doi:10.3772/j.issn.1006-6748.2014.04.007

Stochastic differential equation software reliability growth model with change-point①

Zhang Nan (张 楠)②* **, Cui Gang*, Shu Yanjun*, Liu Hongwei*
(* Deptartment of Computer Science and Technology University of Harbin Institute of Technology, Harbin 150001, P. R. China)
(** Harbin University of Commerce, Harbin 150028, P. R. China)

Abstract

This paper presents software reliability growth models (SRGMs) with change-point based on the stochastic differential equation (SDE). Although SRGMs based on SDE have been developed in a large scale software system, considering the variation of failure distribution in the existing models during testing time is limited. These SDE SRGMs assume that failures have the same distribution. However, in practice, the fault detection rate can be affected by some factors and may be changed at certain point as time proceeds. With respect to this issue, in this paper, SDE SRGMs with change-point are proposed to precisely reflect the variations of the failure distribution. A real data set is used to evaluate the new models. The experimental results show that the proposed models have a fairly accurate prediction capability.

Key words: software reliability, continuous state space, stochastic differential equation (SDE), change-point

0 Introduction

Due to the rapid development of computer technology, computers are being used to control safety-critical and civilian systems. High quality software products are greatly demanded in those application areas. As a result, software reliability becomes one of the main concern issues for software developers and users. Software reliability is defined as the probability of failure-free software operation for a specified period of time in a specified environment[1]. To assess the software reliability, some stochastic models called the software reliability growth models (SRGMs)[2-5] have been developed to describe the software debugging phenomenon. Non-homogeneous Poisson process (NHPP) models, as a class of SRGMs, are extensively used and have progressed successfully. However, NHPP SRGMs are built based on the discrete state space during the failure detection process.

Recently, some SRGMs based on stochastic differential equations (SDEs) have been proposed to consider irregular fluctuation in the rate of failure detection. Yamada et al.[6] at first modeled the behavior of software failure detection processes using an *Itô* type SDE. They asserted that if the size of the software system is large, the number of faults detected during the testing phased becomes large, and the change of the number of faults which are corrected and removed through each debugging becomes sufficiently small compared with the initial fault content at the beginning of the testing phase[6]. For example[7], in the development of a computer operations system, a number of faults are detected and removed during the long testing period, and the system is then released to the market. However, a number of faults are then found by the users, and the software company releases an updated version of the system. Thus, in this case, the software fault detection process is described as a stochastic process with a continuous-state space. Later, Lee, et al.[7] extended Yamada's *Itô* type SDE SRGM. They investigated a delayed S-shaped fault detection rate and an inflection S-shaped fault detection rate instead of the constant. Kapur et al.[8,9] described an SDE-based generalized Erlang model with a logistic error detection function and considered a composite model called generalized SRGM that included three different types of faults. In addition, researchers[10-12] developed SDE SRGMs for open sources software with the assumption that the software failure intensity depends on the time.

① Supported by the International Science & Technology Cooperation Program of China (No. 2010DFA14400), the National Natural Science Foundation of China (No. 60503015) and the National High Technology Research and Development Programme of China (No. 2008AA01A201).
② To whom correspondence should be addressed. E-mail: zn_ftcl@hit.edu.cn
Received on Feb. 28, 2013

SDE SRGMs mentioned above assumed that the rate of failure detection was the same distribution. In fact, this rate strongly depends on many different factors, such as the complex of the failure, the skill of the tester, testing resource allocation, and testing environment, etc. Moreover, these factors may be evenly dispersed throughout the whole testing and debugging processes. Thus, there is a possibility that the failure detection rate may be neither constant nor smooth. Instead, it may be changed at certain moments in time called change-point[13]. In general, a change-point is the time instant when a model's parameter experiences a discontinuity in time, i. e. it is the time at which the parameter changes values[14]. Recently, a number of papers have addressed the problem of change-point in the research field of software reliability[15-17]. Besides, to improve the accuracy of software reliability evaluation and prediction, some researchers incorporated other factors and change-points into SRGMs. For example, Shyur[18] considered the SRGM that incorporated with both imperfect debugging and change-points problem. Zhao, et al.[19] proposed a NHPP SRGM considering the environmental factor and change-point. Kapur, et al.[20] incorporated the testing effort with the change-point concept which was useful in solving the problems of runaway software projects. Lin and Huang[21] incorporated the concept of multiple change-points into Weibull-type testing effort functions.

