Two-Dimensional Software Reliability Growth Modeling Based on a CES Type Time Function

Yuka Minamino¹, Shinji Inoue², Shigeru Yamada¹

¹Tottori University
Minami 4-101, Koyama, Tottori-shi, Tottori 680-8552, Japan
¹{minamino,yamada}@tottori-u.ac.jp
²Kansai University
2-1-1, Ryozenji, Takatsuki-shi, Osaka 569-1095, Japan
²ino@kansai-u.ac.jp

Abstract: Basically, existing SRGMs (software reliability growth models) assume that the software reliability growth process in the testing-phase depends only on the testing-time. It is suggested that the number of detected faults increases unconditionally if only the testing-period is ensured. That is, most of the existing models are not considered the expenditure process of devoted resources, simultaneously. In this paper, we develop new bivariate NHPP (nonhomogeneous Poisson process) models based on a CES (constant elasticity of substitution) type time function. The CES type function is a generalized function of a Cobb-Douglas function, and has a constraint eased for the elasticity of substitution. Concretely, we assume that the testing-time as a software reliability growth factor is expressed by the testing-time and testing-effort factors based on the CES type function. Finally, we show numerical examples by using actual data-sets, and we check the performance of our proposed models. Additionally, we evaluate the relationship for testing-time and testing-effort factors in terms of economics.

Keywords: Software reliability growth model (SRGM), NHPP (nonhomogeneous Poisson process) model, CES (constant elasticity of substitution) type function, testing-resource

1. INTRODUCTION

Generally, software products are developed through the successive phase of specification, design, coding, and testing. The testing-phase is the most important activity to confirm the final quality/reliability of software products, quantitatively. Software reliability growth models (SRGMs) [3,11] are well-known as a basic technique for the quantitative software reliability assessment. They can describe the software failure-occurrence frequency or the software failure occurrence time-intervals as random variables. Also, they treat the time-dependent behavior of the software failure-occurrence phenomenon as the software reliability growth process. Basically, the existing SRGMs assume that the software reliability growth process in the testing-phase depends only the testing-time. It is suggested that the number of detected faults increases unconditionally if only the testing-period is ensured. That is, most of the existing models are not considered the process of devoted resources, simultaneously. From this background, two-dimensional models have developed by several researchers [2,6-10]. Especially, they have proposed the bivariate models based on a Cobb-Douglas type function to express the relationship for the testing-time and testing-effort simply [2,6,7,9].

The Cobb-Douglas type function is known as a production/utility function in economics. Normally, in consideration of productivity in corporate behavior, elasticity of substitution for the capital input and labor input is assumed as 1. However, it is said that the assumption for elasticity of substitution is a severe constraint in economics, practically. Furthermore, the Cobb-Douglas type function is not often applied as an economic production function in the previous software reliability growth modelling studies. Therefore, we need to apply a production function in which the constraint for the elasticity of substitution is eased. Also, we need to try evaluating the relationship for the testing-time and testing-effort factors in terms of economics.

We discuss two-dimensional SRGMs based on a CES (constant elasticity of substitution) type function in this paper. First, we categorize the testing-time in the existing models into the testing-time and testing-effort factors which are related to the software reliability growth. Next, we express the testing-time as a software reliability growth factor based on a CES type function by using them. Then, we treat the testing-time as the CES type function. Concretely, we assume that the testing-time factor is calendar time and the testing-effort is CPU time or execution time. After that, we introduce the CES type function into the existing NHPP (nonhomogeneous Poisson process) models [3,11]. We apply exponential, delayed S-shaped, and inflection S-shaped models, in this paper. Finally, we check the performances of the existing models and our proposed models by using actual data-sets. Furthermore, we estimate the elasticity of substitution in terms of economics.
II. TYPICAL NHPP MODELS

Most of NHPP models assume that the expected number of detectable faults per unit is proportional to the expected number of remaining faults at time $t$.

$$\frac{dN(t)}{dt} = b(t)[a - H(t)](b(t) > 0, t \geq 0), \quad (1)$$

where $a$ is the initial fault content, and $b(t)$ the fault detection rate. In Eq. (1), $H(t)$ represents the mean value function of NHPP. From Eq. (1), the fault detection rate per unit is given by:

$$b(t) = \frac{\frac{dN(t)}{dt}}{[a - H(t)]}. \quad (2)$$

When we assume some initial conditions, we can obtain the following typical SRGMs. Eqs. (3), (4), and (5) are called as exponential (EXP), delayed S-shaped (DSS), and inflection S-shaped (ISS) SRGMs, respectively.

