International Journal of Reliability, Quality and Safety Engineering
2150034 (20 pages)

DOI: 10.1142/S0218539321500340

A Family of Software Reliability Models with Bathtub-Shaped Fault Detection Rate

Maskura Nafreen* and Lance Fiondella†

Department of Electrical and Computer Engineering
University of Massachusetts Dartmouth
North Dartmouth, MA 02747, USA
**mnafreen@umassd.edu*
†lfiondella@umassd.edu

Received 15 November 2020
Revised 17 April 2021
Accepted 18 April 2021
Published

Researchers have proposed several software reliability growth models, many of which possess complex parametric forms. In practice, software reliability growth models should exhibit a balance between predictive accuracy and other statistical measures of goodness of fit, yet past studies have not always performed such balanced assessment. This paper proposes a framework for software reliability growth models possessing a bathtub-shaped fault detection rate and derives stable and efficient expectation conditional maximization algorithms to enable the fitting of these models. The stages of the bathtub are interpreted in the context of the software testing process. The illustrations compare multiple bathtub-shaped and reduced model forms, including classical models with respect to predictive and information theoretic measures. The results indicate that software reliability growth models possessing a bathtub-shaped fault detection rate outperformed classical models on both types of measures. The proposed framework and models may therefore be a practical compromise between model complexity and predictive accuracy.

Keywords: Software reliability; software reliability growth model; bathtub distribution; fault detection rate; expectation conditional maximization algorithm.

Acronyms

SRGM	Software reliability growth model
NHPP	Nonhomogeneous Poisson process
MVF	Mean value function
LL	Log-likelihood function
RLL	Reduced log-likelihood function
MLE	Maximum-likelihood estimation

†Corresponding author.

EM	Expectation-maximization
ECM	Expectation conditional maximization
AIC	Akaike Information Criterion
RL	Relative likelihood
PMSE	Predictive mean square error

Notations

$m(t)$	Mean value function
$\lambda(t)$	Instantaneous failure rate
$b(t)$	Arbitrary bathtub-shaped fault detection function
$F(t)$	Cumulative distribution function
Θ	Vector of model parameters
p	Number of model parameters
n	Number of intervals
T	Vector of failure times
t_i	Number of failures in ith interval
K	Vector of failure counts
k_i	Number of failures in ith interval
$\Theta_i^{(j)}$	ith model parameter in jth iteration

1. Introduction

Software reliability growth models (SRGM)[1,2] offer a quantitative approach to measure the decrease in software failures, especially during the early stages of integration testing when executable code is available. Many SRGM have been proposed and several of the earliest models[3] were relatively simple, while recent models have become progressively more complex. Some individuals, especially practitioners advocate for simple models, but this has not deterred the proliferation of complex models. One legitimate criticism of complex models is that they disregard statistical goodness of fit measures, including predictive accuracy. More concerning, some researchers claim to employ predictive measures of goodness of fit that are, in reality, not predictive. The practice of publishing SRGM without explicitly considering a variety of relevant measures of goodness of fit is not just poor statistics, it is unethical because of the potential for harm to life and property if failures are underestimated when applied to real systems.

Several software reliability researchers have developed a variety of mathematical frameworks to establish relationships between multiple models. Notable examples include the work of Langberg and Singpurwalla[4] who showed how some models can be derived by assigning specific prior distributions in a Bayesian context. Miller[5] showed that several models are special cases of exponential order statistics models. Yamada *et al.*[6] proposed a two-step model fitting procedure, which first fit a curve to testing effort data followed by the mean value function of a nonhomogeneous

Poisson process (NHPP) SRGM to the fault discovery process. Gokhale *et al.*[7] demonstrated that several NHPP models with bounded mean value function are special cases of the enhanced NHPP possessing time-varying test-coverage. Dohi *et al.*[8] introduced an infinite server queueing model to describe software debugging behavior in a framework that accommodates NHPP models possessing a finite or infinite number of faults. Kuo *et al.*[9] proposed a framework to incorporate both testing effort and fault detection rate into SRGM capable of characterizing a wide range of possible fault detection trends. Huang *et al.*[10] presented an NHPP model and derived several existing models through a parametric family of power transformations.

Another common theme is for models to characterize defect detection and correction. For example, Huang and Lin[11] showed how several existing SRGM can be derived by incorporating the ideas of fault dependency and debugging time lag and demonstrated the predictive accuracy of the proposed framework. Xie *et al.*[12] described the failure correction process by delayed failure detection processes with a random or deterministic delay. Wang *et al.*[13] developed a joint likelihood function for failure detection and correction processes, utilizing various correction time distributions, while Wu *et al.*[14] also obtained optimal release times using a modified software cost model from their fault detection and correction model. Kapur *et al.*[15] presented two Generalized Imperfect Non-homogeneous Poisson Process (GINHPP) models to account for imperfect debugging and error generation, and demonstrated that some existing NHPP SRGM are special cases. Okamura and Dohi[16] presented a bivariate modeling framework for SRGM exhibiting time dependent fault detection and correction processes. Zhu and Pham[17] incorporated software fault dependency in NHPP SRGM considering imperfect fault removal with two types of software faults, including type I (independent) and type II (dependent) faults with corresponding debugging processes. Zhu and Pham[18] also developed a multi-release SRGM considering the software faults remaining from the previous release and faults introduced from newly added features.

Other notable works include Inoue and Yamada[19] who developed a generalized discrete software reliability growth model following a binomial process capable of considering the effect of the program size. Song *et al.*[20] proposed a new model considering the Weibull function that relates to the fault detection rate to uncertainty in the operating environment.

This paper presents a family of software reliability models possessing a bathtub-shaped fault detection rate. Several bathtub hazard rates from the hardware reliability literature[21] are interpreted as software fault detection rates. Specifically, the interpretation of the three stages of the bathtub are adapted to the detection of software faults during testing as follows: (i) a burn-in phase characterized by the discovery and correction of superficial faults such as typos and elementary syntax errors; (ii) a requirements verification stage, which exposes more complicated logical errors that require more detailed rework to resolve and (iii) a code comprehension stage characterized by a learning curve,[22,23] where a significant amount of code has been tested, enabling the test team to focus on improving code coverage in order to expose and correct remaining defects. To assess whether the additional complexity introduced by a bathtub-shaped fault detection rate is justified, information theoretic and predictive measures of goodness of fit are computed. This analysis also considers reduced forms of the bathtub model, including classical SRGM.

