Fig. 3 corresponds to a flow of detection of errors with Coxian distribution of length of intervals between the moments of detection of errors with variation coefficient $v=0.5$. The second plot corresponds to a flow with exponential distribution of named intervals length ($v=1$). The distribution of time of fixing of errors is exponential for both cases.

Fig. 4 contains the plots of the probability of fixing all $N=20$ program errors for the same source data.

Fig. 5 illustrates the dependence of the probability of fixing of all program errors from time for the system with the following mean rates of error detection and correction: $\lambda_1=0.144$, $\lambda_2=0.16$, $\lambda_3=\dots=\lambda_7=0.08$, $\lambda_8=\lambda_9=0.02$, $\lambda_{10}=\dots=\lambda_{19}=0.072$, $\lambda_{20}=0.1$; $\mu_1=0.072$, $\mu_2=0.08$, $\mu_3=\dots=\mu_7=0.04$, $\mu_8=\mu_9=0.02$, $\mu_{10}=\mu_{19}=0.08$, $\mu_{20}=0.16$ – and various values of variation coefficient of distribution of length of intervals between the moments of detection of errors ($v=0.1, 0.3, 0.5, 0.7, 1$).

Table 1 contains the calculated values of required debugging time to achieve the given probability of program error absence for the system with mean rates of error detection and correction shown above.

Table 2 contains the calculated values of required debugging time to achieve the given number of remaining program errors with probability 0.99.

The results and plots presented above show that more regular error detection flow tends to be more efficient from the required debugging time point of view.

