# Statistical approach to software reliability certification

# Statistical Approach to Software Reliability Certification

Isaac Corro Ramos, Alessandro Di Bucchianico, Kees van Hee

Department of Mathematics and Computer Science
Technische Universiteit Eindhoven
P.O.Box 513
5600 MB Eindhoven, The Netherlands
{i.corro.ramos,a.d.bucchianico,k.m.v.hee}@tue.nl

**Abstract**

We present a sequential software release procedure that certifies with some confidence level that the next error is not occurring within a certain time interval. Our procedure is defined in such a way that the release time is optimal for single stages and the global risk can be controlled. We assume that the failure detection process can be modeled as a General Order Statistics model where its unknown parameters are studied from a Bayesian point of view. We show how to implement our procedure and study its performance via simulation.

**Keywords:** software reliability, software release, software testing, Bayesian statistics, stopping time, sequential testing.

## 1 Introduction

Statistical procedures to support software release decisions are usually based on a loss function that in general considers the trade-off between the cost of extra testing and the cost of undetected errors. Most of the decision policies found in the literature are based on the optimization of such loss function (cf. Singpurwalla and Wilson (1999), Pham (2006)). Generally, two main decision policies are considered: policies based on observed errors (see e.g. Dalal and Mallows (1988), Morali and Soyer (2003), Özekici and Catkan (1993), Özekici and Soyer (2001), van Dorp et al. (1997)) and policies based on time (see e.g. McDaid and Wilson (2001), Singpurwalla (1991), Singpurwalla and Wilson (1994)). The first approach decides to stop or to continue testing after each error observation while the second one stops always after testing for some time period that is optimal for a certain criterion (no matter how many errors have been observed during such period). Unlike in the previous approaches, we do not base our decisions on a loss function but on a certification criterion. An early approach to certification procedures can be found in Currit et al. (1986). In that paper release decisions are based on the mean time to next failure. However, this approach fails to study certain software reliability growth models for which the mean time to failure is always infinite (see e.g. Goel-Okumoto model presented in Goel and Okumoto (1978)). We aim for a procedure that overcomes this restriction.

In this paper, we present a sequential software release procedure that certifies with some confidence level that the next error is not occurring within a certain time interval. Our procedure is developed assuming that the failure detection process can be modeled as a black box software reliability growth model (discrete or continuous), in particular we have chosen for this paper the Jelinski-Moranda model (cf. Jelinski and Moranda (1972)) and a discrete version of it. The decision of stopping or continuing testing is based on the error free period since the last error observation. Such time periods depend on the test history and, by choosing them properly, we can certify that the software is released in an optimal (local) time and the global risk taken in the procedure ( defined as the probability of stopping testing too early) can be controlled. Our

procedure can be seen as a special case of sequential testing one-stage look ahead policy (see e.g. Singpurwalla and Wilson (1999)) where at each stage of testing we update our statistics with the information obtained from the test history and we decide whether to stop or to continue testing. The main difference with other one-stage policies is that we allow the software to be tested for the time period that is optimal for our certification criterion before stopping, which occurs only in case that no errors are found during such period. Otherwise, we continue testing, repeating this procedure in a dynamic fashion. This approach is similar to the one presented in Di Bucchianico et al. (2008) although their certification criterion (the software is error free) is different from ours. Our problem is studied from a Bayesian point of view but, unlike in Achcar and Brassolatti (1999), we do not focus on a full Bayesian analysis of the parameters of the model. Finally, we show how to implement our procedure and study its performance via simulation.

The rest of the paper is organized as follows. In Section 2 we establish our time model framework and present our release procedure. We implement and study the performance of our procedure for the Jelinski-Moranda model in Section 3. In Section 4 we discuss the results presented in this paper and future work. Finally, in Section 5 we show the proofs of some technical results presented in our paper.

# 2 Model and Release Procedure

## 2.1 Statistical Model

We assume that the failure detection process of a software system can be modeled as a General Order Statistics model. It is assumed that the times between failures can be defined as the differences between two consecutive order statistics. If we denote by $N$ the (unknown) initial number of errors in the system, then the failure times $T_1, \ldots, T_N$ are independent and identically distributed random variables. If we consider that the errors occur in certain order, then the $i^{th}$ order statistic, $T_{(i)}$, defined as

$$\min\{T_1, \ldots, T_N\} = T_{(1)} \leq T_{(2)} \leq \ldots \leq T_{(i)} \leq \ldots \leq T_{(N)} = \max\{T_1, \ldots, T_N\}$$

defines precisely the $i^{th}$ error discovery. Since in our framework failure times are always cumulative and ordered we avoid the notation $T_{(i)}$ and we simply use $T_i$ to denote the corresponding failure time. Therefore, we consider a software system where $n \leq N$ errors have been observed at times $T_1 < T_2 < \ldots < T_n$ and define the time between failures as $X_i = T_i - T_{i-1}$, for all $i = 1, \ldots, n$, with $T_0 = 0$.

When we consider the underlying distribution of the failure times to be the exponential, the corresponding model is known as the Jelinski-Moranda model or the exponential order statistics model (cf. Jelinski and Moranda (1972)). If $\phi > 0$ denotes the parameter of the exponential distribution, then the Jelinski-Moranda model can be characterized by

$$f_{X_i}(x) = \phi(N - i + 1)e^{-\phi(N-i+1)x}\,, \tag{1}$$

where $f_{X_i}$ denotes the density function of the random variable $X_i$. Moreover, the times between failures $X_i$ are independent, for all $i = 1, \ldots, n$. Discrete software reliability growth models have not received as much attention as continuous-time models, being Fries and Sen (1996), Okamura et al. (2004), Yamada et al. (1986) some of the exceptions. A natural discrete version of the Jelinski-Moranda model appears when one considers geometric order statistics. In this case, if $p > 0$ denotes the parameter of the geometric distribution, then we can characterized this discrete model by

$$q_{X_i}(x) = p(N - i + 1)(1 - p)^{(N-i+1)x-1}\,, \tag{2}$$

where $q_{X_i}$ denotes the probability mass function of the random variable $X_i$. In our case, we are not interested in a procedure based on the number of (remaining) errors in the software but we rather concentrate on the survival probability of the system. For that reason, we can interpret the parameter $N$ as an upper bound for the total number of errors (for example, we could consider

the number of statements in a program as an upper bound for $N$). Therefore, we can think of $N$ as if it were known.

Next we present our certification procedure for General Order Statistics models and study it with more detail for the two models described before.