This paper extends earlier work by Fiondella and Gokhale[24] who presented a single software reliability growth model possessing a bathtub-shaped fault detection rate. Our primary contributions include the following:

- A family of software reliability growth models possessing bathtub-shaped fault detection rate, including a visual taxonomy of the mathematical relationships between the models is provided.
- Stable and efficient expectation conditional maximization (ECM) algorithms, which are essential to enable consistent application of these models.

Our results indicate that, for the data sets considered, a software reliability growth model possessing a bathtub-shaped fault detection rate performed best with respect to both information theoretic and predictive measures of goodness of fit. The proposed family of models coupled with efficient ECM algorithms and goodness of fit assessment may therefore be beneficial to the software reliability assessment process.

The remainder of the paper is organized as follows. Section 2 proposes a framework for bathtub-shaped software reliability growth models. Section 3 describes methods to estimate model parameters, while Sec. 4 summarizes methods to assess model goodness of fit. Section 5 compares alternative bathtub and reduced models. Section 6 provides conclusions and future research.

2. Framework for SRGM with Bathtub-Shaped Fault Detection Rate

The nonhomogeneous Poisson process is a stochastic process[25] that counts the number of events that occur by time t. The expected value is characterized by the mean value function (MVF) and denoted $m(t)$. The MVF can take many functional forms. In software reliability, the NHPP counts the number of unique faults detected through testing time t. The MVF of several SRGM can be written in the following general form:

$$m(t) = \omega F(t), \tag{1}$$

where ω is the number of unique faults that would be detected as $t \to \infty$ and $F(t)$ is the cumulative distribution function of a continuous probability distribution, characterizing the software fault detection process.

A family of bathtub-shaped fault detection models possessing the following form is proposed:

$$F(t) = (1 - e^{-b(t)}), \tag{2}$$

where $b(t)$ is an arbitrary fault detection rate possessing a bathtub shape.

2.1. *SRGM with bathtub-shaped fault detection rate*

This section summarizes several bathtub hazard functions from the literature. Since many bathtub distributions simplify to increasing or decreasing trends, we also identify feasible simplifications and their relationships to other well known software reliability growth models, including the Goel–Okumoto[3] and Weibull[26] SRGM. The section concludes with a visual summary of the relationships between the bathtub models and their simplifications. This taxonomy is also used in the illustrations, where it enables explicit comparison of the goodness of fit of bathtub and simpler SRGM in order to objectively assess if the additional complexity is justified.

2.1.1. *Quadratic model*

To model U-shaped or bathtub hazard functions, Bain[27] and Gore *et al.*[28] considered low order polynomial functions. The quadratic hazard function

$$b(t) = \alpha + \beta t + \gamma t^2 \tag{3}$$

is bathtub-shaped when $-2(\alpha\gamma)^{1/2} \le \beta < 0$ and $\alpha, \gamma \ge 0$. Substituting Eq. (3) into Eq. (2) produces the mean value function of the SRGM with bathtub-shaped fault detection rate characterized by the Quadratic Model

$$m(t) = \omega(1 - e^{-\alpha-\beta t-\gamma t^2}). \tag{4}$$

Parameters β, α, and γ, respectively, contribute to the three stages of the bathtub. Specifically, if the coefficient of the linear term β is decreasing, this trend can characterize a decreasing fault detection rate in the earlier stages of testing as simple problems are detected and removed with relative ease. The constant α represents the baseline fault detection rate associated with the second phase of the bathtub. Finally, the coefficient of the quadratic term γ contributes to the third phase, since this final term will eventually dominate the constant and linear terms. In the context of software fault detection, γ can characterize code comprehension as testers increase code coverage[29] and narrow in on remaining sections of untested software resolving logic issues and correcting them to ensure the application conforms to requirements. A low value of γ may indicate that the software is difficult to comprehend or takes longer to achieve a high level of code coverage. In this final stage of testing, faults can no longer elude testers. Setting α and γ to zero reduces the fault detection rate to

$$b(t) = \beta t \tag{5}$$

indicating that the Goel–Okumoto model is a special case of the SRGM with bathtub-shaped fault detection rate characterized by the Quadratic Model.

2.1.2. *Competing risk models*

Hjorth[30] presented a distribution capable of exhibiting increasing, decreasing, constant, and bathtub-shaped rates

$$b(t) = \frac{\alpha}{1+\beta t} + 2\gamma t. \tag{6}$$

The mean value function of the SRGM with bathtub-shaped fault detection rate characterized by Hjorth's competing risk model is

$$m(t) = \omega(1 - e^{-\frac{\alpha}{1+\beta t} - 2\gamma t}) \tag{7}$$

and also contains the Goel–Okumoto model when $\alpha = 0$ and $b = 2\gamma$.

2.1.3. *Modified Weibull (Lai) model*

Lai *et al.*[31] proposed a modified Weibull distribution possessing hazard rate

$$b(t) = a(\alpha + \lambda t)t^{\alpha-1}e^{\lambda t}. \tag{8}$$

The mean value function of the SRGM with bathtub-shaped fault detection rate characterized by Lai's modified Weibull model is

$$m(t) = \omega(1 - e^{-a(\alpha+\lambda t)t^{\alpha-1}e^{\lambda t}}). \tag{9}$$

Substituting $\lambda = 0$ and $a \to \frac{\lambda}{\alpha}$ reduces to the Weibull model, while substituting $\lambda = 0$, $\alpha = 2$, and $\beta = 2a$ produces the Goel–Okumoto model.

2.1.4. *Exponential power and Weibull extension models*

Exponential Power Model. Smith and Bain[32] studied the exponential power model possessing hazard rate

$$b(t) = \beta\alpha(\beta t)^{\alpha-1}e^{(\beta t)^{\alpha}}. \tag{10}$$

The mean value function of the SRGM with bathtub-shaped fault detection rate characterized by the Exponential Power Model is

$$m(t) = \omega(1 - e^{-\beta\alpha(\beta t)^{\alpha-1}e^{(\beta t)^{\alpha}}}). \tag{11}$$

Setting $\alpha = 1$ in Eq. (11) reduces the fault detection rate to

$$b(t) = \beta e^{\beta t}. \tag{12}$$

Weibull Extension (Chen) Model. Chen[33] proposed a distribution with hazard rate

$$b(t) = \lambda\alpha t^{\alpha-1}e^{t^{\alpha}}. \tag{13}$$

The mean value function of the SRGM with bathtub-shaped fault detection rate characterized by the Weibull Extension Model of Chen is

$$m(t) = \omega(1 - e^{-\lambda\alpha t^{\alpha-1}e^{t^\alpha}}). \tag{14}$$

Setting $\lambda = 1$ in Eq. (14) reduces the fault detection rate to

$$b(t) = e^{t^\alpha}\alpha t^{\alpha-1}. \tag{15}$$

Weibull Extension (***Xie***) ***Model.*** Xie *et al.*[34] extended Chen's Weibull Extension Model by incorporating a scale parameter β into Eq. (13) such that

$$b(t) = \lambda\alpha\left(\frac{t}{\beta}\right)^{\alpha-1} e^{(t/\beta)^\alpha}. \tag{16}$$

