A Unified Scheme of Some Nonhomogenous Poisson Process Models for Software Reliability Estimation

Chin-Yu Huang, Michael R. Lyu, *Senior Member*, *IEEE*, and Sy-Yen Kuo, *Fellow*, *IEEE*

Abstract—In this paper, we describe how several existing software reliability growth models based on Nonhomogeneous Poisson processes (NHPPs) can be comprehensively derived by applying the concept of weighted arithmetic, weighted geometric, or weighted harmonic mean. Furthermore, based on these three weighted means, we thus propose a more general NHPP model from the quasi arithmetic viewpoint. In addition to the above three means, we formulate a more general transformation that includes a parametric family of power transformations. Under this general framework, we verify the existing NHPP models and derive several new NHPP models. We show that these approaches cover a number of well-known models under different conditions.

Index Terms—Software reliability growth model (SRGM), weighted arithmetic mean, weighted geometric mean, weighted harmonic mean, mean value function (MVF), power transformation, nonhomogeneous Poisson process (NHPP).

✦

---

1 INTRODUCTION

IN recent years, software has permeated industrial equipment and consumer products. Software reliability may be the most important quality attribute of application software since it quantifies software failures during the software development process. As software reliability represents a customer-oriented view of software quality, it relates to practical operations instead of merely the design of programs. Therefore, it is dynamic rather than static. The aim and objective of software reliability engineering are to increase the probability that a completed program will work as intended by the customers. Hence, measuring and computing the reliability of a software system are very important. Measures of software reliability can be used for planning and controlling testing resources during software development. They can also give us confidence about the correctness of the completed software.

Research efforts in software reliability engineering have been conducted over the past three decades and many software reliability growth models (SRGMs) have been proposed [1], [2], [3]. SRGMs can estimate the number of initial faults, the software reliability, the failure intensity, the mean time-interval between failures, etc. There are some, but only a few, model unification schemes in the literature [4]. Langberg and Singpurwalla first showed that several SRGMs can be comprehensively described by adopting a Bayesian point of view and be derived by

assigning specific prior distributions for the parameters of these models [5]. They tried to describe the model unification through three cases and illustrated the proposed theory with several SRGMs, such as the Goel-Okumoto Model, Jelinski-Moranda Model, and Littlewood-Verrall Model, etc. Later, Miller proposed and presented Exponential Order Statistic models (EOS) based on some assumptions [6]. He observed that failure time of a software reliability growth process could be modeled as order statistics of independent nonidentically distributed exponential random variables and this could provide a natural approach for replicated-run software debugging experiments. Several well-known SRGMs, such as the Jelinski-Moranda Model, Goel-Okumoto Model, Musa-Okumoto logarithmic model, and power law models, can be derived as special cases of EOS. Subsequently, Trachtenberg proposed a general theory of software reliability modeling that software failure rates are the product of the average size of remaining faults, apparent fault density, and software workload [7]. He showed that some classical software models, such as the linear model, Rayleigh model, and geometric model, are special cases of the general theory.

From our studies, many existing SRGMs can be unified under a more general formulation. In fact, model unification is an insightful investigation for the study of general models without making many assumptions. Therefore, in this paper, first we will briefly discuss the unification of SRGMs based on Nonhomogeneous Poisson processes (NHPPs) from a new viewpoint. The arithmetic, geometric, and harmonic means are three well-known methods to obtain the average figures of results. Extended from them, we consider three more general means—the weighted arithmetic, weighted geometric, and weighted harmonic means. The objective of this paper is then to show how several existing discrete and continuous NHPP-based SRGMs can be derived by applying the concept of weighted

---

● *C.-Y. Huang and S.-Y. Kuo are with the Department of Electrical Engineering, National Taiwan University, No. 1, Sec. 4, Roosevelt Rd., Taipei, Taiwan.*
E-mail: cy2huang@mail.cbc.gov.tw, sykuo@cc.ee.ntu.edu.tw.
● *M.R. Lyu is with the Computer Science and Engineering Department, The Chinese University of Hong Long, Shatin, Hong Kong.*
E-mail: lyu@cse.cuhk.edu.hk.

arithmetic, weighted geometric, and weighted harmonic means. In addition, we also present a new general NHPP model by incorporating the idea of power transformation in the model unification process. From these approaches, we can not only obtain existing NHPP models but also develop new NHPP models.

The remainder of the paper is organized as follows: Section 2 describes several SRGMs based on nonhomogeneous Poisson processes (NHPPs). In Section 3, the weighted arithmetic, weighted geometric, weighted harmonic, and quasi arithmetic means are introduced. The general forms of the discrete and continuous NHPP models based on the quasi arithmetic mean are proposed and discussed in Sections 4 and 5, respectively. Finally, the conclusions are given in Section 6.

2 NONHOMOGENEOUS POISSON PROCESS MODELS

In this paper, we will show that several existing SRGMs based on NHPPs are special cases in our new general model. While other NHPP models exist, it is our suggestion that most NHPPs-based SRGMs are also special cases of the general model. Let $\{N(t), t \geq 0\}$ denote a counting process representing the cumulative number of faults detected by the time t. An SRGM based on an NHPP with the mean value function (MVF) $m(t)$ can be formulated as [1]

$$P\{N(t) = n\} = \frac{m(t)^n}{n!} e^{-m(t)}, \ n = 0, 1, 2, ..., \qquad (1)$$

where $m(t)$ represents the expected cumulative number of faults detected by the time t. The MVF $m(t)$ is nondecreasing with respect to testing time t under the bounded condition $m(\infty) = a$, where a is the expected total number of faults to be eventually detected. Knowing its value can help us to determine whether the software is ready to be released to the customers and how much more testing resources are required. It can also provide an estimate of the number of failures that will eventually be encountered by the customers. Generally, we can get distinct NHPP models by using different nondecreasing mean value functions. The failure intensity function at testing time t is [2]

$$\lambda(t) = \frac{dm(t)}{dt} = m'(t). \qquad (2)$$

The software reliability, i.e., the probability that no failures occur in $(s, s + t)$ given that the last failure occurred at testing time $s(s \geq 0, t > 0)$, is

$$R(t \mid s) = \exp[-(m(t + s) - m(t))]. \qquad (3)$$

The fault detection rate per fault at testing time t is given by

$$d(t) = \frac{m'(t)}{a - m(t)} = \frac{\lambda(t)}{a - m(t)}. \qquad (4)$$

Equation (4) means the detectability of a fault in the current fault content.

There are several existing well-known NHPP models with different MVFs, as described below.