In this paper, we will propose extended *Itô* type SDE SRGMs with change-point to assess more practice case. The proposed models may provide some contributions as follows:

(1) The proposed models may deal with reliability analysis for a large and complex software system.

(2) The proposed models may deal with the fault detection rate changes by using the basic equations of non-linear type.

(3) The proposed models introduce the nonlinear state dependency of software debugging processes directly into the software reliability assessment model, not as a mean behavior like software reliability growth models based on an NHPP.

(4) The proposed models improve the accuracy of evaluation of software reliability and help developers to measure software reliability.

The rest of this paper is organized as follows. In Section 1, *Itô* type SDE SRGMs with change-point are demonstrated. Section 2 discusses the assessment measures of SRGMs used in this paper. Section 3 presents a numerical example to illustrate new models. Finally, Section 4 draws the conclusion remarks.

1 *Itô* type SDE SRGMs with change-point

We will propose new *Itô* type SDE SRGMs that incorporate S-shaped fault detection rate and the change-point problem. The formulation of the proposed models is based on the following assumptions[9, 17]

(1) The software failure detection process is modeled as a stochastic process with a continuous-state space.

(2) The software is subject to failures caused by the manifestation of remaining faults in the system.

(3) Failures are independent and each failure is caused by a single error.

(4) The failure detection rate is not just a constant or in some case it may be changed at some time moment τ called a change-point.

(5) During the fault correction process, no new fault is introduced into the system and the faults are debugged perfectly.

Based on the assumptions, we have the following differential equation[6]:

$$\frac{\mathrm{d}M(t)}{\mathrm{d}t} = -r(t) \cdot M(t) \tag{1}$$

where $M(t)$ denotes the number of fault remaining at time t and gradually decreases as the testing and debugging process go on, $r(t)$ represents a failure detection rate per unit time. However, the behavior of $r(t)$ is not completely known since it is subject to several random effects. So, $r(t)$ can be written as

$$r(t) = b(t) + \sigma\xi(t) \tag{2}$$

where $\xi(t)$ is a standard Gaussian white noise and σ is a positive constant representing a magnitude of the irregular fluctuations. Hence, Eq. (1) becomes

$$\frac{\mathrm{d}M(t)}{\mathrm{d}t} = -[b(t) + \sigma\xi(t)] \cdot M(t) \tag{3}$$

Further, we extend Eq. (3) to derive the following *Itô* type SDE[6, 22]

$$\mathrm{d}M(t) = -\left[b(t) - \frac{\sigma^2}{2}\right] \cdot M(t)\,\mathrm{d}t - \sigma M(t)\,\mathrm{d}W(t) \tag{4}$$

where $W(t)$ is a one-dimensional Wiener process which is formally defined as an integration of the white noise $\xi(t)$ with respect to time t. By applying *Itô* type formula[6, 22], $M(t)$ is got as

$$M(t) = a\exp\left(-\int_0^t b(x)\,\mathrm{d}x - \sigma W(t)\right) \tag{5}$$

where a is the expected total number of faults. Considering that the Wiener process $W(t)$ is a Gaussian process, the transition probability distribution of $M(t)$ is given as

$$Pr[M(t) \leqslant m \mid M(0) = a]$$

$$= \Phi\left(\frac{\log \frac{m}{a} + \int_0^t b(x)\,dx}{\sigma\sqrt{t}}\right) \quad (6)$$

where $\Phi(\cdot)$ is the standardized normal distribution function and defined as

$$\Phi(x) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{x} \exp\left(-\frac{y^2}{2}\right)dy \quad (7)$$

