The block diagram of reliability analysis usage for analysis of safety critical systems

D. GABRIŠKA

Abstract

Reliability of the technological processes or reliability of devices used in different industries is an important part of designing safety critical systems. The failure of such systems leads to economic losses, health damage or environmental pollution. An important role in the development of safety critical systems is therefore the reliability analysis, the assessment of the risks associated with the use of the technical means and the consequent reduction of this risk. The actual level of risk considered tolerable will vary depending on a number of factors such as the level of human control over the circumstances, the voluntary or unintentional nature of the risk, the number of people at risk in each individual case, the degree of responsibility placed on safety and critical systems reflects the need for quality design and ensure of software safety. Various standards and methods are used to achieve the desired level of safety. One of the methods used for reliability analysis is the use of a block diagram of reliability.

Mathematics Subject Classification 2010: 97P50, 68N30
Additional Key Words and Phrases: Safety functions, risk analysis, E/E/PE systems, safety critical system

1. INTRODUCTION

The safety system includes all components (hardware, hardware, software, personnel) needed to perform one or more safety features. The necessary task is a review and evaluation of the various safety requirements imposed on software from hardware to software architecture [6,8]. Failure of a safety feature may significantly increase the risk of a safety threat to personnel and / or the surrounding environment [2,7,9]. The safety-related system may include an autonomous device that is designed to perform an individual safety function (for example, a fire system) or may be integrated into another device (for example, a machine speed control system in the machine). The development of dependable software also needs to include protection measures against external worst-case scenario network attacks that could compromise the availability and security of the system [11].

The functional safety of the application software is part of the overall safety system that meets the requirements of the proper functioning of the system or device

10.1515/jamsi-2017-0007

in response to input signals [1,3,5]. Functional safety is achieved when each of these safety functions is performed and the required performance level of each function is achieved.

2. EVALUATING THE RELIABILITY OF SOFTWARE SAFETY AND CRITICAL SYSTEMS

One of the options to ensure functional safety is to evaluate the reliability of the safety-critical software [4]. Allows you to determine the probability of trouble-free operation within a specified time of operation.

Operation between individual failures - the random variable $T^{(i)}$– can be represented by the sum of two random variables [6]:

$$T^{i} = T^{(i-1)} + \Delta T^{(i)} \tag{1}$$

Gradually, applying the relation (1) to all the operating periods between the disturbances, we get the relation:

$$T^{(i)} = T^{(0)} + \sum_{v=1}^{i} \Delta T^{(v)} \tag{2}$$

The random variable T_n runs until the *n-th* program fails

$$T_n = \sum_{i=0}^{n} T^{(i)} = \sum_{i=0}^{n} \left[T^{(0)} + \sum_{v=0}^{n} T^{(v)} \right] \tag{3}$$

We introduce the following assumptions:

1. All random variables $\Delta T^{(v)}$ are independent and have the same mathematical expectations m_Δt and the standard deviation $\sigma_{\Delta t}$;
2. The random variable $T^{(0)}$ is negligibly small compared to the sum $\sum_{v=1}^{i} \Delta T^{(v)}$.

The basis of second assumptions may be the following conclusion. At the full initial period of runtime of the program errors occur very often, i.e., time T_0 is

small. For the sum (3) grows rapidly with increasing n and the ratio T_0 is rapidly decreasing. We will assume that $T^{(0)} \approx \Delta T^{(0)}$.

