1993-11

Software reliability model with optimal selection of failure data

Schneidewind, Norman F.

IEEE

IEEE Transactions on Software Engineering, Vol. 19, No. 11, November 1993.
http://hdl.handle.net/10945/45152

Software Reliability Model with Optimal Selection of Failure Data

Norman F. Schneidewind, *Fellow, IEEE*

Abstract— **In the use of software reliability models it is not necessarily the case that all the failure data should be used to estimate model parameters and to predict failures. The reason for this is that old data may not be as representative of the current and future failure process as recent data. Therefore, it may be possible to obtain more accurate predictions of future failures by excluding or giving lower weight to the earlier failure counts. Although "data aging" techniques such as moving average and exponential smoothing are frequently used in other fields, such as inventory control, we did not find use of data aging in the various models we surveyed. One model that includes the concept of selecting a subset of the failure data is the Schneidewind Non-Homogeneous Poisson Process (NHPP) software reliability model. In order to use the concept of data aging, there must be a criterion for determining the optimal value of the starting failure count interval. We evaluated four criteria for identifying the optimal starting interval for estimating model parameters. Three of the criteria are novel. Two of these treat the failure count interval index as a parameter by substituting model functions for data vectors and optimizing on functions obtained from maximum likelihood estimation techniques. The third one uses weighted least squares to maintain constant variance in the presence of the decreasing failure rate assumed by the model. The fourth criterion is the familiar mean square error. Our research showed that significantly improved reliability predictions can be obtained by using a subset of the failure data, based on applying the appropriate criteria, and using the Space Shuttle On-Board software as an example.**

Index Terms—**NHPP software reliability model, optimal selection of failure data, Space Shuttle application.**

I. Introduction

In the use of software reliability models it is not necessarily the case that all the failure data should be used to estimate model parameters and to predict failures. The reason for this is that old data may not be as representative of the current and future failure process as recent data (i.e., the reliability of the software may change over time). More specifically, changes in reliability trends could be caused by dependency of faults (i.e., some faults mask others) and variation in the time between failure occurrence and fault correction. If the failure process remains the same over a long series of observations, we should use a great deal (or all) of the failure data; if there is a significant change in the process, we should use only the most recent observations [3]. Therefore, it may be possible to obtain more accurate predictions of future failures by excluding or giving lower weight to the earlier failure counts. Although "data aging" techniques such as moving average and exponential smoothing are frequently used in other fields, such as inventory control, we did not find use of data aging in the many models we examined in various papers and reports that contain surveys of models [1], [2], [6]–[9], [14]. However, trend analysis for selecting failure data for displaying trends in accordance with a model's assumptions has been studied [11], [16] and neural networks have been applied to trend analysis [12]. Another approach is to use a filter and window to select a specified number of the most recent predictions when deciding on the weights to use for combining predictions from various reliability growth models [15]. While these approaches involve evaluating model trends or predictions for selecting reliability data, one software reliability model that has a *built-in* method for optimally selecting a subset of the failure data is the Schneidewind Non-Homogeneous Poisson Process (NHPP) software reliability model [17], [18], [21]. In order to use the concept of data aging (i.e., giving more weight to recent failure counts), there must be a criterion for determining the optimal value of s, an index in the range $1 \leq s \leq t$, which is the starting value of equal-length failure count intervals. In this model one may choose to use all the failure counts in the execution intervals from 1 to t (Method 1), exclude counts from 1 to $s-1$ (Method 2), or use an aggregate count from 1 to $s-1$ and individual counts from s to t (Method 3).

A. Importance of Research

The importance of this research is that significant improvements were obtained in the accuracy of predicting failure count and time to failure by *not using all the observed failure data* as we will illustrate in the examples. The focus of this paper is on the development, evaluation, and application of criteria for: first, estimating the optimal value of s, s^*, for a *given* criterion, where "optimal" will be defined for each criterion; second, evaluating the accuracy of predictions that are obtained by using each criterion. This allows us to identify s^{**} the value of s^* that produces the most accurate predictions. We note that to evaluate the criteria, s^* is found before predicting reliability, using observed failure data, and s^{**} is identified after the predictions, using predicted data and a criterion which is independent of the criteria being evaluated. Once the best criterion is found, we use its s^* to make *future* reliability predictions.

This research was conducted on the Schneidewind model and the criteria were applied to the Space Shuttle on-board flight software. Since this model is used to assist IBM-

Manuscript received January 1993; revised July 1993. Recommended by F. Bastani.

The author is with the Naval Postgraduate School, Monterey, CA 93943.

IEEE Log Number 9213117.