Yamada and Ohba[23] have proposed the S-shaped model. The S-shaped model can be explained as the learning process. Moreover, Schneidewind[24] pointed that S-shaped model provided the best prediction accuracy for one of the Shuttle's failure data sets. So, $b(x)$ is described as two types of S-shaped models as follows[23]:

(1) the failure detection rate of a delay S-shaped model:

$$b(t) = b\left(1 - \frac{1}{1+bt}\right),\ (b > 0) \quad (8)$$

(2) the failure detection rate of an inflection S-shaped model:

$$b(t) = \frac{b}{1 + c\exp(-bt)},\ (b > 0,\ c > 0) \quad (9)$$

As noted above, the failure distribution is not possible to be kept constant and changed. Hence, considering the problem of change-point in failure detection process is intended to be more in reality. As pointed out by Zou[16], the actual number of change-point should be determined by the real data, and the sequence of failure time is divided into many small pieces with few data points in each piece when using too many change-points. As a result, the overall piecemeal fitting of the sequence appears very well, but the fitting function actually has very little value in predicting the future behavior of the product. Therefore, it is reasonable to use fewer change-points in reality. According to data set in Ref. [25], we consider that there exists only one apparent change-point τ[21]. The failure detection rate has a common statistic with parameter b_1 before τ and parameter b_2 after τ. So, the failure detection rate of S-shaped model with one-change-point is

(1) the detection rate of a delay S-shaped model with one change-point:

$$b(t) = \begin{cases} b_1\left(1 - \frac{1}{1+b_1 t}\right) & 0 \leqslant t \leqslant \tau \\ b_2\left(1 - \frac{1}{1+b_2 t}\right) & \tau < t \end{cases} \quad (10)$$

(2) the detection rate of an inflection S-shaped model with one change-point:

$$b(t) = \begin{cases} \frac{b_1}{1 + c_1\exp(-b_1 t)} & 0 \leqslant t \leqslant \tau \\ \frac{b_2}{1 + c_2\exp(-b_2 t)} & \tau < t \end{cases} \quad (11)$$

Substituting Eq. (10) and Eq. (11) into Eq. (5), we have

$$M(t) = \begin{cases} a(1+b_1 t)\exp(-b_1 t - \sigma_1 W_1(t)) & 0 \leqslant t \leqslant \tau \\ a(1+b_1\tau)\left(\frac{1+b_2 t}{1+b_2\tau}\right)\exp[-b_1\tau - b_2(t-\tau) - \sigma_2 W_2(t)] & \tau < t \end{cases} \quad (12)$$

and

$$M(t) = \begin{cases} a\left(\frac{1+c_1}{1+c_1\exp(-b_1 t)}\right)\exp(-b_1 t - \sigma_1 W_1(t)) & 0 \leqslant t \leqslant \tau \\ a\left(\frac{1+c_1}{1+c_1\exp(-b_1\tau)}\right)\left(\frac{1+c_2\exp(-b_2\tau)}{1+c_2\exp(-b_2 t)}\right)\exp[-b_1\tau - b_2(t-\tau) - \sigma_2 W_2(t)] & \tau < t \end{cases} \quad (13)$$

Let $N(t)$ denote the cumulative number of failures detected by time t, and we get

$$N(t) = \begin{cases} a[1 - (1+b_1 t)\exp(-b_1 t - \sigma_1 W_1(t))] & 0 \leqslant t \leqslant \tau \\ a\left[1 - (1+b_1\tau)\left(\frac{1+b_2 t}{1+b_2\tau}\right)\exp[-b_1\tau - b_2(t-\tau) - \sigma_2 W_2(t)]\right] & \tau < t \end{cases} \quad (14)$$

and

$$N(t) = \begin{cases} a\left[1 - \left(\frac{1+c_1}{1+c_1\exp(-b_1 t)}\right)\exp(-b_1 t - \sigma_1 W_1(t))\right] & 0 \leqslant t \leqslant \tau \\ a\left[1 - \left(\frac{1+c_1}{1+c_1\exp(-b_1\tau)}\right)\left(\frac{1+c_2\exp(-b_2\tau)}{1+c_2\exp(-b_2 t)}\right)\exp(-b_1\tau - b_2(t-\tau) - \sigma_2 W_2(t))\right] & \tau < t \end{cases} \quad (15)$$