## 2.2 Software Release Procedure

Our goal is to certify with high confidence that the next software error is not occurring within a certain time interval, say $\Delta > 0$. Suppose that at certain stage of testing we have discovered and repaired $n \geq 0$ software errors. Thus, if $X^{(n)}$ denotes the random vector $(X_1, \ldots, X_n)$ and $x^{(n)}$ denotes its realization, then the conditional reliability function for the next stage is defined by

$$R_{n+1}(y, x^{(n)}) = \mathbb{P}_\phi \left[ X_{n+1} \geq y \mid X^{(n)} = x^{(n)} \right] , \tag{3}$$

for all $y > 0$. Since $n$ errors were found, we define the *one step risk* taken at stage $n+1$ as follows

$$S_{n+1}(y, x^{(n)}) = \mathbb{P}_\phi \left[ X_{n+1} \geq y + \Delta \mid X_{n+1} \geq y, X^{(n)} = x^{(n)} \right] . \tag{4}$$

Note that the one step risk is just the probability of surviving $\Delta$ given that the system has already survived some time $y > 0$ after $n \geq 0$ error discoveries. If we fix a reliability level $1 - \delta$ for some $\delta > 0$, then our release procedure consists on finding the error free period after the $n^{th}$ error detection, denoted by $b_{n+1}$ (which depends on the test history $x^{(n)}$), such that the one step risk at stage $n + 1$ is greater than or equal to $1 - \delta$. Besides the existence of $b_{n+1}$, we will prove in Section 2.3.1 that it is a local optimal release time for our procedure. Thus, our release procedure can be seen as a special case of sequential testing one-stage look ahead policy (see e.g. Singpurwalla and Wilson (1999)) where after each error detection we update our statistics with the information obtained from the past and then, after observing the system for an optimal error free period, we decide to stop testing. Based on the monotonicity (with respect to $y$) of the one step risk function, we will show that the procedure should stop at the exact time where the reliability level $1 - \delta$ is reached. Moreover, the *global risk* taken in the procedure can also be controlled. If $b^{(n)}$ denotes the random vector $(b_1, \ldots, b_n)$ and $\Delta$ is given, we can define the global risk as

$$G_n(b^{(n)}) = \mathbb{P}_\phi \left[ \bigcup_{n \geq 1} (X_n \leq b_n + \Delta, X_1 \leq b_1, \ldots, X_{n-1} \leq b_{n-1}, X_n \geq b_n) \right] . \tag{5}$$

Note that (5) defines the probability of stopping testing too early since we want to survive at least $b_n + \Delta$ for all $n \geq 1$. In the next theorem we show that if the one step risk is bounded, then we can keep the global risk also under control. The proof can be found in Section 5.

**Theorem 2.1** (Global risk bound). *Suppose that $S_n(b_n, x^{(n-1)}) \geq 1 - \delta$ for some $\delta > 0$. Then $G_n(b^{(n)}) \leq \delta$, for all $n \geq 1$.*

In our procedure, the decision of releasing the software is based on the error free period since the last error observation. The following result concerns with the computation of such (conditional) survival probabilities.

**Lemma 2.2** (One step risk). *Let $X_1, \ldots, X_{n+1}$ be continuous random variables whose distributions depends on the non-negative random parameter $\Phi$. Suppose that the joint densities $f_{X_{n+1}, X_1, \ldots, X_n}$ and $f_{X_{n+1}, X_1, \ldots, X_n, \Phi}$ exist. If $X_{n+1}$ is conditionally independent of $X_1, \ldots, X_n$ given $\Phi$, then*

$$R_{n+1}(y, x^{(n)}) = \int_0^\infty R_{n+1}(y, \phi)\ f_\Phi(\phi \mid X^{(n)} = x^{(n)})\ d\phi$$

*and*

$$S_{n+1}(y, x^{(n)}) = \frac{\displaystyle\int_0^\infty R_{n+1}(y + \Delta, \phi)\ f_\Phi(\phi \mid X^{(n)} = x^{(n)})\ d\phi}{\displaystyle\int_0^\infty R_{n+1}(y, \phi)\ f_\Phi(\phi \mid X^{(n)} = x^{(n)})\ d\phi}$$

*for all* $y > 0$, *where* $R_{n+1}(y, \phi) = \mathbb{P}\left[X_{n+1} \geq y \mid \Phi = \phi\right]$ *and* $f_\Phi(\phi \mid X^{(n)} = x^{(n)})$ *is the posterior density function of* $\Phi$.

We can find a proof of this lemma in Section 5. Note that conditional independency is required. For its definition and some characterizations we refer to Lauritzen (1996).

Next we study the one step risk function for the Jelinski-Moranda and the geometric order statistics models.

## 2.3 One Step Risk

### 2.3.1 Jelinski-Moranda Model

Suppose that the failure detection process of a software system can be described by the Jelinski-Moranda model where $N$ represents an upper bound for the total number of errors in the system. Then, for $\Phi = \phi$, the density function of the times between failures is given by (1). We assume that $\Phi$ has a prior gamma distribution with parameters $k > 0$ and $w > 0$ while $N$ is supposed to be known and fixed. We have chosen this prior since it is a conjugate prior for exponential likelihoods. Therefore, the posterior distribution of $\Phi$ is also gamma with parameters $n + k > 0$ and $\left(\frac{1}{w} + \sum_{i=1}^{n}(N - i + 1)x_i\right)^{-1}$ (see e.g. Lee (1989)). In this case, the one step risk as given in Lemma 2.2 has the following closed-form (cf. Singpurwalla (1991))

$$S_{n+1}(y, x^{(n)}) = \left(\frac{\sum_{i=1}^{n}(N - i + 1)x_i + \frac{1}{w} + (N - n)\, y}{\sum_{i=1}^{n}(N - i + 1)x_i + \frac{1}{w} + (N - n)(y + \Delta)}\right)^{n+k} . \tag{6}$$

The next lemma shows two important monotonicity properties of the one step risk. For its proof we refer to Section 5.

**Lemma 2.3.** *The one step risk* $S_{n+1}(y, x^{(n)})$ *given in (6) is monotone increasing with respect to* $y$ *and monotone decreasing with respect to* $N$.

We have defined our procedure in such a way that it stops as soon as $S_{n+1}(y, x^{(n)}) \geq 1 - \delta$. Since $S_{n+1}(y, x^{(n)})$ is monotone increasing with respect to $y$, then there exists a unique $b_{n+1} > 0$ (which depends on $x^{(n)}$) such that $S_{n+1}(b_{n+1}, x^{(n)}) = 1 - \delta$. Thus, $b_{n+1}$ is an optimal one step stopping time. Although the existence of $b_{n+1}$ is proved, it may occur that to wait for it would not be feasible in practice. For example, we may fix a value of $\Delta$ big enough to make $b_{n+1}$ unrealistically large to wait for it. This phenomenon will be explained in Section 3. On the other hand, the one step risk is monotone decreasing with respect to $N$. This confirms what one may expect: a system with more errors is less reliable. In this case, to consider a worst case scenario (lowest one step risk) would mean to take $N = \infty$. However, for such case our model is not defined. This supports our assumption of considering $N$ fixed as an upper bound for the initial number of errors.