- **Goel-Okumoto Model**. This model, first proposed by Goel and Okumoto, is one of the most popular NHPP model in the field of software reliability

modeling [1], [2], [3]. It is also called the exponential NHPP model. Considering failure detection as a Nonhomogeneous Poisson process with an exponentially decaying rate function, the mean value function is hypothesized in this model as

$$m(t) = a(1 - \exp[-bt]), \ a > 0, b > 0,$$

where a is the expected total number of faults to be eventually detected and b represents the fault detection rate.

- **Gompertz Growth Curve Model**. Gompertz Growth Curve Model is used in the Fujitsu and Numazu work [3]. Many Japanese computer manufacturers and software houses have adopted this model since it is one of the simplest S-shaped software reliability growth models. Its mean value function is

$$m(t) = ak^{b^t}, \ a > 0, 0 < b < 1, 0 < k < 1,$$

where a is the expected total number of faults to be eventually detected and b and k are parameters whose values are estimated using regression analysis.

- **Logistic Growth Curve Model**. In general, software reliability tends to improve and it can be treated as a growth process during the testing phase. That is, the reliability growth occurs due to fixing faults. Therefore, under some conditions, the models developed to predict economic population growth could also be applied to predict software reliability growth. These models simply fit the cumulative number of detected faults at a given time with a function of known form. Logistic growth curve model is one of them and it has an S-shaped curve [3]. Its mean value function is

$$m(t) = \frac{a}{1 + k \exp[-bt]}, \ a > 0, b > 0, k > 0,$$

where a is the expected total number of faults to be eventually detected and k and b are parameters which can be estimated by fitting the failure data.

    Similar to the analysis given for the Gompertz curve, the point of inflection is $t_{\inf} = \frac{\ln k}{b}$. If $k > 1$, then $t_{\inf} > 0$ and we have an S-shaped results; when $0 < k \leq 1$, then $t_{\inf} \leq 0$ and no S-shaped curve is present. Therefore, the logistic reliability growth curve is S-shaped when $k > 1$ and $b > 0$.

- **Generalized Goel NHPP Model**. In order to describe the situation that software failure intensity increases slightly at the beginning and then begins to decrease, Goel proposed a simple generalization of the Goel-Okumoto model with an additional parameter c [3]. The mean value function is

$$m(t) = a(1 - \exp[-bt^c]), \ a > 0, b > 0, c > 0,$$

where a is the expected total number of faults to be eventually detected and b and c are parameters that reflect the quality of testing.

- **Yamada Delayed S-Shaped Model**. The Yamada Delayed S-Shaped model is a modification of the Nonhomogeneous Poisson process to obtain an S-shaped curve for the cumulative number of failures detected such that the failure rate initially

increases and later (exponentially) decays [1], [3]. It can be thought of as a generalized exponential model with failure rate first increasing and then decreasing. The software error detection process described by such an S-shaped curve can be regarded as a learning process because the testers' skills will gradually improve as time progresses. The mean value function is

$$m(t) = a(1 - (1 + bt)\exp[-bt]), \ a > 0, b > 0,$$

where a and b are is the expected total number of faults to be eventually detected and the fault detection rate, respectively.

- **Inflected S-Shaped Model**. This model solves a technical problem in the Goel-Okumoto model. It was proposed by Ohba and its underlying concept is that the observed software reliability growth becomes S-shaped if faults in a program are mutually dependent, i.e., some faults are not detectable before some others are removed [3]. The mean value function is

$$m(t) = a\frac{1 - \exp[-bt]}{1 + \psi(r)\exp[-bt]}, \ \psi(r) = \frac{1 - r}{r},$$
$$a > 0, b > 0, r > 0.$$

The parameter r is the inflection rate that indicates the ratio of the number of detectable faults to the total number of faults in the software, a is the expected total number of faults to be eventually detected, b is the fault detection rate, and ψ is the inflection factor.

- **Modified Duane Model**. Duane published a report that presented failure data of several systems during their developments in 1962 by analyzing the data. It was observed that the cumulative Mean-Time-Between-Failures versus the cumulative operating time becomes close to a straight line if plotted on log-log paper. Later, a modified Duane model was proposed and its hypothesized mean value function is [3]

$$m(t) = a\left\{1 - \left(\frac{b}{b + t}\right)^c\right\}, \ a > 0, b > 0, c > 0,$$

where a is the expected total number of faults to be eventually detected.

- **Two-Error-Type Model**. Yamada and Osaki proposed a software reliability growth model with nonhomogeneous fault detection rate by assuming that the errors can be divided into type 1 and type 2 errors [3]. They assume that type 1 errors are easy to detect and type 2 errors are difficult to detect. The mean value function is

$$m(t) = a\sum_{i=1}^{2} p_i(1 - \exp[-b_i t], \ a > 0, 0 < b_2 < b_1 < 1,$$
$$p_1 + p_2 = 1, 0 < p_1 < 1, 0 < p_2 < 1,$$

where a is the expected total number of faults to be eventually detected, b_i is the fault detection rate of

type i error $(i = 1, 2)$, and p_i is the percentage of type i error $(i = 1, 2)$.

- **Weibull-Type Testing-Effort Function Model**. Yamada et al. [3] proposed a software reliability growth model incorporating the amount of test-effort expended during the software-testing phase. The mean value function is

$$m(t) = a(1 - \exp[-b\alpha(1 - \exp[-\beta \times t^\gamma])]),$$
$$a, b, \alpha, \beta, \gamma, > 0,$$

where a is the expected total number of faults to be eventually detected, α is the total amount of test-effort required by software testing, β is the scale parameter, γ is the shape parameter, and b is the fault detection rate.

3 WEIGHTED ARITHMETIC, WEIGHTED GEOMETRIC, AND WEIGHTED HARMONIC MEANS

In this section, we first introduce three well-known means: arithmetic, geometric, and harmonic means. Then, we consider several more general means [4], [8].

