# Early estimating the number of errors encountered during program testing

G. KONOPACKI, K. WORWA
e-mail: gkonopacki@wat.edu.pl

Institute of Information Systems
Cybernetics Faculty, Military University of Technology
Kaliskiego Str. 2, 00-908 Warsaw

An approach to estimate the number of errors encountered during the program testing process is proposed in the paper. Considerations are based on some program reliability growth model which is constructed for assumed scheme of program testing process. In this model the program under the testing is characterized by means of so-called characteristic matrix and the program testing process is determined by means of so-called testing strategy. The formula for determining the mean value of the predicted number of errors encountered during the program testing is obtained. This formula can be used if the characteristic matrix and the testing strategy are known. Formula for estimating this value when the program characteristic matrix is not known are also proposed in the paper.

**Keywords:** Software testing; Software reliability; Program correctness; Reliability growth model

## 1. Introduction

The testing of a newly developed program, prior to its practical use, is a commonly followed practice. The program testing process involves the execution of the program with many sets of input data with the intention of finding errors. Testing is done to lower the chances of in-service failures which are defined as an unacceptable departure from a program operation. A long period of testing results in increasing the chances of detecting program errors and decreasing the chances of in-service failures, but it also results in increasing the cost of the program testing process.

It is known that testing is the most significant money consuming stage of the program development. The cost of the program testing process can make 50-70 percent of the total cost of the program development, especially for complex program systems [3]. Considering the essential impact of the testing cost on the whole program development cost, the testing process ought to be prudently planned and organized. Decisions relative to the testing process organization should be made on the basis of the results of testing efficiency analysis. In order to make such an analysis easier it may be convenient to estimate the number of program errors that could be encountered during the process of program testing. The knowledge of this estimation makes it possible to evaluate the duration and the cost of the program testing process, e.g. by means of formal, mathematical expressions. Such evaluations can be very useful in practice, e.g. for comparing the effectiveness of different ways of program testing process organizations (i.e. in order to find an optimal organization).

The number of program errors encountered during the testing process depends on many factors, such as the testing process organization (which defines the manner of the testing process realization), the duration of the testing, the qualifications and professional of testers experience, and the reliability level of the program at the beginning of the testing process. The duration of the program testing process can be determined by a time spent on testing activities (it may be a calendar time or so-called execution time) or by the cardinality of the set of input data used for the testing. The first way of the two mentioned above is characteristic for so-called time-domain models of software testing and the other way is specific to so-called data-domain models [2,7].

This paper attempts to describe a probabilistic model of the program testing process for some scheme of the program testing and to determine a formula to estimate the number of program errors encountered during the testing process.

## 2. Description of the program testing scheme

An organization of a program testing process depends on the program testing strategy which was selected for the testing. In particular, this strategy defines the way of the testing process realization and the set of program input data which is used for the testing.

It is assumed that the program testing process under investigation consists of a number of organizational units of the program testing process that are called the testing stages. Every stage of the program testing process consists of the two following parts of testing:

- the testing of the program with a prepared set of program input data (tests),
- the comparison of the results with the expected outputs for the data used and the correction of encountered errors.

Let *S* mean the program testing strategy which is defined as follows

$$S = (K, (L_1, L_2, ..., L_k, ..., L_K)), \qquad (1)$$

where:

$K$ – the number of stages of the program testing process,

$L_k$ – the number of program input data (tests) used in the $k$-th program testing stage, $L_k > 0,\ k = \overline{1,K}$.

Let *S* denote the set of all strategies that have the form (1). The strategy (1) defines a program testing scheme. In accordance with this scheme, the process of correction of program errors which were encountered during the *k*-th program testing stage can be started after the execution of the program on all $L_k$ tests is finished. A situation that a number of different tests of all $L_k$ tests executed during the *k*-th stage encounter the same error in the program under the testing is possible. It is assumed that every time when the program execution on the single test leads to encounter a program error, a so-called error message is generated. So, according to the note mentioned above, it is possible to obtain several times the same message signalling the same program error.

Let $M_k(S)$ denote the number of error messages generated during the *k*-th stage of program testing process according to the testing strategy *S*.

Let $N_k(S)$ mean the number of errors encountered during the *k*-th stage of that process.