2 Software reliability assessment measures

In this section, expressions for software reliability assessment measures are presented. The current number of failures detected in the system is a random variable. Given that there are K pairs of observed data as (t_j, n_j) $(j = 1, 2, \cdots, K;\ 0 < t_1 < t_2 < \cdots t_K)$, where n_j is the cumulative number of faults detected up to testing time t_j. Therefore, its expected value can be useful in software reliability measures. The expected value can be obtained as follows[6, 22]:

(1) the mean number of failures detected of a delay S-shaped SDE SRGM is given as

$$E[N(t)] = \begin{cases} a\left\{1-(1+b_1t)\exp\left[-\left(b_1-\frac{\sigma_1^2}{2}\right)t\right]\right\} & 0 \leqslant t \leqslant \tau \\ a\left\{1-(1+b_1\tau)\left(\frac{1+b_2t}{1+b_2\tau}\right)\exp\left[-\left(b_1-\frac{\sigma_1^2}{2}\right)\tau-\left(b_2-\frac{\sigma_2^2}{2}\right)(t-\tau)\right]\right\} & \tau < t \end{cases} \quad (16)$$

(2) the mean number of failures detected of an inflection S-shaped SDE SRGM is given as

$$E[N(t)] = \begin{cases} a\left\{1-\left(\frac{1+c_1}{1+c_1\exp(-b_1t)}\right)\exp\left[-\left(b_1-\frac{\sigma_1^2}{2}\right)t\right]\right\} & 0 \leqslant t \leqslant \tau \\ a\left\{1-\left(\frac{1+c_1}{1+c_1\exp(-b_1\tau)}\right)\left(\frac{1+c_2\exp(-b_2\tau)}{1+c_2\exp(-b_2t)}\right)\exp\left[-\left(b_1-\frac{\sigma_1^2}{2}\right)\tau-\left(b_2-\frac{\sigma_2^2}{2}\right)(t-\tau)\right]\right\} & \tau < t \end{cases} \quad (17)$$

The instantaneous meantime to between failure ($MTBF$) is the average time between failures in an interval dt and is defined as Ref. [9]

(1) a delay S-shaped model:

$$MTBF(t) = \begin{cases} \left\{a\left(b_1^2t-\frac{\sigma_1^2}{2}-\frac{\sigma_1^2b_1t}{2}\right)\exp\left[-\left(b_1-\frac{\sigma_1^2}{2}\right)t\right]\right\}^{-1} & 0 \leqslant t \leqslant \tau \\ \left\{a\left(b_2^2t-\frac{\sigma_2^2}{2}-\frac{\sigma_2^2b_2t}{2}\right)\left(\frac{1+b_1\tau}{1+b_2\tau}\right)\exp\left[-\left(b_1-\frac{\sigma_1^2}{2}\right)\tau-\left(b_2-\frac{\sigma_2^2}{2}\right)(t-\tau)\right]\right\}^{-1} & \tau < t \end{cases} \quad (18)$$

(2) an inflection S-shaped model:

$$MTBF(t) = \begin{cases} \left\{\frac{a(1+c_1)}{(1+c_1\exp(-b_1t))^2}\left(b_1-\frac{\sigma_1^2}{2}-\frac{\sigma_1^2c_1\exp(-b_1t)}{2}\right)\exp\left[-\left(b_1-\frac{\sigma_1^2}{2}\right)t\right]\right\}^{-1} & 0 \leqslant t \leqslant \tau \\ \left\{\frac{a(1+c_1)}{(1+c_2\exp(-b_2t))^2}\left(\frac{1+c_2\exp(-b_2\tau)}{1+c_1\exp(-b_1\tau)}\right)\left(b_2-\frac{\sigma_2^2}{2}-\frac{\sigma_2^2c_2\exp(-b_2t)}{2}\right)\exp\left[-\left(b_1-\frac{\sigma_1^2}{2}\right)\tau-\left(b_2-\frac{\sigma_2^2}{2}\right)(t-\tau)\right]\right\}^{-1} & \tau < t \end{cases} \quad (19)$$

