Friedrich-Alexander-Universität Erlangen-Nürnberg
Wirtschafts- und Sozialwissenschaftliche Fakultät

Discussion Paper

66 / 2004

On a Method for Mending Time to Failure Distributions

Michael Grottke, Kishor S. Trivedi

Lehrstuhl für Statistik und Ökonometrie
Lehrstuhl für Statistik und empirische Wirtschaftsforschung
Lange Gasse 20 · D-90403 Nürnberg

On a Method for Mending Time to Failure Distributions

Michael Grottke[*] **and Kishor S. Trivedi**

Department of Electrical & Computer Engineering

Duke University

Durham, NC 27708-0291, USA

grottke@ee.duke.edu

kst@ee.duke.edu

Discussion Paper 66 / 2004

Abstract

Many software reliability growth models assume that the time to next failure may be infinite; i.e., there is a chance that no failure will occur at all. For most software products this is too good to be true even after the testing phase. Moreover, if a non-zero probability is assigned to an infinite time to failure, metrics like the mean time to failure do not exist. In this paper, we try to answer several questions: Under what condition does a model permit an infinite time to next failure? Why do all finite failures non-homogeneous Poisson process (NHPP) models share this property? And is there any transformation mending the time to failure distributions? Indeed, such a transformation exists; it leads to a new family of NHPP models. We also show how the distribution function of the time to first failure can be used for unifying finite failures and infinite failures NHPP models.

Keywords: software reliability growth model, non-homogeneous Poisson process, defective distribution, (mean) time to failure, model unification

[*]Corresponding author, on leave of absence from the Chair of Statistics and Econometrics, University of Erlangen-Nuremberg, Germany. This work was supported by a fellowship within the Postdoc Program of the German Academic Exchange Service (DAAD).

1 Introduction

Despite the advances made with respect to the development of techniques and tools supporting the requirements analysis, the design and the implementation of software, the correctness of computer programs cannot be guaranteed. It is always possible that a piece of software contains faults (e.g., buggy lines of code) leading to deviations of the actual software behavior from its specification. Such observed deviations are referred to as failures.

Since the number of software faults, their location in the code and the sequence of user inputs are not pre-determined, the times at which failures are experienced are random. Let the continuous random variable X_i represent the time between the $(i-1)^{st}$ and the i^{th} failure occurrence, also called the i^{th} time to failure (TTF). For a program that has already been released, we hope that all realizations of the TTFs are large values; i.e., the software should only fail rarely. This means that due to the characteristics of the software and the execution profile each random variable X_i should have a density function assigning a large fraction of the probability mass to long inter-failure times. In an ideal scenario, in which the software does not even fail once, the entire probability mass of the first TTF X_1 is assigned to infinity. This may happen either if the software is fault-free or if the existing faults are located in parts of the software that will never be executed. If there is a certain chance that no fault is contained in those regions of the software (eventually) used according to the operational profile, then a probability between zero and one is attached to infinity. As long as the software may not fail at all, the distribution function of X_1 does not reach the value one for x approaching infinity:

$$\lim_{x \to \infty} F_{X_1}(x) < 1.$$

Distributions with this characteristic are called "improper" [13] or "defective" [25, p. 146].

While a defective TTF distribution is desirable in the operational phase, during the testing phase of software development (from initial unit tests up to integration and system tests) it is not. Many testing professionals and researchers follow Myers in considering testing to be "a *destructive* process, even a sadistic process" [22, p. 5] and a test case finding a fault to be successful. For increasing the efficiency of fault detection various systematic testing strategies have been proposed, see for example Myers' classical monograph cited above, or [12]. In contrast with these approaches is the operational testing technique [18, 19], which aims at mimicking the user behavior in order to uncover those faults that are most dangerous from a user perspective and to assess the current operational reliability. But even within operational testing concepts like the testing compression factor [20, pp. 233–234] are introduced in order to account for efforts to amplify the speed with which the code and the faults contained in it are covered during testing. (For a more detailed discussion of systematic and operational testing see [7, pp. 6–14].)

However, many existing software reliability growth models (SRGMs) used for modeling and predicting failure occurrences during the integration and system test phase share the property that *all* TTF distributions are defective - implying the possibility that no failure will occur at all. This does not only seem to be in disagreement with the approaches to testing sketched above, it also entails problems for the application of the models. For an SRGM in which all TTF distributions are defective, the moments of these distributions are infinite. Therefore, important metrics like the mean time to failure $E(X_i)$ or the variance $Var(X_i)$ do not exist for all values of i. Even if the probability for an infinite TTF is very small, it drowns any useful information about the distribution that these measures might convey.

This paper investigates why certain SRGMs imply defective TTF distributions. Its main contribution is the derivation of a generic method for transforming non-homogeneous Poisson process (NHPP) models of the finite failures category; in the resulting model class all TTF distributions are proper. An additional result of our research is a mean value function unifying all NHPP models.

The remaining parts are organized as follows: In section 2 we investigate the general class of continuous-time Markov chain SRGMs. NHPP models in particular are studied in section 3. Based on the insight gained, we are able to find an approach for transforming NHPP models of the finite failures category such that all TTF distributions of the resulting models are non-defective; this approach is explained in section 4. With respect to continuous-time Markov chain models not belonging to the class of NHPP models section 5 identifies those sub-classes for which the TTF distributions may be defective. In section 6 we apply our generic transformation to the well-known Goel-Okumoto model. This leads us to a new SRGM which we call "truncated Goel-Okumoto model", and we use this model for fitting and predicting a real failure data set. Section 7 concludes this paper.

2 Defective TTF distributions in SRGMs - General condition

For many SRGMs the stochastic process counting the number of failure occurrences over time, $\{M(t), t \geq 0\}$, is a continuous-time Markov chain (CTMC).[1] Its structure is shown in figure 1.

Assuming that only one failure can occur at a time and taking into account that a failure occurrence cannot be undone, from each state $i-1$ a transition is merely possible to the next state i; the counting process is a pure birth process. The dashed transition out of state u_0 indicates that some models assume that the total number of failure occurrences is bounded by a certain value u_0. For these models state u_0 is absorbing, and the CTMC terminates at that state.

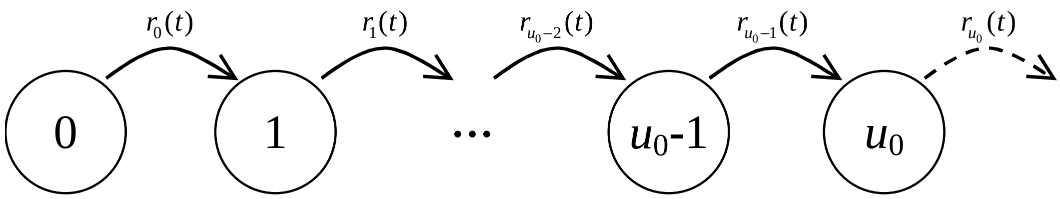

Figure 1: *Counting process as a continuous-time Markov chain*

[1] A more general model class containing additional SRGMs is the self-exciting point process (SEPP). The following discussion of the relationships between the transitions rates, the program hazard rate and the failure intensity function is based on the software reliability literature dealing with SEPPs, see [2, 4, 8, 15, 24].

According to the Markov property, the only part of the history of the counting process that may affect its future is the current state. In addition, the time t may have an influence. Since the transition rate between state $i-1$ and state i is in general both time-dependent and state-dependent, we denote it by $r_{i-1}(t)$. If all transition rates are not time-dependent but only state-dependent, then the SRGM is a homogeneous CTMC model such as the Jelinski-Moranda model [11]; if they are all time-dependent but not state-dependent, then the SRGM belongs to the class of NHPP models.