### 2.3.2 Geometric Order Statistics Model

Suppose now that a software system is subjected to series of test runs until an error is found. Then, for $\Phi = p$ fixed, the probability mass function of the times between failures is given by (2). We assume that $\Phi$ has a prior beta distribution with parameters $\gamma > 0$ and $\lambda > 0$. We have chosen the beta distribution as prior because it is a conjugate prior. Moreover, the posterior distribution of $\Phi$ is a beta distribution with parameters $n + \gamma > 0$ and $\lambda + \sum_{i=1}^{n}((N - i + 1)x_i - 1) > 0$ (cf. Lee (1989)). In this case, the one step risk has no simple closed-form like in the Jelinski-Moranda model. First note that

$$R_{n+1}(y, p) = \mathbb{P}\left[X_{n+1} \geq y \mid \Phi = p\right] = (N - n)\, p\, \frac{(1 - p)^{(N-n)y-1}}{1 - (1 - p)^{N-n}} . \tag{7}$$

Then, by Lemma 2.2

$$R_{n+1}(y, x^{(n)}) = c \int_0^1 p^{n+\gamma}\, \frac{(1 - p)^{(N-n)y-2+\lambda+\sum_{i=1}^{n}((N-i+1)x_i-1)}}{1 - (1 - p)^{N-n}}\, dp , \tag{8}$$

where $c = \frac{(N-n)}{\mathrm{B}\left(n+\gamma, \lambda+\sum_{i-1}^{n}((N-i+1)x_i-1)\right)}$ does not depend on $p$. Thus, the one step risk in Lemma 2.2 is given by

$$S_{n+1}(y, x^{(n)}) = \frac{\displaystyle\int_0^1 p^{n+\gamma} \frac{(1-p)^{(N-n)(y+\Delta)-2+\lambda+\sum_{i=1}^{n}((N-i+1)x_i-1)}}{1-(1-p)^{N-n}}\, dp}{\displaystyle\int_0^1 p^{n+\gamma} \frac{(1-p)^{(N-n)y-2+\lambda+\sum_{i=1}^{n}((N-i+1)x_i-1)}}{1-(1-p)^{N-n}}\, dp}\,. \tag{9}$$

We have already discussed in the continuous case the importance of the monotonicity of the one step risk in order to compute an optimal one step stopping time. The next lemma shows that the one step risk is also an increasing function of $y$ in this case. For its proof we refer to Section 5.

**Lemma 2.4.** *The one step risk $S_{n+1}(y, x^{(n)})$ given in (9) is monotone increasing with respect to $y$.*

Note that we have not included a similar result about monotonicity with respect to $N$. Although one may expect the same behaviour as in the continuous case, the analytical proof does not seem to be as straightforward as before. Since we do not consider this an important issue for the purposes of this paper, we have decided to skip it.

# 3 Simulation

In this section we study the performance of our procedure for the Jelinski-Moranda model. The distribution of the times between failures is determined by the parameters $\phi$ and $N$ introduced in (1). Moreover, we have to take into account two more parameters, namely the time period $\Delta$ to be certified as error free (see the one step risk definition in (4)) and the reliability level $1-\delta$ we wish to reach in our procedure. Finally, we also consider the parameter *ntests* defining the number of tests to be performed. Although we are interested in a Bayesian approach to our problem, we first study the case where $\phi$ is suppose to be known since some interested conclusions can also be extracted from there.

## 3.1 Known parameter $\phi$

Suppose that $\phi$ is known and fixed. This is not completely unrealistic since we may have some previous information based on similar systems behaviour observed before. In this case, and due to the memoryless property of the exponential distribution, the one step risk after $n < N$ error detections is given by

$$D_{n+1}(\phi) = e^{-\phi(N-n)\Delta}\,. \tag{10}$$

Note that the one step risk does not depend on $y$ (there is no $b_{n+1}$ to be computed) and the only aspect of the test history taken into account is the number of errors discovered so far (but not when).

We first study which values of $\Delta$ are reasonable choices for our problem. The one step risk is increasing as soon as the number of remaining errors in the system decreases. Therefore, except for the case where the system is error free, the situation with the highest reliability corresponds to only one error left. Thus, the maximum value of $\Delta$ to reach certain reliability level can easily be computed from (10) when $n = N-1$. Some of these $\Delta$ values are shown in Table 1 for different values of $\phi$ when the reliability level $1-\delta = 0.90$. It is straightforward to deduce from Table 1 that

| | $\phi = 0.00025$ | $\phi = 0.0005$ | $\phi = 0.001$ | $\phi = 0.002$ | $\phi = 0.004$ |
|---|---|---|---|---|---|
| $\max \Delta$ | 421.44 | 210.72 | 105.36 | 52.68 | 26.34 |

Table 1: Maximum feasible $\Delta$ when the reliability level is 0.90 for different values of $\phi$.

the election of $\Delta$ must be done carefully since there exist many values of $\Delta$ for which certification is impossible.

Given a feasible $\Delta$ we are interested in computing the mean number of errors as a fraction of $N$ to be observed before we stop testing. We calculate this for different values of $\Delta$ and $N$ when $\phi = 0.0005$, $1 - \delta = 0.90$ and $ntests = 1000$. The results are shown in Table 2. The

| | $\Delta = 10$ | $\Delta = 25$ | $\Delta = 50$ | $\Delta = 100$ | $\Delta = 150$ |
|---|---|---|---|---|---|
| $N = 10$ | 0.00 | 0.20 | 0.60 | 0.80 | 0.90 |
| $N = 25$ | 0.16 | 0.68 | 0.84 | 0.92 | 0.96 |
| $N = 50$ | 0.58 | 0.84 | 0.92 | 0.96 | 0.98 |
| $N = 100$ | 0.79 | 0.92 | 0.96 | 0.98 | 0.99 |

Table 2: Mean number of errors as a fraction of $N$ to be observed to reach reliability 0.90 for different values of $N$ and $\Delta$ when $\phi = 0.0005$ and $ntests = 1000$.

main conclusion we can extract is that for a fixed $\Delta$, the system containing more errors requires more effort (a bigger amount of errors has to be repaired) to certify $\Delta$ as error free period, or equivalently the more errors a system contains the smaller the $\Delta$ admitted. In the next section we study the case where $\phi$ is a random variable and compare the results we obtain with the ones obtained here.

### 3.2 Random Parameter $\phi$

We now study our problem from a Bayesian point of view assuming that $\phi$ is a random variable following a gamma prior distribution. In this case, the one step risk after $n$ error detections is given by (6). Unlike in the deterministic case, the one step risk depends not only on the number of discovered errors but also on the observation times themselves.

We saw in the deterministic case that, for $\phi$ and $\delta$ fixed , there exists a maximum value of $\Delta$ such that if we ask for a $\Delta$ larger than this maximum, then certification is impossible. This is not the case now since in theory any value of $\Delta$ is feasible. However, we will show later in this section what the cost of choosing a high $\Delta$ is. In addition to the mean number of errors as a fraction of $N$ to be observed before we stop testing, denoted by $s$, there are some more interesting metrics that we compute. We denote by $b_s$ the mean last error free period before we stop testing, $\bar{b}$ is the mean error free period after error detection, $\hat{\delta}$ is the observed reliability and finally $ttt$ denotes the mean total time of test. These metrics have to be computed for some values of $N$ and $\Delta$. Table 3 shows the results when $\phi$ has been sampled from a gamma distribution with mean 0.0005, $N = 50$, $1 - \delta = 0.90$ and $ntests = 1000$.

| | $s$ | $b_s/\Delta$ | $\hat{\delta}$ | $\bar{b}$ | $ttt$ |
|---|---|---|---|---|---|
| $\Delta = 10$ | 0.34 | 4.51 | 0.95 | 22.79 | 825.66 |
| $\Delta = 25$ | 0.68 | 2.81 | 0.92 | 1481.72 | 2953.44 |
| $\Delta = 50$ | 0.83 | 1.43 | 0.89 | 5585.98 | 4984.53 |
| $\Delta = 100$ | 0.90 | 0.68 | 0.87 | 16385.90 | 7338.40 |
| $\Delta = 150$ | 0.93 | 0.26 | 0.85 | 28058.15 | 8290.67 |
| $\Delta = 200$ | 0.92 | 0.18 | 0.86 | 39382.70 | 9039.62 |
| $\Delta = 250$ | 0.87 | 0.20 | 0.77 | 49591.35 | 9246.99 |

Table 3: Mean number of errors as a fraction of $N$ to be observed before we stop testing ($s$), mean last error free period as a fraction of $\Delta$ before we stop testing ($b_s/\Delta$), observed reliability ($\hat{\delta}$), mean error free period after error detection ($\bar{b}$ ) and mean total time of test ($ttt$) for different values of $\Delta$ when the reliability level is fixed to 0.90 and $N = 50$.