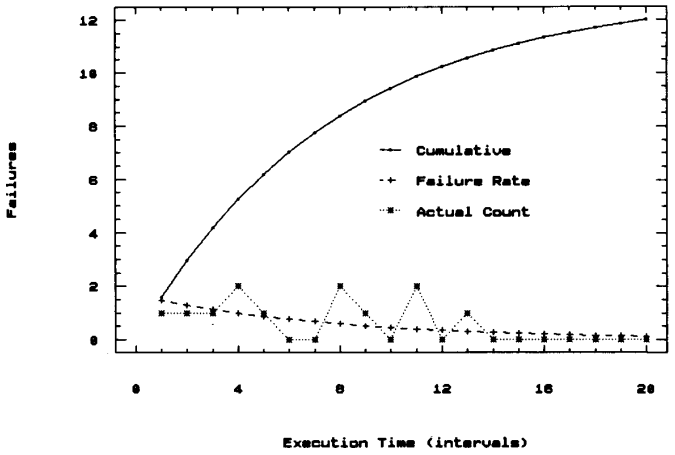


Fig. 1. Method 1: Predicted failures and failure rate, and actual failure count, for $s = 1$.

Houston in making software reliability predictions for the Space Shuttle software, we were motivated to find generic methods for optimal failure data selection and to apply these methods to obtain the most accurate predictions possible for the Space Shuttle [2], [20]. The concepts developed here have general applicability to other models but in order to realize the advantages of optimal data selection, it would be necessary to modify the parameter estimation methods used in those models to explicitly allow for subsets of the failure data to be used.

Our research had four objectives: 1) find a criterion that can consistently identify the optimal s^*, s^{**}, where "optimal" is defined as the value of s^* *across* criteria that produces the most accurate failure predictions; 2) develop a direct solution for s^* such that the criterion function would not have to be evaluated for every value of s; 3) find a criterion that is simple to compute (related to 2); and 4) demonstrate that $s^* > 1$ can produce more accurate failure predictions than $s = 1$ for the Space Shuttle software.

Before discussing the criteria for selecting s, we provide an overview of the Schneidewind model parameter estimation in order to establish the rationale for data aging. After the overview, we develop four criteria for optimal selection of s for Method 2. Each criterion is evaluated with respect to three Space Shuttle modules by computing the values of the criteria for various values of s and plotting the results. We then assess which criterion provides the most accurate cumulative failure predictions; also, we apply one of the criteria to time to next failure predictions. As a by-product of this analysis we show that dramatic improvements can be made in prediction accuracy by *not using all the failure data*. We close with conclusions about the utility of the data aging approach and the best criterion to use for data aging, and with a discussion of extensions to this research.

II. Overview of Schneidewind Model Parameter Estimation

The method of maximum likelihood is used to estimate the model parameters α and β, for a given s, where α is the failure rate at $t = 0$ and β is the failure rate time constant (i.e., a measure of how fast the failure rate decays—the smaller the value of β, the faster the failure rate decreases).

We define three interval count ranges that are pertinent to parameter estimation and reliability prediction, given that $1 \leq i \leq t$ is the range of observed failure data:

Parameter Estimation Range (Observed): The subset of $1 \leq i \leq t$ that is used for estimating α and β.

Prediction Range (Observed): The subset of $1 \leq i \leq t$ that is used for making predictions that are compared to observed failure data for goodness of fit analysis.

Prediction Range (Future): The range $t < i \leq T$ that is used for future predictions.

A. Parameter Estimation: Method 1

Use all of the failure counts from interval 1 through $t (1 = s \leq t)$. This method is used if it is assumed that all of the historical failure counts from 1 through t are representative of the future failure process. With Method 1 we give equal weight to counts in intervals $1, \cdots, t$ when we estimate α and β so that each interval has a weight of $1/t$. Fig. 1 shows predicted cumulative failures and failure rate and actual failure count, all beginning at $t = 1$ signifying the prediction range (*observed*) starts at $t \geq 1$. Equations (1) and (2) are used to estimate β and α, respectively [6]–[8], [17], [18].

$$\frac{1}{\exp(\beta) - 1} - \frac{t}{\exp(\beta t) - 1} = \sum_{k=0}^{t-1} k \frac{x_{k+1}}{X_t} \tag{1}$$

$$\alpha = \frac{\beta X_t}{1 - \exp(-\beta t)} \tag{2}$$

where x_{k+1} are *actual* failure counts in $1, 2, \cdots, k+1, \cdots, t$ and X_t is the *actual* cumulative failure count in $1, t$.