As long as the current state $m(t)$ of the counting process is unknown, the program hazard rate Z, representing the instantaneous danger of a failure occurrence, is a function of the random variable $M(t)$ as well as time:

$$Z(t, M(t)) = r_{M(t)}(t).$$

Since its realization $z(t, m(t))$ is pieced together from the individual transition rates $r_0(t)$, $r_1(t)$, ..., the program hazard rate is also referred to as "concatenated hazard rate" (or "concatenated failure rate function" [2]). Its expected value with respect to $M(t)$ is a function of time [15], the so-called failure intensity function,

$$\lambda(t) = E(Z(t, M(t))) = \sum_{i=0}^{\infty} r_i(t) \cdot P(M(t) = i).$$

Integrating the failure intensity function from zero to t yields the mean value function $\mu(t)$, representing the expected number of failure occurrences in the interval $(0, t]$:

$$\mu(t) = \int_0^t \lambda(y)\, dy = \sum_{i=0}^{\infty} i \cdot P(M(t) = i) = E(M(t)).$$

Given that $i-1$ failures have been experienced by time t, the reliability in the interval $(t, t+x]$ is

$$R(x \mid t, M(t) = i-1) = \exp\left(-\int_t^{t+x} z(y, i-1)\, dy\right) = \exp\left(-\int_t^{t+x} r_{i-1}(y)\, dy\right).$$

Let the random variables T_1, T_2, ... denote the times of the first, second, ... failure occurrence. We will use t_i $(i = 1, 2, ...)$ for referring to the realization of the i^{th} failure time; $t_0 \equiv 0$ is not a failure time but the beginning of testing.

Given t_{i-1}, the distribution function of X_i is

$$F_{X_i}(x) = 1 - R(x \mid t_{i-1}, M(t_{i-1}) = i-1) = 1 - \exp\left(-\int_{t_{i-1}}^{t_{i-1}+x} r_{i-1}(y)\, dy\right).$$

This distribution of X_i is defective if $r_{i-1}(t)$ converges to zero fast enough for

$$\lim_{x\to\infty} \int_{t_{i-1}}^{t_{i-1}+x} r_{i-1}(y)\, dy = c < \infty, \tag{1}$$

because in this case

$$\lim_{x\to\infty} R(x \mid t_{i-1}, M(t_{i-1}) = i-1) = \exp(-c) > 0 \quad \text{and} \quad \lim_{x\to\infty} F_{X_i}(x) = 1 - \exp(-c) < 1.$$

A possible explanation as to why $r_{i-1}(t)$ may decrease at all although no failure occurs (and hence no fault is corrected) is a subjective one: The longer the software has been running without showing a failure, the higher is the confidence that it will not fail in the future.

3 Defective TTF distributions in NHPP models

3.1 General considerations

For non-homogeneous Poisson process (NHPP) models, all transition rates $r_0(t)$, $r_1(t)$, ... are functions of time t, but they are independent of the number of previous failure occurrences $M(t)$. Therefore, they are the same function $r(t)$. As a consequence, the program hazard rate $Z(t, M(t))$ is not a random variable, but a deterministic function $z(t)$ of time, and it is identical to the function $r(t)$. Moreover, it is identical to the failure intensity $\lambda(t)$. Hence,

$$\lambda(t) = z(t) = r(t) = r_0(t) = r_1(t) = \dots. \tag{2}$$

The model assumptions imply that $M(t)$ follows a Poisson distribution with expectation given by the mean value function $\mu(t)$ connected to equation (2). Specifying either the failure intensity function or the mean value function fully determines the NHPP model.

Given the observed value t_{i-1}, the reliability of the software in the interval $(t_{i-1}, t_{i-1}+x]$ is

$$R(x \mid t_{i-1}, M(t_{i-1}) = i-1) = \exp\left(-\int_{t_{i-1}}^{t_{i-1}+x} \lambda(y)\, dy\right) = \exp\left(-\mu(t_{i-1}+x) + \mu(t_{i-1})\right), \tag{3}$$

and the distribution function of X_i is

$$F_{X_i}(x) = 1 - \exp\left(-\mu(t_{i-1}+x) + \mu(t_{i-1})\right). \tag{4}$$

Whether the distribution of the time to the i^{th} failure is defective or not depends on the behavior of $\mu(t_{i-1}+x)$ as x approaches infinity.

3.2 Finite failures category NHPP models

Musa et al. [20, pp. 250–251] refer to SRGMs for which the expected number of failures experienced in infinite time is finite as "finite failures category models". We follow Kuo and Yang [14] in calling NHPP models of this category "NHPP-I" models. The mean value function of these models has the general form [20, p. 269]

$$\mu(t) = \nu G(t). \tag{5}$$

Assuming perfect fault removal, ν represents the expected number of inherent software faults, and the initial number of faults, N, follows a Poisson distribution with parameter ν [20, p. 268]. The continuous function $G(t)$ can be interpreted as the distribution function of the time until a specific fault causes a failure [20, p. 261], or as a coverage function [6, 23]. Since at the beginning of testing no failure has occurred with probability one, $G(0) = 0$.

Moreover, it is usually assumed that $G(t)$ is non-defective, implying that each fault will eventually lead to a failure. In the well-known Goel-Okumoto model [5], for example, $G(t)$ is the non-defective function

$$G(t) = 1 - \exp(-\phi t). \tag{6}$$

However, the coverage function does not have to be proper. In many SRGMs with a time-varying testing-effort, e.g. the one with a Weibull testing-effort proposed by Yamada et al. [26, 27] and the one with a logistic testing-effort by Huang et al. [9, 10], the coverage function $G(t)$ is

$$G(t) = 1 - \exp(-\phi\gamma W^*(t)). \tag{7}$$

In this equation, $\phi > 0$ represents the fault detection rate per fault and unit of testing-effort, while $\gamma > 0$ stands for the total amount of testing-effort required by software testing. $W^*(t)$ is a (non-defective) distribution function modeling the dispersion of testing-effort over time. Since the total testing-effort is limited by γ,

$$\lim_{t\to\infty} G(t) = 1 - \exp(-\phi\gamma) < 1,$$

which means that the coverage function (7) is defective.

In the following, we will assume that $G(t)$ is a *non-defective* distribution function.

According to equation (2), for an NHPP-I model all transition rates are identical to the failure intensity,

$$r_0(t) = r_1(t) = r_2(t) = ... = \lambda(t) = \nu g(t), \tag{8}$$

where $g(t)$ is the first derivative of $G(t)$ with respect to t. Therefore, the structure of the counting process can be depicted as in figure 2.

Since the expected number of failures experienced during an infinite amount of testing is equal to the expected number of inherent faults ν, the limit of the reliability in the interval $(t_{i-1}, t_{i-1}+x]$ for x approaching infinity is

$$\lim_{x\to\infty} R(x \mid t_{i-1}, M(t_{i-1}) = i-1) = \exp\left(-\nu + \mu(t_{i-1})\right) = \exp\left(-\nu(1 - G(t_{i-1}))\right) > 0. \tag{9}$$

Whatever the number of previous failures $i - 1$ may be, there is always a non-zero probability that the software will not fail an i^{th} time. Therefore, all TTF distributions $F_{X_i}(x)$ connected to NHPP-I models are defective.

An intuitive proposition is that the event of no further failure occurrence in the future is related to the event that no additional fault is left in the software. In fact, the conditional probability mass function of the initial number of faults N, given that $i - 1$ failures have been experienced by time t, turns out to be

$$\begin{aligned} P(N = n \mid M(t) = i-1) &= \frac{P(M(t) = i-1 \mid N = n) \cdot P(N = n)}{\sum_{k=i-1}^{\infty} P(M(t) = i-1 \mid N = k) \cdot P(N = k)} \\ &= \frac{\binom{n}{i-1} G(t)^{i-1}[1 - G(t)]^{n-(i-1)} \cdot \frac{\nu^n}{n!} \cdot \exp(-\nu)}{\sum_{k=i-1}^{\infty} \binom{k}{i-1} G(t)^{i-1}[1 - G(t)]^{k-(i-1)} \cdot \frac{\nu^k}{k!} \cdot \exp(-\nu)} \\ &= \frac{[\nu(1 - G(t))]^{n-(i-1)}}{(n - (i-1))!} \exp(-\nu(1 - G(t))) \qquad \text{for } n \geq i-1. \end{aligned} \tag{10}$$