We have chosen such a prior distribution in order to compare the results obtained in this section with those obtained in Table 2 (in the deterministic case we chose $\phi = 0.0005$). In particular, we have taken $k = 2$ and $w = 0.00025$. Note that the choice of the prior distribution determines the scale of the data in the problem (small values of $\phi$ produce large times between failures). However, the posterior distribution has no great effect since it does not deviate much from the prior. This can be seen in Figure 1. On its left-hand side we observe how the posterior mean

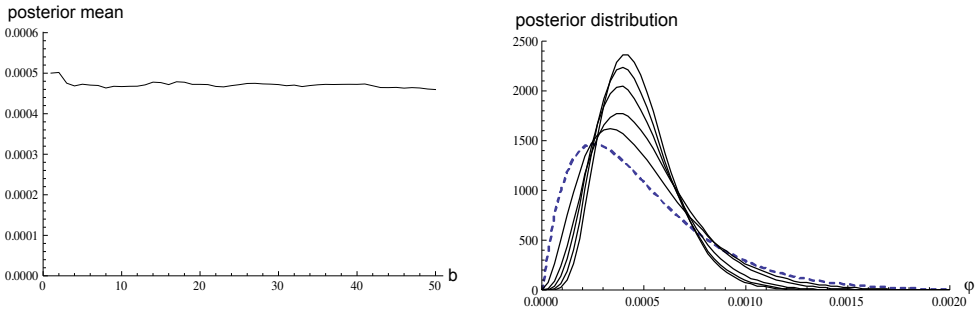


Figure 1: Posterior mean (left) and posterior distribution (right) when $\phi$ follows a prior (dashed) distribution with parameters $k = 2$, $w = 0.00025$ and $N = 50$.

remains close to the prior mean (0.0005) for all $n$. On the right-hand side we show the prior distribution (dashed) and the evolution of the posterior distribution after a few updates. Note how the posterior distribution becomes sharper near 0.0005 as long as new data is observed and a new posterior distribution is calculated.

We can conclude from Table 3 that choosing a large $\Delta$ is expensive. Increasing $\Delta$ produces an increase of the total time of test ($ttt$) and a decrease of reliability ($\hat{\delta}$). Thus, we would have to test longer to get a less reliable result. This behaviour is not surprising given the results obtained in the deterministic case. The increase of the value of $\bar{b}$ with $\Delta$ has an important interpretation. Although it is possible to provide a theoretical solution for our problem (there always exists a value of $b_i$ for which (6) is greater than or equal to $1 - \delta$, for all $i = 1, \ldots, N$), this is useless in practice since the values of $b_i$ are extremely large with respect to the scale of the data. This can be observed in Figure 2 where we have plotted the mean error free period after error detection and the mean times between failures for $N = 50$ and $\Delta = 25$. Note that even for this reasonably

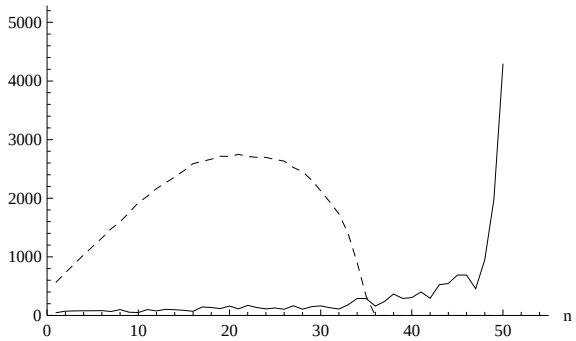


Figure 2: Mean error free period after error detection (dashed) and mean times between failures (solid), when $N = 50$, $\Delta = 25$ and $\phi$ follows a prior distribution with parameters $k = 2$ and $w = 0.00025$.

small value of $\Delta$ the difference of magnitude between the error free period after error detection

and the times between failures is extremely large in early stages of testing (making thus any value of $b_i$ unfeasible for such stages). Only when at the end of testing $b_i$ approaches the times between failures, we can use the computed value of $b_i$ in practice and at the time where the curves intersect we stop testing. Figure 2 also shows the typical behaviour of the error free periods after error detection. In early stages of testing, where only a few errors have been discovered and repaired, these are increasing and after a certain period they start to decrease to eventually become 0. This behaviour is not completely intuitive since one may expect that as long as the system is repaired the time you have to wait to release it becomes smaller. This can be explained as follows. We proved in Section 2.3.1 that the one step risk is monotone increasing with respect to $y$. We now study its behaviour with respect to the number of discovered errors $n$. After reparation of one error, the next one step risk function improves, i.e., $S_{n+1}(0, x^{(n)}) \geq S_n(0, x^{(n-1)})$, for all $n \geq 1$ (see Section 5 for a proof). However, this monotonicity with respect to $n$ is not kept for all $y > 0$ since some risks functions grow rapidly than others. This can be observed in Figure 3. On the left-hand

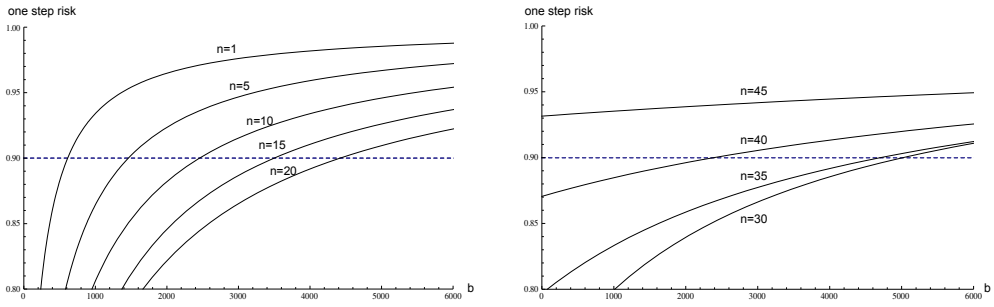


Figure 3: Left: one step risk functions for $n = 1$, $n = 5$, $n = 10$, $n = 15$ and $n = 20$: $b_n$ increases with respect to $n$. Right: one step risk functions for $n = 30$, $n = 35$, $n = 40$, and $n = 45$: $b_n$ decreases with respect to $n$.

side we observe the one step risk function for $n = 1$, $n = 5$, $n = 10$, $n = 15$ and $n = 20$. If $b_n$ denotes the values of $y$ for which $S_n(y, x^{(n-1)}) = 1 - \delta$, then we see that $b_1 < b_5 < b_{10} < b_{15} < b_{20}$. On the right-hand side we observe the one step risk for $n = 30$, $n = 35$, $n = 40$, and $n = 45$. In this case, we see that $b_n$ decreases for such values of $n$. If we plot the one step risk functions for all $n = 1, \ldots, 50$, we would observe precisely the behaviour of $b_n$ depicted in Figure 2 (increasing until $n = 21$ approximately and then decreasing).