While planning the program testing process it is reasonable to consider the values $M_k(S)$, $N_k(S)$, $k = \overline{1,K}$, as random variables. According to earlier remarks the value $N_k(S) \le M_k(S)$. So we can write

$$N_k(S) \in \{0,1,2,...,M_k(S)\},\ k = \overline{1,K}. \qquad (2)$$

Let $p_{nm}$ define the probability of an event that *n* errors will be encountered during a single stage of the program testing if there are *m* error messages in that stage, i.e.

$$p_{nm} = Pr\left\{N_k(S) = n / M_k(S) = m\right\},\ k=\overline{1,K}. \qquad (3)$$

If we assume that every execution of the program on a single test can lead to at most one error message and this message means encounter of exactly one program error, we will have:

$$p_{00} = p_{11} = 1,$$
$$p_{nm} = 0 \quad \text{if} \quad n > m, \qquad (4)$$
$$p_{nm} > 0 \quad \text{if} \quad n \le m,\ n > 0,$$

and

$$\sum_{n=0}^{\infty} p_{nm} = 1, \qquad m \in \{0,1,2,...\}. \qquad (5)$$

Probabilities $p_{nm}$ $n\in\{0,1,...,m\}$, $m\in\{0,1,...\}$, form an infinite matrix *P*=[ $p_{nm}$ ] as follows

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & ... \\ 0 & 1 & p_{12} & p_{13} & p_{14} & \cdots \\ 0 & 0 & p_{22} & p_{23} & p_{24} & \cdots \\ 0 & 0 & 0 & p_{33} & p_{34} & \cdot\cdot \\ 0 & 0 & 0 & 0 & p_{44} & \cdots \\ & & \cdots & & & \end{bmatrix}. \qquad (6)$$

where values $p_{nm}$ are defined by (3).

The matrix *P* contains the values 0 below the main diagonal because it is not possible to encounter more different errors than the signalized number of error messages.

The dependence (3) means that the values of probabilities $p_{nm}$ for every program testing stage only depend on the number of tests which are used during that stage. The matrix *P* will be called the characteristic matrix of the program under testing.

The probability of a correct execution of the program under investigation with a single input data (test) is assumed to be a program reliability coefficient in the paper.

If a number of errors is encountered in the program and they are successfully corrected then the program reliability level will increase. On the basis of facts described in literature [2,3,4,5] it is assumed that an increase of the program reliability coefficient, obtained through encountering and correcting *m* errors, is of the form

$$\Delta r(m) = (1-r)(1-e^{-\alpha m}), \qquad (7)$$
$m \in \{0,1,2,...\}$,

where:

$r$ – an initial value of the program reliability coefficient, i.e. at the beginning of the program testing process,

$\alpha$ – a parameter that characterizes both the internal structure of the program under testing and the impact of one error removal on the increase of the program reliability.

## 3. Estimating the mean value of the number of errors encountered during the program testing process

Let $N(S,P)$ mean the total number of errors encountered during the process of the program testing in accordance with the testing strategy $S$ and the characteristic matrix $P$. The value $N(S,P)$ is a random variable.

On the basis of previous assumptions and (7) it can be shown that the probability distribution function of the random variable $N(S,P)$ has the form as follows [6]

$$Pr\{N(S,P)=n\} = \sum_{n_1+n_2+...+n_K=n} Pr\{N_1(S)=n_1,...,N_K(S)=n_K\} =$$
$$= \sum_{n_1+n_2+...+n_K=n} \prod_{l=1}^{K} \sum_{m_l=0}^{L_l} p_{n_l m_l} A_{m_l}(\sum_{i=1}^{l-1} n_i, L_l),$$
$$n=\overline{0,L(S)} \qquad (8)$$

where

$$A_{m_1}(\sum_{i=1}^{l-1} n_i, L_l) = \binom{L_l}{m_l} \cdot \left[ e^{-\alpha \sum_{i=1}^{l-1} n_i} (1-r) \right]^{m_l} \cdot \qquad (9)$$
$$\cdot \left[ 1 - e^{-\alpha \sum_{i=1}^{l-1} n_i} (1-r) \right]^{L_l - m_l}$$

and value $L(S)$ is the maximal – according to the testing strategy $S$ – number of encountered errors, i.e.

$$L(S) = \sum_{k=1}^{K} L_k . \qquad (10)$$

The probability $Pr\{N_1(S)=n_1,...,N_K(S)=n_K\}$, $n_k=\overline{0,L_k}$, $k=\overline{1,K}$, denotes a distribution of multidimentional random variable $(N_1(S)=n_1,...,N_K(S)=n_K)$.