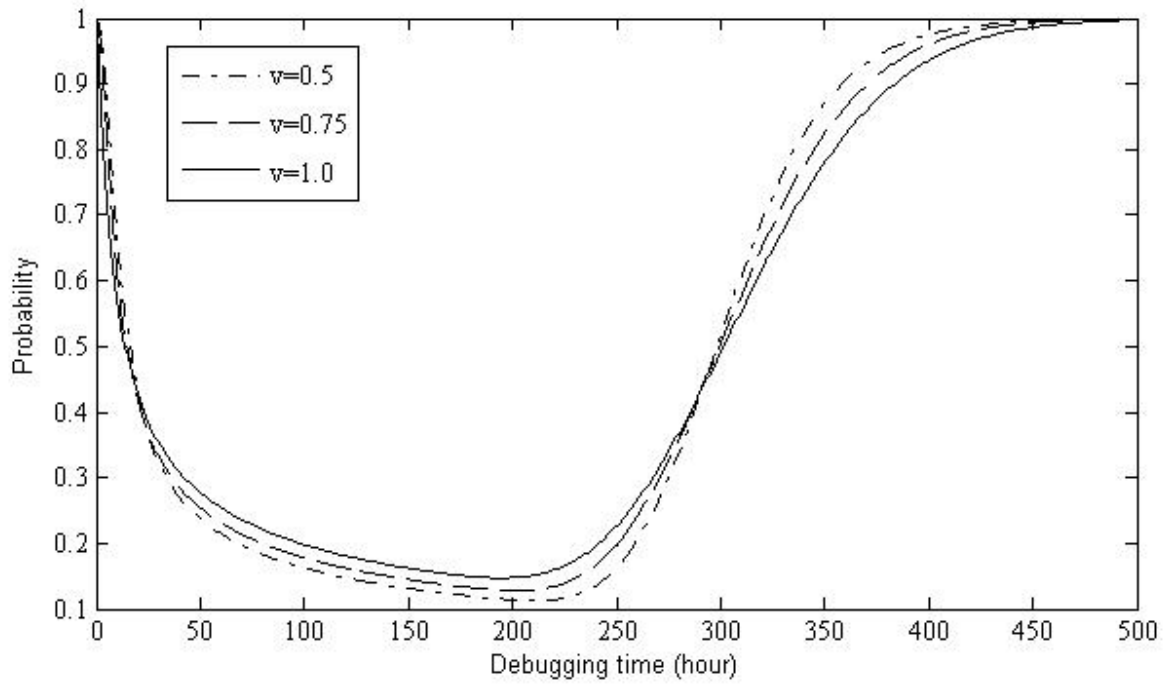


Figure 3. The probability that no program errors have been detected

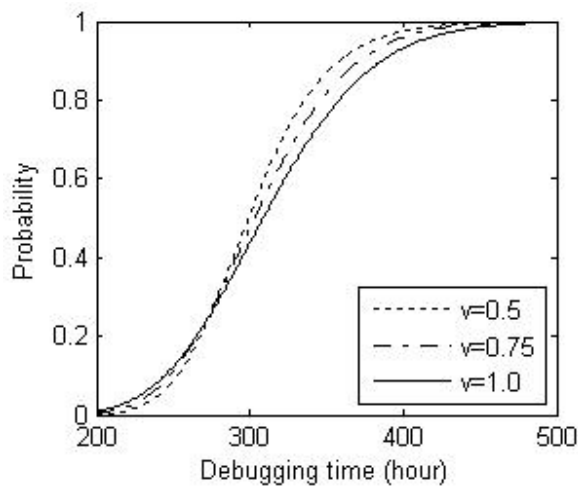


Figure 4. The probability of fixing all program errors

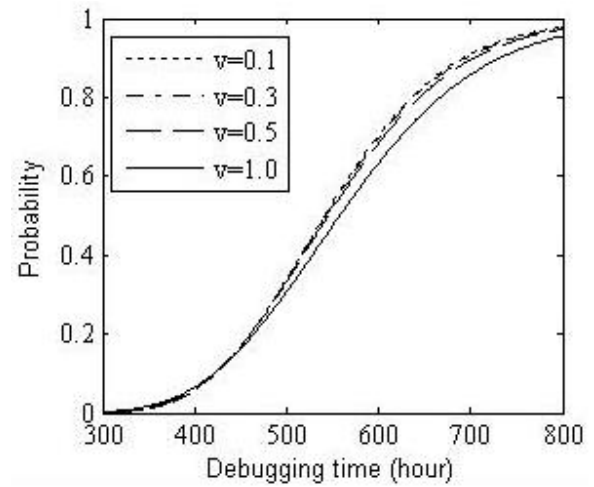


Figure 5. The probability of fixing of all program errors

TABLE I. TIME TO FIX ALL PROGRAM ERRORS

Probability of program error absence	Required debugging time, hour				
	Variation coefficient				
	0.1	0.3	0.5	0.7	1.0
0.8	637	641	645	654	668
0.9	692	696	706	715	734
0.95	745	750	757	769	791
0.99	856	860	870	881	912

TABLE II. TIME TO FIX GIVEN NUMBER OF PROGRAM ERRORS

Number of remaining errors	Required debugging time, hour				
	Variation coefficient				
	0.1	0.3	0.5	0.7	1.0
3	656	665	680	698	736
2	756	760	771	788	819
1	842	847	853	869	898

V. CONCLUSION

In paper the nonstationary model of reliability of the software with two-phase Coxian distribution of length of intervals between the moments of detection of errors is proposed. The model generalizes known software reliability models with exponential or other (for example, Veibull) distribution of length of intervals between the moments of detection of errors which have some limits of use. The results of calculations with use of proposed model are presented. The presented results can be used to forecast the software reliability characteristics and to make plans of software development and testing processes in common with technique of planning of software testing with use of software object-oriented metrics as described in papers [14, 15].

REFERENCES

- [1] R. Lyu, Handbook of Software Reliability Engineering. McGraw Hill, 1996.
- [2] B. Zachariah, and R. N. Rattihalli "Failure Size Proportional Models and an Analysis of Failure Detection Abilities of Software Testing Strategies," IEEE Transactions on Reliability, vol. 56, n. 2, pp. 246-253, June 2007.
- [3] Y. P. Wu, Q. P. Hu, M. Xie, and S. H. Ng "Modeling and Analysis of Software Fault Detection and Correction Process by Considering Time Dependency," IEEE Transactions on Reliability, vol. 56, n. 4, pp. 629-642, December 2007.
- [4] P. Moranda, and Z. Jelinski, Final Report on Software Reliability Study, MADC Report Number 63921. McDonnell Douglas Astronautics Company, 1972.
- [5] J. Musa, A. Iannino, and K. Okumoto, Software Reliability — Measurement, Prediction, Application. McGraw Hill, New York, 1987.
- [6] B. Littlewood "The Littlewood-Verrall Model for Software Reliability Compared with Some Rivals," Journal of Systems and Software, vol. 1, n. 3, pp. 251-258, 1980.
- [7] P.R. Cox "A use of Complex Probabilities in the Theory of Stochastic Processes," Proc. Cambr. Soc., vol.51, pp. 313-319, 1955.
- [8] A.D. Khomonenko, and V.P. Bubnov "A use of Coxian Distribution for Iterative Solution of $M/G/n/R \leq \infty$ Queueing Systems," Probl. of Control and Inform. Theory, vol. 14, n. 2, pp. 143-153, 1985.
- [9] H. Okamura, and T. Dohi "Building Phase-Type Software Reliability Models," 17th International Symposium on Software Reliability Engineering (ISSRE'06), pp.289-298, 2006.
- [10] H. Okamura, Y. Watanabe, and T. Dohi "Estimating Mixed Software Reliability Models Based on the EM Algorithms," Proc. of 2002 Int. Sympo. Empirical Software Eng. (ISESE-2002), IEEE CS Press, pp. 69-78, 2002.
- [11] L. Takács, Introduction to the Theory of Queues. Oxford University Press, 1962.
- [12] V.P. Bubnov, V.I. Safonov, and V.A. Smagin "About Computational System Utilization with Changing Intensity of Task Arrival," Automatics and Computer Engineering, n. 6, pp. 19-22, 1987 (in Russian).
- [13] V.P. Bubnov, and V.I. Safonov, Development of Dynamic Models of Nonstationary Service Systems. Lan', St. Petersburg, 1999 (in Russian).
- [14] A.V. Tyrva, and A.D. Khomonenko, "The Method for Software Complexes Testing Scheduling on Design and Development Phases," Scientific and Technical Herald of StPSPU / Informatics. Telecommunications. Control, n. 4(82), pp. 125-131, 2009, (in Russian).
- [15] V.V. Kuznecov, and V.A. Smagin, "Direct and Reverse Tasks of Reliability of Software Complexes," Reliability and quality control, n.10, pp. 56-62, 1997.