In Table 4 we compare the mean number of errors as a fraction of $N$ to be observed before we stop testing for both Bayesian and deterministic procedures when the reliability level is fixed at 0.90. We observe that the values obtained using the Bayesian approach are smaller than those

| | $\Delta = 10$ | $\Delta = 25$ | $\Delta = 50$ | $\Delta = 100$ | $\Delta = 150$ |
|---|---|---|---|---|---|
| $\phi$ random | 0.34 | 0.68 | 0.83 | 0.90 | 0.93 |
| $\phi$ known | 0.58 | 0.84 | 0.92 | 0.96 | 0.98 |

Table 4: Mean number of errors as a fraction of $N$ to be observed before we stop testing for the Bayesian ($\phi$ random) and the deterministic ($\phi$ known) procedures for different values of $\Delta$ when the reliability level is fixed to 0.90 and $N = 50$.

obtained in the deterministic case for all $\Delta$. Therefore, we can conclude that in the Bayesian approach the decision of releasing the software requires less error discoveries than in the deterministic case. In this sense, the software can be released earlier. We prove in Section 5 a general result showing that the Bayesian approach performs always better than the deterministic one with respect to this metric.

As one may expect, an increase of the number of errors $N$ in the system has the same effect as in the deterministic case, i.e., the system containing more errors requires more effort to certify $\Delta$ as error free period. Moreover, the more errors we have the worse is our system. This will be reflected in an increase of the total time of test and a decrease of the observed reliability. Thus, we would have to test longer to get a less reliable result. Table 5 shows the results when $\phi$ has been sampled from a gamma distribution with mean 0.0005, $\Delta = 50$, $1 - \delta = 0.90$ and $ntests = 1000$. Note that Table 5 behaves in a similar way as Table 3. Thus, the conclusions that can be drawn

| | $s$ | $b_s/\Delta$ | $\hat{\delta}$ | $\bar{b}$ | $ttt$ |
|---|---|---|---|---|---|
| $N = 10$ | 0.35 | 1.81 | 0.93 | 128.52 | 685.07 |
| $N = 25$ | 0.67 | 2.35 | 0.91 | 1633.90 | 2776.39 |
| $N = 50$ | 0.83 | 1.43 | 0.89 | 5585.98 | 4984.53 |
| $N = 100$ | 0.90 | 0.90 | 0.89 | 15481.30 | 7488.62 |

Table 5: Mean number of errors as a fraction of $N$ to be observed before we stop testing ($s$), mean last error free period as a fraction of $\Delta$ before we stop testing ($b_s/\Delta$), observed reliability ($\hat{\delta}$), mean error free period after error detection ($\bar{b}$ ) and mean total time of test ($ttt$) for different values of $N$ when the reliability level is fixed to 0.90 and $\Delta = 50$.

in this case are like those obtained before. The only difference lies on the column $b_s/\Delta$. From Table 3 one may think that as long as $\Delta$ increases the last waiting period $b_s$ becomes smaller with respect to $\Delta$ which is indeed a desirable behaviour. However, this monotonicity does not completely hold for all values of $\Delta$ (see e.g. $\Delta = 250$). In Table 5 no real trend can be observed. Therefore, no conclusion can be drawn from this respect.

Finally, we study the effect of changing the reliability level of our procedure. In Table 6 we show the mean number of errors as a fraction of $N$ to be observed before we stop testing, the observed reliability and the mean total time of test for different reliability levels when $N = 50$, $\Delta = 50$ and $ntests = 1000$. We can conclude (as one may expect) that for lower reliability levels release

| | $s$ | $\hat{\delta}$ | $ttt$ |
|---|---|---|---|
| $1 - \delta = 0.99$ | 0.97 | 0.98 | 11869.90 |
| $1 - \delta = 0.95$ | 0.90 | 0.93 | 7185.98 |
| $1 - \delta = 0.90$ | 0.82 | 0.91 | 4926.88 |
| $1 - \delta = 0.75$ | 0.57 | 0.82 | 2021.48 |

Table 6: Mean number of errors as a fraction of $N$ to be observed before we stop testing ($s$), observed reliability ($\hat{\delta}$) and mean total time of test ($ttt$) for different reliability levels $1 - \delta$ when $N = 50$, $\Delta = 50$ and $ntests = 1000$.

can be done earlier, i.e., less errors need to be discovered and therefore less testing effort. On the other hand, when the reliability level is very high (see e.g. $1 - \delta = 0.99$) it seems more difficult to reach such level in practice and in this particular example it is not reached for $ntests = 1000$.

# 4 Conclusion and Future Work

We have presented a sequential software release procedure that certifies with a certain confidence level that the next software error is not occurring within a certain time interval. The main difference with current approaches is that we allow testing for a certain time period after the last error observation and then we release the software but only if we do not find any error in such period. We have shown that the software is released in an optimal (local) time and that the global risk taken in the procedure (the probability of stopping testing too early) remains under control.

We have studied our problem from a Bayesian point of view assuming that the failure detection process can be modeled as the Jelinski-Moranda and the geometric order statistics models. Finally, we have investigated the performance of our procedure via simulation for the Jelinski-Moranda model.

An additional step that could also be considered would be to investigate a full Bayesian approach to our problem. Moreover, we would like to extend this work studying the performance of our procedure for more complicated General Order Statistics models (like for example the Schick-Wolverton model where the underlying distribution is Weibull) and for some Non Homogeneous Poisson Process models (the other main family of software reliability growth models) to finally compare the results obtained in any case.

# 5 Appendix

• Proof of Theorem 2.1.

*Proof.* Denote the event $\{X_1 \leq b_1, \ldots, X_{n-1} \leq b_{n-1}\}$ by $\{\prod_{i=1}^{n-1} X_i \leq b_i\}$ and write the global risk as

$$G_n(b^{(n)}) = \mathbb{P}_\phi \left[ \bigcup_{n\geq 1} \left( X_n \leq b_n + \Delta, \prod_{i=1}^{n-1} X_i \leq b_i, X_n \geq b_n \right) \right] . \tag{11}$$

The times between failures are independent so do the events in the right-hand side of (11), therefore

$$G_n(b^{(n)}) = \sum_{n\geq 1} \mathbb{P}_\phi \left[ X_n \leq b_n + \Delta, \prod_{i=1}^{n-1} X_i \leq b_i, X_n \geq b_n \right] . \tag{12}$$

If we define $Y_n = I_{\{X_n \geq b_n\}}$, where $I_A$ is the indicator function of an event $A$, then (12) is equivalent to

$$G_n(b^{(n)}) = \sum_{n\geq 1} \mathbb{P}_\phi \left[ X_n \leq b_n + \Delta, \prod_{i=1}^{n-1} X_i \leq b_i, Y_n = 1 \right] . \tag{13}$$