The mean value function of the SRGM with bathtub-shaped fault detection rate characterized by the Weibull Extension Model of Xie is

$$m(t) = \omega(1 - e^{-\lambda\alpha(\frac{t}{\beta})^{\alpha-1}e^{(t/\beta)^\alpha}}). \tag{17}$$

Setting $\alpha = 1$ in Eq. (16), reduces the fault detection rate to

$$b(t) = \lambda e^{(t/\beta)}. \tag{18}$$

Double Exponential Power Model. Paranjpe and Rajarshi[35] considered the double exponential power model possessing failure rate

$$b(t) = \beta\alpha t^{\alpha-1}e^{\beta t^\alpha}e^{e^{\beta t^\alpha}-1}. \tag{19}$$

The mean value function of the SRGM with bathtub-shaped fault detection rate characterized by the Double Exponential Power Model is

$$m(t) = \omega(1 - e^{-\beta\alpha t^{\alpha-1}e^{\beta t^\alpha}e^{e^{\beta t^\alpha}-1}}). \tag{20}$$

Setting $\alpha = 1$ in Eq. (19) simplifies to

$$b(t) = \beta e^{\beta t + e^{\beta t} - 1}. \tag{21}$$

Weibull Extension (***Lee***) ***Model.*** Lee[36] proposed a three parameter model with hazard rate

$$b(t) = \lambda\gamma t^{\gamma-1}e^{\phi t} \tag{22}$$

which is bathtub-shaped when $\lambda > 0$, $\gamma < 1$, and $\phi > 0$. The mean value function of the SRGM with bathtub-shaped fault detection rate characterized by the Weibull Extension Model of Lee is

$$m(t) = \omega(1 - e^{-\lambda\gamma t^\gamma e^{\phi t}}). \tag{23}$$

Substituting $\phi = 0$, $\lambda \to \frac{\lambda}{\gamma}$ reduces to the Weibull model, while substituting $\phi = 0$, $\gamma = 1$, and $\beta = \lambda$ reduces to the Goel–Okumoto model.

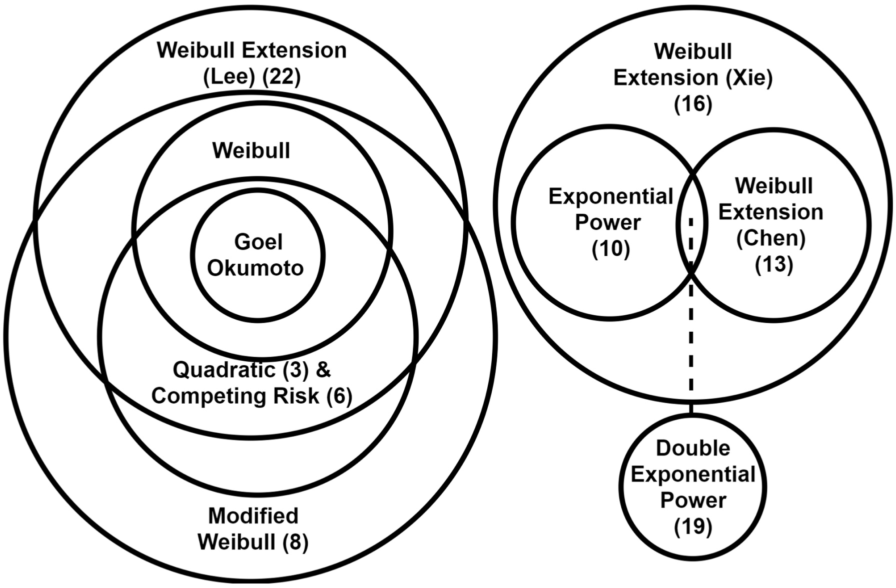

Fig. 1. Framework for a family of Bathtub models.

2.2. *Bathtub-shaped models framework*

Figure 1 depicts the relationships among the SRGM possessing a bathtub-shaped fault detection rate as well as simplifications that correspond to an existing model.

For example the left half of Fig. 1 illustrates how Lee's Weibull Extension Model reduces to the Weibull SRGM and further simplifies to the Goel–Okumoto model. Similarly, the quadratic and competing risk models also contain the Goel–Okumoto model as a special case. Furthermore, Lai's modified Weibull contains the Weibull and Goel–Okumoto models.

The right half of Fig. 1 indicates Xie's Weibull Extension contains the Exponential power model and Chen's Weibull Extension, and that the Double Exponential Power model can only be reduced by deleting the double exponential term $e^{e^{\beta t^{\alpha}}-1}$ because setting $\beta = 0$ or $\alpha = 0$ produces the degenerate expression $b(t) = 0$.

3. Parameter Estimation Methods

This section describes various methods to estimate the parameters of a SRGM with the method of maximum-likelihood estimation, including Newton's method[37] as well as initial parameter estimation with the EM algorithm[38] and ECM[39] algorithms.

3.1. *Maximum-likelihood estimation*

Maximum-likelihood estimation maximizes the likelihood function or joint distribution of the failure data. Typically, the log-likelihood function is maximized. This is possible because the logarithm is a monotonic function, which ensures that the maximum of the log-likelihood function maximizes the likelihood function. Identities of logarithms simplify the product series terms and the resulting set of equations.

Failure count or grouped data consists of a vector of times $T = \langle t_1, t_2, \ldots, t_n\rangle$ at which the intervals ended and failure counts $K = \langle k_1, k_2, \ldots, k_n\rangle$ for these intervals. Failure time models are also possible. However, failure count models are presented here because several historical failure count data sets are well characterized by bathtub models.

The log-likelihood function of a failure count dataset is

$$\mathrm{LL}(\omega, \Theta \,|\, T, K) = \sum_{i=1}^{n} k_i \log\omega + \sum_{i=1}^{n} k_i \log(e^{-b(t_{i-1})} - e^{-b(t_i)}) - \omega(1 - e^{-b(t_n)}) - \sum_{i=1}^{n} \ln k_i!, \tag{24}$$

where Θ is the vector of model parameters contained in $F(t)$.