In accordance with the second assumption of relationship (2), we get:

$$T^{(n)} = \sum_{\upsilon=0}^{n} \Delta T^{(\upsilon)} \tag{4}$$

$$T_n = \sum_{i=0}^{n} \sum_{\upsilon=0}^{i} \Delta T^{(\upsilon)} \ = \mathrm{n}\Delta T^{(0)} + (n-1)\Delta T^{(1)} + \cdots + \Delta T^{(n)} \tag{5}$$

At the same of values ΔT_0 of operation between (n-1) and n-th fault, the random of value $T^{(n)}$ acquires the arithmetic mean:

$$m_t^{(n)} = M\left[T^{(n)}\right] = n m_{\Delta t} \tag{6}$$

and the standard deviation is:

$$\sigma_{n(t)} = \sigma_{\Delta t}\sqrt{n} \tag{7}$$

The random variable T_n is the arithmetic mean:

$$m_{t_n} = m_{\Delta t}\frac{n(n+1)}{2} \tag{8}$$

The standard deviation is:

$$\sigma_{t_n} = \sigma_{\Delta t}\sqrt{\frac{1}{6}n(n+1)(2n+1)} \tag{9}$$

For the calculation of the values $m_t^{(n)}, m_{\Delta t}\ and\ \sigma_{t_n}$ it is necessary to find statistical evaluation of the numerical characteristics of the random difference $\Delta T^{(i)} = T^{(i)} - T^{(i-1)}$ according to the program error data during the observed period t_i.

$$m_{\Delta t}^{*} = \frac{1}{n_H}\sum_{i=0}^{n_H}\Delta t_i = \frac{1}{n_H}\sum_{i=1}^{n_H}[t_i - t_{i-1}]; \tag{10}$$

$$[\sigma_{\Delta t}^2]^* = \frac{1}{n_H - 1}\sum_{i=0}^{n_i}[\Delta t_i - m_{\Delta t}^*]^2\,, \tag{11}$$

where n_H - number of program failure during operation $(0, t_H)$.

Since, in the case $t > t_H$ of the number of faults $n_H \gg 1$ from formulas (8) and (9), we get:

$$m_{t_n} \approx m_{\Delta t}\ \frac{n^2}{2;}\ ,\sigma_{t_n} \approx \sigma_{\Delta t}\sqrt{\frac{n^3}{3}} \tag{12}$$

Since the random values T^n and $T_n$with respect to formulas (4) and (5) are equal to the sum of many random variables, the values $T^{(n)}$ and T_n can be considered as normally distributed with arithmetic mean and scatter. As the operation is positive, practically a limited normal distribution (0, ∞) is used. Normally the normalization factor is ≈ 1 .

For $n > n_H$ the probability distribution of the probability of operation between the other (*n*-1) and *n*-th faults is:

$$f^{(n)}(\tau) = \frac{1}{\sigma_{\Delta t}\sqrt{2\pi n}}\ exp\left[-\frac{1}{2}\frac{(\tau - nm_{\Delta t})^2}{n\sigma_{\Delta t}^2}\right], \tag{13}$$

where τ is calculated from the moment of the last (*n*-1) fault.

Corresponding function of probability distribution between failures:

$$F^{(n)}(\tau) = \frac{1}{2} + \Phi\left(\frac{\tau - nm_{\Delta t}}{\sigma_{\Delta t}\sqrt{n}}\right), \tag{14}$$

where Φ(u) – table function, $\Phi(u) = -\frac{1}{\sqrt{2n}}\int_0^u exp\left(-\frac{v^2}{2}\right)\,dv$.

To calculate the probability of a trouble-free operation of the program, it is more convenient to use the agreed probability function (probability that the random operating time in the fault will be greater than the specified operating time, calculated from the moment of the last fault (*n*-1)):

$$p^{(n)}(\tau) = \frac{1}{2} - \Phi\left(\frac{\tau - nm_{\Delta t}}{\sigma_{\Delta t}\sqrt{n}}\right) \tag{15}$$

The probability of trouble-free operation within the specified operating time *(τ1, τ2)* after the (*n*-1) fault is calculated as:

$$p^{(n)}(\tau_1, \tau_2) = \frac{p^n(\tau_2)}{p^n(\tau_1)} \tag{16}$$

Due to the previous program failure assumptions, they create a diminishing random current. The main function of the current, i.e. the average number of failures that have occurred during operation (0, *t*) at $t > t_H$ is:

$$\Omega(t) = \sum_{n=1}^{\infty} F_n(t) = \sum_{n=1}^{\infty}\left[\frac{1}{2} + \Phi\left(\frac{t - \frac{n^2 m_{\Delta t}}{2}}{\sigma_{\Delta t}\sqrt{\frac{n^3}{3}}}\right)\right] \tag{17}$$