Application of the Law of Total Probability to the right-hand side of (13) yields

$$G_n(b^{(n)}) = \sum_{n\geq 1} \mathbb{E} \left[ \mathbb{P}_\phi \left[ X_n \leq b_n + \Delta, \prod_{i=1}^{n-1} X_i \leq b_i, Y_n = 1 \mid X^{(n-1)}, Y_n \right] \right] . \tag{14}$$

Note that (14) is equivalent to

$$G_n(b^{(n)}) = \sum_{n\geq 1} \mathbb{E} \left[ \mathbb{E} \left[ I_{\{X_n \leq b_n + \Delta\}} \cdot I_{\{\prod_{i=1}^{n-1} X_i \leq b_i\}} \cdot I_{\{Y_n = 1\}} \mid X^{(n-1)}, Y_n \right] \right] . \tag{15}$$

Since only $I_{\{\prod_{i=1}^{n-1} X_i \leq b_i\}}$ and $I_{\{Y_n=1\}}$ belong to the $\sigma-$algebra generated by $X^{(n-1)}, Y_n$, we may write (15) as

$$G_n(b^{(n)}) = \sum_{n\geq 1} \mathbb{E} \left[ I_{\{\prod_{i=1}^{n-1} X_i \leq b_i\}} \cdot I_{\{Y_n=1\}} \, \mathbb{E} \left[ I_{\{X_n \leq b_n + \Delta\}} \mid X^{(n-1)}, Y_n \right] \right] \tag{16}$$

or equivalently,

$$G_n(b^{(n)}) = \sum_{n\geq 1} \mathbb{E} \left[ I_{\{\prod_{i=1}^{n-1} X_i \leq b_i\}} \cdot I_{\{Y_n=1\}} \, \mathbb{P}_\phi \left[ X_n \leq b_n + \Delta \mid X^{(n-1)}, Y_n \right] \right] . \tag{17}$$

Note that $Y_n$ is the indicator function of $\{X_n \leq b_n\}$ . Hence,

$$\begin{aligned} \mathbb{P}_\phi \left[ X_n \leq b_n + \Delta \mid X^{(n-1)}, Y_n \right] = {} & \mathbb{P}_\phi \left[ X_n \leq b_n + \Delta \mid X^{(n-1)}, Y_n = 1 \right] \, \mathbb{P}_\phi \left[ Y_n = 1 \right] \\ & + \mathbb{P}_\phi \left[ X_n \leq b_n + \Delta \mid X^{(n-1)}, Y_n = 0 \right] \mathbb{P}_\phi \left[ Y_n = 0 \right] . \end{aligned} \tag{18}$$

Note that $\{Y_n = 0\}$ implies $\{X_n \leq b_n\}$ and therefore, $\mathbb{P}_\phi\left[X_n \leq b_n + \Delta \mid X^{(n-1)}, Y_n = 0\right] = 1$. But this also implies that $I_{\{Y_n=1\}} = 0$. Thus, in (18) we only count the term for which $\{Y_n = 1\}$. Thus, we may write (17) as

$$G_n(b^{(n)}) = \sum_{n\geq 1} \mathbb{E}\left[I_{\{\prod_{i=1}^{n-1} X_i \leq b_i\}} \cdot \mathbb{P}_\phi\left[X_n \leq b_n + \Delta \mid X^{(n-1)}, Y_n = 1\right] \cdot \mathbb{P}_\phi\left[Y_n = 1\right]\right] \tag{19}$$

Note now that $\mathbb{P}_\phi\left[X_n \leq b_n + \Delta \mid X^{(n-1)}, Y_n = 1\right] \leq \delta$ since this probability is the complement probability of the one step risk at stage $n$. Hence,

$$G_n(b^{(n)}) \leq \delta \sum_{n\geq 1} \mathbb{E}\left[I_{\{\prod_{i=1}^{n-1} X_i \leq b_i\}} \cdot \mathbb{P}_\phi\left[Y_n = 1\right]\right] . \tag{20}$$

Note also that

$$\sum_{n\geq 1} \mathbb{E}\left[I_{\{\prod_{i=1}^{n-1} X_i \leq b_i\}} \cdot \mathbb{P}_\phi\left[Y_n = 1\right]\right] = \sum_{n\geq 1} \mathbb{P}_\phi\left[\prod_{i=1}^{n-1} X_i \leq b_i, Y_n = 1\right] \tag{21}$$

and since the times between failures are independent, we may conclude from (21) that

$$\sum_{n\geq 1} \mathbb{E}\left[I_{\{\prod_{i=1}^{n-1} X_i \leq b_i\}} \cdot \mathbb{P}_\phi\left[Y_n = 1\right]\right] = \mathbb{P}_\phi\left[\bigcup_{n\geq 1}\left(\prod_{i=1}^{n-1} X_i \leq b_i, Y_n = 1\right)\right] \leq 1 . \tag{22}$$

Finally, substitution in (20) yields $G_n(b^{(n)}) \leq \delta$. □

• Proof of Lemma 2.2.

*Proof.* Let $f_{X_{n+1}}(x \mid X^{(n)} = x^{(n)})$ denotes the density function of $X_{n+1}$ given $X^{(n)}$. Then, we may write

$$R_{n+1}(y, x^{(n)}) = \mathbb{P}_\phi\left[X_{n+1} \geq y \mid X^{(n)} = x^{(n)}\right] = \int_y^\infty f_{X_{n+1}}(u \mid X^{(n)} = x^{(n)})\, du . \tag{23}$$

Let $f_{X^{(n)}}(x^{(n)})$ be the marginal density function of the random vector $X^{(n)}$, then we can define

$$f_{X_{n+1}}(u \mid X^{(n)} = x^{(n)}) = \frac{f_{X_{n+1},X^{(n)}}(u, x^{(n)})}{f_{X^{(n)}}(x^{(n)})}$$

and substitute in (23) to get

$$R_{n+1}(y, x^{(n)}) = \int_y^\infty \frac{f_{X_{n+1},X^{(n)}}(u, x^{(n)})}{f_{X^{(n)}}(x^{(n)})}\, du . \tag{24}$$

Since the joint density $f_{X_{n+1},X^{(n)},\Phi}(u, x^{(n)}, \phi)$ exists, we may write the joint density function of $X_{n+1}$ and $X^{(n)}$ as

$$f_{X_{n+1},X^{(n)}}(u, x^{(n)}) = \int_0^\infty f_{X_{n+1},X^{(n)},\Phi}(u, x^{(n)}, \phi)\, d\phi . \tag{25}$$

We can condition on $x^{(n)}$ and $\phi$ and substitute in (25). Since $X_{n+1}$ is independent of $X^{(n)}$ given $\Phi = \phi$, we get

$$\begin{aligned}\int_0^\infty f_{X_{n+1},X^{(n)},\Phi}(u, x^{(n)}, \phi)\, d\phi &= \int_0^\infty f_{X_{n+1}}(u \mid X^{(n)} = x^{(n)}, \Phi = \phi)\, f_{X^{(n)},\Phi}(x^{(n)}, \phi)\, d\phi \\ &= \int_0^\infty f_{X_{n+1}}(u \mid \Phi = \phi)\, f_{X^{(n)},\Phi}(x^{(n)}, \phi)\, d\phi .\end{aligned} \tag{26}$$