Let $x \geq 0$ and $y \geq 0$. The *arithmetic mean* z of x and y is defined as

$$z = \frac{1}{2}x + \frac{1}{2}y.$$

More generally, the *weighted arithmetic mean* z of x and y with weights w and $1 - w$ is defined as

$$z = wx + (1 - w)y, \ \ 0 < w < 1. \quad (5)$$

The *geometric mean* z of x and y is defined as

$$z = \sqrt{xy}.$$

That is,

$$\ln z = \frac{1}{2}\ln x + \frac{1}{2}\ln y.$$

Similarly, the *weighted geometric mean* z of x and y with weights w and $1 - w$ is defined as

$$\ln z = w\ln x + (1 - w)\ln y, \ \ 0 < w < 1. \quad (6)$$

The *harmonic mean* z of x and y is defined as

$$\frac{1}{z} = \frac{1}{2x} + \frac{1}{2y}.$$

Similarly, the *weighted harmonic mean* z of x and y with weights w and $1 - w$ is defined as

$$\frac{1}{z} = w\frac{1}{x} + (1 - w)\frac{1}{y}, \ \ 0 < w < 1. \quad (7)$$

Proposition 1. Let z_1, z_2, and z_3, respectively, be the weighted arithmetic, the weighted geometric, and the weighted harmonic means of two nonnegative real numbers x and y with weights w and $1 - w$, where $0 < w < 1$. Then,

$$\min(x, y) \leq z_3 \leq z_2 \leq z_1 \leq \max(x, y),$$

where equality holds if and only if $x = y$.

A more general mean than the ones defined above is considered in the following.

Definition 1. *Let g be a real-valued and strictly monotone function. Let x and y be two nonnegative real numbers. The quasi arithmetic mean z of x and y with weights w and $1 - w$ is defined as*

$$z = g^{-1}(wg(x) + (1-w)g(y)), \ \ 0 < w < 1, \quad (8)$$

where g^{-1} is the inverse function of g.

Obviously, from (8), we can get the weighted arithmetic (where $g(x) = x$), the weighted geometric (where $g(x) = 1/x$), and the weighted harmonic means (where $g(x) = \ln x$).

Proposition 2. *Let z be the quasi arithmetic mean of two nonnegative real numbers x and y with weights w and $1 - w$ and $0 < w < 1$. Then,*

$$\min(x, y) \le x \le \max(x, y), \quad (9)$$

where equality holds if and only if $x = y$.

4 A GENERAL DISCRETE SOFTWARE RELIABILITY GROWTH MODEL

For software reliability growth models, either the calendar time or the machine execution time is often used as the unit of software fault detection period. However, the appropriate unit of software fault detection period may sometimes be measured in terms of the number of test runs [1]. Therefore, Yamada et al. [1], [3] proposed a general description of a discrete software reliability growth model based on an NHPP in which a random variable is defined as the number of faults detected by i test runs $(i = 0, 1, 2, \ldots)$.

Let $\{N(i), i = 0, 1, 2, \ldots\}$ denote a discrete counting process representing the cumulative number of faults detected by i test runs. An SRGM based on an NHPP with mean value function $m(i)$ can be formulated as

$$P[N(i) = n] = \frac{[m(i)]^n \exp[-m(i)]}{n!}, \quad (10)$$
$$i = 0, 1, 2, ., n = 0, 1, 2, \ldots,$$

where $m(i)$ represents the expected cumulative number of faults detected by i test runs. In this section, we propose and discuss a general discrete NHPP model. For the discrete Goel-Okumoto model [3], suppose that the expected number of faults detected per test run is proportional to the current fault content of a software system; that is,

$$m(i+1) - m(i) = b\{a - m(i)\}, \ \ a > 0, 0 < b < 1, \quad (11)$$

where $a = m(\infty)$ is the expected number of software faults to be eventually detected and b is the fault detection rate which is a constant. Taking $w = 1 - b$, we have

$$m(i+1) = wm(i) + (1-w)a, \ \ a > 0, 0 < w < 1. \quad (12)$$

Equation (12) indicates that $m(i+1)$ is equal to the weighted arithmetic mean of $m(i)$ and a with weights w and $1 - w$, respectively. That is, the discrete Goel-Okumoto model can also be derived based on the weighted arithmetic mean. Since the *weighted arithmetic*, *weighted geometric*, and

weighted harmonic means are all well-known means, we thus try to use the other two means to derive other existing NHPP models. First, consider the case that $m(i + 1)$ is equal to the weighted geometric mean of $m(i)$ and a with weights w and $1 - w$, respectively, then

$$\frac{1}{m(i+1)} = w\frac{1}{m(i)} + (1-w)\frac{1}{a}, \ \ 0 < w < 1, a > 0. \quad (13)$$

Next, consider the case that $m(i+1)$ is equal to the weighted harmonic mean of $m(i)$ and a with weights w and $1 - w$, respectively, then

$$\ln m(i+1) = w\ln m(i) + (1-w)\ln a, 0 < w < 1, a > 0. \quad (14)$$

More generally, let g be a real-valued and strictly monotone function and $m(i + 1)$ be equal to the quasi arithmetic mean of $m(i)$ and a with weights w and $1 - w$, respectively, then

$$g(m(i+1)) = wg(m(i) + (1-w)g(a), \ \ 0 < w < 1, a > 0. \quad (15)$$

Solving (15) yields

$$g(m(i+1))$$
$$= w^i g(m(0)) + (1 + w + w^2 + \ldots + w^{i-1})(1-w)g(a) \quad (16)$$
$$= w^i g(m(0)) + (1 - w^i)g(a).$$

Therefore,

$$m(i) = g^{-1}(w^i g(m(0)) + (1 + w^i)g(a)), \quad (17)$$

where $a = m(\infty)$ is the expected number of faults to be eventually detected.

Moreover, by (15) and Proposition 2, $m(i)$ in (17) is nondecreasing in i; that is, the growth curve of the cumulative number of detected faults is nondecreasing. In the following, we show that three classical SRGMs based on NHPPs can be derived by assigning $g(x) = x$, $g(x) = 1/x$, and $g(x) = \ln x$ in (17), respectively.

- **Goel-Okumoto Model**. Take $g(x) = x$ (i.e., $g^{-1}(x) = x$) and $w = e^{-b}$, then

  $$m(i) = w^i g(m(0)) + (1 - w^i)g(a) = a(1 - k\exp[-bi]),$$

  where $k = 1 - m(0)/a$, $b > 0$, and $0 < k$.
  If $k = 1$ (i.e., $m(0) = 0$), then

  $$m(i) = a(1 - \exp[-bi]), \ \ a > 0, a > 0.$$

- **Gompertz Growth Curve**. Take $g(x) = \ln x$ (i.e., $g^{-1}(x) = e^x$) and $w = b$, then

  $$m(i) = \exp[w^i \ln m(0) + (1 - w^i)\ln a] = ak^{b^i},$$

  where $k = m(0)/a$, $a > 0$, $0 < b < 1$, and $0 < k <$.

- **Logistic Growth Curve**. Take $g(x) = 1/x$ (i.e., $g^{-1}(x) = 1/x$) and $w = e^{-b}$, then

  $$m(i) = \frac{1}{w^i/m(0) + (1 - w^i)/a} = \frac{a}{1 + k\exp[-bi]},$$

  where $k = a/m(0) - 1$, $b > 0$, and $k > 0$.

