Change-Point Modeling for Software Reliability Measurement Depending on Two-Dimensional Software Reliability Growth Factors

Shinji Inoue¹ and Shigeru Yamada¹

¹ Department of Social Management Engineering, Graduate School of Engineering, Tottori University, Japan
E-mail: {ino, yamada}@sse.tottori-u.ac.jp

accepted February 21, 2012

Summary
It is known that the trend of the software reliability growth process changes during testing or operation phases in an actual environment due to a change of specification, software upgrading, a change of fault-target, and so forth. This paper discusses an extended two-dimensional software reliability growth modeling, in which the software reliability growth process is assumed to depend on the simultaneous testing-time and testing-effort factors, with the effect of the change and the uncertainty of the magnitude of the effect on software reliability growth process for more plausible two-dimensional software reliability assessment. Finally, we show examples of the applications of software reliability assessment based on the proposed model by using actual data.

Key words: Software reliability assessment, change-point, software reliability growth model, testing-environmental function, counting process, two-dimensional software reliability measurement.

1. INTRODUCTION

Quantitative software reliability assessment is one of the important activities for ensuring software quality/reliability of the final product. Software reliability growth model [1-3] is a mathematical tool for measuring software reliability by describing the software failure-occurrence phenomenon. Ordinarily, the SRGM is developed by treating the software failure-occurrence time- or the fault-detection time-intervals as random variables.
Needless to say, it is preferable that the SRGM is developed under feasible modeling assumptions, which reflect actual software failure-occurrence phenomena. Most of SRGMs proposed so far were developed under the following assumptions: (1) the software reliability growth process depends only on the testing-time essentially, (2) the stochastic characteristics for the software failure-occurrence or the software fault-detection phenomenon does not change throughout the testing-phase. In an actual testing-phase, it is not necessarily that the common assumptions mentioned above is always appropriate. That means, it is natural to consider the software reliability growth process observed in the actual testing-phase depends on not only the testing-time but also the other software reliability growth factors, such as the testing-coverage, the testing-effort expenditures, the number of executed test-cases [4-6]. And the stochastic characteristics for the software failure-occurrence or the software fault-detection phenomenon changes due to a change of specification, software upgrading, a change of fault-target, and so forth. Especially, testing-time when the stochastic characteristic for the software failure-occurrence or the software fault-detection phenomenon notably changes is called change-point [7-12].

Under the background mentioned above, Inoue et al. proposed a two-dimensional SRGM with the effect of a change of the software reliability growth factors [13]. This paper discusses more plausible two-dimensional software reliability growth modeling with the effect of the change on the software reliability growth process and the uncertainty of the magnitude of the effect. Further, we show numerical examples of our two-dimensional change-point by using actual data.

2. ONE-DIMENSIONAL MODELING FRAMEWORK

First of all, we discuss one-dimensional software reliability growth modeling framework. Basically, it is known that almost one-dimensional SRGMs in which the total number of detectable faults is finite are developed under the following basic assumptions [14]:

1. Whenever a software failure is observed, the fault which caused it will be detected immediately and no new faults are introduced in the fault-removing activities.
2. Each software failure occurs at independently and identically distributed random times with the probability distribution, \( F(s,u) = \Pr\{S \leq s, U \leq u\} \), where \( \Pr\{A\} \) represents the probability of event \( A \).
3. The initial number of faults in the software, \( N_0(>0) \), is a random variable, and is finite.