The maximum likelihood estimates (MLE) that determine the numerical values of the parameters that best fit the data is found by numerically solving the following system of simultaneous equations:

$$\frac{\partial}{\partial\Theta}\mathrm{LL}(\Theta) = \mathbf{0} \tag{25}$$

since models of the form given in Eq. (1) possess a closed form

$$\hat{\omega} = \frac{\sum_{i=1}^{n} k_i}{F(t_n)} \tag{26}$$

which can be substituted into Eq. (24) to reduce the set of simultaneous equations by one.

For example, the log-likelihood of the SRGM with bathtub-shaped fault detection rate characterized by the Quadratic Model is $\mathrm{LL}(\omega, \alpha, \beta, \gamma \,|\, T, K)$

$$= \sum_{i=1}^{n} k_i \log\omega + \sum_{i=1}^{n} k_i \log(e^{-(\alpha+\beta t_{i-1}+\lambda t_{i-1}^2)} - e^{-(\alpha+\beta t_i+\lambda t_i^2)}) - \omega(1 - e^{-(\alpha+\beta t_n+\lambda t_n^2)}) - \sum_{i=1}^{n} \log(k_i)! \tag{27}$$

with

$$\hat{\omega} = \frac{\sum_{i=1}^{n} k_i}{1 - e^{-(\alpha+\beta t_n+\gamma t_n^2)}}, \tag{28}$$

so that the reduced log-likelihood is $\mathrm{RLL}(\alpha, \beta, \gamma \,|\, T, K)$

$$= \sum_{i=1}^{n} k_i \log\left(\frac{\sum_{i=1}^{n} k_i}{1 - e^{-(\alpha+\beta t_n+\gamma t_n^2)}}\right) + \sum_{i=1}^{n} k_i \log(e^{-(\alpha+\beta t_{i-1}+\lambda t_{i-1}^2)} - e^{-(\alpha+\beta t_i+\lambda t_i^2)}) - \sum_{i=1}^{n} k_i - \sum_{i=1}^{n} \log(k_i)!. \tag{29}$$

Traditionally, the Newton-Raphson method[37] has been employed to identify the MLE. However, this numerical method may not converge when the initial estimates chosen as input are not close to the maximum. Without the stability of the Expectation Conditional Maximization algorithm described in this paper it would not be possible to consistently apply these SRGM with bathtub-shaped fault detection rate.

3.2. *Initial parameter estimation*

The EM algorithm[40] provides a systematic calculus-based method to identify initial parameter estimates for some or all parameters of a model. For a mean value function of the form specified in Eq. (1), the observed number of faults is an initial estimate of the number of faults such that $\omega^{(0)} = \sum_{i=1}^{n} k_i$. Initial estimates of the remaining parameters (Θ) can be determined by maximizing the log-likelihood function of the probability density function

$$\Theta^{(0)} = \sum_{i=1}^{n} \frac{\partial}{\partial \Theta} \log[f(t_i; \Theta)] = \mathbf{0} \tag{30}$$

and solving to obtain closed-form expressions.

For example, the initial parameter estimates of the SRGM with bathtub-shaped fault detection rate characterized by the Quadratic Model are

$$\alpha = 0, \tag{31}$$

$$\beta = \sum_{i=1}^{n} \frac{1}{t_i} - 2\gamma \sum_{i=1}^{n} t_i, \tag{32}$$

$$\gamma = \sum_{i=1}^{n} \frac{1}{t_i^2} - \beta \sum_{i=1}^{n} \frac{1}{2t_i}. \tag{33}$$

The parameters β and γ can be estimated in multiple ways. The first is to solve Eqs. (32) and (33) as a pair of simultaneous equations. One alternative approach is to substitute the closed form solution for β on the left-hand side of Eq. (32) into Eq. (33), solving for γ, then substituting the estimate of γ into Eq. (32) to determine β. A second alternative substitutes the closed form expression of γ into Eq. (32) and proceeds in a similar manner.

3.3. *Expectation conditional maximization algorithm*

This section provides a brief overview of the expectation conditional maximization algorithm,[41] which is an extension of the expectation maximization (EM) algorithm that simplifies computation by dividing a single M-step into p conditional-maximization (CM)-steps, where p denotes the number of model parameters. The CM-steps are the partial derivatives of the log-likelihood function $\frac{\partial \mathrm{LL}}{\partial \Theta_i}$ or reduced log-likelihood function $\frac{\partial \mathrm{RLL}}{\partial \Theta_i}$. The ECM algorithm updates one parameter at a time holding all others constant, reducing the maximum likelihood estimation process to p distinct 1-dimensional problems. Thus, in each CM-step, the ECM algorithm searches a single dimension of the parameter space to improve the log-likelihood. Successive CM-steps determine $\Theta_i^{(j)}$, which is the updated value of the ith parameter in the jth iteration.

Without loss of generality, the CM-step which updates the ith parameter in the jth iteration takes

$$\Theta^{jp+i} = \langle \Theta_1^{(j+1)}, \Theta_2^{(j+1)}, \ldots, \Theta_{i-1}^{(j+1)}, \Theta_i^{(j)}, \ldots, \Theta_p^{(j)} \rangle \tag{34}$$

as input, holds all values but $\Theta_i^{(j)}$ constant, and maximizes the partial derivative of the LL or RLL function with respect to Θ_i to produce $\Theta^{jp+(i+1)}$ containing Θ_i^{j+1}. Each CM-step improves the LL or RLL function monotonically. After the CM-step for each parameter is applied, a convergence criterion such as

$$|\mathrm{LL}_j - \mathrm{LL}_{j-1}| < \varepsilon \tag{35}$$

is tested, where $\varepsilon > 0$ is an arbitrarily small constant. If satisfied, the ECM algorithm terminates.

For example, the CM-steps of the SRGM with bathtub-shaped fault detection rate characterized by the Quadratic Model are computed from Eq. (29)

$$\alpha = \sum_{i=1}^{n} k_i \frac{(e^{\alpha} - e^{-(\chi-\alpha)})\left(\frac{e^{2\chi-\alpha}(e^{-\sigma}-e^{\tau})}{(e^{\chi}-1)^2}\right)}{e^{-\tau} - e^{-\sigma}}, \tag{36}$$

$$\beta = \sum_{i=1}^{n} k_i (1 - e^{-\chi}) \frac{\frac{e^{-\sigma}t_i - e^{-\tau}t_{i-1}}{1-e^{-\chi}} + \frac{e^{\chi}(e^{-\tau}+e^{-\sigma})t_n}{(e^{\chi}-1)^2}}{e^{-\tau} - e^{-\sigma}}, \tag{37}$$

$$\gamma = \sum_{i=1}^{n} k_i (1 - e^{-\chi}) \frac{\frac{e^{-\sigma}t_i^2 - e^{-\tau}t_{i-1}^2}{1-e^{-\chi}} + \frac{e^{\chi}(e^{-\sigma}-e^{-\tau})t_n^2}{(e^{\chi}-1)^2}}{e^{-\tau} - e^{-\sigma}}, \tag{38}$$

where, $\chi = \alpha + \beta t_n + \gamma t_n^2$, $\tau = \beta t_{i-1} + \gamma t_{i-1}^2$, and $\sigma = \beta t_i + \gamma t_i^2$. Thus, when the CM-step for α in Eq. (36) is applied all instances of β and γ in χ, τ, and σ are held constant at their most recent estimates and the expression is solved for α. Similarly, the CM-step for β in Eq. (37) holds all instances of α and γ constant and solves for α.