The knowledge of the probability distribution function (8) makes it possible to obtain a formula for the mean value of the number of errors encountered during the program testing process according to the strategy $S$. We have

$$E[N(S,P)] = \sum_{n=0}^{L(S)} nPr\{N(S,P)=n\},$$

and then, according to (8)

$$E[N(S,P)] =$$
$$= \sum_{n=0}^{L(s)} n \sum_{n_1+n_2+...+n_K=n} \prod_{l=1}^{K} \sum_{m_l=0}^{L_l} p_{n_l m_l} A_{m_l}(\sum_{i=1}^{l-1} n_i, L_l) =$$
$$= \sum_{k=1}^{K} \sum_{n_1=0}^{L_1} \sum_{n_2=0}^{L_2} \cdots$$
$$\cdots \sum_{n_K=0}^{L_K} n_K \prod_{l=1}^{K} \sum_{m_l=0}^{n_l} p_{n_l m_l} A_{m_l}(\sum_{i=1}^{l-1} n_i, L_l) \qquad (11)$$

The formula (11) for estimating the mean value of the number of errors encountered during the program testing process can be used in practice if the program characteristic matrix $P$ is known. If the probabilities $p_{n_k m_k}$, $n_k$, $m_k \in \{0,1,2,...L_k\}$, $k=\overline{1,K}$, are unknown, it is possible to estimate the boundary values of this estimation by means of the following theorem.

**Theorem 1.** Let $P^*$, $P^{**}$ denote characteristic matrixes of the program under the testing that have forms:

$$P^* = \begin{bmatrix} 1 & 0 & 0 & 0 & \dots \\ 0 & 1 & 0 & 0 & \dots \\ 0 & 0 & 1 & 0 & \dots \\ 0 & 0 & 0 & 1 & \dots \\ & & \dots & & \end{bmatrix},$$

$$P^{**} = \begin{bmatrix} 1 & 0 & 0 & 0 & \dots \\ 0 & 1 & 1 & 1 & \dots \\ 0 & 0 & 0 & 0 & \dots \\ 0 & 0 & 0 & 0 & \dots \\ & & \dots & & \end{bmatrix}. \quad (12)$$

Then, for any program testing strategy *S* and any characteristic matrix *P* is

$$E[N(S,P^{**})] \le E[N(S,P)] \le E[N(S,P^*)]. \quad (13)$$

Using the previous denotations and assumptions we can also prove the following theorem.

**Theorem 2.** Let $P^*, P^{**}$ mean the program characteristic matrixes that have the forms (12). Let $S', S'' \in S$ denote the following program testing strategies:

$$S' = (L, \underbrace{(1,1,\dots,1)}_{L\ times}), \quad (14)$$

$$S'' = (1, L). \quad (15)$$

Then we have:

$$\Delta r(S', P^*) = \min_{S \in S(L)} \Delta r(S, P^*) \quad (16)$$

$$\Delta r(S'', P^*) = \max_{S \in S(L)} \Delta r(s, P^*) \quad (17)$$

and

$$\Delta r(S', P^{**}) = \max_{S \in S(L)} \Delta r(S, P^{**}) \quad (18)$$

$$\Delta r(S', P^{**}) = \min_{S \in S(L)} \Delta r(S, P^{**}) \quad (19)$$

*where*

$$S(L) = \{S = (K, (L_1, L_2, \dots, L_k, \dots, L_K)) \in \in S: \sum_{k=1}^{K} L_k = L\} \quad (20)$$

Both the theorem 1 and theorem 2 can be proved on the basis of the results that have been obtained in [5,6].