The cumulative $MTBF$ is the average time between failures from the beginning of the test up to time t and defined as Ref. [9]

(1) the delay S-shaped model:

$$MTBF(t) = \begin{cases} ta\left\{1-(1+b_1t)\exp\left[-\left(b_1-\frac{\sigma_1^2}{2}\right)t\right]\right\}^{-1} & 0 \leqslant t \leqslant \tau \\ ta\left\{1-(1+b_2t)\left(\frac{1+b_2t}{1+b_2\tau}\right)\exp\left[-\left(b_1-\frac{\sigma_1^2}{2}\right)\tau-\left(b_2-\frac{\sigma_2^2}{2}\right)(t-\tau)\right]\right\}^{-1} & \tau < t \end{cases} \quad (20)$$

(2) the inflection S-shaped model:

$$MTBF(t) = \begin{cases} ta\left\{1-\left(\frac{1+c_1}{1+c_1\exp(-b_1t)}\right)\exp\left[-\left(b_1-\frac{\sigma_1^2}{2}\right)t\right]\right\}^{-1} & 0 \leqslant t \leqslant \tau \\ ta\left\{1-\left(\frac{1+c_1}{1+c_1\exp(-b_1\tau)}\right)\left(\frac{1+c_2\exp(-b_2\tau)}{1+c_2\exp(-b_2t)}\right)\exp\left[-\left(b_1-\frac{\sigma_1^2}{2}\right)\tau-\left(b_2-\frac{\sigma_2^2}{2}\right)(t-\tau)\right]\right\}^{-1} & \tau < t \end{cases} \quad (21)$$

3 Numerical and data analysis

In this section, the effectiveness of the proposed model is evaluated by a real data which is collected during the testing of the program designed for monitoring and real-time control[25]. The size of the real-time control system is large, which consists of about 200 modules and each module has on average 1000 lines of computer coding language. The test data is recorded by day. The data set is reported "per day" and the 481 faults were detected during 111 days of the testing phase. In order to find and locate the change-point, Huang et al. [21] applied the Laplace trend test to the failure data. Negative values of the Laplace factor indicate decreasing failure intensity; however, positive values suggest increasing failure intensity. From Ref. [21], we can obtain values of the Laplace which were positive before the 25th day. However, after the 47th day, all values of Laplace became negative. This means that the reliability decreased in the beginning and then increased after the 47th day. That is to say, the reliability increased monotonously after the 47th day for the data set. So, this change-point indeed exists and may be located around the 47days[21].

3.1 Criteria for model comparison

The comparison criteria useed to judge model's performance are described as follows

(1) Mean squared errors (MSE)

The MSE is used to measure the derivation between the predicted values with the actual observations. It is defined as[24]:

$$MSE = \sum_{j=1}^{K} (n_j - \hat{E}[N(t_j)])^2 / K \quad (22)$$

where n_j is the cumulative number of faults detected up to testing time t_j and $\hat{E}[N(t_j)]$ is the estimated cumulative number of faults by testing time t_j obtained from the fitted expected function. The lower MSE indicates less fitting error, thus better goodness-of-fit.

(2) Accuracy of estimation (AE)

The AE can reflect the difference between the estimated numbers of all errors with the actual number of all detected errors[24]. It is defined as

$$AE = \left| \frac{a - \hat{a}}{a} \right| \quad (23)$$

Where a is the actual cumulative number of detected errors after the test, and $\hat{a}$ is the estimated number of all errors.