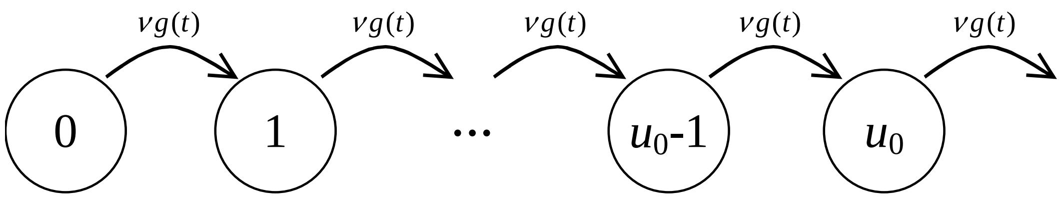

Figure 2: *The counting process connected to an NHPP-I model*

Hence, the conditional distribution of the number of faults remaining $N - M(t)$, given that $M(t) = i-1$, is Poisson with expected value $\nu(1-G(t))$. If the $(i-1)^{st}$ failure occurred at time t_{i-1}, then the conditional probability for the event that this failure was caused by the last of $i-1$ initial faults is

$$P(N = i-1 \mid M(t_{i-1}) = i-1) = \exp(-\nu(1-G(t_{i-1}))),$$

which is indeed identical to the limiting reliability in equation (9). This seems to corroborate our assumption that the defectiveness of the TTF distributions in NHPP-I models is linked to the possibility of no fault remaining in the software. In section 4 we will study how this insight can be used for mending TTF distributions.

3.3 Infinite failures category NHPP models

Kuo and Yang [14] introduced the term "NHPP-II" for infinite failures category [20, pp. 250–251] NHPP models. The models in this class share the property that $\mu(t)$ approaches infinity as $t \to \infty$. For these NHPP-II models Kuo and Yang showed that the mean value function can be written as

$$\mu(t) = -\ln[1 - H(t)], \tag{11}$$

where $H(t)$ is a non-defective distribution function. The failure times generated by such a model are the record values of independent outcomes with identical density function $h(t) = dH(t)/dt$.

Since $\mu(t)$ approaches infinity as $t \to \infty$, all TTF distributions are non-defective:

$$\lim_{x\to\infty} F_{X_i}(x) = 1 - \lim_{x\to\infty} \exp\left(-\mu(t_{i-1}+x) + \mu(t_{i-1})\right) = 1.$$

However, this does not necessarily mean that the expected values $E(X_i)$ are finite. A prominent example for this phenomenon is the Musa-Okumoto model [21], whose mean value function and failure intensity are given by

$$\mu(t) = \frac{1}{\theta}\ln(\lambda_0\theta t + 1) \tag{12}$$

and

$$\lambda(t) = \frac{\lambda_0}{\lambda_0\theta t + 1}, \tag{13}$$

respectively. In this model, only for $0 < \theta < 1$ the mean time to the i^{th} failure is finite:[2]

$$\begin{aligned} E(X_i) &= \int_0^\infty R(x \mid t_{i-1}, M(t_{i-1}) = i-1)\, dx = \int_0^\infty \left(\frac{\lambda_0\theta t_{i-1}+1}{\lambda_0\theta(t_{i-1}+x)+1}\right)^{1/\theta} dx \\ &= (\lambda_0\theta t_{i-1}+1)^{1/\theta} \cdot \left[\frac{(\lambda_0\theta(t_{i-1}+x)+1)^{1-1/\theta}}{\lambda_0(\theta-1)}\right]_0^\infty \overset{0<\theta<1}{=} \frac{\lambda_0\theta t_{i-1}+1}{\lambda_0(1-\theta)}. \end{aligned} \tag{14}$$

While Kuo and Yang used the generic mean value function (11) only for the unification of NHPP-II models, we find that taking defective distribution functions into account allows us to include NHPP-I models as well. According to equation (3) the relationship

$$R(t \mid 0, M(0) = 0) = \exp\left(-\mu(t)\right)$$

[2] Musa et al. [20, p. 291] correctly point out that the mean time to failure only exists for $\theta < 1$. However, their equation for calculating it in this case does not seem to be correct.

holds for all NHPP models. Consequently, $H(t)$ in equation (11) is nothing but the distribution function of the time to first failure:

$$H(t) = 1 - R(t \mid 0, M(0) = 0) = F_{X_1}(t).$$

This result shows that both NHPP-II models and NHPP-I models can be unified via the mean value function

$$\mu(t) = -\ln[1 - F_{X_1}(t)]. \tag{15}$$

If a non-defective TTF distribution $F_{X_1}(t)$ is plugged into this equation, then an NHPP-II model is obtained. A defective distribution $F_{X_1}(t)$, on the other hand, leads to an NHPP-I model.

4 Truncating Poisson distributions

4.1 Truncating the distribution of the number of inherent faults

From equation (9) we see that for an NHPP-I model at the beginning of testing the probability that even infinite testing will never lead to a failure is given by

$$\lim_{x \to \infty} R(x \mid 0, M(0) = 0) = \exp\left(-\nu\right). \tag{16}$$

According to equation (10) the conditional probability for no inherent software fault given that no failure has occurred at the beginning of testing is

$$P(N = 0 \mid M(0) = 0) = \frac{\nu^0}{0!} \exp(-\nu) = \exp(-\nu). \tag{17}$$

This is identical to the unconditional probability $P(N = 0)$, since $M(0) = 0$ with probability one.

The equality of (16) and (17) suggests that the defectiveness of the distribution of the time to first failure can be healed by removing the possibility that the number of inherent software faults is zero.

In a different context, Trivedi [25, p. 261] proposes to do this by left-truncating the distribution of N. The probability mass function of the zero-truncated Poisson distribution is

$$P(N = n) = \frac{\nu^n}{n!} \frac{\exp(-\nu)}{1 - \exp(-\nu)} = \frac{\nu^n}{n!} \frac{1}{\exp(\nu) - 1} \qquad \text{for } n = 1, 2, ..., \tag{18}$$

and its expected value is given by

$$E(N) = \sum_{n=1}^{\infty} n \cdot \frac{\nu^n}{n!} \frac{\exp(-\nu)}{1 - \exp(-\nu)} = \frac{\nu}{1 - \exp(-\nu)} > \nu. \tag{19}$$

Adopting this idea to our problem leads to the following reliability of the software in the interval $(0, x]$, bearing in mind that $M(0) = 0$:

$$\begin{aligned} R(x \mid 0, M(0) = 0) &= \sum_{n=1}^{\infty} [1 - G(x)]^n \cdot \frac{\nu^n}{n!} \frac{\exp(-\nu)}{1 - \exp(-\nu)} \\ &= \frac{\exp(-\nu)}{1 - \exp(-\nu)} \cdot \{\exp\left[\nu(1 - G(x))\right] - 1\} \\ &= \frac{\exp\left[\nu(1 - G(x))\right] - 1}{\exp\left(\nu\right) - 1}. \end{aligned} \tag{20}$$

Since this reliability expression approaches zero as $x \to \infty$, the defectiveness of the distribution of the time to first failure has indeed been mended.

Truncating the distribution of the number of inherent faults implicitly replaces the original transition rate from state 0 to state 1 given by (8) with the following one connected to the reliability function (20):

$$r_0(t) = \frac{-dR(t \mid 0, M(0) = 0)/dt}{R(t \mid 0, M(0) = 0)} = \frac{\nu g(t)}{1 - \exp\left[-\nu(1 - G(t))\right]}.$$