In the following, we consider a more general case that w in (15) is not a constant for all i. That is, let $m(i + 1)$ be equal to the quasi arithmetic mean of $m(i)$ and a with weights $w(i)$ and $1 - w(i)$, respectively. Equation (15) is then

$$g(m(i + 1)) = w(i)g(m(i)) + (1 - w(i))g(a). \quad (18)$$

Solving (18) yields

$$g(m(i)) = \prod_{j=1}^{i} w(j)g(m(0)) + \left(1 - \prod_{j=1}^{i} w(j)\right)g(a) \quad (19)$$
$$= u_i g(m(0)) + (1 - u_i)g(a),$$

where $u_0 = 1$ and $u_i = \prod_{j=1}^{i} w(j)$ for $i \geq 1$. Therefore,

$$m(i) = g^{-1}(u_i g(m(0)) + (1 - u_i)g(a)) \quad (20)$$

is the general discrete NHPP model if g is a real-valued and strictly monotone function. By (18) and Proposition 2, $m(i)$ in (20) is nondecreasing in i. Obviously, $0 < u_i < 1$ for $i \geq 1$ and u_i decrease to 0 as $i \to \infty$. In addition, (20) can be interpreted as that $m(i)$ is equal to the quasi arithmetic mean of $m(0)$ and a with weights u_i and $1 - u_i$, respectively. If $g(x) = x$ in (20), then

$$m(i) = u_i m(0) + (1 - u_i)a = a(1 - m(0)/a)u_i) = a(1 - ku_i), \quad (21)$$

where $k = 1 - m(0)/a$ and $0 < k \leq 1$.

For $g(x) = \ln x$ in (20), we have

$$m(i) = a\left(\frac{m(0)}{a}\right)^{u_i} = ak^{u_i}, \quad (22)$$

where $k = m(0)/a$ and $0 < k < 1$.

If $g(x) = 1/x$ in (20), then

$$m(i) = \frac{a}{1 + (a/m(0) - 1)u_i} = \frac{a}{1 + ku_i}, \quad (23)$$

where $k = a/m(0) - 1$ and $k > 1$.

We now show that other SRGMs based on NHPPs can also be directly derived from (21).

- **Generalized Goel NHPP Model**. Take $g(x) = x$ and $u_i = \exp[-bi^c]$ (i.e., $w(i) = \exp[-b\{i^c - (i-1)^c\}]$), then

  $$m(i) = a(1 - k\exp[-bi^c]), a > 0, b > 0, c > 0, 0 < k \leq 1.$$

  If $k = 1$, then

  $$m(i) = a(1 - \exp[-bi^c]), a > 0, b > 0, c > 0, 0 < k \leq 1.$$

  In this model, $w(i)$ is increasing with respect to i if $c < 1$, $w(i)$ is a constant if $c = 1$, and $w(i)$ is decreasing if $c > 1$.
- **Delayed S-Shaped Model**. Take $g(x) = x$ and $u_i = (1 + bi)\exp[-bi]$ (i.e., $w(i) = \frac{1+bi}{1+b(i-1)}\exp[-b]$), then

  $$m(i) = a(1 - k(1 + bi)\exp[-bi]), a > 0, b > 0, 0 < k \leq 1.$$

  If $k = 1$, then

  $$m(i) = a(1 - (1 + bi)\exp[-bi]), a > 0, b > 0.$$

  In this model, $w(i)$ is increasing with respect to i.

- **Inflected S-Shaped Model**. Take $g(x) = x$ and

  $$u_i = \frac{(1 + c)\exp[-bi]}{1 + c\exp[-bi]}$$

  (i.e., $w(i) = \frac{1+c\exp[-b(i-1)]}{1+c\exp[-bi]}$), then

  $$m(i) = a\left(1 - k\frac{(1 + c)\exp[-bi]}{1 + c\exp[-bi]}\right),$$
  $$a > 0, b > 0, c > 0, 0 < k \leq 1.$$

  If $k = 1$, then

  $$m(i) = a\frac{1 - \exp[-bi]}{1 + c\exp[-bi]}, a > 0, b > 0, c > 0.$$

  In this model, $w(i)$ is increasing with respect to i.

5 A GENERAL CONTINOUS SOFTWARE RELIABILITY GROWTH MODEL

In this section, we propose a general continuous NHPP model. As in the above discussion for the discrete case, let $m(t + t)$ be equal to the quasi arithmetic mean of $m(t)$ and a with weights $w(t, \Delta t)$ and $1 - w(t, \Delta t)$, respectively, i.e.,

$$g(m(t + \Delta t)) = w(t, \Delta t)g(m(t)) + (1 - w(t, \Delta t))g(a), \quad (24)$$

where $0 < w(t, \Delta t) < 1$ and g is a real-valued, strictly monotone, and differentiable function. That is,

$$\frac{g(m(t + \Delta t)) - g(m(t))}{\Delta t} = \frac{1 - w(t, \Delta t)}{\Delta t}(g(a) - g(m(t))).$$

Suppose $(1 - w(t, \Delta t))/\Delta t \to b(t)$ as $\Delta t \to 0$, we get the differential equation

$$\frac{d}{dt}g(m(t)) = b(t)\{g(a) - g(m(t))\}. \quad (25)$$

If $g(x) = x$ in (25) (i.e., the weighted arithmetic mean is considered), then

$$\frac{d}{dt}m(t) = b(t)\{a - m(t)\}. \quad (26)$$

Here, $b(t)$ is the fault detection rate. Furthermore, if $b(t) = b$, then the Goel-Okumoto model can be derived from (26). The differential equations for $g(x) = \ln x$ and $g(x) = 1/x$ can also be derived from (25), respectively.