(3) R-squared value (R-squared)

The R-square can measure how successful the fit is in explaining the variation of the data. The R-square can take on any value between 0 and 1, with a value closer to 1 indicating a better fit, and it can be formulated as follows[24]:

$$R - square = \frac{\sum_{j=1}^{K} \left(\hat{E}[N(t_j)] - \frac{1}{K} \sum_{j=1}^{K} n_j \right)^2}{\sum_{j=1}^{K} \left(n_j - \frac{1}{K} \sum_{j=1}^{K} n_j \right)^2} \quad (24)$$

3.2 Performance analysis

For analysing the performance of the new SDE SRGMs, we compare those with two S-shaped NHPP models and two S-shaped SDE models in terms of goodness-of-fit. Two S-shaped NHPP models are well-known for NHPP SRGMs[23]. Two S-shaped SDE models are extended Yamada's *Itô* type SDE models[7]. We first estimate the parameters of all selected models using the least squares estimation (LSE). LSE minimizes the sum of the squares of the deviations between what is expected and what actually observed. For the real failure data set, the change-point τ indeed exists and is located around the 47 days[21]. The estimated parameters of the proposed model and results of the three comparison criteria: MSE, AE, and R-squared are listed in Table 1.

Table 1 comparison of different SRGMs for the data set

Model	Estimation	MSE	AE	R-square
Equation (16)	$\hat{a} = 479.54$, $\hat{b}_1 = 0.0660$, $\hat{b}_2 = 0.1353$, $\hat{\sigma}_1 = 0.0002$, $\hat{\sigma}_2 = 0.0542$	265.9279	0.0030	0.9939
Equation (17)	$\hat{a} = 484.66$, $\hat{b}_1 = 0.0853$, $\hat{b}_2 = 0.1093$, $\hat{c}_1 = 0.5714$, $\hat{c}_2 = 0.9773$, $\hat{\sigma}_1 = 0.2950$, $\hat{\sigma}_2 = 0.0018$	260.7438	0.0076	0.9879
SDE delay S-shaped[7]	$\hat{a} = 487.83$ $\hat{b} = 0.06643$, $\hat{\sigma} = 0.3924$	325.9189	0.0142	0.9937
SDE inflection S-shaped[7]	$\hat{a} = 483.04$, $\hat{b} = 0.06827$, $\hat{c} = 3.91$, $\hat{\sigma} = 0.3945$	293.4414	0.0042	0.9828
NHPP delay S-shaped[23]	$\hat{a} = 488.12$, $\hat{b} = 0.0663$	340.6937	0.0148	0.9920
NHPP inflection S-shaped[23]	$\hat{a} = 484.55$, $\hat{b} = 0.0668$, $\hat{c} = 3.6479$	295.4234	0.0074	0.9783

From Table1, it can be seen that the values of MSE of proposed models are lower than other the existing SDE models and two NHPP models, which means the proposed models fit best among the models. The failure data set and the fitted curves of the proposed models and other SRGMs are shown in Fig. 1 and Fig. 2. As a comparison, the analyzed results show that the proposed models fit the data set very well. The estimated $MTBF(t)$ for the data sets are plotted in Fig. 3 and Fig. 4, respectively. Both of them show that the software reliability grows as the testing procedures go on. From these figures and tables, we can get a conclusion that the proposed models provide a better goodness-of-fit and prediction than other SRGMs.

4 Conclusions

In this paper, a new *Itô* type of SDE SRGMs incorporated the problems of change-point has been presented. On the one hand, the use of the *Itô* type SDE is