$$H(t) \equiv m(t) = a(1 - \exp[-bt]). \quad (3)$$

$$H(t) \equiv M(t) = a(1 - (1 + bt)\exp[-bt]). \quad (4)$$

$$H(t) \equiv l(t) = \frac{a(1 - \exp[-bt])}{(1 + c\exp[-bt])}. \quad (5)$$

where $b$ is the fault detection rate, $c = (1 - l)/l$, and $l$ the inflection coefficient ($0 \leq l \leq 1$).

III. CES TYPE TIME FUNCTION

We apply the CES type function for expressing the testing-time as the software reliability growth factor in this paper. It is a generalized function of the Cobb-Douglas type function. The Cobb-Douglas type and CES type functions as a production function are defined as Eqs. (6) and (7), respectively.

$$Y = K^\alpha L^{1-\alpha}, \quad (6)$$

$$Y = (\alpha K^\rho + (1 - \alpha)L^\rho)^\frac{1}{\rho}, \quad (7)$$

where $Y$, $K$, and $L$ represent the total production, the capital input, and the labor input, respectively. In Eqs. (6) and (7), $\alpha$ is the distribution parameter, and $\rho$ the substitution parameter. The CES type function constrains the following functions depending on the substitution parameter $\rho$. When $\rho \rightarrow 1$, it becomes a linear form function. When $\rho \rightarrow 0$, it becomes the Cobb-Douglas type function. When $\rho \rightarrow \infty$, it becomes the Leontief type function. Also, the elasticity of substitution for the CES type function is defined as:

$$e = \frac{1}{\rho}. \quad (8)$$

When we assume that the testing-time $t$ in the existing models is expressed as a reliability growth factor by using Eqs. (6) and (7), the Cobb-Douglas type time function is defined as:

$$t \equiv s^\rho u^{1-\rho}. \quad (9)$$

The CES type time function is defined as:

$$t \equiv (as^\rho + (1 - a)u^{\frac{1}{\rho}})^{\frac{1}{\rho}}. \quad (10)$$

where $s$ is the testing-time factor (calendar time), and $u$ the testing-effort factor (CPU time or execution time).

IV. TWO-DIMENSIONAL MODELS BASED ON TIME FUNCTIONS

Next, we define the bivariate NHPP. Now let $\{N(s, u), s \geq 0, u \geq 0\}$ denote a counting process representing the total number of faults detected up to testing-time $s$ and testing-effort $u$. The probability mass function that $m$ faults are detected up to testing-time $s$ and testing-effort $u$ is derived as:

$$P\{N(s, u) = m\} = \frac{[H(s, u)]^m}{m!}\exp[-H(s, u)]. \quad (11)$$

From Eqs. (9) and (10), we can provide the following two-dimensional SRGMs depending on the Cobb-Douglas type and CES type time functions. We can provide two-dimensional exponential (Cobb-EXP), delayed S-shaped (Cobb-DSS), and inflection S-shaped (Cobb-ISS) SRGMs based on the Cobb-Douglas type time function in Eqs. (12), (13), and (14), respectively [7].

$$m_{E1}(s, u) = a(1 - \exp[-bs^\rho u^{1-\alpha}]). \quad (12)$$

$$m_{D1}(s, u) = a(1 - (1 + bs^\rho u^{1-\alpha})\exp[-bs^\rho u^{1-\alpha}]). \quad (13)$$

$$m_{I1}(s, u) = \frac{a(1 - \exp[-bs^\rho u^{1-\alpha}])}{(1 + c\exp[-bs^\rho u^{1-\alpha}])}. \quad (14)$$

Also, we can provide two-dimensional exponential (CES-EXP), delayed S-shaped (CES-DSS), and inflection S-shaped (CES-ISS) SRGMs based on the CES type time function in Eqs. (15), (16), and (17), respectively.