Parameter fault current programs (calculated according to the service)

$$\varpi(t) = \sum_{n=1}^{\infty} f_n\ (t) \tag{18}$$

The density of the time division to the *n*-th event is:

$$f_n(t) = \frac{\sqrt{3}}{n\sigma_{\Delta t}\sqrt{2\pi n}}\ exp\left[-\frac{3}{2}\frac{\left(t - \frac{1}{2}n^2 m_{\Delta t}\right)^2}{n^3\sigma_{\Delta t}^2}\right] \tag{19}$$

From the relation (14) for $t > t_H$, we have the expression for the program fault current parameter:

$$\omega(t) = \sum_{n=1}^{\infty}\frac{\sqrt{\frac{3}{2n}}}{n\sigma_{\Delta t}\sqrt{\pi}} exp\left[-\frac{3}{2n}\frac{\left(t - \frac{n^2}{2}m_{\Delta t}\right)^2}{n^2\sigma_{\Delta t}^2}\right] \tag{20}$$

Given the complexity of the terms (17) and (20), it is appropriate to approximate them by simpler terms. Practically, it makes sense to use the least squares method. In accordance with this method, the approximate function (for $\varpi(t)$ is appropriate to use $Aexp(-vt)$)is best represented at intervals (t_H, t_1) with a function that is determined by the relation (15):

$$\int_{t_H}^{t_1} \left\{ Aexp(-vt) - \sum_{n=1}^{\infty} \frac{\sqrt{\frac{3}{2n}}}{n\sigma_{\Delta t}\sqrt{\pi}} exp\left[-\frac{3}{2n} \frac{\left(t - \frac{n^2}{2} m_{\Delta t}\right)^2}{n^2 \sigma_{\Delta t}^2} \right] \right\} dt = min \tag{21}$$

When the partial derivations of the integral „I_1“ equal zero at „A“ and „v“, we obtain a system of equations for determining these numerical characteristics. Analogously, it can also be done when approximating the $\Omega(t)$ function of $1 - Bexp(-\gamma t)$.

2.1. Program for readiness evaluation

When evaluating the program's readiness, we discuss the process of restoring the program's operational capability. Running time between other program restorations:

$$T_0^{(i)} = T^{(i)} + T_B^{(i)} \tag{22}$$

where $T^{(i)}, T_B^{(i)}$ - Independent random variables.

The value $T_0^{(i)}$ is determined according to the formula (22). Due to the accumulation of program restoration experience, the magnitude can be expressed in the following form:

$$T_0^{(i)} = T_B^{(i-1)} - \Delta T^{(i)} \tag{23}$$

Then, using the relationship (18) to all other restorations, we get:

$$T_B^{(i)} = T_B^{(0)} - \sum_{V=1}^{i} \Delta T_B^{(V)} \tag{24}$$

After replacing the terms in accordance with the expressions (22) and (23) in (24), we obtain:

$$T_0^{(i)} = T^{(0)} + T_B^{(i)} + \sum_{V=1}^{i} \left[\Delta T^{(V)} - \Delta T_B^{(V)}\right] \tag{25}$$

Random running time variable until the *n*-th disorder programs

$$T_{on} = \sum_{i=0}^{n} T_0^{(i)} = \sum_{i=0}^{n} \left[T_0^{(0)} + \sum_{V=1}^{i} \Delta T_0^{(V)}\right] \tag{26}$$

where indicated

$$T_0^{(0)} = T^{(0)} + T_B^{(0)}; \quad T_0^{(V)} = \nabla T^{(0)} - T_B^{(0)}. \tag{27}$$