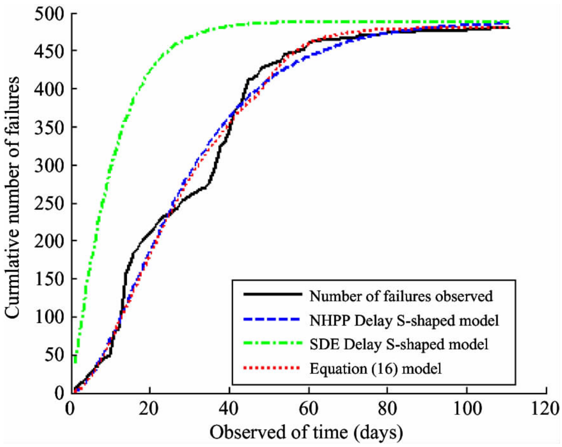

Fig. 1 $\hat{E}[N(t)]$ for Eq. (16) and other models

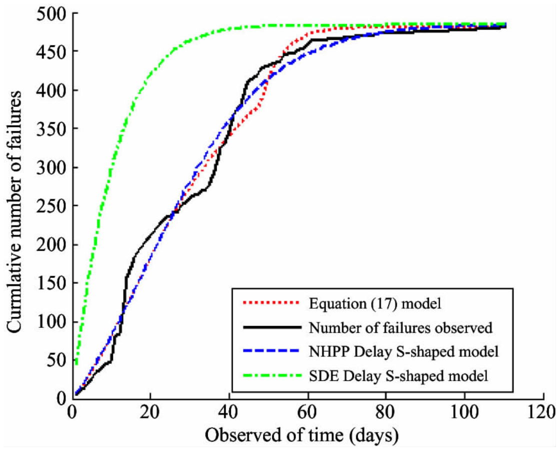

Fig. 2 $\hat{E}[N(t)]$ for Eq. (17) and other models

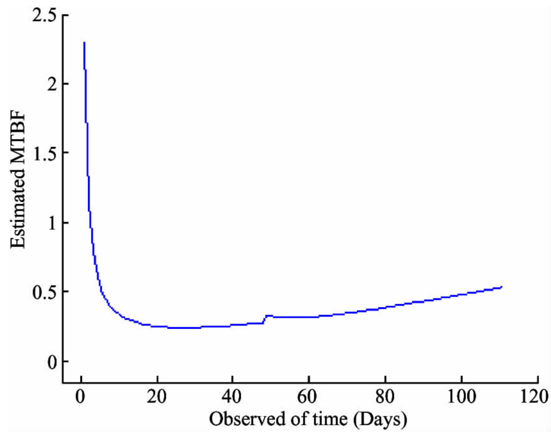

Fig. 3 Instantaneous $MTBF(t)$ for Eq. (18)

to consider the irregular fluctuations of the failure detection process in a large scale software system testing. On the other hand, SRGMs with change-point can accurately reflect the real environment during the fault testing and debugging process. Moreover, we have applied this model on the real data set. The Experimental results show that the accuracy of evaluation of software reliability is improved and has a better goodness-of-fit.

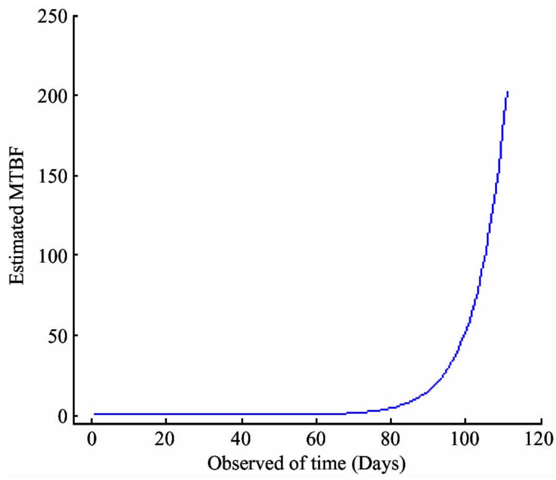

Fig. 4 Cumulative $MTBF(t)$ for Eq. (20)

References

[1] Musa J D. Software Reliability Engineering More Reliable Software Faster and Cheaper. New York: McGrwa-Hill, 2004. 1-680

[2] Musa J D, Annino A, Okumoto K. Software Reliability, Measurement, Prediction and Application. New York: McGram-Hill, 1987. 227-252

[3] Xie M. Software Reliability Modeling. Singapore: World Scientific, 1991. 85-111