4. Model Assessment

Model assessment evaluates how well a model performs on a data set. Two complementary measures are the Akaike Information Criterion (AIC)[42] and Predictive Mean Square Error (PMSE).[43] The AIC is an information theoretic method to compare the performance of multiple models on a single data set, while PMSE measures the disagreement between a model's predictions and future observations. Ideally, a single "best" model will perform better than all other models under consideration on both measures. In many cases, however, no single model will be most highly recommended by all measures. In such cases, the user must make a subjective decision based on factors such as the amount of data available, stage of testing, and predictive horizon.

4.1. *Akaike information criterion*

The Akaike Information Criterion is an information theoretic measure of a statistical model's goodness-of-fit to a dataset. It is grounded in the concept of entropy, offering a relative measure of the information lost when a given model is applied. The AIC quantifies the tradeoff between a model's characterization of the observed data and the model's complexity. The AIC of model i is a function of the maximized log-likelihood and the number of model parameters (p)

$$\mathrm{AIC}_i = 2p - 2\mathrm{LL}(\hat{\Theta} \mid T). \tag{39}$$

The term $2p$ is a penalty function, which deters a model with an excessive number of parameters that fits the observed data well, but compromises its predictive ability.

Given the AIC of two or more models the probability that the ith model minimizes the information loss or relative likelihood (RL) is

$$\mathrm{RL}_i = e^{\frac{\mathrm{AIC}_{\min} - \mathrm{AIC}_i}{2}}, \tag{40}$$

where $\mathrm{AIC}_{\min}$ is the minimum AIC value among all models considered.

4.2. *Predictive mean square error*

The k-step predictive mean square error of the mean value function measures a model's predictive ability. It is calculated by fitting a model to the data in the first $n - \ell$ intervals and computing computed by

$$\mathrm{PMSE}_\ell = \sum_{i=(n-\ell)+1}^{n} (K_i - \hat{m}(t_i))^2 \tag{41}$$

which is the sum of the squared differences between the cumulative number of faults observed ($K_i = \sum_{j=1}^{i} k_i$) and the cumulative faults predicted by the fitted mean value function ($\hat{m}(t_i)$) for the last ℓ observations not used to fit the model.

5. Illustrations

This section illustrates the application of the ECM algorithm to the SRGM with bathtub-shaped fault detection rate characterized by the Quadratic model. A comparative analysis of bathtub-shaped fault detection rate models and their simplified forms is then performed to assess these models with respect to information theoretic and predictive measures of goodness of fit.

5.1. *Quadratic LL-ECM application*

This first example explains how the ECM algorithm is applied in the context of the SRGM with bathtub-shaped fault detection rate characterized by the Quadratic Model on the PL/I data set,[44] which consists of over 1.3 million lines of code and exhibited 328 faults over nineteen ($n = 19$) weeks of testing. As noted in Sec. 3, the EM and ECM algorithms provide closed form expressions for the initial value of the parameters. The initial estimate of $\alpha^{(0)}$ is 0, while solving Eqs. (32) and (33) produces initial estimates for the remaining parameters such as $\beta^{(0)} = 0.010203$ and $\gamma^{(0)} = 0.000954$. The initial value of the log-likelihood function specified in Eq. (27) is therefore -133.49. The first iteration applies Eq. (36), holding β and γ constant and solving for $\alpha^{(1)} = 0.012757$, which increases the log-likelihood value to -126.16. Successive CM-steps update γ and β with Eqs. (38) and (37), respectively.

Figure 2 shows the CM-steps in the β and γ parameters superimposed on a contour plot of the log-likelihood function. The 90° angle movements illustrate

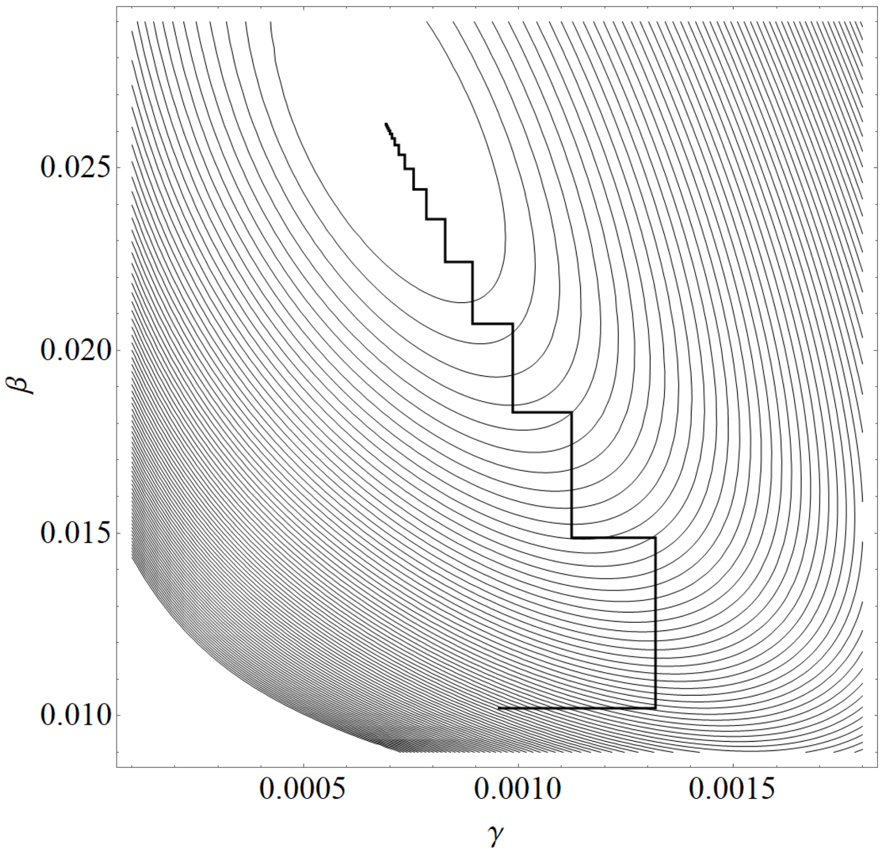

Fig. 2. Iterations of γ and β CM-steps superimposed on contour plot of log-likelihood function of SRGM with bathtub-shaped fault detection rate characterized by Quadratic Model.

how only one parameter is updated at a time. The α parameter is also updated, but not shown here in order to present the process more clearly in two dimensions. Moreover, the value of α used to produce the contour plot is the eventual maximum likelihood estimate such that the convergence of the β and γ parameters shown in Fig. 2 is to the overall maximum likelihood estimate.