Now, let \( \{N(t), t \geq 0\} \) denote a counting process representing the total number of faults detected up to testing-time. From the basic assumptions above, the probability that \( m \) faults are detected up to testing-time \( t \) is derived as:

\[
\Pr\{N(t) = m\} = \sum_{n} \binom{n}{m} \{F(t)\}^n \{1 - F(t)\}^{n-m} \Pr\{N_0 = n\} \quad (m = 0,1,2,\cdots).
\]

As a well-known result, if we assume that the initial fault content, \( N_0 \), follows a Poisson distribution with mean \( \omega \), the counting process \( \{N(t), t \geq 0\} \) can be rewritten as:

\[
\Pr\{N(t) = m\} = \sum_{n} \binom{n}{m} \{oF(t)\}^n \frac{\omega^n}{m!} \exp[-\omega] = \exp[-\omega] \frac{\{oF(t)\}^n}{m!} \sum_{n} \frac{\omega^n (1 - F(t))^{n-m}}{(n-m)!} (n = 0,1,2,\cdots).
\]
The above equation is equivalent to a non-homogeneous Poisson process (NHPP) with mean value function $\omega F(t)$. We need to give a suitable software failure-occurrence times distribution to develop a specific NHPP model. For an example, we obtain an exponential SRGM \[2\], $E[N(t)] = \omega(1 - e^{-\lambda t})$, which is one of the representative NHPP models, if we assume that the software failure-occurrence times distribution follows an exponential distribution with parameter $\lambda$.

3. TWO-DIMENSIONAL MODELING FRAMEWORK

One dimensional software reliability growth modeling describes a software reliability growth process depending on only the testing time. However, it is natural to consider that the software reliability growth process observed in an actual testing-phase depends on not only the testing time but also the other software reliability growth factors, such as the testing-coverage, the testing-effort expenditure, the number of executed test-cases. Under the background, two-dimensional software reliability growth models have been proposed \[4-6\]. The two-dimensional software reliability growth model in which the number of detectable faults in a software system is assumed to be finite can be developed by the following modeling assumptions:

(A1) Whenever a software failure is observed, the fault is detected immediately, and no new faults are introduced in the fault-detection procedures.

(A2) Each software failure occurs at independently and identically distributed random times with the bivariate probability distribution function $F(s,u) = \Pr\{S \leq s, U \leq u\}$, where $S$ and $U$ are the random variables representing the testing-time and cumulative testing-effort expenditure, respectively. And $\Pr\{A\}$ represents the probability of event $A$.

(A3) The initial number of faults in the software system, $N_0(>0)$, is a random variable, and is finite.

Now we define the two-dimensional stochastic process \(\{N(s,u), s \geq 0, u \geq 0\}\) representing the number of faults detected in the two-dimensional space \([0,s] \times [0,u]\). Then, we have a two-dimensional probability mass function that $m$ faults are detected in the two-dimensional space \([0,s] \times [0,u]\) as:

\[
\Pr\{N(s,u) = m\} = \sum_{n=0}^{\infty} \binom{n}{m} \left(1 - F(s,u)\right)^n \times \Pr\{N_0 = n\} \times \Pr\{A\}
\]

\(m = 0, 1, 2, \cdots\),

from the modeling assumptions. From Eq. (1), we can say that the stochastic behavior of the software failure-occurrence or the software fault-detection phenomenon can be characterized by assuming the probability mass function for the initial number of faults in the software system $N_0$.

In this paper, we assume that the initial fault content follows a Poisson distribution with parameter $\omega$. Then we have:

\[
\Pr\{N(s,u) = m\} = \frac{\omega^m F(s,u)^m}{m!} \exp\{-\omega F(s,u)\},
\]

from Eq. (1). Eq. (2) is essentially equivalent to a two-dimensional nonhomogeneous Poisson process (abbreviated as a two-dimensional NHPP) \[5,6\] with mean value function. The two-dimensional mean value function can be developed by assuming suitable bivariate software failure-occurrence time distribution in Eq. (2).
4. TWO-DIMENSIONAL CHANGE-POINT MODELING

We discuss a two-dimensional change-point modeling for software reliability assessment. In our two-dimensional change-point modeling, we consider the uncertainty of the effect of change-point on software reliability growth process. Especially in this paper, we assume that change-point is observed once \((\tau_s, \tau_u) (0 < \tau_s < S, 0 < \tau_u < U)\) throughout the testing-phase. Figure 1 shows the stochastic quantities for our two-dimensional change-point modeling.