Theorem 1. *Let $\frac{d}{dt}g(m(t)) = b(t)\{g(a) - g(m(t))\}$, where g is a real-valued, strictly monotone, and differentiable function. We have*

$$m(t) = g^{-1}(g(a) + \{g(m(0)) - g(a)\}\exp[-B(t)]), \quad (27)$$

where

$$B(t) = \int_0^t b(u)du. \quad (28)$$

Proof. Multiply both sides of (25) by $\exp[B(t)]$, then

$$\exp[B(t)]\frac{d}{dt}g(m(t)) + \exp[B(t)]b(t)g(m(t))$$
$$= \exp[B(t)]b(t)g(a).$$

Therefore,

$$\frac{d}{dt}(\exp[B(t)]g(m(t))) = \frac{d}{dt}(g(a)\exp[B(t)]).$$

Hence,

$$g(m(t)) = g(a) + c\exp[-B(t)].$$

Since $B(0) = 0$, $c = g(m(0)) - g(a)$ and Theorem 1 is then proven.   □

Note that $m(t)$ in (27) is a general form of the continuous NHPP models if g is a real-valued, strictly monotone, and differentiable function. In addition, by (24) and Proposition 2, $m(t)$ is nondecreasing. The failure intensity function, the software reliability, and the fault detection rate of this general NHPP model are shown below, respectively,

a. $\lambda(t) = -\{g'(m(t))\}^{-1}(g(m(0)) -$
       $g(a))b(t)\exp[-B(t)],$ (29)

b. $R(t \mid s) = \exp[g^{-1}(g(a) + (g(m(0)) - g(a)) \times$
       $\exp[-B(s)]) - g^{-1}(g(a) + (g(m(0)) - g(a)) \times$
       $\exp[-B(t + s)])],$ (30)

c. $d(t) = \frac{b(t)}{a - m(t)}\frac{g(a) - g(m(t))}{G'(m(t))}.$ (31)

The values of $m(t)$, $\lambda(t)$, $R(t \mid s)$, and $d(t)$ for $g(x) = x$, $g(x) = 1/x$, and $g(x) = \ln x$ are depicted in the following three corollaries, respectively.

Corollary 1. *Based on the weighted arithmetic mean, take $g(x) = x$ in (27) and let $k = 1 - m(0)/a$, then*

1. $m(t) = a(1 - k\exp[-B(t)]), a > 0, 0 < l \le 1;$
2. $\lambda(t) = akb(t)\exp[-B(t)];$
3. $R(t \mid s) = \exp[-ak(\exp[-B(s)] - \exp[-B(t + s)])];$
4. $d(t) = b(t).$

Proof. The proof is omitted since it is quite straightforward. □

Corollary 2. *Based on the weighted geometric mean, take $g(x) = \ln x$ in (27) and let $k = m(0)/a$, then*

1. $m(t) = ak^{\exp[-B(t)]}, a > 0, 0 < k < 1;$
2. $\lambda(t) = -a(\ln k)b(t)\exp[-B(t)]k^{\exp[-B(t)]};$
3. $R(t \mid s) = \exp[-a(k^{\exp[-B(t+s)]} - k^{\exp[-B(s)]});$
4. $d(t) = b(t)m(t)(\ln a - \ln m(t))/(a - m(t));$
5. *if $b(t)$ is nondecreasing in t, then $d(t)$ is nondecreasing in t.*

Proof. Items 1-4 are straightforward. To prove item 5, let

$$h(t) = \frac{m(t)(\ln a - \ln m(t))}{a - m(t)}.$$

In the following, we want to show that $h(t)$ is nondecreasing in t. Taking the derivative of $h(t)$, we have

$$h'(t) =$$
$$\frac{m'(t)\{(\ln a - \ln m(t) - 1)(a - m(t) + m(t)(\ln a - \ln m(t)))\}}{[a - m(t)]^2}.$$
        (32)

The numerator of the above equation is equal to

$$m'(t)\{m(t) + a(\ln a - \ln m(t) - 1)\}.$$

Let $f(t) = m(t) + a(\ln a - \ln m(t) - 1)$. By simple calculation, it can be shown that 0 and $f'(t) \le 0$ and $\lim_{t\to\infty} f(t) = 0$. Hence, we have $f(t) \le 0$ for all t. Since $m'(t) \ge 0$ and $[a - m(t)]^2 > 0$, we have $h'(t) \ge 0$, that is, $h(t)$ is nondecreasing. Thus, item 5 of Corollary 2 is proven.   □

Corollary 3. *Based on the weighted harmonic mean, take $g(x) = 1/x$ in (27) and let $k = a/m(0) - 1$, then*

1. $m(t) = \frac{a}{1 + k\exp[-B(t)]}, a > 0, k > 0;$
2. $\lambda(t) = \frac{akb(t)}{(1 + k\exp[-B(t)])^2} \times \exp[-B(t)];$
3. $R(t \mid s) = \exp\left[-a\left(\frac{1}{1 + k\exp[-B(t+s)]} - \frac{1}{1 + k\exp[-B(s)]}\right)\right];$
4. $d(t) = b(t)m(t)/a.$

Proof. This corollary can be easily proven as well.   □

Proposition 3. *Let $m_A(t)$, $m_G(t)$, and $m_H(t)$ be three mean value functions defined by (26), (27), and (28), respectively. Moreover, suppose the values of a, $m(0)$, and $b(t)$ in the above three mean value functions are all equal. We have*

$$m_A(t) > m_G(t) > m_H(t).$$

Proof. It can be easily proven based on Proposition 1.   □

In the following, we show that some classical SRGMs based on NHPPs can be directly derived from Corollary 1, 2, or 3.

- **Goel-Okumoto Model**. Take $g(x) = x$ and $b(t) = b$, then, from Corollary 1,

    $$m(t) = a(1 - k\exp[-bt]), a > 0, b > 0, 0 < k \le 1.$$

    If $k = 1$, then

    $$m(t) = a(1 - \exp[-bt]), a > 0, b > 0.$$

    In this model, $b(t)$ is a constant.

- **Gompertz Growth Curve**. Take $g(x) = \ln x$ and $b(t) = b$, then, from Corollary 2,

    $$m(t) = ak^{\exp[-bt]}, a > 0, b > 0, 0 < k < 1.$$

    In this model, $b(t)$ is a constant.

- **Logistic Growth Curve**. Take $g(x) = 1/x$ and $b(t) = b$, then, from Corollary 3,

    $$m(t) = \frac{a}{1 + k\exp[-bt]}, a > 0, b > 0, k > 0.$$

    In this model, $b(t)$ is a constant.

- **Generalized Goel NHPP Model**. Take $g(x) = x$ and $b(t) = bct^{c-1}$, then, from Corollary 1,

    $$m(t) = a(1 - k\exp[-bt^c]), a > 0, b > 0, c > 0, 0 < k \le 1.$$

    If $k = 1$, then

    $$m(t) = a(1 - \exp[-bt^c]), a > 0, b > 0, c > 0.$$

    In this model, $b(t)$ is increasing if $c > 1$, a constant if $c = 1$, and decreasing if $c < 1$.

- **Delayed S-shaped Model**. Take $g(x) = x$ and $b(t) = b^2t/(1 + bt)$, then, from Corollary 1,

$$m(t) = a(1 - k(1 + bt)\exp[-bt]), a > 0, b > 0, 0 < k \le 1.$$

If $k = 1$, then

$$m(t) = a(1 - (1 + bt)\exp[-bt]), a > 0, b > 0.$$

In this model, $b(t)$ is increasing in t.