Figure 3 shows the monotonic improvements made by the ECM algorithm in each of the 44 iterations until convergence when Eq. (35) is less than $\varepsilon = 10^{-15}$. Substituting the maximum likelihood estimates of α, β and γ into Eq. (28) produces the MLE of the initial number of faults $\hat{\omega} = 349.402$.

Figure 4 shows PL/I data as well as the plot of the mean value function produced by substituting the maximum likelihood estimates in to Eq. (4).

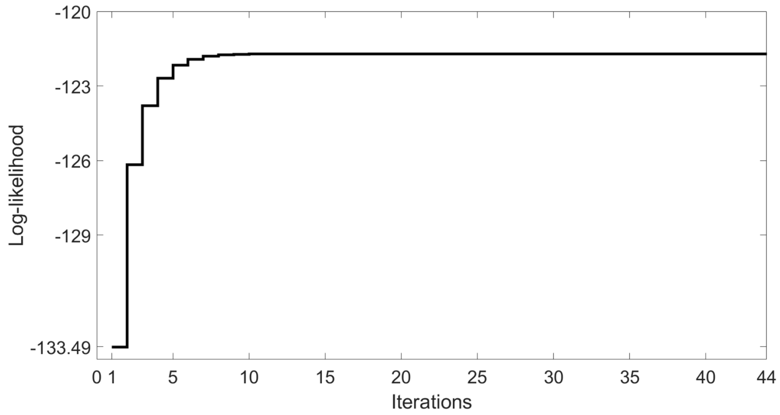

Fig. 3. Improvement of log-likelihood function in iterations of ECM algorithm for SRGM with bathtub-shaped fault detection rate characterized by Quadratic Model.

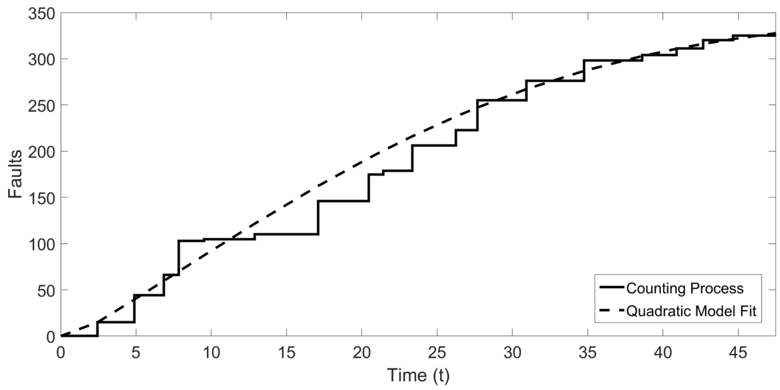

Fig. 4. SRGM with bathtub-shaped fault detection rate characterized by Quadratic Model fit to PL/I data.

5.2. *Model assessment*

Table 1 summarizes the Akaike Information Criterion and corresponding relative likelihood as well as the predictive sum of squares error computed with $k = 2$ of $n = 19$ weeks of data withheld from model fitting for each of the nine bathtub-shaped distributions considered. Values in bold indicate models preferred by specific measures. The AIC and hence the relative likelihood prefer the SRGM with bathtub-shaped fault detection rate characterized by Lee's Weibull extension. Lai's Modified Weibull and the Quadratic Model rank second and third with respect to AIC and RL. The remaining six models are not competitive candidates to characterize this data set, but may perform better on other data sets not considered here. However, PSSE prefers the Double Exponential Power Model, followed by Chen's Weibull extension, Lai's Modified Weibull, and Lee's Weibull Extension. While no model performs best on all measures, Lee's Weibull Extension and Lai's Modified Weibull perform relatively well on both measures.

To assess if the full bathtub models are necessary, Table 2 reports the results of a similar analysis performed on the reduced models identified in Sec. 2, with the relative likelihood computed based on the AIC of Lee's Weibull extension reported in Table 1. Table 2 indicates that among the reduced models, Lai's Modified Weibull and Lee's Weibull Extension reductions (classical Weibull SRGM) perform best, but that their relative likelihood compared the full bathtub of Lee's Weibull Extension is just 0.052. Moreover, the PMSE results of the Exponential Power Model as well as Xie's and Chen's Weibull Extensions predict reasonably well, despite the fact that their relative likelihood is virtually zero.

Table 1. Goodness of fit of bathtub-shaped models.

Function name	AIC	RL	PMSE
Quadratic	251.41	0.208	2439.23
Competing risk	430.43	0.000	2132.20
Modified Weibull (Lai)	250.12	0.418	476.23
Exponential power	480.57	0.000	30023.50
Weibull extension (Chen)	339.22	0.000	129.01
Weibull extension (Xie)	285.69	0.000	5654.02
Double exponential power	300.01	0.000	**9.43**
Weibull extension (Lee)	**248.33**	**1.000**	1369.00

Table 2. Goodness of fit of reduced bathtub-shaped models.

Function name	AIC	RL	PMSE
Quadratic	264.63	0.000	5821.29
Competing risk	264.63	0.000	5821.29
Modified Weibull (Lai)	**254.17**	**0.052**	3013.01
Exponential power	333.23	0.000	**143.76**
Weibull extension (Chen)	289.46	0.000	479.65
Weibull extension (Xie)	326.39	0.000	230.10
Double exponential power	380.90	0.000	46049.00
Weibull extension (Lee)	**254.17**	**0.052**	3013.01

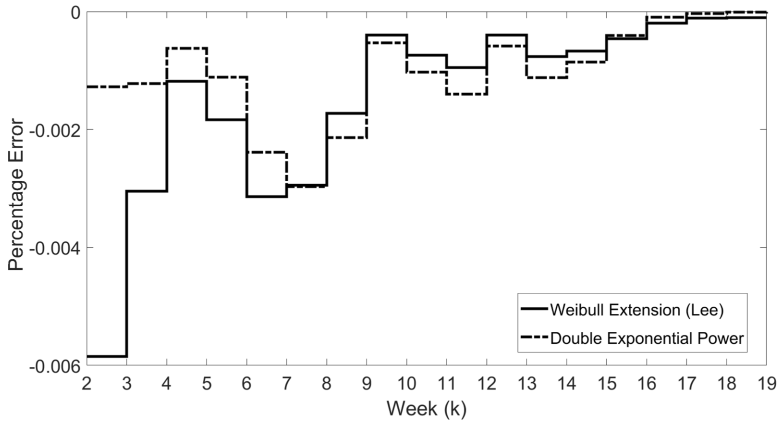

Fig. 5. Predictive error (%) of SRGM with bathtub-shaped fault detection rate.