The transition rates between the other states of the counting process $\{M(t) \mid t \geq 0\}$ remain unchanged, however. This can be seen by studying the reliability of the software after the failure number $i-1 \geq 1$ has occurred at time t_{i-1}. The reliability in the interval $(t_{i-1}, t_{i-1}+x]$ is derived as

$$\begin{aligned}
&R(x \mid t_{i-1}, M(t_{i-1}) = i-1) \\
&\quad= P(M(t_{i-1}+x) - M(t_{i-1}) = 0 \mid M(t_{i-1}) = i-1) \\
&\quad= \frac{P(M(t_{i-1}+x) - M(t_{i-1}) = 0 \text{ and } M(t_{i-1}) = i-1)}{P(M(t_{i-1}) = i-1)} \\
&\quad= \frac{\sum_{n=i-1}^{\infty} \left(\frac{1-G(t_{i-1}+x)}{1-G(t_{i-1})}\right)^{n-(i-1)} \cdot \binom{n}{i-1} G(t_{i-1})^{i-1}[1-G(t_{i-1})]^{n-(i-1)} \cdot \frac{\nu^n}{n!} \cdot \frac{\exp(-\nu)}{1-\exp(-\nu)}}{\sum_{n=i-1}^{\infty} \binom{n}{i-1} G(t_{i-1})^{i-1}[1-G(t_{i-1})]^{n-(i-1)} \cdot \frac{\nu^n}{n!} \cdot \frac{\exp(-\nu)}{1-\exp(-\nu)}} \\
&\quad= \frac{\exp(\nu(1 - G(t_{i-1}+x)))}{\exp(\nu(1 - G(t_{i-1})))} \\
&\quad= \exp(-\nu G(t_{i-1}+x) + \nu G(t_{i-1})) \\
&\quad= \exp(-\mu(t_{i-1}+x) + \mu(t_{i-1})) \qquad \text{for } i-1 \geq 1.
\end{aligned} \tag{21}$$

This result is identical to equation (3), the reliability in the original NHPP model. Therefore, the transition rates $r_1(t), r_2(t), ...$ connected to equation (21) are the same as in (8),

$$\begin{aligned}
r_{i-1}(t) &= \frac{-dR(t - t_{i-1} \mid t_{i-1}, M(t_{i-1}) = i-1)/dt}{R(t - t_{i-1} \mid t_{i-1}, M(t_{i-1}) = i-1)} = \frac{\nu g(t) \exp(-\nu G(t) + \nu G(t_{i-1}))}{\exp(-\nu G(t) + \nu G(t_{i-1}))} \\
&= \nu g(t) \qquad \text{for } i-1 \geq 1.
\end{aligned}$$

Adapting the generic NHPP-I model with mean value function (5) by zero-truncating the distribution of N leads to a new family of SRGMs, which we will refer to as "first-stage truncated models". The counting processes connected to these models feature the common structure shown in figure 3.

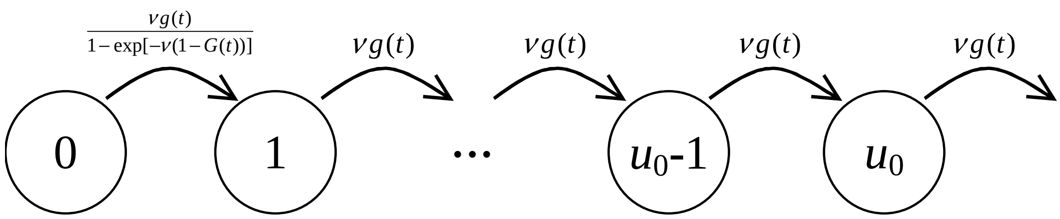

Figure 3: *The counting process connected to a first-stage truncated model*

Since $r_0(t)$ differs from all the other transition rates, the model family does not belong to the class of NHPP models, and $M(t)$ does not follow a Poisson distribution. Rather, the probability for $M(t) = 0$ is given by

$$P(M(t) = 0) = R(t \mid 0, M(0) = 0) = \frac{\exp\left[\nu(1 - G(t))\right] - 1}{\exp(\nu) - 1},$$

while the probabilities for $M(t)$ taking values greater than zero are

$$\begin{aligned} P(M(t) = m) &= \sum_{n=m}^{\infty} \binom{n}{m} G(t)^m [1 - G(t)]^{n-m} \cdot \frac{\nu^n}{n!} \frac{\exp(-\nu)}{1 - \exp(-\nu)} \\ &= \frac{(\nu G(t))^m}{m!} \cdot \frac{\exp(-\nu G(t))}{1 - \exp(-\nu)} \qquad \text{for } m \geq 1. \end{aligned}$$

From this probability mass function, we derive the generic mean value function of the first-stage truncated models as

$$\mu(t) = \sum_{m=0}^{\infty} m \cdot P(M(t) = m) = \sum_{m=1}^{\infty} m \cdot \frac{(\nu G(t))^m}{m!} \cdot \frac{\exp(-\nu G(t))}{1 - \exp(-\nu)} = \frac{\nu G(t)}{1 - \exp(-\nu)}.$$

Obviously, truncating the distribution of the number of inherent faults scales the original mean value function (5) by the factor $(1 - \exp(-\nu))^{-1} > 1$ for each value of t. Specifically, the expected number of failure occurrences after an infinite amount of testing is

$$\lim_{t \to \infty} \mu(t) = \frac{\nu}{1 - \exp(-\nu)},$$

which is exactly the same as the expected number of inherent faults (19) connected to the zero-truncated Poisson distribution.

Since the transition rates and reliability functions attached to the states 1, 2, ... of the counting process are not affected by the truncation, the distribution of the time to second, third, ... failure is still defective. In the following section, we investigate how the defectiveness of *all* TTF distributions can be mended.

4.2 Truncating the conditional distributions of the number of faults remaining

From section 4.1 we can see that the defectiveness of the distribution of the time to first failure in NHPP-I models is caused by the fact that as long as no failure has occurred - i.e., as long as the counting process resides in state 0 - it is possible that the software does not contain any fault at all. Truncating the Poisson distribution of N, the number of inherent faults, fixes this problem.

More generally, equation (10) tells us that the conditional distribution of $N - M(t) \mid M(t) = i - 1$ is Poisson. The meaning of this is as follows: The number of faults currently remaining in the software, calculated as the difference between the number of initial faults and the number of previous failure occurrences (the actual state of the counting process), follows a Poisson distribution. Since the Poisson distribution always assigns a non-zero probability to the value 0, after the correction of the $(i-1)^{st}$ fault there is a chance that the software is fault-free.

Left-truncating all the conditional distributions of $N \mid M(t) = i-1$ therefore seems to be a natural extension to the approach employed in the last section. The zero-truncated conditional distributions have the probability mass functions

$$\begin{aligned} P(N = n \mid M(t) = i-1) &= \frac{[\nu(1-G(t))]^{n-(i-1)}}{(n-(i-1))!} \cdot \frac{\exp(-\nu(1-G(t)))}{1-\exp(-\nu(1-G(t)))} \\ &= \frac{[\nu(1-G(t))]^{n-(i-1)}}{(n-(i-1))!} \cdot \frac{1}{\exp(\nu(1-G(t)))-1} \quad \text{for } i-1 \geq 0, n \geq i. \end{aligned} \tag{22}$$

For $i-1 = 0$ and $t = 0$, equation (22) specializes to the probability mass function of the zero-truncated (unconditional) distribution of N, equation (18). For $i-1 > 0$, as soon as the $(i-1)^{st}$ failure has been experienced the truncated conditional probability mass function (22) rules out the possibility that the number of inherent faults was merely $i-1$.

The reliability in the interval $(t_{i-1}, t_{i-1}+x]$ is then given by

$$\begin{aligned} &R(x \mid t_{i-1}, M(t_{i-1}) = i-1) \\ &= \sum_{n=i}^{\infty} P(M(t_{i-1}+x) - M(t_{i-1}) = 0 \mid N = n, M(t_{i-1}) = i-1) \cdot P(N = n \mid M(t_{i-1}) = i-1) \\ &= \sum_{n=i}^{\infty} \left(\frac{1-G(t_{i-1}+x)}{1-G(t_{i-1})}\right)^{n-(i-1)} \cdot \frac{[\nu(1-G(t_{i-1}))]^{n-(i-1)}}{(n-(i-1))!} \cdot \frac{1}{\exp(\nu(1-G(t_{i-1})))-1} \\ &= \frac{\exp(\nu(1-G(t_{i-1}+x)))-1}{\exp(\nu(1-G(t_{i-1})))-1} \quad \text{for } i-1 \geq 0. \end{aligned} \tag{23}$$

Regardless the previous number of failures $i-1$, reliability function (23) approaches zero for $x \to \infty$. Therefore, all distributions $F_{X_1}(x), F_{X_2}(x), ...$ are non-defective. Unlike the truncation of only the unconditional distribution of N, truncating each conditional distribution mends all TTF distributions. Moreover, since the truncation is carried out at each state of the counting process, the transition rates $r_0(t), r_1(t), ...$ connected to equation (23) are all identical:

$$\begin{aligned} r_{i-1}(t) &= \frac{-dR(t-t_{i-1} \mid t_{i-1}, M(t_{i-1}) = i-1)/dt}{R(t-t_{i-1} \mid t_{i-1}, M(t_{i-1}) = i-1)} \\ &= \frac{\nu g(t)}{1-\exp(-\nu(1-G(t))} \quad \text{for } i-1 \geq 0. \end{aligned} \tag{24}$$

The structure of the counting process related to the family of "all-stages truncated models" is shown in figure 4.