Let us assume the independence $\Delta T_0^{(V)}$, the equivalence of their arithmetic mean and dispersion and the small value $T_0^{(0)}$ as compared to the sum $\Delta T_0^{(V)}$ at large values. Additionally, usually $T_B^{(i)} \ll T^{(i)}$ can be used. When we give $T_0^{(0)} \approx \Delta T_0^{(0)}$, we get:

$$T_{0n} = \sum_{i=0}^{n} \sum_{V=0}^{i} \Delta T_0^{(V)} = n\Delta T_0^{(0)} + (n-1)\Delta T_0^{(1)} + \cdots + \Delta T_0^{(n)} \tag{28}$$

For the same $\Delta T_0^{(V)}$, the random variable T_{0_n} has an arithmetic mean:

$$m_{t_{on}} = m_{\Delta t_0} \frac{n(n+1)}{2}; \tag{29}$$

and the standard deviation:

$$\sigma_{t_{0n}} = \sigma_{\Delta_{t0}} \sqrt{\frac{1}{6} n(n+1)(2n+1)}, \tag{30}$$

where $m_{\Delta t_0}$, $\sigma_{\Delta_{t0}}$ – arithmetic mean and standard deviation ΔT_0.

Considering that according to the formulas (29), (30)

$$m_{\Delta t_o} = m_{\Delta t} - m_{\Delta t_n};\ \ \sigma_{\Delta t_0} = \sqrt{\sigma_{\Delta t}^2 + \sigma_{\Delta t_B}^2}\,, \tag{31}$$

We get:

$$m_{t_{on}} = \frac{1}{2} n(n+1)\left(m_{\Delta t} - m_{\Delta t_B}\right); \tag{32}$$

$$\sigma_{t_{0n}} = \sqrt{\frac{1}{6} n(n+1)(2n+1)\left(\sigma_{\Delta t}^2 + \sigma_{\Delta t_B}^2\right)} \tag{33}$$

At $n \gg 1$

$$m_{t_{on}} \approx \frac{n^2}{2}\left(m_{\Delta t} - m_{\Delta t_B}\right);\quad \sigma_{t_{0n}} = \sqrt{\frac{n^2}{3}\left(\sigma_{\Delta t}^2 + \sigma_{\Delta t_B}^2\right)} \tag{34}$$

The values $m_{t_{on}}$ and $\sigma_{t_{0n}}$ are evaluated according to the statistical data at the recovery time (remove the error) of the programs analogous to the values $m_{\Delta t}$ and $\sigma_{\Delta t}^2$.

$$\omega_0(t) = \sum_{n=1}^{\infty} f_{0_n}(t), \tag{35}$$

After calculation $m_{t_{on}}$ and $\sigma_{t_{0n}}$ it can be found the recovery flow parameter:

$$H(t) = p\left\{\bigcup_{n=1}^{\infty} A_n\right\} = \sum_{n=0}^{\infty} p\{A_n\}\,, \tag{36}$$

The $H(t)$ is a preparedness function, which expresses the probability of the program occurring in the operational state at time *t*.

For each A_n , fault and reset occurred in the moment *t* and the program is operational at time *t*.

To estimate the probability of occurrence of A_n, let us consider a small interval$(\Theta, \Theta + \mathrm{d}\Theta)$, that precedes *t*. The probability that the last nth renewal will

end at this interval and the program does not fail several times in the remaining time $(t-\Theta)$ is equal:

$$f_{0n}(\Theta)d\Theta\left[1-F^{(n+1)}(t-\Theta)\right] \tag{37}$$

The equal $F^{(n+1)}(t-\Theta)$ is a function of the time division between the end of the nth renewal and the (n + 1) disorder.