$$m_{E2}(s, u) = a(1 - \exp[-b(\alpha s^\rho + (1 - \alpha)u^{\frac{1}{\rho}})^{\frac{1}{\rho}}]). \quad (15)$$

$$m_{D2}(s, u) = a(1 - (1 + b(\alpha s^\rho + (1 - \alpha)u^{\frac{1}{\rho}})\exp[-b(\alpha s^\rho + (1 - \alpha)u^{\frac{1}{\rho}})^{\frac{1}{\rho}}]). \quad (16)$$
We estimate the parameters for each model by using the method of maximum-likelihood. Suppose that we have observed \( (s_k, u_k, y_k) \) \( (k = 0, 1, 2, \ldots, K) \), the log-likelihood function is given as:

\[
\ln L(\theta) = \sum_{k=1}^{K} \ln \left[ \frac{1}{\theta} (y_k - y_{k-1}) \ln[H(s_k, u_k; \theta) H(s_{k-1}, u_{k-1}; \theta)] - H(s_k, u_k; \theta) - \sum_{k=1}^{K} \ln[(y_k - y_{k-1})] \right].
\]  

(18)

where \( s_k \) is the measured on the basis of weeks, \( u_k \) the measured CPU time or execution time, and \( y_k \) the total number of faults detected during \( [0, s_k], [0, u_k] \). The estimated parameters are obtained by solving the log-likelihood equation with respect to the parameters as:

\[
\frac{\partial \ln L(\theta)}{\partial \theta} = 0.
\]  

(19)

In Eq. (19), \( \theta \) means a set of the parameters in the models.

VI. RELIABILITY ASSESSMENT MEASURES

We estimate the software reliability function and expected number of remaining faults. The software reliability function is defined as the probability which a software failure does not occur in the time-interval \( (s, s + x) \) \( (s \geq 0, x \geq 0) \) given that the testing or operation is executed by testing-time \( s \). Note that the value of the testing-effort has been attained up to \( u \) by testing-termination time\( s \). Therefore, the software reliability function is given by:

\[
R(x|s, u) = \exp[-\{H(s + x, u) - H(s, u)]
\]  

(20)

The expected number of remaining faults by arbitrary testing-time \( s \) and testing-effort \( u \) is given by:

\[
M(s, u) = E[N(s, u)] = E[N(\infty, \infty) - N(s, u)] = H(\infty, \infty) - H(s, u)
\]  

(21)

VII. NUMERICAL EXAMPLES

We show numerical examples by using the following actual data-sets \([1, 4, 5]\).

DS1: \((s_k, u_k, y_k)\)
\((k = 1, 2, \ldots, 19; s_{19} = 19, u_{19} = 47.65, y_{19} = 328)\)

DS2: \((s_k, u_k, y_k)\)
\((k = 1, 2, \ldots, 21; s_{21} = 21, u_{21} = 8736, y_{21} = 43)\)

DS3: \((s_k, u_k, y_k)\)
\((k = 1, 2, \ldots, 20; s_{20} = 20, u_{20} = 10000, y_{20} = 100)\)

DS4: \((s_k, u_k, y_k)\)
\((k = 1, 2, \ldots, 19; s_{19} = 19, u_{19} = 10272, y_{19} = 120)\)

DS5: \((s_k, u_k, y_k)\)
\((k = 1, 2, \ldots, 12; s_{12} = 12, u_{12} = 5053, y_{12} = 61)\)

DS6: \((s_k, u_k, y_k)\)
\((k = 1, 2, \ldots, 19; s_{19} = 19, u_{19} = 11305, y_{19} = 42)\)

Also, we conduct the goodness-of-fit comparison in terms of MSE (mean squared errors) in Eq. (22).

\[
MSE = \frac{1}{k} \sum_{k=1}^{K} (y_k - \hat{H}(s_k, u_k))^2.
\]  

(22)

where \( k \) is the total number of data pairs \((s_k, u_k, y_k)\).

We focus on the numerical examples for DS2. Figs. 1, 2, and 3 show the estimated expected number of detected faults for two-dimensional exponential, delayed S-shaped, and inflection S-shaped SRGMs based on the CES type time function, respectively. Figs. 4, 5, and 6 show the estimated number of remaining faults for DS2. Figs. 7, 8, and 9 show the estimated software reliability function for DS2.
Table 1 The result of goodness-of-fit comparison.