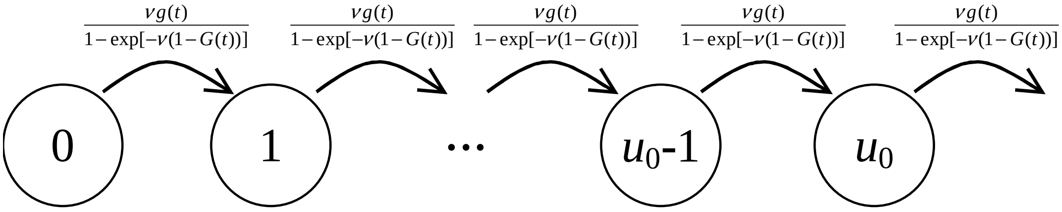

Figure 4: *The counting process connected to an all-stages truncated model*

This model family belongs to the class of NHPP models, because all transition rates are identical. The number of failure occurrences at time t, $M(t)$, follows a Poisson distribution with expected value

$$\mu(t) = -\ln(R(t \mid 0, M(0) = 0)) = \ln\left[\frac{\exp(\nu) - 1}{\exp[\nu(1 - G(t))] - 1}\right]. \tag{25}$$

Since $\mu(t) \to \infty$ for $t \to \infty$, the models are NHPP-II models. This result is not unexpected. The zero-truncated conditional probability mass functions (22) ensure that regardless the previous number of failure occurrences there is always at least one undiscovered fault remaining in the software. Due to the non-defectiveness of $G(t)$ each fault will eventually lead to a failure. Consequently, there is no upper bound for the expected number of failures to be experienced during infinite testing.

From the unifying mean value function (15) we can derive the family of all-stages truncated models by plugging in the generic non-defective distribution function of the time to first failure

$$F_{X_1}(t) = \frac{1 - \exp(-\nu G(t))}{1 - \exp(-\nu)}.$$

The structure of this distribution is similar to the one of the coverage function in the software reliability models with a time-varying testing-effort, cf. equation (7). However, while the latter one is defective, our time to first failure distribution is non-defective because of the normalizing denominator.

5 Defective TTF distributions in other models

In section 2 we have seen that the distribution of the time to the i^{th} failure is defective if equation (1) holds, i.e. if the area below the transition rate $r_{i-1}(t)$ is finite. Focusing on NHPP models, our investigations in section 3 have shown that due to the identity of all transition rates and the failure intensity the defectiveness of the TTF distributions is linked to the asymptotic behavior of the mean value function: All TTF distributions are defective for NHPP-I models, while they are all proper for NHPP-II models. In this section we will briefly discuss in which other sub-classes of CTMC models defective TTF distributions may occur. Our classification criteria are the time-dependence and/or state-dependence of the transition rates on the one hand and the fact whether a model belongs to the finite-failures category or the infinite-failures category on the other hand. (Models in which the transition rates are neither time- nor state-dependent are too simplistic to model software reliability growth, and we therefore omit them.) In figure 5 sub-classes containing models with at least one (non-trivially) defective distribution are shaded in gray. Moreover, examples of models are listed in italics. The class of NHPP models, covered in sections 3 and 4, is shown on the left-hand side of the figure.

Let us proceed with those models for which the transition rates are not merely *time-dependent* (like for the NHPP models), but *also state-dependent*. Here the asymptotic behavior of the mean value function does not determine the defectiveness of the TTF distributions.

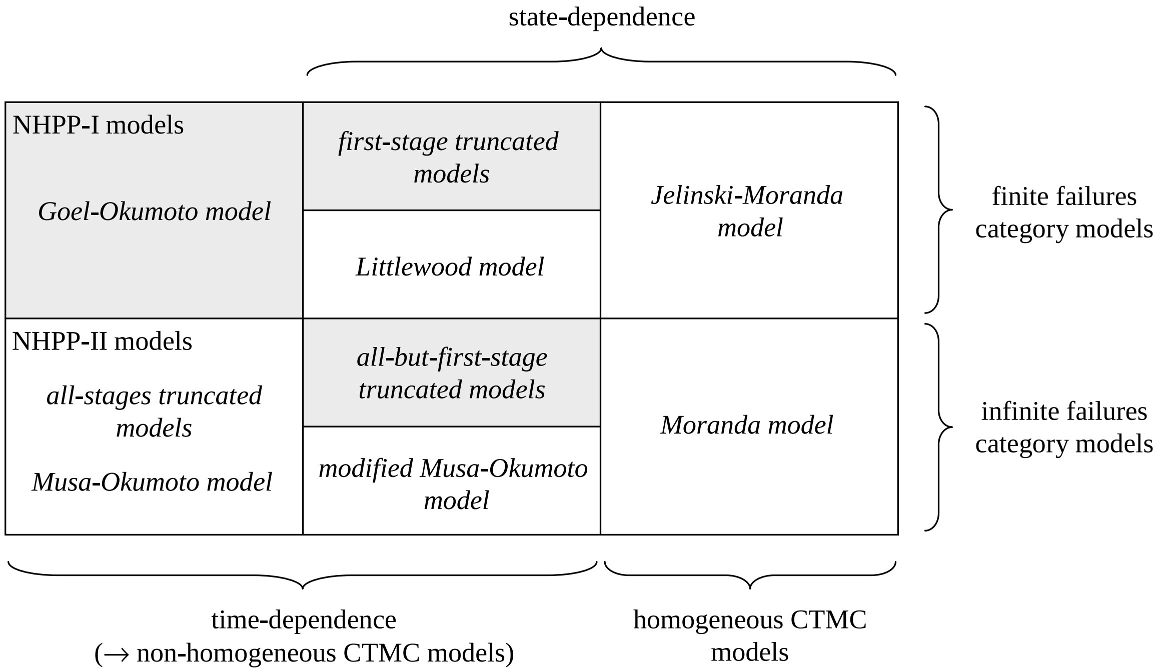

Figure 5: *Classification of CTMC software reliability growth models*

First of all, while *some* TTF distributions of finite failures category models belonging to this class may be defective, this is not necessarily true for *all* TTF distributions. An example for such models is the family of first-stage truncated models derived in section 4.1.

Moreover, it is even possible that all TTF distributions of a finite failures category model are proper, as the example of the Littlewood model [16] shows. This model proposes that the software initially contains u_0 faults, where u_0 is a fixed but unknown integer value. All of these faults have time-independent hazard rates that are independently sampled from the same Gamma(α, β) distribution. These assumptions entail the time- and state-dependent transition rates

$$r_{i-1}(t) = (u_0 - (i-1)) \cdot \frac{\alpha}{\beta + t} \qquad \text{for } 0 \leq i - 1 \leq u_0 - 1.$$

For these transition rates equation (1) is not satisfied, and therefore the distributions of X_1, X_2, ..., X_{u_0} are proper. The transition rate $r_{u_0}(t)$ is constant at zero, which means that the entire probability mass of the distribution of X_{u_0+1} is attached to infinity. However this defectiveness is trivial and can already be seen from the structure of the counting process: The Littlewood model is one of those models for which the CTMC representing the counting process terminates at the absorbing state u_0, see figure 1.

Time- and state-dependent CTMC models belonging to the infinite failures category are not very common. However, it is not difficult to construct examples in order to prove that such models may or may not feature defective distributions, just like those models of the finite failures category.