[4] Lyu M R. Handbook of Software Reliability Engineering. New York: McGram-Hill, 1996. 71-117

[5] Pham H. Software Reliability. Singapore: Springer-Verlag, 2000. 73-102

[6] Yamada S, Kimura M, Tanaka H, et al. Software reliability measurement and assessment with stochastic differential equations. *IEICE Transactions on Fundamentals*, 1994, E77-A (1): 109-115

[7] Lee C H, Kim Y T, Park D H. S-shaped software reliability growth models derived from stochastic differential equations. *IIE Transactions*, 2004, 36: 1193-1199

[8] Kapur P K, Anand S, Yadavalli V S S. A generalized software growth model using stochastic differential equation. *Communications in Dependability and Quality Management*, 2007, 10(3): 82-96

[9] Kumar A, Kapur P K. SRGMs based on stochastic differential equations. In: Proceedings of the 2nd International Conference on Communication Theory, Reliability, and Quality of Service, Colmar, France, 2009. 1-7

[10] Tamura Y, Yamada S. Software reliability growth model based on stochastic differential equations for open source software. In: Proceedings of International Conference on Mechatronics, Kumamoto, Japan, 2007. 1-6

[11] Tamura Y, Yamada S. Optimization analysis for reliability assessment based on stochastic differential equation modeling for open source software. *International Journal of System Science*, 2009, 40(4): 429-438

[12] Tamura Y, Yamada S. A flexible stochastic differential equation model in distributed development environment. *European Journal of Operational Research*, 2006, 168: 143-152

[13] Zhao M. Change-point problems in software and hardware

reliability. *Communications in Statistics-Theory and Methods*, 1993, 22 (3): 757-768

[14] Huang C Y, Hung T Y. Software reliability analysis and assessment using queuing models with multiple change-points. *Computers and Mathematics with Applications*, 2010, 60: 2015-2030

[15] Chang Y P. Estimation of parameters for non-homogeneous Poisson process software reliability with change-point model. *Communications in Statistics- Simulation and Computation*, 2001, 30: 623-635

[16] Zou F Z. A change-point perspective on the software failure process. *Software Testing, Verification and Reliability*, 2003, 13: 85-93

[17] Zhao J, Wang J D. Testing the existence of change-point in NHPP software reliability models. *Communications in Statistics- Simulation and Computation*, 2007, 36: 607-619

[18] Shyur H J. A stochastic software reliability model with imperfect debugging and change-point. *Journal of Systems and Software*, 2003, 66: 135-141

[19] Zhao J, Liu H W, Cui G, et al. Software reliability growth model with change-point and environmental function. *Journal of Systems and Software*, 2006, 79: 1578-1587

[20] Kapur P K, Singh V B, Anand S, et al. Software reliability growth model with change-point and effort control using a power function of the testing time. *International Journal of Production Research*, 2008, 46 (3): 771-787

[21] Lin C T, Huang C Y. Enhancing and measuring the predictive capabilities of testing-effort dependent software reliability models. *Journal of Systems and Software*, 2008, 81: 1025-1038

[22] ØKsendal B. Stochastic Differential Equations: An Introduction with Applications. Berlin: Springer, 2005. 63-74

[23] Yamada S, Ohba M, Osaki S. S-shaped software reliability growth models and their applications. *IEEE Tractions on Reliability*, 1984, R-33 (4): 289-292

[24] Schneidewind N F. Comparison of reliability and testing models. *IEEE Transactions on Reliability*, 2008, 57 (4): 607-615

[25] Tohma Y, Yamano H, Ohba M, et al. The estimation of parameters of the hyper geometric distribution and its application to the software reliability growth model. *IEEE Transactions on Software Engineering*, 1991, 17 (5): 483-489

Zhang Nan, born in 1980. She received her M.S. degree from Northeast Forestry University in 2008. She also received her B. S. and Ph. D degrees from Harbin Institute of Technology in 2003 and 2013 respectively. Her research interests include software reliability evaluation, software testing and fault tolerance computing.