Note that the last equality in (26) comes from conditional independency. Therefore,

$$f_{X_{n+1},X^{(n)}}(u, x^{(n)}) = \int_0^\infty f_{X_{n+1}}(u \mid \Phi = \phi)\, f_{X^{(n)},\Phi}(x^{(n)}, \phi)\, d\phi\,.$$

Substitution in (24) yields

$$R_{n+1}(y, x^{(n)}) = \int_x^\infty \int_0^\infty \frac{f_{X_{n+1}}(u \mid \Phi = \phi)\, f_{X^{(n)},\Phi}(x^{(n)}, \phi)}{f_{X^{(n)}}(x^{(n)})}\, d\phi\, du\,. \tag{27}$$

We now write

$$\frac{f_{X^{(n)},\Phi}(x^{(n)}, \phi)}{f_{X^{(n)}}(x^{(n)})} = f_\Phi(\phi \mid X^{(n)} = x^{(n)})\,.$$

Substitution in (27) and posterior application of Fubini's theorem (cf. Pitt (1985)) yields

$$\begin{aligned} R_{n+1}(y, x^{(n)}) &= \int_y^\infty \int_0^\infty f_{X_{n+1}}(u \mid \Phi = \phi)\, f_\Phi(\phi \mid X^{(n)} = x^{(n)})\, d\phi\, du \\ &= \int_0^\infty \int_y^\infty f_{X_{n+1}}(u \mid \Phi = \phi)\, du\, f_\Phi(\phi \mid X^{(n)} = x^{(n)})\, d\phi \\ &= \int_0^\infty R_{n+1}(y, \phi)\, f_\Phi(\phi \mid X^{(n)} = x^{(n)})\, d\phi\,. \end{aligned} \tag{28}$$

Finally, note that

$$\begin{aligned} S_{n+1}(y, x^{(n)}) &= \mathbb{P}_\phi\left[X_{n+1} \geq x + \Delta \mid X_{n+1} \geq x, X^{(n)}\right] \\ &= \frac{\mathbb{P}_\phi\left[X_{n+1} \geq x + \Delta \mid X^{(n)}\right]}{\mathbb{P}_\phi\left[X_{n+1} \geq x \mid X^{(n)}\right]}\,. \end{aligned}$$

Application of (28) yields the desired result. □

• Proof of Lemma 2.3.

*Proof.* First we study the monotonicity of the one step risk with respect to $y$. Note that

$$\frac{d\, S_{n+1}(y, x^{(n)})}{d\, y} = \frac{(n+k)(N-n)^2\Delta\left(\frac{\sum_{i=1}^n (N-i+1)x_i + \frac{1}{w} + (N-n)\, y}{\sum_{i=1}^n (N-i+1)x_i + \frac{1}{w} + (N-n)(y+\Delta)}\right)^{n+k}}{\left(\sum_{i=1}^n (N-i+1)x_i + \frac{1}{w} + (N-n)(y+\Delta)\right)^2}\,. \tag{29}$$

Since all the factors in the right-hand side of (29) are positive, the one step risk is increasing with respect to $y$.

We now study the monotonicity of the one step risk with respect to $N$. In this case it is convenient to write $\sum_{i=1}^n (N-i+1)x_i = N\sum_{i=1}^n x_i - \sum_{i=1}^n (i-1)x_i$. Substitution in (6) yields

$$S_{n+1}(y, x^{(n)}) = \left(\frac{N(\sum_{i=1}^n x_i + y) - \sum_{i=1}^n (i-1)x_i + \frac{1}{w} - n\, y}{N(\sum_{i=1}^n x_i + y + \Delta) - \sum_{i=1}^n (i-1)x_i + \frac{1}{w} - n(y+\Delta)}\right)^{n+k}\,. \tag{30}$$

To simplify our computations, let us define the quantities which do not depend on $N$ in (30) as follows $A = \sum_{i=1}^n x_i + y$, $B = \sum_{i=1}^n x_i + y + \Delta$, $C = -\sum_{i=1}^n (i-1)x_i + \frac{1}{w} - ny$ and $D = -\sum_{i=1}^n (i-1)x_i + \frac{1}{w} - n(y+\Delta)$. Then it follows that

$$\frac{d\, S_{n+1}(y, x^{(n)})}{d\, N} = \frac{(n+k)(\frac{AN+C}{BN+D})^{n+k-1}(AD - BC)}{(BN+D)^2}\,. \tag{31}$$

Note that the sign of (31) depends only on the factor $AD - BC$, since all the remaining ones are positive. Adequate manipulation of this factor shows that the sign of $AD - BC$ is the same as the sign of $-\sum_{i=1}^n (n-i+1)x_i - \frac{1}{w}$, thus negative. Hence, the one step risk is a decreasing function of $N$. □

• Proof of Lemma 2.4.

*Proof.* First define $1-p=\varphi$ and $h(\varphi)=\frac{(1-\varphi)^{n+\gamma}}{1-\varphi^{N-n}}\varphi^{\lambda-2+\sum_{i=1}^{n}((N-i+1)x_i-1)}$. Then, we may write the one step risk defined in (9) as

$$S_{n+1}(y,x^{(n)})=\frac{\displaystyle\int_0^1 h(\varphi)\;\varphi^{(N-n)\Delta}\;\varphi^{(N-n)y}\;d\varphi}{\displaystyle\int_0^1 h(\varphi)\;\varphi^{(N-n)y}\;d\varphi}\;. \tag{32}$$

Note that (32) is a quotient and the denominator of a derivative in this case is always positive. Therefore, we need only to study the sign of the numerator of its derivative with respect to $y$ to determine its monotonicity. Thus, the condition we need to check is

$$\begin{aligned}&\int_0^1 h(\varphi)\;\varphi^{(N-n)\Delta}\;(N-n)\log(\varphi)\;\varphi^{(N-n)y}\;d\varphi\int_0^1 h(\varphi)\;\varphi^{(N-n)y}\;d\varphi\\&-\int_0^1 h(\varphi)\;(N-n)\log(\varphi)\;\varphi^{(N-n)y}\;d\varphi\int_0^1 h(\varphi)\;\varphi^{(N-n)\Delta}\;\varphi^{(N-n)y}\;d\varphi\geq 0\;.\end{aligned} \tag{33}$$

First note that (33) is equivalent to

$$\frac{\displaystyle\int_0^1 h(\varphi)\;\varphi^{(N-n)\Delta}\;(N-n)\log(\varphi)\;\varphi^{(N-n)y}\;d\varphi}{\displaystyle\int_0^1 h(\varphi)\;\varphi^{(N-n)\Delta}\;\varphi^{(N-n)y}\;d\varphi}\geq\frac{\displaystyle\int_0^1 h(\varphi)\;(N-n)\log(\varphi)\;\varphi^{(N-n)y}\;d\varphi}{\displaystyle\int_0^1 h(\varphi)\;\varphi^{(N-n)y}\;d\varphi}\;. \tag{34}$$

The general form of Chebyshev's inequality (cf. Mitrinović et al. (1993), Chapter 9) shows that (34) holds and thus $S_{n+1}(y,x^{(n)})$ is increasing in $y$. □

• Bayesian procedure vs. deterministic procedure.
We now proof the following claim. If we denote by $s_b$ and $s_d$ the mean number of errors to be observed before stopping testing for the Bayesian and deterministic procedure, respectively, then $s_b\leq s_d$.