"Inverting" the structure of the first-stage truncated models in figure 3 by setting the transition rate out of state 0 to

$$r_0(t) = \nu g(t)$$

and all other transition rates to

$$r_1(t) = r_2(t) = ... = \frac{\nu g(t)}{1 - \exp[-\nu(1 - G(t))]},$$

where $G(t)$ is again a non-defective distribution function and $g(t)$ is its derivative, results in a model in which both the failure intensity

$$\begin{aligned}
\lambda(t) &= \nu g(t) \cdot P(M(t) = 0) + \frac{\nu g(t)}{1 - \exp[-\nu(1 - G(t))]} \cdot P(M(t) > 0) \\
&= \nu g(t) \exp(-\nu G(t)) + \frac{\nu g(t)[1 - \exp(-\nu G(t))]}{1 - \exp[-\nu(1 - G(t))]} = \frac{\nu g(t)(1 - \exp(-\nu))}{1 - \exp[-\nu(1 - G(t))]}
\end{aligned}$$

and mean value function

$$\mu(t) = \int_0^t \lambda(y)\, dy = (1 - \exp(-\nu)) \cdot \ln \left[\frac{\exp(\nu) - 1}{\exp[\nu(1 - G(t))] - 1} \right]$$

are scaled versions of the respective functions attached to the family of all-stages truncated models. Obviously, $\mu(t)$ approaches infinity for $t \to \infty$. Moreover, our previous analyses have shown that the TTF distribution related to the transition rate $r_0(t)$ is defective, while this is not the case for all other TTF distributions. Therefore, this generic "all-but-first-stage truncated model" belongs to the infinite failures category and contains exactly one defective distribution.

An infinite failures category model in which all TTF distributions are proper can be derived from the Musa-Okumoto model by setting

$$r_0(t) = \frac{\lambda_0 \theta}{\lambda_0 \theta t + 1}.$$

The reliability in the interval $(0, x]$ implied by this transition rate,

$$R(x \mid 0, M(0) = 0) = \exp\left(-\int_0^x \frac{\lambda_0 \theta}{\lambda_0 \theta y + 1}\, dy \right) = \frac{1}{\lambda_0 \theta x + 1},$$

approaches zero for $x \to \infty$. Therefore, the distribution of the time to first failure is not defective. All other transition rates are kept identical to the failure intensity (13) of the original model. Since the Musa-Okumoto model is an NHPP-II model, the TTF distributions related to these transition rates are proper as well. For the modified model the failure intensity becomes

$$\begin{aligned}
\lambda(t) &= \frac{\lambda_0 \theta}{\lambda_0 \theta t + 1} \cdot P(M(t) = 0) + \frac{\lambda_0}{\lambda_0 \theta t + 1} \cdot P(M(t) > 0) \\
&= \frac{\lambda_0 \theta}{(\lambda_0 \theta t + 1)^2} + \frac{\lambda_0^2 \theta t}{(\lambda_0 \theta t + 1)^2} = \frac{\lambda_0 \theta + \lambda_0^2 \theta t}{(\lambda_0 \theta t + 1)^2},
\end{aligned}$$

which leads to the mean value function

$$\mu(t) = \frac{1}{\theta} \ln(\lambda_0 \theta t + 1) + \frac{\lambda_0 \theta t - \lambda_0 t}{\lambda_0 \theta t + 1}.$$

As expected, this modified Musa-Okumoto model is indeed of the infinite failures category.

The last class of CTMC models to be discussed contains those models for which the transition rates are *merely state-dependent*. Examples include both finite failures category models like the well-known Jelinski-Moranda model [11] and infinite failures category models like Moranda's geometric model [17]. Due to the time-homogeneity all these models share the common property that all transition rates are constant over time. As a consequence, for each failure *that can occur at all* equation (1) does not hold, and the TTF distribution is non-defective. The italicized qualification in the last sentence is required in order to allow for the fact that homogeneous CTMC models of the finite failures category necessarily feature an absorbing state at which the Markov chain terminates. As seen in the discussion of the Littlewood model, for the time out of this state the entire probability mass is allocated to infinity.

6 A specific all-stages truncated model and its application

The derivation of the all-stages truncated models in section 4.2 is valid for any (non-defective) coverage function $G(t)$. Therefore, $G(t)$ and consequently the initial NHPP-I model have not been specified so far. In this section, we apply our approach to the well-known Goel-Okumoto model [5] and show how to estimate the parameters of the resulting all-stages truncated model. We then employ this model for fitting and predicting a classic failure data set, and we compare its performance to the one of the original Goel-Okumoto model and the Musa-Okumoto model.

6.1 The truncated Goel-Okumoto model

The mean value function and the failure intensity of the NHPP-I model introduced by Goel and Okumoto [5] are

$$\mu(t) = \nu(1 - \exp(-\phi t)) \tag{26}$$

and

$$\lambda(t) = \nu\phi \exp(-\phi t), \tag{27}$$

respectively, implying the non-defective coverage function (6). Plugging equation (6) into equation (25), we obtain the mean value function of the all-stages truncated Goel-Okumoto model (in the following referred to as the "truncated Goel-Okumoto model"):

$$\mu(t) = \ln\left[\frac{\exp(\nu) - 1}{\exp[\nu\exp(-\phi t)] - 1}\right]. \tag{28}$$

Its derivative with respect to time, the failure intensity, is

$$\lambda(t) = \frac{\nu\phi\exp(-\phi t)}{1 - \exp[-\nu\exp(-\phi t)]}. \tag{29}$$

From (3) and (28), the reliability in the interval $(t_{i-1}, t_{i-1} + x]$ is derived as

$$\begin{aligned} R(x \mid t_{i-1}, M(t_{i-1}) = i - 1) &= \exp\left(-\mu(t_{i-1} + x) + \mu(t_{i-1})\right) \\ &= \frac{\exp(\nu\exp(-\phi(t_{i-1} + x))) - 1}{\exp(\nu\exp(-\phi t_{i-1})) - 1}, \end{aligned} \tag{30}$$

which approaches zero for $x \to \infty$. Thus, all TTF distributions are non-defective. Moreover, it can be shown that all mean times to failure are finite: The mean time to the i^{th} failure implied by the truncated Goel-Okumoto model is

$$\begin{aligned}
E(X_i) &= \int_0^\infty R(x \mid t_{i-1}, M(t_{i-1}) = i-1)dx = \int_0^\infty \frac{\exp(\nu\exp(-\phi(t_i + x))) - 1}{\exp(\nu\exp(-\phi t_i)) - 1}\,dx \\
&= \int_0^{\exp(-\phi t_{i-1})} \frac{\exp(\nu z) - 1}{\exp(\nu\exp(-\phi t_{i-1})) - 1} \cdot \frac{dz}{\phi z} \\
&= \frac{1}{\phi[\exp(\nu\exp(-\phi t_{i-1})) - 1]} \int_0^{\exp(-\phi t_{i-1})} \frac{1}{z} \sum_{j=1}^{\infty} \frac{(\nu z)^j}{j!}\,dz \\
&= \frac{1}{\phi[\exp(\nu\exp(-\phi t_{i-1})) - 1]} \sum_{j=1}^{\infty} \frac{(\nu\exp(-\phi t_{i-1}))^j}{j \cdot j!} \qquad \forall\ i. \qquad (31)
\end{aligned}$$

The transition to line two is done via the substitution $z := \exp(-\phi(t_{i-1} + x))$. The sum in the last line of the equation converges to a finite value, as can be seen by comparing it to the Taylor series expansion of the exponential function. This means that for each failure $i = 1, 2, ...$ the mean time to failure is finite. Since the summands vanish rather quickly, the mean time to failure can easily be calculated based on equation (31). However, it is interesting to note that by a two-fold approximation we find:

$$\begin{aligned}
E(X_i) &= \frac{1}{\phi[\exp(\nu\exp(-\phi t_{i-1})) - 1]} \sum_{j=1}^{\infty} \frac{(\nu\exp(-\phi t_{i-1}))^j}{j \cdot j!} \\
&\approx \frac{1}{\nu\phi\exp(-\phi t_{i-1})[\exp(\nu\exp(-\phi t_{i-1})) - 1]} \sum_{j=1}^{\infty} \frac{(\nu\exp(-\phi t_{i-1}))^{j+1}}{(j+1)!} \\
&\approx \frac{1}{\nu\phi\exp(-\phi t_{i-1})} \qquad \forall\ i. \qquad (32)
\end{aligned}$$

The mean time to the i^{th} failure is roughly the reciprocal of the hazard rate (or, equivalently, the failure intensity) of the original Goel-Okumoto model, equation (27), evaluated at the time of the $(i-1)^{st}$ failure occurrence.