- **Inflected S-Shaped Model**. Take $g(x) = x$ and $b(t) = b/(1 + c\exp[-bt])$, then, from Corollary 1,

$$m(t) = a\left(1 - k\frac{(1 + c)\exp[-bt]}{1 + c\exp[-bt]}\right),$$
$$a > 0, b > 0, c > 0, 0 < k \le 1.$$

If $k = 1$, then

$$m(t) = a\frac{1 - \exp[-bt]}{1 + c\exp[-bt]}, a > 0, b > 0, c > 0.$$

In this model, $b(t)$ is increasing in t.

- **Modified Duane Model**. Take $g(x) = x$ and $b(t) = c/(b + t)$, then ,from Corollary 1,

$$m(t) = a\left\{1 - k\left(\frac{b}{b + t}\right)^c\right\}, a > 0, b > 0, c > 0, 0 < k \le 1.$$

If $k = 1$, then

$$m(t) = a\left\{1 - \left(\frac{b}{b + t}\right)^c\right\}, a > 0, b > 0, c > 0.$$

In this model, $b(t)$ is decreasing in t.

- **Two-Error-Type Model**. Take $g(x) = x$ and

$$b(t) = \left(\sum_{i=1}^{2} p_i b_i \exp[-b_i t]\right) / \left(\sum_{i=1}^{2} p_i \exp[-b_i t]\right),$$

then, from Corollary 1,

$$m(t) - a\sum_{i=1}^{2} p_i(1 - k\exp[-b_i t]), 0 < k \le 1.$$

If $k = 1$, then

$$m(t) = a\sum_{i=1}^{2} p_i(1 - \exp[-b_i t]),$$

where $a > 0, 0 < b_2 < b_1 < 1, p_1 + p_2 = 1, 0 < p_1 < 1$, and $0 < p_2 < 1$. In this model, $b(t)$ is decreasing in t.

- **Weibull-Type Testing-Effort Function Model**. Take $g(x) = x$ and $b(t) = b\alpha\beta\gamma \times t^{r-1}\exp[-\beta \times t^\gamma]$, then, from Corollary 1,

$$m(t) = a(1 - k\exp[-b\alpha(1 - \exp[-\beta \times t^\gamma])]), 0 < k \le 1.$$

If $k = 1$, then

$$m(t) = a(1 - \exp[-b\alpha(1 - \exp[-\beta t^\gamma])]), a, b, \alpha, \beta, \gamma > 0.$$

- **Log-Logistic Software Reliability Growth Model**. This model was proposed by Gokhale and Trivedi [9]. They offered a decomposition of the mean value function of a finite failure NHPP model, which can capture the increasing/decreasing nature of the hazard function. That is, the increasing/decreasing behavior of the failure occurrence rate per fault can be captured by the hazard function of the log-logistic distribution. Take $g(x) = 1/x$, $b(t) = k/t$, and use some dummy variables during the derivation, then, from Corollary 3,

$$m(t) = a \times \frac{ct^k}{1 + d(ct^k)}, a > 0, c > 0, d > 0, k > 0. \quad (33)$$

Let $c = \lambda k$ and $d = 1$, we get the log-logistic form model:

$$m(t) = a\frac{\lambda^k t^k}{1 + \lambda^k t^k} = a\frac{(\lambda t)^k}{1 + (\lambda t)^k}, a > 0, \lambda > 0, k > 0. \quad (34)$$

- **SRGM with Logistic Testing-Effort Function**. In the field of software reliability modeling, Musa first discussed the validity of execution time theory by taking data sets from real software systems as testing effort can be faithfully represented by execution times [2], [3]. However, most existing software reliability models do not take testing effort into consideration. Recently, a simple and new software reliability growth model with logistic testing-effort function is proposed [10], [11], [12], [13]. This model attempts to account for the relationship between the amount of testing-effort and the number of software faults detected during testing. In fact, the testing effort can be measured as the human power, the number of test cases, the number of CPU hours, etc. [15], [16], [17], [18], [19], [20]. The MVF $m(t)$ can be described as the following:

$$m(t) = a(1 - \exp[-b(W(t) - W(0))])$$
$$= a\left(1 - \exp\left[-b\left\{\frac{N}{1 + A\exp[-\alpha t]} - \frac{N}{1 + A}\right\}\right]\right), \quad (35)$$

where a is the expected number of initial faults in the software, b is the fault detection rate per unit testing-effort at testing time t that satisfies $b > 0$, N is the total amount of testing effort to be eventually consumed, α is the consumption rate of testing-effort expenditures, and A is a constant. If we take $g(x) = x$ and

$$b(t) = \frac{bNA\alpha e^{-\alpha t}}{(1 + Ae^{-\alpha t})^2} = \frac{bNA\alpha}{\left(e^{\frac{\alpha t}{2}} + Ae^{\frac{-\alpha t}{2}}\right)^2},$$

then, from Corollary 1,

$$m(t) = a\left(1 - k\exp\left[-b\left\{\frac{N}{1 + A\exp[-\alpha t]} - \frac{N}{1 + A}\right\}\right]\right). \quad (36)$$

If $k = 1$, then

$$m(t) = a\left(1 - \exp\left[-b\left\{\frac{N}{1 + A\exp[-\alpha t]} - \frac{N}{1 + A}\right\}\right]\right).$$

Finally, in addition to the previously discussed three known means, we propose a more general transformation that includes a parametric family of power transformations [14]:

$$g(x) = \begin{cases} \frac{x^\alpha - 1}{\alpha}, & \alpha \neq 0 \\ \ln x, & \alpha = 0. \end{cases} \qquad (37)$$

Note that $g(x)$ is only one of many parametric families of transformations that can be used for data analysis.