The formula (11) can be simplified if the program characteristic matrix *P* has a specific form. For example, if the program characteristic matrix has the form (12) we will obtain:

$$E[N(S,P^*)] = = (1-r) \sum_{k=1}^{K} L_k \sum_{n_1=0}^{L_1} \sum_{n_2=0}^{L_2} \dots \sum_{n_{k-1}=0}^{L_{k-1}} e^{-\alpha \sum_{i=1}^{k-1} n_i} \prod_{l=1}^{k-1} A^*_{n_l}(\sum_{i=1}^{l-1} n_i, L_l), \quad (21)$$

and

$$E[N(S,P^{**}] = \sum_{k=1}^{K} \sum_{n_1=0}^{1} A^{**}_{n_1}(0, L_1) \cdots \sum_{n_{k-1}=0}^{1} A^{**}_{n_{k-1}}\left(\sum_{i=1}^{k-2} n_i, L_{k-1}\right) A^{**}_1\left(\sum_{i=1}^{k-1} n_i, L_k\right), \quad (22)$$

where

$$A^*_{n_l}(\sum_{i=1}^{l-1} n_i, L_l) = \left[ e^{-\alpha \sum_{i=1}^{l-1} n_i} (1-r) \right]^{n_l} \cdot \left[ 1 - e^{-\alpha \sum_{i=1}^{l-1} n_i} (1-r) \right]^{L_l - n_l} \quad (23)$$

and

$$A^{**}_{n_k}(\sum_{i=1}^{k-1} n_i, L_k) = = \left\{ 1 - \left[ 1 - e^{-\alpha \sum_{i-1}^{k-1} n_i} (1-r) \right]^{L_k} \right\}^{n_k} \cdot \left\{ 1 - e^{-\alpha \sum_{i-1}^{k-1} n_i} (1-r) \right\}^{L_k(1-n_k)} \quad (24)$$

For example, if $P = P^*$ and *S*=(*2,*(*1,1*))*,* we will have

$$E[N(S,P^*)] = (1-r)[1 + r + e^{-\alpha}(1-r)].$$

## 4. Conclusions

The formula for determining the mean value of the number of errors encountered during the program testing process (11) has been obtained under the assumption that both the program testing scheme and the program reliability growth model are of the forms presented in chapter 2. It is noteworthy that the assumed program testing scheme and the program testing strategy are both very popular in the software testing. The assumptions concerning assumed program reliability growth model, including the formula (7), have been made according to some literature facts, e.g. [1,2,3,4]. The formula (7) can be obtained on the basis of some so-called software reliability models, e.g. those proposed by Shooman and Jelinski-Moranda [3].

The knowledge of the mean value of the predicted number of errors encountered during the program testing process is very useful from the practical point of view. In particular, it makes possible to reasonably estimate both the duration and the cost of the program testing process. These estimations can be very useful for the planning of the program testing process. The mean value of the number of errors encountered during the program testing can be treated as a measure of the increase of the program reliability that was caused by the testing. An example of such a use of the mean value of the number of errors encountered during the testing process to determine the value of both the program testing cost and the program reliability coefficient after testing can be found in [6].

## 5. References


[1] A. Csenki, Bayes predictive analysis of a fundamental software reliability model. *IEEE Trans. Software Engrg.* Vol.39, No. 2 (1990) 177–183.

[2] J.D. Musa, A. Iannino, K. Okumoto., Software reliability. Measurement, prediction, application. McGraw-Hill, Inc. (1987).

[3] T.A. Thayer, M. Lipov, E.C. Nelson, Software reliability. *North-Holland Publishing Company. Amsterdam* (1978).

[4] M. Trechtenberg, A general theory of software reliability modeling. *IEEE Trans. Software Engrg.* Vol.39, No. 1 (1990) 92–96.

[5] K. Worwa, Estimation of the program testing strategy. Part 1 – The same errors can be encountered. *Cybernetics Research and Development*, R. 18, Z. 3-4 (1995) 155–173.

[6] K. Worwa, Estimation of the program testing strategy. Part 2 – The same errors cannot be encountered. *Cybernetics Research and Development*, R. 18, Z. 3-4 (1995) 175–188.

[7] F. Zahedi, N. Ashrafi, Software reliability based on structure, utility, price and cost. *IEEE Trans. on Software Engrg.* Vol.17, No. 4 (1991) 345–356.

[8] K. Worwa, G. Konopacki, Bicriterial Optimization of the Program Testing Strategy.Biuletyn WAT, No. 4 (1994).