<table>
<thead>
<tr>
<th>Data-Sets</th>
<th>EXP</th>
<th>Cobb-EXP</th>
<th>CES-EXP</th>
<th>DSS</th>
<th>Cobb-DSS</th>
<th>CES-DSS</th>
<th>ISS</th>
<th>Cobb-ISS</th>
<th>CES-ISS</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS1</td>
<td>221.995</td>
<td>206.239</td>
<td>205.978</td>
<td>209.196</td>
<td>96.651</td>
<td>101.855</td>
<td>96.609</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DS5</td>
<td>27.200</td>
<td>6.405</td>
<td>23.558</td>
<td>10.931</td>
<td>6.600</td>
<td>5.839</td>
<td>2.310</td>
<td>2.505</td>
<td>2.104</td>
</tr>
<tr>
<td>DS6</td>
<td>6.009</td>
<td>2.791</td>
<td>2.636</td>
<td>1.0948</td>
<td>6.281</td>
<td>1.089</td>
<td>0.952</td>
<td>1.064</td>
<td>0.951</td>
</tr>
</tbody>
</table>

Table 2 The estimated elasticity of substitution.

<table>
<thead>
<tr>
<th>Data-Sets</th>
<th>CES-EXP</th>
<th>CES-DSS</th>
<th>CES-ISS</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS1</td>
<td>2.415</td>
<td>0.639</td>
<td>0.076</td>
</tr>
<tr>
<td>DS2</td>
<td>1.026</td>
<td>0.903</td>
<td>1.395</td>
</tr>
<tr>
<td>DS3</td>
<td>0.021</td>
<td>0.424</td>
<td>2.604</td>
</tr>
<tr>
<td>DS4</td>
<td>1.401</td>
<td>5</td>
<td>1.222</td>
</tr>
<tr>
<td>DS5</td>
<td>0.923</td>
<td>1.754</td>
<td>0.816</td>
</tr>
<tr>
<td>DS6</td>
<td>0.956</td>
<td>0.190</td>
<td>0.169</td>
</tr>
</tbody>
</table>

Fig. 3 The behavior of our proposed model (CES-ISS, DS2) 
\((a = 47.99, b = 0.036, \bar{a} = 0.85, \bar{p} = 0.283)\)
Fig. 4 The estimated number of remaining faults (CES-EXP, DS2). $\tilde{M}(21,8736) \approx 59.9$.

Fig. 5 The estimated number of remaining faults (CES-DSS, DS2). $\tilde{M}(21,8736) \approx 14.05$.

Fig. 6 The estimated number of remaining faults (CES-EXP, DS2). $\tilde{R}(1,21,8736) \approx 4.99$.

Fig. 7 The estimated software reliability function (CES-EXP, DS2). $\hat{R}(2.0|21,8736) = 0.053$.

Fig. 8 The estimated software reliability function (CES-DSS, DS2). $\hat{R}(2.0|21,8736) = 0.085$.

Fig. 9 The estimated software reliability function (CES-EXP, DS2). $\hat{R}(2.0|21,8736) = 0.42$. 
From these figures, when the testing-time and the amount of testing-effort expenditures increase, the expected number of detected faults increases. However, even if the testing-time is long, the number of detected faults cannot increase without the testing-effort expenditures. Therefore, software development managers need to determine the testing-period and the amount of testing-effort expenditures, accordingly.

Table 1 shows the result of goodness-of-fit comparison. Bold letters represent the highest goodness-of-fit among each model which is exponential, delayed S-shaped, and inflection S-shaped. On the other hand, underlines represent the highest goodness-of-fit among all models for each data. From these results, we can see that the CES-DSS model has better performance than the DSS and Cobb-DSS models. In particular, the CES-ISS model has better performance than the other models.

Table 2 shows the estimated elasticity of substitution. When we focus on the case of the CES-ISS model, we can see that the testing-time and testing-effort factors in DS2, DS3, and DS4 are easy to substitute.

VIII. Conclusion

We have discussed on the two-dimensional SRGMs based on the CES type time function. Also, we have shown numerical examples of software reliability assessment by using the actual data-sets, and checked the performances of our proposed models in terms of the MSE. From the result of goodness-of-fit comparison for the existing SRGMs, Cobb-Douglas type, and CES type SRGMs, we have shown that our CES type SRGMs capture the software reliability growth curve data well. Furthermore, we have evaluated the elasticity of substitution for the testing-time and testing-effort factors in terms of economics. As future study, we need to check the effectiveness of our proposed models by using more actual data-sets. Furthermore, we need to consider various software reliability growth factors.

Acknowledgment

This work was supported in part by The Telecommunications Advancement Foundation in Japan.

References