\[
\begin{align*}
&\eta_s = \eta_u = \eta, \\
&M_i = \eta_s X_i, \quad C_i = \eta_s Z_i, \\
&K_i = \eta_s Y_i, \quad D_i = \eta_u W_i.
\end{align*}
\]

And we assume the following relationship [15] between the software failure-occurrence time or time-interval before change-point and those after change-point:

\[
\begin{align*}
&M_i = \eta_s X_i, \quad C_i = \eta_s Z_i, \\
&K_i = \eta_s Y_i, \quad D_i = \eta_u W_i.
\end{align*}
\]

where \(\eta_s\) and \(\eta_u\) are the testing-environmental function and we assume \(\eta_s = \eta_u = \eta\) for simplicity. Regarding the testing-environmental coefficient, we might be better to treat these coefficient as a random variable because the magnitude of the effect of change-point does not take constant value rather than that the magnitude is stochastically behaved. Then, we assume

\[
\eta = \int_0^\infty \int_0^\tau_s \int_{\tau_s}^{\tau_u} \eta f(\eta) d\eta d\tau_u d\tau_s.
\]

Figure 1. Stochastic quantities in our two-dimensional change-point modeling

\[
\begin{align*}
&M_i = \eta_s X_i, \quad C_i = \eta_s Z_i, \\
&K_i = \eta_s Y_i, \quad D_i = \eta_u W_i.
\end{align*}
\]
where \( f(\eta) \) is the probability density function for the testing-environment coefficient. In this paper, we assume a gamma distribution with the shape parameter:

\[
f(\eta) = \frac{\theta^{\gamma} \eta^{\gamma-1} \exp[-\theta \eta]}{\Gamma(\gamma)} \quad (\theta > 0, \gamma > 0),
\]

(5)

where \( \Gamma(\gamma) \) is the gamma function:

\[
\Gamma(\gamma) = \int_0^{\infty} e^{-x} x^{\gamma-1} \, dx.
\]

Now we suppose that \( n(\geq 1) \) faults have been detected up to change-point and their fault data from the test-beginning have been observed as \((x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n)\), where \( x_0 = 0, y_0 = 0, 0 < x_1 < x_2 \ldots < x_n \leq \tau_s \) and \( 0 < y_1 < y_2 \ldots < y_n \leq \tau_u \), where \( y_i \) (\( i = 0, 1, 2, \ldots, n \)) represents the testing-effort factor expended by the testing time \( x_i \). Let us suppose that the number of faults detected in the testing-territory before change-point, \([0, s] \times [0, u] \) \((s \leq \tau_s, u \leq \tau_u)\) follows the two-dimensional NHPP in Eq. (2) with mean value function. Then, the two-dimensional probability distribution function for the \((M_1, K_1)\), can be derived by its cofunction. The cofunction is derived as the following conditional probability:

\[
\Pr\{M_1 > s, K_1 > u\} = \frac{\Pr\{X_{n+1} > \tau_s - x_n + s \theta / \gamma, Y_{n+1} > \tau_u - y_n + u \theta / \gamma\}}{\Pr\{X_{n+1} > \tau_s - x_n, Y_{n+1} > \tau_u - y_n\}}.
\]

(6)

From Eq. (6), the expected number of faults detected after change-point, \( \Lambda_A(s, u) \), is derived as

\[
\Lambda_A(s, u) = -\log[\Pr\{M_1 > s - \tau_s, K_1 > u - \tau_u\}]
\]

\[
= \Lambda_B(\tau_s + \frac{(s - \tau_s)\theta}{\gamma}, \tau_u + \frac{(u - \tau_u)\theta}{\gamma}) - \Lambda_B(\tau_s, \tau_u + \frac{(u - \tau_u)\theta}{\gamma}) - \Lambda_B(\tau_s + \frac{(s - \tau_s)\theta}{\gamma}, \tau_u) + \Lambda_B(\tau_s, \tau_u).
\]

(7)