Maximum likelihood estimation (MLE) can be employed for calculating point estimates of the two model parameters ν and ϕ. Based on the m_e failure times $t_1, t_2, ..., t_{m_e}$ collected while testing the software from time 0 to t_e (where t_e may be identical to or larger than t_{m_e}), for NHPP models the log-likelihood function to be maximized with respect to the parameter vector $\boldsymbol{\delta}$ takes the general form [20, p. 324]

$$\ln \mathcal{L}(\boldsymbol{\delta}; t_1, ..., t_{m_e}, t_e) = \sum_{i=1}^{m_e} \ln(\lambda(t_i)) - \mu(t_e). \qquad (33)$$

With equations (28) and (29) the log-likelihood of the truncated Goel-Okumoto model becomes

$$\begin{aligned}
\ln \mathcal{L}(\nu, \phi; t_1, ..., t_{m_e}, t_e) &= m_e \ln(\nu\phi) - \phi \sum_{i=1}^{m_e} t_i - \sum_{i=1}^{m_e} \ln\left[1 - \exp(-\nu\exp(-\phi t_i))\right] \\
&\quad + \ln[\exp(\nu\exp(-\phi t_e)) - 1] - \ln[\exp(\nu) - 1].
\end{aligned}$$

Its maximization entails the simultaneous solution of the equations

$$\frac{\partial \ln \mathcal{L}}{\partial \nu} = \frac{m_e}{\nu} - \sum_{i=1}^{m_e} \frac{\exp(-\phi t_i)}{\exp(\nu \exp(-\phi t_i)) - 1} + \frac{\exp(-\phi t_e)}{1 - \exp(-\nu \exp(-\phi t_e))} - \frac{1}{1 - \exp(-\nu)} = 0 \quad \text{and}$$

$$\frac{\partial \ln \mathcal{L}}{\partial \phi} = \frac{m_e}{\phi} - \sum_{i=1}^{m_e} t_i + \sum_{i=1}^{m_e} \frac{\nu t_i \exp(-\phi t_i)}{\exp(\nu \exp(-\phi t_i)) - 1} - \frac{\nu t_e \exp(-\phi t_e)}{1 - \exp(-\nu \exp(-\phi t_e))} = 0.$$

6.2 Numerical example

For illustrating the application of the truncated Goel-Okumoto model we use the "System 40" data set collected by Musa in the mid 1970s and available at the web site of the Data & Analysis Center for Software [3]. The data set consists of the wall-clock times of 101 failures experienced during the system test phase of a military application containing about 180,000 delivered object code instructions.

Estimation of the parameters of the truncated Goel-Okumoto model is carried out according to the procedure described in the last section. We also employ MLE for fitting the Goel-Okumoto model and the Musa-Okumoto model to the data set. This is done by maximizing the log-likelihood derived from combining equations (26), (27), (33) and (12), (13), (33), respectively. Figure 6 shows the development of the cumulative number of failure occurrences over time for System 40 as well as the mean value functions of the three models, with parameters estimated based on the complete data set. Obviously, the truncated Goel-Okumoto model does the best job in fitting the actual data. This is corroborated by the log-likelihood values attained by the three models, which are listed in table 1.

The maximum log-likelihood value achieved by a model during MLE can be viewed as a measure for the possibility that the data were generated by the respective model. Since adding parameters to a model cannot worsen its fit, selecting the "best" model based on the log-likelihood value would in general favor overtly complex models. Indeed, Akaike's [1] information criterion derived from the Kullback-Leibler distance essentially adjusts the log-likelihood value by penalizing for the number of model parameters. However, since all three models considered here contain two parameters, we can simply compare the log-likelihood values. The model ranking implied by table 1 coincides with the visual impression given by figure 6: The truncated Goel-Okumoto model attains the largest log-likelihood value and is therefore most capable in explaining the collected failure data; it is followed by the Musa-Okumoto model and the original Goel-Okumoto model.

Table 1: *Log-likelihood values of the fitted models*

Model	log-likelihood value
Goel-Okumoto model	-1282.362
Musa-Okumoto model	-1251.290
truncated Goel-Okumoto model	-1239.508

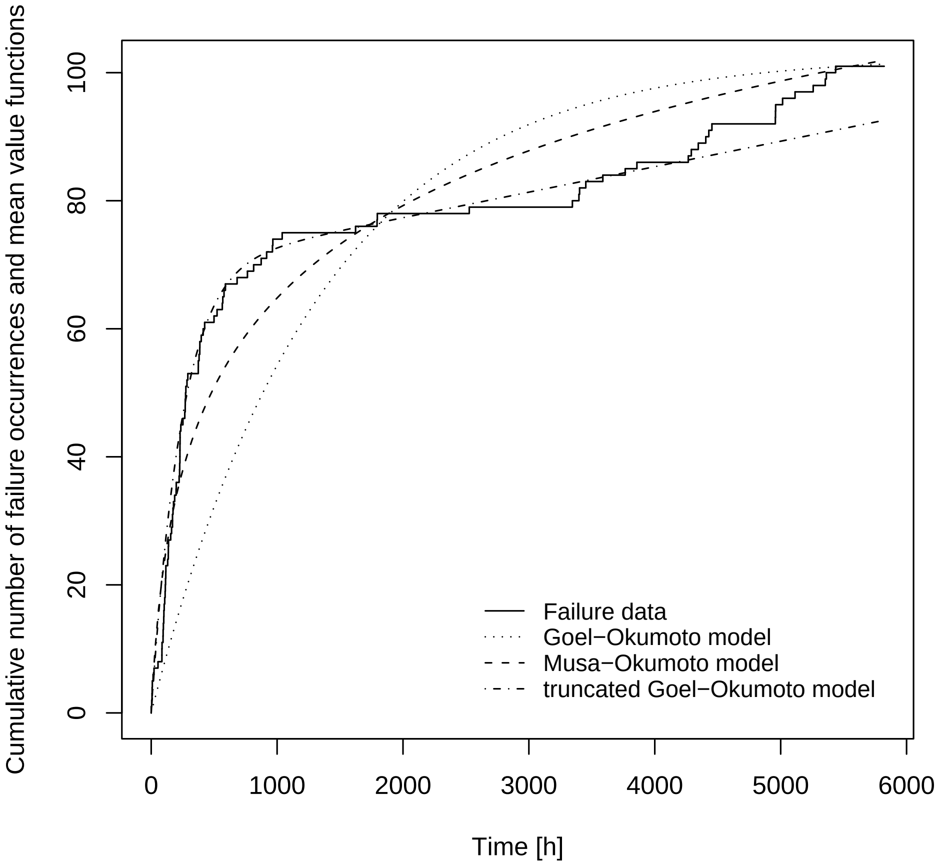

Figure 6: *Cumulative number of failure occurrences and mean value functions of the fitted models*

As shown in the last section, in the truncated Goel-Okumoto model all mean times to failure are finite. For this data set this is also the case for the Musa-Okumoto model, because the estimate of the parameter θ is smaller than one. We can therefore contrast the predicted mean times to failure according to both models with the failure data. For each model, we start out with the first five data points, estimate the model parameters and predict the time to the sixth failure based on the parameter estimates and the fifth failure time, using equations (14) and (31). This procedure is repeated, each time adding one data point, until the end of the data set is reached. The predicted mean times to next failure and the actual times to failure are depicted in figure 7.

The development in the predicted $E(X_i)$ values is quite similar for the two models. While the mean time to failure predictions of the truncated Goel-Okumoto model are slightly more optimistic, they seem to be less volatile than the ones of the Musa-Okumoto model. Moreover, the former model does not only respond to the long inter-failure times experienced by increasing the mean times to failure predictions (as the Musa-Okumoto model does), but it already predicts this increasing trend before the first TTF exceeding 100 hours is observed.