When integrating this function according to Θ from 0 to t we get:

$$p\{A_n\}=\int_0^t\left[1-F^{(n+1)}(t-\Theta)\right]>d\Theta \tag{38}$$

Replacing the expression for probability $p\{A_n\}$we get:

$$H(t)=1-F^{(1)}(t)+\sum_{n=1}^{\infty}\int_0^t[1-F^{n+1}(t-\theta)]f_{0_n}(\theta)d\theta\,. \tag{39}$$

Given that the practical value has only a value $t>t_H$, where several tens of failures have occurred and taking into account that $n\geq n_i\gg 1$, we get:

$$\begin{aligned}&H(t)\\&=\sum_{n=1}^{\infty}\int_0^t\left[\frac{1}{2}-\Phi\left(\frac{t-\Theta-nm_{\Delta t}}{\sigma_{\Delta t}\sqrt{n}}\right)\right]\\&\cdot\langle\frac{\sqrt{\frac{3}{2n}}}{n\sigma_{\Delta t}\sqrt{\pi}}exp\left\{-\frac{3}{2n}\frac{\left[\theta-\frac{n^2}{2}(m_{\Delta t}-m_{\Delta t_n})\right]^2}{n^2\left(\sigma_{\Delta t}^2+\sigma_{\Delta t_B}^2\right)}\right\}\rangle\,d\theta\end{aligned} \tag{40}$$

Since $t\rightarrow\infty$ of the $H(t)\rightarrow\infty$ value, it is appropriate to approximate the complex expression (40) by simpler expressions.

For example, for $H(t)=1-C\cdot\exp(-\delta t)$, it is suitable to find C and δ using the least squares method by analogy (21). In this way, for easy use, it will be possible to use simple relationships that take into account the improvement of programs and the training of staff.

CONCLUSION

The planning and design of safety-critical systems is performed by various methods. One method for assessing the reliability of a safety-critical software is to determine the probability of trouble-free operation within the specified time of operation. Determining the probability of trouble-free operation enables safety critical systems to determine which probability of malfunctioning software meets the requirements set for a particular level of critical safety.

REFERENCES

[1] MUDRONČÍK, D., GÁLIK, M. 2009. Standards for creating software of control systems. Available on the Internet: http://www.odbornecasopisy.cz/res/pdf/38879.pdf

[2] MUDRONČÍK, D., ZOLOTOVÁ, I. 2000. Industrial programmable controllers. Elfa, s. r. o., Košice.

[3] STN EN ISO 13849-1. 2015. Safety of machinery. Safety-related parts of control systems. Part 1:General principles for design

[4] RÁSTOČNÝ, K., ŽDÁNSKY, J. 2007. Control systems with safety PLC. Žilina: University of Žilina.

[5] Standard IEC 61508-1 2010. Functional safety of electrical/electronic/ programmable electronic safety-related systems. Part 1: General requirements.

[6] SHISHMAREV, V. 2010. Reliability of technical systems. Moskva, Akademia, p.308. Available on the Internet: http://www.twirpx.com/file/525173/

[7] GABRIŠKA, D. 2017. Functional safety of safety-critical programmable electronic systems. Faculty of Natural Sciences in Trnava. ISBN 978-80-8105-869-1.

[8] SMITH, D. J. Reliability, Maintainability and Risk: Practical Methods for Engineers. Butterworth-Heinemann, 2017, 478p.

[9] REUS T., GARBSCH M. Functional safety - Safety Integrity Level. [online], JUMO, 2016, 22 p. ISBN 978-3-935742-18-4. Available on the Internet: http://www.jumo.net/zpv/attachment/01/de/de_DE//attachmentdownload?id=5881

[10] GABRIŠKA, D. 2016. Software requirements for the control systems according to the level of functional safety, Journal of Applied Mathematics, Statistics and Informatics, 2016, Vol.12, pages 25-32

[11] ŠIMON, M., DIRGOVÁ LUPTÁKOVÁ I., HURAJ, L., HOSŤOVECKÝ, M., AND POSPÍCHAL, J. 2017. Combined Heuristic Attack Strategy on Complex Networks, Mathematical Problems in Engineering, Article ID 6108563, 9 pages, doi:10.1155/2017/6108563

Darja Gabriška
Department of Applied Informatics and Mathematics,
University of SS. Cyril and Methodius, 917 01 Trnava,
Slovak Republic
Email: darja.gabriska@ucm.sk