Corollary 4. *Based on the power transformation, if we take*

$$g(x) = \begin{cases} \frac{x^\alpha - 1}{\alpha}, & \alpha \neq 0 \\ \ln x, & \alpha = 0. \end{cases}$$

into (27) and let $k = 1 - \left(\frac{m(0)}{a}\right)^\alpha$, *then*

1. $m(t) - a(1 - ke^{-Bt(t)})^{1/\alpha}, \alpha \neq 0;$
2. $\lambda(t) = \frac{1}{\alpha} akb(t)e^{-B(t)}(1 - ke^{-B(t)})^{\frac{1-\alpha}{\alpha}} \alpha \neq 0;$
3. $R(t \mid s) = \exp\{a[(1 - ke^{-B(s)})^{1/\alpha} - (1 - ke^{-B(t+s)})^{1/\alpha} \ \alpha \neq 0;$
4. $d(t) = \frac{m'(t)}{a - m(t)} = \frac{\frac{1}{\alpha} kb(t)e^{-B(t)}(1 - ke^{-B(t)})^{\frac{1-\alpha}{\alpha}}}{1 - (1 - ke^{-B(t)})^{1/\alpha}}$, *where*

$$k = 1 - \left(\frac{m(0)}{a}\right)^\alpha \text{ and } \alpha \neq 0;$$

5. *if* $\alpha = 0$, *the result is the same as Corollary 2.*

Proof. 1) Since $g(x) = \frac{x^\alpha - 1}{\alpha}$, $\alpha \neq 0$, that is, $g^{-1}(y) = (\alpha y + 1)^{1/\alpha}$. From (27) and through some simple calculations, we obtain

$$g(a) + \left(\frac{m(0)^\alpha - a^\alpha}{\alpha}\right) \exp[-B(t)]$$
$$= \frac{a^\alpha - 1}{\alpha} + k^* \times \exp[-B(t)],$$

where $k^* = \frac{m(0)^\alpha - a^\alpha}{\alpha}$, $g(m(0)) = \frac{m(0)^\alpha - 1}{\alpha}$, and $g(a) = \frac{a^\alpha - 1}{\alpha}$. Consequently,

$$m(t) = g^{-1}(g(a) + \{g(m(0)) - g(a)\}\exp[-B(t)])$$
$$= \left(\alpha \times \left(\frac{a^\alpha - 1}{\alpha} + k' \times \exp[-B(t)]\right) + 1\right)^{1/\alpha}$$
$$= (a^\alpha + (m(0)^\alpha - a^\alpha) \times \exp[-B(t)])^{1/\alpha}$$
$$= \left(a^\alpha \times \left(1 - \frac{1}{a^\alpha} \times (a^\alpha - (m(0)^\alpha) \times \exp[-B(t)])\right)\right)^{1/\alpha}$$
$$= (a^\alpha \times (1 - k \times \exp[-B(t)]))^{1/\alpha}$$
$$= a \times (1 - k \times \exp[-B(t)])^{1/\alpha},$$

where $k = 1 - \left(\frac{m(0)}{a}\right)^\alpha$. The proofs for 2 to 5 are straightforward and, thus, omitted. □

Altogether, based on Corollary 4, we can generate several new models with various $b(t)$:

1. If $b(t) = b$, i.e., $B(t) = bt$, then the new MVF is

$$m(t) = a(1 - e^{-bt})^{1/\alpha}. \qquad (38)$$

2. If $b(t) = bct^{c-1}$, i.e., $B(t) = bt^c$, then the new MVF is

$$m(t) = a(1 - ke^{-bt^c})^{1/\alpha}, \alpha \neq 0. \qquad (39)$$

3. If $b(t) = \frac{c}{b+t}$, i.e., $B(t) = c\ln\frac{b+t}{b}$, then the new MVF is

$$m(t) = a\left(1 - k\left(\frac{b+t}{b}\right)^c\right)^{1/\alpha}, \alpha \neq 0. \qquad (40)$$

4. If $b(t) = \frac{b^2 t}{1+bt}$, i.e., $B(t) = bt = \ln(1 + bt)$, then the new MVF is

$$m(t) = a(1 - (1 + bt)ke^{-bt})^{1/\alpha}, \alpha \neq 0. \qquad (41)$$

5. If $b(t) = \frac{b}{1+ce^{-bt}}$, i.e., $B(t) = bt + \ln\frac{1+ce^{-bt}}{1+c}$, then the new MVF is

$$m(t) = a\left(1 - \frac{k(1+c)e^{-bt}}{1+ce^{-bt}}\right)^{1/\alpha}, \alpha \neq 0. \qquad (42)$$

6 CONCLUSIONS

In this paper, we first showed how several existing software reliability growth models based on NHPPs can be derived based on the concept of *weighted arithmetic*, *weighted geometric*, and *weighted harmonic means*. Next, we proposed a general NHPP model based on the quasi arithmetic mean, which is a more general mean compared with the above three weighted means. We also developed a more general NHPP model using the *power transformation*. Many existing SRGMs based on NHPPs are special cases of this general NHPP model. One unique contribution of this paper is that we do not add any new models to the already large collection of SRGMs, rather we emphasize new approaches for model development and classification. Based on the integrated theoretical foundation, the techniques and approaches presented in this paper offer a unified and consistent software reliability modeling and evaluation scheme.

REFERENCES

[1] M.R. Lyu, *Handbook of Software Reliability Engineering.* McGraw-Hill, 1996.
[2] J.D. Musa, A. Iannino, and K. Okumoto, *Software Reliability, Measurement, Prediction and Application.* McGraw-Hill, 1987.
[3] M. Xie, *Software Reliability Modeling.* World Scientific Publishing, 1991.
[4] R.H. Hou, S.Y. Kuo, and Y.P. Chang, "On a Unified Theory of Some Nonhomogenous Poisson Process Models for Software Reliability," *Proc. 1998 Int'l Conf. Software Eng., Education & Practice (SEEP '98),* pp. 60-67, Jan. 1998.
[5] N. Langberg and N.D. Singpurwalla, "A Unification of Some Software Reliability Models," *SIAM J. Scientific and Statistical Computing,* vol. 6, no. 3, pp. 781-790, Mar. 1985.
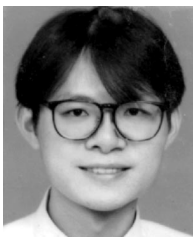
[6] D.R. Miller, "Exponential Order Statistic Models of Software Reliability Growth," *IEEE Trans. Software Eng.,* vol. 12, no. 1, pp. 12-24, Jan. 1986.

[7] M. Trachtenberg, "A General Theory of Software-Reliability Modeling," *IEEE Trans. Reliability,* vol. 39, no. 1, pp. 92-96, Jan. 1990.

[8] P.S. Bullen, D.S. Mitrinovic, and P.M. Vasic, *Means and Their Inequalities.* Dordrecht, Holland: D. Reidel Publishing, 1988.

[9] S.S. Gokhale and K.S. Trivedi, "Log-Logistic Software Reliability Growth Model," *Proc. Third IEEE Int'l High-Assurance Systems Eng. Symp.,* pp. 34-41, 1998.