Accordingly, we have a two-dimensional mean value function with effect of change-point as

\[
\Lambda(s, u) = \begin{cases} 
\Lambda(s, u) = \Lambda_B(s, u) & (for \quad 0 \leq s \leq \tau_s, 0 \leq u \leq \tau_u) \\
\Lambda(s, u) = \Lambda_B(\tau_s, \tau_u) + \Lambda_A(s, u) & \\
= \Lambda_B(\tau_s + \frac{(s - \tau_s)\theta}{\gamma}, \tau_u + \frac{(u - \tau_u)\theta}{\gamma}) - \Lambda_B(\tau_s, \tau_u + \frac{(u - \tau_u)\theta}{\gamma}) - \Lambda_B(\tau_s + \frac{(s - \tau_s)\theta}{\gamma}, \tau_u) + 2\Lambda_B(\tau_s, \tau_u) & (for \quad s > \tau_s, u > \tau_u). 
\end{cases}
\]

(8)
Eq. (8) implies that our two-dimensional change-point model for software reliability assessment can be developed by assuming the two-dimensional mean value function before change-point, \( \Lambda_s(s, u) \).

5. SOFTWARE RELIABILITY ASSESSMENT MEASURES

Software reliability assessment measures are derived by stochastic properties of the SRGM, and play an important role in quantitative software reliability assessment based on the SRGM. In this paper, we discuss an expected number of remaining faults and a software reliability function. First of all, the expected number of remaining faults is derived as:

\[
M(s, u) \equiv E[\bar{N}(s, u)] = \Lambda(\infty, \infty) - \Lambda(s, u). \tag{9}
\]

An operational software reliability [4] is defined as the probability that a software failure does not occur in the time-interval \( (s_e, s_e + \eta, u) \) \( (s_e \geq 0, \eta \geq 0) \) given that the testing has been going up to testing-time \( s_e \) and the testing-effort has been expended up to \( u_e \) by testing-time \( s_e \). From the basic notion of this measure and the stochastic properties in Eq. (1), we can generally formulate the operational software reliability as

\[
R(\eta | s_e, u_e) = \sum_k \Pr\{ N(s_e + \eta, u_e) = k \} \cdot \Pr\{ N(s_e, u_e) = k \} \tag{10}
\]

\[
= \sum_k \left[ \frac{(F(s_e, u_e))^k}{k!} \cdot (1 - F(s_e + \eta, u_e))^n \cdot \sum_n \left( \frac{n!}{k!} \right) \right].
\]

Assuming that \( N_0 \) follows a Poisson distribution with parameter \( \omega \), we can derive the operational software reliability function as:

\[
R(\eta | s_e, u_e) = \exp[-\{ \Lambda(s_e + \eta, u_e) - \Lambda(s_e, u_e) \}]. \tag{11}
\]

6. PARAMETER ESTIMATION

Parameters of our two-dimensional SRGM with change-point for software reliability assessment can be estimated by the method of maximum-likelihood. Suppose that we have observed \( K \) data pairs \((s_k, u_k, y_k)\) \((k = 0,1,2,\ldots,K)\) with respect to the number of fault \( y_k \), which have been detected in the testing-territory \([0, \tau_s]\times[0, \tau_u] + (\tau_s, s_k] \times (\tau_u, u_k] \). The logarithmic likelihood function, \( \ln L(\Xi | \tau) \), for the two-dimensional stochastic process \{\(N(s, u), s \geq 0, u \geq 0\) given the change-point \( \tau \) can be derived as:

\[
\ln L(\Xi | \tau) = \sum_{k=1}^{K} (y_k - y_{k-1}) \cdot \ln[\Lambda(s_k, u_k; \Xi | \tau) - \Lambda(s_{k-1}, u_{k-1}; \Xi | \tau)] - \Lambda(s_K, u_K; \Xi | \tau) - \sum_{k=1}^{K} \ln[(y_k - y_{k-1})!], \tag{12}
\]

where \( \Xi \) represents the set of the parameter in \( \Lambda(s, u) \). Then, we obtain the following equation:
\[
\frac{\partial \ln L(\Xi | \tau)}{\partial \Xi} = 0. \tag{13}
\]

The maximum-likelihood estimates are obtained by solving Eq. (13) numerically in terms of each parameter.