5.3. *Online prediction assessment*

The results observed so far suggest that the SRGM with bathtub-shaped fault detection rate characterized by Lee's Weibull Extension best characterizes the data, but that its predictive accuracy was not a good as some alternatives for the last two weeks of testing. To assess if the predictive accuracy of the SRGM with bathtub-shaped fault detection rate characterized by Lee's Weibull Extension is significantly poorer, this example compares the model's predictive accuracy with the SRGM with bathtub-shaped fault detection rate characterized by the Double Exponential Power Model.

Figure 5 shows the prediction error of the two models that performed best with respect to AIC and PSSE, respectively, where between two and seventeen weeks of data were used for model fitting and error computed between the model estimates and faults observed two weeks into the future. Figure 5 indicates that the SRGM with bathtub-shaped fault detection rate characterized by Lee's Weibull Extension underpredicts more significantly during the first few weeks of testing, but performs as well as or better than the SRGM with bathtub-shaped fault detection rate characterized by the Double Exponential Power Model in week eight and beyond. This observation suggests that, even though the PMSE was less than ideal toward the end of testing, using Lee's Weibull Extension to perform software reliability tracking throughout the testing process may be reasonable.

6. Conclusions and Future Research

This paper presented a family of software reliability models possessing a bathtub-shaped fault detection rate. Several bathtub hazard rates from the hardware reliability literature were considered. The interpretation of the three stages of the bathtub were adapted to the detection of software faults during testing, namely,

(i) a burn-in phase characterized by the discovery and correction of superficial faults such as typos and elementary syntax errors; (ii) a requirements verification stage that exposes more complicated logical errors that require more detailed rework to resolve and (iii) a code comprehension stage characterized by a learning curve. To assess whether the additional complexity introduced by a bathtub-shaped fault detection rate was justified, information theoretic and predictive measures of goodness of fit were computed. This analysis also considered reduced forms of the bathtub model, including classical SRGM. Our results indicated that SRGM possessing a bathtub-shaped fault detection rate outperformed classical and reduced models on both types of measures. The framework and models may therefore be a reasonable compromise between model complexity and predictive accuracy to track software reliability during testing.

Future research will seek to identify the causes of the bathtub shape such as test procedures and application architecture.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant Number (#1749635). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

1. M. Lyu (ed.), *Handbook of Software Reliability Engineering* (McGraw-Hill, Hightstown, NJ, USA, 1996).
2. S. Yamada, Software reliability modeling, *Fund. Appl.* **1** (2014) 1–38.
3. A. Goel and K. Okumoto, Time-dependent error-detection rate model for software reliability and other performance measures, *IEEE Trans. Reliab.* **R-28** (1979) 206–211, 10.1109/TR.1979.5220566.
4. N. Langberg and N. Singpurwalla, A unification of some software reliability models, *SIAM J. Sci. Stat. Comput.* **6**(3) (1985) 781–790.
5. D. Miller, Exponential order statistic models of software reliability growth, *IEEE Trans. Softw. Eng.* **SE-12** (1986) 12–24, 10.1109/TSE.1986.6312915.
6. S. Yamada, H. Ohtera and H. Narihisa, Software reliability growth models with testing-effort, *IEEE Trans. Reliab. Eng.* **R-35** (1986) 19–23.
7. S. Gokhale, T. Philip, P. Marinos and K. Trivedi, Unification of finite failure non-homogeneous poisson process models through test coverage, in *Proc. Int. Symp. Software Reliability Engineering*, White Plains, NY, USA, 1996, pp. 299–307.
8. T. Dohi, S. Osaki and K. Trivedi, An infinite server queueing approach for describing software reliability growth: Unified modeling and estimation framework, in *Proc. Asia-Pacific Software Engineering Conf.*, Busan, Korea (South), 2004, pp. 110–119.
9. S. Kuo, C. Huang and M. Lyu, Framework for modeling software reliability, using various testing-efforts and fault-detection rates, *IEEE Trans. Reliab.* **50**(3) (2001) 310–320.
10. C. Huang, M. Lyu and S. Kuo, A unified scheme of some nonhomogenous poisson process models for software reliability estimation, *IEEE Trans. Softw. Eng.* **29**(3) (2003) 261–269.

11. C. Huang and C. Lin, Software reliability analysis by considering fault dependency and debugging time lag, *IEEE Trans. Reliab.* **55**(3) (2006) 436–450.
12. M. Xie, Q. Hu, Y. Wu and S. Ng, A study of the modeling and analysis of software fault-detection and fault-correction processes, *Quality Reliab. Eng. Int.* **23**(4) (2007) 459–470.
13. L. Wang, Q. Hu and J. Liu, Software reliability growth modelling and analysis with dual fault detection and correction processes, *IIE Trans.* **48** (2015) 359–370.
14. Y. Wu, Q. Hu, M. Xie and S. Ng, Modeling and analysis of software fault detection and correction process by considering time dependency, *IEEE Trans. Reliab.* **56**(4) (2007) 629–642.
15. P. Kapur, H. Pham, S. Anand and K. Yadav, A unified approach for developing software reliability growth models in the presence of imperfect debugging and error generation, *IEEE Trans. Reliab.* **60** (2011) 331–340.
16. H. Okamura and T. Dohi, A generalized bivariate modeling framework of fault detection and correction processes, in *Proc. Int. Symp. Software Reliability Engineering*, 2017, pp. 35–45.
17. M. Zhu and H. Pham, A two-phase software reliability modeling involving with software fault dependency and imperfect fault removal, *Comput. Languages Syst. Struct.* **53** (2018) 27–42.
18. M. Zhu and H. Pham, A multi-release software reliability modeling for open source software incorporing dependent fault detection process, *Ann. Oper. Res.* **269** (2018) 773–790.
19. S. Inoue and S. Yamada, Generalized discrete software reliability modeling with effect of program size, *IEEE Trans. Syst. Man Cybern. Part A: Syst. Humans* **37** (2007) 170–179, 10.1109/TSMCA.2006.889475.
20. K. Song, I. Chang and H. Pham, A software reliability model with a weibull fault detection rate function subject to operating environments, *Appl. Sci.* **7** (2017) 983.
21. C.-D. Lai and M. Xie, *Stochastic Ageing and Dependence for Reliability* (Springer-Verlag, Berlin, Heidelberg, 2006).
22. R. Hou, S. Kuo and Y. Chang, Efficient allocation of testing resources for software module testing based on the hyper-geometric distribution software reliability growth model, in *Proc. Int. Symp. Software Reliability Engineering*, White Plains, NY, USA, 1996, pp. 289–298.
23. R. Duffey and L. Fiondella, Software, hardware, and procedure reliability by testing and verification: Evidence of learning trends, *IEEE Trans. Human-Mach. Syst.* **44**(3) (2014) 395–405.
24. L. Fiondella and S. Gokhale, Software reliability model with bathtub-shaped fault detection rate, in *Proc. Annual Reliability and Maintainability Symp.*, Lake Buena Vista, FL, USA, 2011, pp. 1–6.
25. S. Ross, *Introduction to Probability Models*, 8th edn., Academic Press, Copyright ©Elsevier Inc. (2003).
26. S. Yamada and S. Osaki, Reliability growth models for hardware and software systems based on nonhomogeneous poisson processes: A survey, *Microelectron. Reliab.* **23**(1) (1983) 91–112.
27. L. Bain, Analysis for the linear failure-rate life-testing distribution, *Technometrics* **16**(4) (1974) 551–559.
28. A. Gore, S. Paranjape, M. Rajarshi and M. Gadgil, Some methods for summarizing survivorship data in nonstandard situations, *Biomet. J.* **28**(5) (1986) 577–586.