[10] C.Y. Huang and S.Y. Kuo, "Analysis and Assessment of Incorporating Logistic Testing Effort Function into Software Reliability Modeling," *IEEE Trans. Reliability,* vol. 51, no. 3, pp. 261-270, Sept. 2002.

[11] S.Y. Kuo, C.Y. Huang, and M.R. Lyu, "Framework for Modeling Software Reliability, Using Various Testing-Efforts and Fault-Detection Rates," *IEEE Trans. Reliability,* vol. 50, no. 3, pp. 310-320, Sept. 2001.

[12] C.Y. Huang, S.Y. Kuo, M.R. Lyu, and J.H. Lo, "Quantitative Software Reliability Modeling from Testing to Operation," *Proc. 11th Int'l Symp. Software Reliability Eng., (ISSRE 2000),* pp. 72-82, Oct. 2000.

[13] C.Y. Huang, S.Y. Kuo, and M.R. Lyu, "Effort-Index-Based Software Reliability Growth Models and Performance Assessment," *Proc. 24rd Ann. Int'l Computer Software and Applications Conf. (COMPSAC 2000),* pp. 454-459, Oct. 2000.

[14] G.E. Box and G.M. Jenkins, *Time Series Analysis: Forecasting and Control.* San Francisco: Holden-Day, 1976.

[15] J.D. Musa, *Software Reliability Engineering: More Reliable Software, Faster Development and Testing.* McGraw-Hill, 1999.

[16] B. Littlewood, "Software Reliability Modeling: Achievements and Limitations," *Proc. Fifth Ann. European Computer Conf. Advanced Computer Technology, Reliable Systems, and Applications (CompEuro'91),* pp. 36-344, May 1991.

[17] Y.K. Malaiya and P.K. Srimani, *Software Reliability Models: Theoretical Developments, Evaluation and Applications.* IEEE Press, 1990.

[18] M. Ohba, "Software Reliability Analysis Models," *IBM J. Research and Development,* vol. 28, no. 4, pp. 428-443, July 1984.

[19] A.L. Goel, "Software Reliability Models: Assumptions, Limitations, and Applicability," *IEEE Trans. Software Eng.,* vol. 11, no. 12, Dec. 1985.

[20] H. Pham, *Software Reliability.* Springer-Verlag, 2000.

Chin-Yu Huang received the honorary degree (1989) in electronic engineering from Hocking College, Nelsonville, Ohio. He received the MS (1994) and the PhD (2000) degree in electrical engineering from the National Taiwan University, Taipei. From 1990 to 1992, he studied in the Department of Transportation Engineering and Management at National Chiao Tung University, Hsin-Chu. He was with the Bank of Taiwan from 1994 to 1999 and was a senior software engineer at Taiwan Semiconductor Manufacturing Company from 1999 to 2000. He is now a division chief at the Central Bank of China, Taipei. His research interests are software reliability, software testing, and fault-tolerant computing.

Michael R. Lyu (S'84-M'88-SM'97) received the BS degree in electrical engineering from National Taiwan University, Taipei, in 1981, the MS degree in computer engineering from University of California, Santa Barbara, in 1985, and the PhD degree in computer science from the University of California, Los Angeles, in 1988. He is currently a professor in the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong. He was with the Jet Propulsion Laboratory, Altadena, California, as a technical staff member from 1988 to 1990. From 1990 to 1992, he was with the Department of Electrical and Computer Engineering, the University of Iowa, Iowa City, as an assistant professor. From 1992 to 1995, he was a member of the technical staff in the applied research area of Bell Communications Research (Bellcore), Morristown, New Jersey. From 1995 to 1997, he was a research member of the technical staff at Bell Laboratories, Murray Hill, New Jersey. His research interests include software reliability engineering, distributed systems, fault-tolerant computing, wireless communication networks, Web technologies, digital libraries, and e-commerce systems. He has published more than 120 refereed journal and conference papers in these areas. He has participated in more than 30 industrial projects, and helped to develop many commercial systems and software tools. He was the editor of two book volumes: *Software Fault Tolerance* (New York: Wiley, 1995) and *The Handbook of Software Reliability Engineering* (Piscataway, NJ: IEEE and New York: McGraw-Hill, 1996). He has frequently been invited as a keynote or tutorial speaker to conferences and workshops in the US, Europe, and Asia. Dr. Lyu initiated the First International Symposium on Software Reliability Engineering (ISSRE) in 1990. He was the program chair for ISSRE'96, and has served in program committees for many conferences, including ISSRE, SRDS, HASE, ICECCS, ISIT, FTCS, DSN, ICDSN, EUROMICRO, APSEC, PRDC, PSAM, ICCCN, ISESE, and WWW. He was the general chair for ISSRE2001, and the WWW10 program cochair. He is an associate editor of the *IEEE Transactions on Reliability*, *IEEE Transactions on Knowledge and Data Engineering*, and the *Journal of Information Science and Engineering*. He is a senior member of the IEEE.

Sy-Yen Kuo received the BS degree (1979) in electrical engineering from National Taiwan University, Taipei, the MS degree (1982) in electrical and computer engineering from the University of California at Santa Barbara, and the PhD degree (1987) in computer science from the University of Illinois at Urbana-Champaign. Since 1991, he has been with National Taiwan University, where he is currently a professor and the chairman in the Department of Electrical Engineering. He spent his sabbatical year as a visiting researcher at AT&T Labs-Research, New Jersey, from 1999 to 2000. He was the chairman of the Department of Computer Science and Information Engineering, National Dong Hwa University, Taiwan, from 1995 to 1998, a faculty member in the Department of Electrical and Computer Engineering at the University of Arizona from 1988 to 1991, and an engineer at Fairchild Semiconductor and Silvar-Lisco, both in California, from 1982 to 1984. In 1989, he also worked as a summer faculty fellow at the Jet Propulsion Laboratory of the California Institute of Technology, Altadena. His current research interests include software reliability engineering, mobile computing, dependable systems and networks, and optical WDM networks. Dr. Kuo is an IEEE fellow. He has published more than 180 papers in journals and conferences. He received the distinguished research award (1997 to 2001) from the National Science Council, Taiwan. He was also a recipient of the Best Paper Award in the 2001 Pacific Rim International Symposium on Dependable Computing, the Best Paper Award in the 1996 International Symposium on Software Reliability Engineering, the Best Paper Award in the simulation and test category at the 1986 IEEE/ACM Design Automation Conference (DAC), the National Science Foundation's Research Initiation Award in 1989, and the IEEE/ACM Design Automation Scholarship in 1990 and 1991.

▷ **For more information on this or any computing topic, please visit our Digital Library at** http://computer.org/publications/dlib.