7. NUMERICAL EXAMPLES

We show numerical examples of two-dimensional change-point model and its application to an optimal software release problem. In this paper, we use actual data consisted of 19 data pairs: \((s_k, u_k, y_k) (k = 0, 1, 2, \ldots, 19; t_{19} = 19 \text{ weeks}, s_{19} = 47.65 \text{ (CPU hours)}, y_{19} = 328)\) [14]. And we assume that the software failure-occurrence time distribution before change-point follows the following bivariate probability distribution function [15]:

\[
F(s, u) = (1 - e^{-as})(1 - e^{-bu})(1 + ze^{-as-bu})
\]

\(a > 0, b > 0, -1 \leq z \leq 1\). \tag{14}

We set \(\tau = \{r_s, r_u\} = \{6, 12.89\}\) by following the actual behavior of the software failure-occurrence phenomenon. Then, we estimate the parameters as \(\hat{\omega}, \hat{a}, \hat{b}, \hat{z}, \hat{\theta}\), and \(\hat{\gamma}\), which are the estimates of \(\omega, a, b, z, \theta, \) and \(\gamma\) by the method of maximum likelihood.

Figure 2 shows the estimated our two-dimensional mean value function in Eq. (8), in which the testing-environment coefficient follows the gamma distribution and \(r_s = 6\) and \(r_u = 12.89\). In Figure 2, \(\hat{\omega} = 700.0, \hat{a} = 0.061, \hat{b} = 0.058, \hat{z} = 0.70, \hat{\theta} = 2.069 \times 10^{-9}, \) and \(\hat{\gamma} = 1.034 \times 10^{-9}\). In Figure 2, the dotted line represents the actual behavior of the cumulative number of detected faults and the curved surface the estimated behavior. We see that the expected number of detected faults is estimated to be zero outside the software failure-occurrence territory, which has been explained in Figure 1. In Figure 2, we see that our change-point model fits well to the actual behavior of the cumulative number of detected faults.

![Figure 2. Estimated our two-dimensional mean value function with change-point](image)
Figure 3 shows the estimated gamma distribution representing the uncertainty of the testing-environment coefficient. Further, Figure 4 shows the estimated time-dependent behavior of the expected number of remaining faults. From Figure 4, we can see the probability density of the testing-coefficient, which follows the gamma distribution. Figure 4 shows the estimated expected number of remaining faults based on our two-dimensional change-point model in Eq. (8). From Figure 4, we can see that the residual fault content decreases along with the testing-time and the testing-effort factors. As one of the examples, we can estimate the expected residual fault content at the termination time of the testing, ˆ M(19,47.65), as around 371.

![Figure 3. Estimated the probability density of the testing-environment coefficient following the gamma distribution](image1)

Finally, Figure 5 shows the estimated operational software reliability based on our two-dimensional change-point model, ˆ R(η |19,47.65). From Figure 5, we can estimate the operational software reliability, ˆ R(0.5 |19,47.65), as about 0.031.

![Figure 4. Estimated expected number of remaining faults based on our two-dimensional change-point model](image2)
8. CONCLUDING REMARKS

This paper discussed two-dimensional software reliability growth modeling with the effect of change-point. Especially in our modeling, we considered the relationship the software failure-occurrence times before change-point and after change-point and the uncertainty of the magnitude of the testing-environmental change at the change-point. Further, this paper developed a specific model based on our modeling approach, and showed numerical examples for software reliability measurement based on our two-dimensional change-point model. In the further studies, we need to develop other two-dimensional change-point models by following our modeling framework discussed in this paper and to investigate the effectiveness of your change-point modeling framework by applying a lot of actual data.

Acknowledgement

This work was supported in part by Grant-in-Aid for Scientific Research (C), Grant No. 22510150 from the Ministry of Education, Culture, Sports, Science, and Technology of Japan.

REFERENCES