*Proof.* Let us define $j_s=\min\{i:S_i(0,x^{(i-1)})\geq 1-\delta\}$. Note that our Bayesian procedure is defined in such a way that if it reaches the stage $j_s$, then it stops. But we may have stopped before, therefore $s_b\leq j_s$.
We show in Section 2.3.1 that $S_i(y,x^{(i-1)})$ is increasing with respect to $y$. Therefore, $S_i(0,x^{(i-1)})\leq S_i(y,x^{(i-1)})$, for all $y\geq 0$. Thus, if $S_i(0,x^{(i-1)})\geq 1-\delta$, then $S_i(y,x^{(i-1)})\geq 1-\delta$, for all $y\geq 0$. Consider now the Bayesian one step risk at time $y=0$. Direct application of Lemma 2.2 yields

$$S_i(0,x^{(i-1)})=\int_0^\infty e^{-\phi(N-i+1)\Delta}\;f_\Phi(\phi\mid X^{(i-1)}=x^{(i-1)})\;d\phi\;. \tag{35}$$

Define the function $g_i(\phi)=e^{-\phi(N-i+1)\Delta}$ and write (35) as follows

$$\int_0^\infty e^{-\phi(N-i+1)\Delta}\;f_\Phi(\phi\mid X^{(i-1)}=x^{(i-1)})\;d\phi=\mathbb{E}[g_i(\Phi)\mid X^{(i-1)}=x^{(i-1)}]\;. \tag{36}$$

First note that $g_i(\phi)\leq g_{i+1}(\phi)$, for all $i\geq 1$. Then, it follows that

$$S_i(0,x^{(i-1)})=\mathbb{E}[g_i(\Phi)\mid X^{(i-1)}=x^{(i-1)}]\leq\mathbb{E}[g_{i+1}(\Phi)\mid X^{(i)}=x^{(i)}]=S_{i+1}(0,x^{(i)})\;. \tag{37}$$

Note also that $g_i$ is convex ($g_i''(\phi)>0$, for all $\phi>0$). Therefore, we can apply Jensen's inequality to (36) and write

$$\mathbb{E}[g_i(\Phi)\mid X^{(i-1)}=x^{(i-1)}]\geq g_i\left(\mathbb{E}[\Phi\mid X^{(i-1)}=x^{(i-1)}]\right)\;.$$

We also prove in Section 2.3.1 that if we consider $\Phi$ to follow a prior gamma distribution, then the posterior distribution of $\Phi$ is also gamma. Therefore, if $\bar{\phi} = \mathbb{E}[\Phi \mid X^{(i-1)} = x^{(i-1)}]$, then

$$g_i\left(\mathbb{E}[\Phi \mid X^{(i-1)} = x^{(i-1)}]\right) = e^{-\bar{\phi}(N-i+1)\Delta} , \tag{38}$$

which is precisely the deterministic one step risk $D_i(\bar{\phi})$ defined in (10). Thus,

$$S_i(0, x^{(i-1)}) \geq D_i(\bar{\phi}) , \tag{39}$$

for all $i \geq 1$. Combination of (37) and (39) yields

$$S_{j_s}(0, x^{(j_s-1)}) \geq 1 - \delta \geq S_{j_s-1}(0, x^{(j_s-2)}) \geq D_{j_s-1}(\bar{\phi}) .$$

Therefore, $D_{j_s-1}(\bar{\phi}) \leq 1 - \delta$, i.e., the deterministic procedure never stops before finding $j_s$ errors. In particular, if $D_{j_s}(\bar{\phi}) \geq 1 - \delta$, then $s_d = j_s$ and if $D_{j_s}(\bar{\phi}) \leq 1 - \delta$, then $s_d > j_s$. Thus, in combination with $s_b \leq j_s$, we have $s_b \leq j_s \leq s_d$. □

# References


J.A. Achcar and D. Brassolatti. Jelinski and Moranda software reliability model: A Bayesian approach. *Revista de Matemática e Estatística*, 17:61–84, 1999.

P.A. Currit, M. Dyer, and H.D. Mills. Certifying the reliability of software. *IEEE Trans. Softw. Eng.*, 12(1):3–11, 1986. ISSN 0098-5589.

S. R. Dalal and C. L. Mallows. When should one stop testing software? *J. Amer. Statist. Assoc.*, 83(403):872–879, 1988.

A. Di Bucchianico, Groote J. F., van Hee K., and R. Kruidhof. Statistical certification of software systems. *Comm. Stat. C*, 37:1–14, 2008.

A. Fries and A. Sen. A survey of discrete reliability-growth models. *IEEE Trans. Rel.*, 45:582–604, 1996.

A.L. Goel and K. Okumoto. An analysis of recurrent software failures on a real-time control system. In *Proceedings of the ACM Annual Technical Conference*, pages 496–500, 1978.

Z. Jelinski and P. Moranda. Software reliability research. In W. Freiberger, editor, *Statistical Computer Performance Evaluation*, pages 465–497. Academic Press, 1972.

S. L. Lauritzen. *Graphical Models*. Clarendon Press, Oxford, 1996.

P. M. Lee. *Bayesian Statistics: an Introduction*. Oxford University Press, 1989.

K. McDaid and S. P. Wilson. Deciding how long to test software. *The Statistician, Journal of the Royal Statistical Society - Series D*, 50(2):117–134, 2001.

D. Mitrinović, J. Pečarić, and A. Fink. *Classical and New Inequalities in Analysis*, volume 61 of *Mathematics and Its Applications*. Kluwer Academic Publishers Group, Dordrecht, 1993.

N. Morali and R. Soyer. Optimal stopping in software testing. *Naval Res. Logist.*, 50(1):88–104, 2003.

H. Okamura, A. Murayama, and T. Dohi. Em algorithm for discrete software reliability models: a unified parameter estimation method. *High Assurance Systems Engineering, 2004. Proceedings. Eighth IEEE International Symposium on*, pages 219–228, March 2004.

S. Özekici and A. Catkan. A dynamic software release model. *Computational Economics*, 6:77–94, 1993.

S. Özekici and R. Soyer. Bayesian testing strategies for software with an operational profile. *Naval Res. Logist.*, 48(8):747–763, 2001.

H. Pham. *System Software Reliability*. Springer Series in Reliability Engineering. Springer, London, 2006.

H. R. Pitt. *Measure and Integration for use*. Clarendon Press, Oxford, 1985.

N.D. Singpurwalla. Determining an optimal time interval for testing and debugging software. *IEEE Trans. Soft. Eng.*, 17(4):313–319, 1991.

N.D. Singpurwalla and S.P. Wilson. Software reliability modeling. *International Statistical Review*, (62):289–317, 1994.

N.D. Singpurwalla and S.P. Wilson. *Statistical Methods in Software Engineering*. Springer Series in Statistics. Springer-Verlag, New York, 1999.

J. R. van Dorp, T. A. Mazzuchi, and R. Soyer. Sequential inference and decision making for single mission systems development. *Journal of Statistical Planning and Inference*, 62(2):207–218, 1997.

S. Yamada, S. Osaki, and H. Narihisa. Discrete models for software reliability evaluation. In *Reliability and quality control (Columbia, Mo., 1984)*, pages 401–412. North-Holland, Amsterdam, 1986.