29. J. Horgan and S. London, A data flow coverage testing tool for *c*, in *Proc. Symp. Assessment of Quality Software Development Tools*, New Orleans, LA, USA, 1992, pp. 2–10.
30. U. Hjorth, A reliability distribution with increasing, decreasing, constant and bathtub-shaped failure rates, *Technometrics* **22**(1) (1980) 99–107.
31. C. Lai, M. Xie and D. Murthy, A modified weibull distribution, *IEEE Trans. Reliab.* **52** (2003) 33–37.
32. R. Smith and L. Bain, An exponential power life-testing distribution, *Commun. Stat.* **4**(5) (1975) 469–481.
33. Z. Chen, A new two-parameter lifetime distribution with bathtub shape or increasing failure rate function, *Stat. Probab. Lett.* **49**(2) (2000) 155–161.
34. M. Xie, Y. Tang and T. Goh, A modified weibull extension with bathtub-shaped failure rate function, *Reliab. Eng. Syst. Safety* **76**(3) (2002) 279–285.
35. S. Paranjpe and M. Rajarshi, Modelling non-monotonic survivorship data with bathtub distributions, *Ecology* **67**(6) (1986) 1693–1695.
36. L. Lee, Testing adequacy of the weibull and log linear rate models for a poisson process, *Technometrics* **22**(2) (1980) 195–199.
37. R. Burden and J. Faires, *Numerical Analysis*, The Prindle, Weber and Schmidt Series in Mathematics, 4th edn. (PWS-Kent Publishing Company, Boston, 1989).
38. H. Okamura, Y. Watanabe and T. Dohi, An iterative scheme for maximum likelihood estimation in software reliability modeling, in *Proc. Int. Symp. Software Reliability Engineering*, Denver, CO, USA, 2003, pp. 246–256.
39. X. Meng and D. Rubin, Maximum likelihood estimation via the ecm algorithm: A general framework, *Biometrika* **80**(2) (1993) 267–278.
40. A. Dempster, N. Laird and D. Rubin, Maximum likelihood from incomplete data via the em algorithm, *J. Roy. Stat. Soc. Ser. B* **39**(1) (1977) 1–38.
41. V. Nagaraju, L. Fiondella, P. Zeephongsekul, C. Jayasinghe and T. Wandji, Performance optimized expectation conditional maximization algorithms for nonhomogeneous poisson process software reliability models, *IEEE Trans. Reliab.* **66**(3) (2017) 722–734.
42. Y. Sakamoto, M. Ishiguro and G. Kitagawa, *Akaike Information Criterion Statistics* (KTK Scientific Publishers, Tokyo, 1986).
43. S. Narula, Predictive mean square error and stochastic regressor variables, *J. Roy. Stat. Soc. Ser. C* **23**(1) (1974) 11–17.
44. M. Ohba, Software reliability analysis models, *IBM J. Res. Develop.* **28** (1984) 428–443.

About the Author

Maskura Nafreen Maskura Nafreen is a PhD candidate in the Department of Electrical and Computer Engineering (ECE) at the University of Massachusetts Dartmouth (UMassD) in MA, USA. She received her BSc (2018) in Electrical and Electronics Engineering (EEE) from Ahsanullah University of Science and Technology (AUST) in Dhaka, Bangladesh. Since 2018, she has worked as a graduate research assistant in the Dependable Software and Systems Lab at UMassD on projects funded by the NSF, NASA, and the Massachusetts Department of Transportation. Her research interests include statistical data analysis and modeling, autonomous systems, software reliability, and systems performance. She worked for Energypac Engineering Limited in summer 2017, Crossline in fall 2017, Motional (Formerly

Hyundai-Aptiv Joint Venture) in summer 2020, and Aerovironment in summer 2021. Ms. Nafreen served as the Chair of the IEEE AUST student branch, Program Coordinator of IEEE Women in Engineering (WIE) AUST student branch, and as an Ambassador of IEEE Extreme. She is a member of the Society of Reliability Engineers (SRE) and Society of Women Engineers (SWE).

Lance Fiondella received the Ph.D. degree in Computer Science and Engineering from the University of Connecticut, Storrs, CT, USA, in 2012. He is an associate professor of Electrical and Computer Engineering at the University of Massachusetts Dartmouth. His peer-reviewed conference papers have been the recipient of 12 awards, including five as first author and seven with his students as first author. His research has been funded by the Department of Homeland Security, NASA, U.S. Department of Defense, and National Science Foundation, including a CAREER Award. He is an associate editor of the Military Operations Research Journal and the North American Regional Editor of the International Journal of Performability Engineering. Dr. Fiondella served as the vice-chair of IEEE Standard 1633: Recommended Practice on Software Reliability from 2013-15 and a three year term as a Member of the Administrative Committee of the IEEE Reliability Society from 2015-2017. He presently serves as a technical committee chair of the Annual IEEE Symposium on Technologies for Homeland Security.