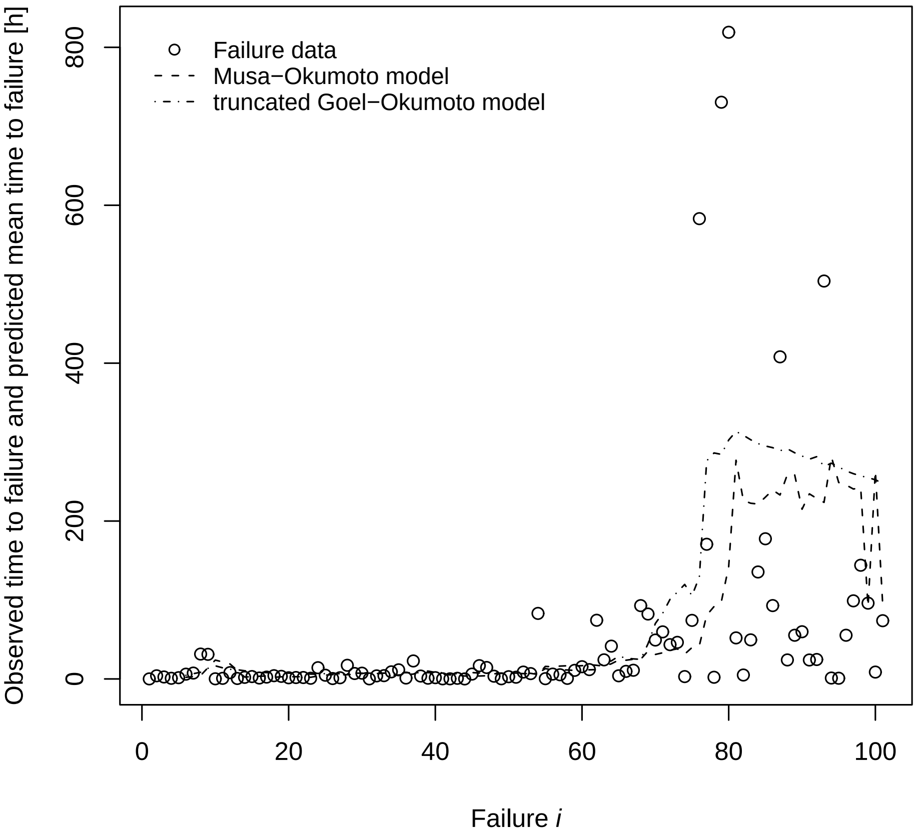

Figure 7: *Observed times to failure and predicted mean times to failure*

7 Conclusions

Defective time to failure distributions are often unrealistic, and they entail infinite mean times to failure, making this metric useless. In the course of our investigations, we have been able to answer the questions listed in the abstract: The i^{th} time to failure distribution is defective if the transition rate into state i decreases so quickly in time that the area below it is finite. While this can never happen for homogeneous CTMC models, it is possible for non-homogeneous ones. NHPP models are a special case of the latter, and due to the identity between all transition rates and the failure intensity the areas below the transition rates are related to the mean value function. If this function is bounded as t approaches infinity, i.e. for NHPP-I models, all time to failures distributions are defective. However, there is a generic approach with which an NHPP-I model can be transformed into an NHPP-II model. Its application to the Goel-Okumoto model has turned out to be both feasible and worthwhile, since it led us to a new SRGM with desirable properties, including all mean times to failure being finite.

References

[1] H. Akaike. Information theory and an extension of the maximum likelihood principle. In S. Kotz and N. L. Johnson, editors, *Breakthroughs in Statistics - Volume III*, pages 611–624. Springer, New York, 1992. (Reprint of the original 1973 paper).

[2] Y. Chen and N. D. Singpurwalla. Unification of software reliability models by self-exciting point processes. *Advances in Applied Probability*, 29:337–352, 1997.

[3] Data & Analysis Center for Software. The software reliability dataset. Available at `http://www.dacs.dtic.mil/databases/sled/swrel.shtml`. (Link verified on 2004-11-16).

[4] O. Gaudoin. *Outils statistiques pour l'évaluation de la fiabilité des logiciels*. Thèse de doctorat, Université de Joseph Fourier - Grenoble 1, Grenoble, 1990.

[5] A. L. Goel and K. Okumoto. Time-dependent error-detection rate model for software reliability and other performance measures. *IEEE Trans. Reliability*, 28:206–211, 1979.

[6] S. S. Gokhale and K. S. Trivedi. A time/structure based software reliability model. *Annals of Software Engineering*, 8:85–121, 1999.

[7] M. Grottke. *Modeling Software Failures during Systematic Testing - The Influence of Environmental Factors*. Shaker Verlag, Aachen, 2003.

[8] M. Grottke. Prognose von Softwarezuverlässigkeit, Softwareversagensfällen und Softwarefehlern. In P. Mertens and S. Rässler, editors, *Prognoserechnung*, pages 459–487. Physica, Heidelberg, 6th edition, 2005.

[9] C.-Y. Huang and S.-Y. Kuo. Analysis of incorporating logistic testing-effort function into software reliability modeling. *IEEE Trans. Software Engineering*, 51:261–270, 2002.

[10] C.-Y. Huang, S.-Y. Kuo, and I.-Y. Chen. Analysis of a software reliability growth model with logistic testing-effort function. In *Proc. Eighth International Symposium on Software Reliability Engineering*, pages 378–388, 1997.

[11] Z. Jelinski and P. Moranda. Software reliability research. In W. Freiberger, editor, *Statistical Computer Performance Evaluation*, pages 465–484. Academic Press, New York, 1972.

[12] C. Kaner, J. Falk, and H. Q. Nguyen. *Testing Computer Software*. Van Nostrand Reinhold, New York, 2nd edition, 1993.

[13] S. Kotz, N. L. Johnson, and C. B. Read. Improper distributions. In S. Kotz, N. L. Johnson, and C. B. Read, editors, *Encyclopedia of Statistics*, volume 4, pages 25–26. John Wiley & Sons, New York, 1983.

[14] L. Kuo and T. Y. Yang. Bayesian computation for nonhomogeneous Poisson processes in software reliability. *Journal of the American Statistical Association*, 91:763–773, 1996.

[15] J. Ledoux. Software reliability modeling. In H. Pham, editor, *Handbook of Reliability Engineering*, pages 213–234. Springer, London, 2003.

[16] B. Littlewood. Stochastic reliability growth: A model for fault-removal in computer-programs and hardware-design. *IEEE Trans. Reliability*, 30:313–320, 1981.

[17] P. B. Moranda. Event-altered reliability rate models for general reliability analysis. *IEEE Trans. Reliability*, 28:376–381, 1979.

[18] J. D. Musa. Operational profiles in software-reliability engineering. *IEEE Software*, pages 14–32, Mar. 1993.

[19] J. D. Musa. *Software Reliability Engineering*. McGraw-Hill, New York, 1998.

[20] J. D. Musa, A. Iannino, and K. Okumoto. *Software Reliability - Measurement, Prediction, Application*. McGraw-Hill Series in Software Engineering and Technology. McGraw-Hill, New York, 1987.

[21] J. D. Musa and K. Okumoto. A logarithmic Poisson execution time model for software reliability measurement. In *Proc. Seventh International Conference on Software Engineering*, pages 230–238, 1984.

[22] G. J. Myers. *The Art of Software Testing*. John Wiley & Sons, New York, 1979.

[23] H. Pham and X. Zhang. Software release policies with gain in reliability justifying the costs. *Annals of Software Engineering*, 8:147–166, 1999.

[24] N. D. Singpurwalla and S. P. Wilson. *Statistical Methods in Software Engineering - Reliability and Risk*. Springer Series in Statistics. Springer, New York, 1999.

[25] K. S. Trivedi. *Probability and Statistics with Reliability, Queuing, and Computer Science Applications*. John Wiley & Sons, New York, 2001.

[26] S. Yamada, J. Hishitani, and S. Osaki. Software-reliability growth with a Weibull test-effort. *IEEE Trans. Reliability*, 42:100–106, 1993.

[27] S. Yamada, H. Ohtera, and H. Narihisa. Software reliability growth models with testing-effort. *IEEE Trans. Reliability*, 35:19–23, 1986.