B. Parameter Estimation: Method 2

Use failure counts only in the intervals s through $t (1 \leq s \leq t)$. This method is used if it is assumed that only the

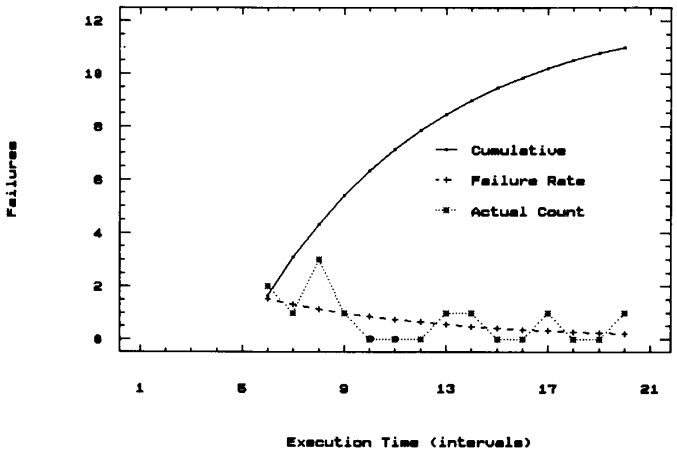


Fig. 2. Method 2: Predicted failures and failure rate, and actual failure count, for $s = 6$.

historical failure counts from s through t are representative of the future failure process. With Method 2 we give zero weight to counts in intervals $1, \cdots, s-1$ and weight $1/(t-s+1)$ to counts in intervals $s, \cdots, t$ when we estimate α and β; thus the more recent counts in intervals $s, \cdots, t$ are given more weight than in Method 1. Fig. 2 shows the predicted cumulative failures and failure rate and actual failure count, all beginning at $t = s$ signifying the prediction range (*observed*) starts at $t \geq s$. Equations (3) and (4) are used to estimate β and α, respectively [6]–[8], [17], [18].

$$\frac{1}{\exp(\beta)-1} - \frac{t-s+1}{\exp(\beta(t-s+1))-1} = \sum_{k=0}^{t-s} k \frac{x_{s+k}}{X_{s,t}} \quad (3)$$

$$\alpha = \frac{\beta X_{s,t}}{1-\exp(-\beta(t-s+1))} \quad (4)$$

where x_{s+k} are *actual* failure counts in $s, s+1, \cdots, s+k, \cdots, t$ and $X_{s,t}$ is the *actual* cumulative failure count in s, t. We note that Method 2 is equivalent to Method 1 for $s = 1$.

C. Parameter Estimation: Method 3

Use the cumulative failure count in the interval 1 through $s-1$ and individual failure counts in the intervals s through $t\,(2 \leq s \leq t)$. This method is used if it is assumed that the historical cumulative failure count from 1 through $s-1$ and the individual failure counts from s through t are representative of the future failure process. With Method 3 we give weight $(s-1)/t$ to the *cumulative* count in the intervals $1, \cdots, s-1$ (i.e., equivalent to the count in a *single* interval of length $s-1$) and weight $1/t$ to counts in the intervals $s, \cdots, t$ when we estimate α and β; thus the more recent counts in intervals $s, \cdots, t$ are given the same weight as Method 1. Although the weight of $(s-1)/t$ for a single interval of length $s-1$ is equivalent to a weight of $1/t$ for $s-1$ intervals, effectively the counts in intervals $1, \cdots, s-1$ are given less emphasis because they are aggregated. This method is intermediate to Method 1, which uses all the data, and Method 2, which discards "old" data. Fig. 3 shows predicted cumulative failures and failure rate and actual failure count, all beginning at $t = 1$ signifying that the prediction range (*observed*) starts at $t \geq 1$. Equations (5) and (6) are used to estimate β and α, respectively [6]–[8], [17], [18].

TABLE I
PARAMETER AND PREDICTION RANGES

Method	Parameter Range (Observed)	Prediction Range (Observed)	Prediction Range (Future)
1	$s = 1$	$1 \leq i \leq t$	$t < i \leq T$
2	$1 \leq s \leq t$	$s \leq i \leq t$	$t < i \leq T$
3	$2 \leq s \leq t$	$1 \leq i \leq t$	$t < i \leq T$

$$\frac{(s-1)X_{s-1}}{\exp(\beta(s-1))-1} + \frac{X_{s,t}}{\exp(\beta)-1} - \frac{tX_t}{\exp(\beta t)-1} = \sum_{k=0}^{t-s}(s+k-1)x_{s+k} \quad (5)$$

$$\alpha = \frac{\beta X_t}{1-\exp(-\beta t)} \quad (6)$$

where X_{s-1} is the *actual* cumulative failure count in $1, s-1$. We note that Method 3 is equivalent to Method 1 for $s = 2$.

The treatment of failure counts for parameter estimation purposes is elaborated in Fig. 4 where all the actual counts of Fig. 1 are used for Method 1, only the actual counts starting at s of Fig. 2 are used for Method 2, and the actual cumulative count for $1, \cdots, s-1$ and individual counts in $s, \cdots, t$ of Fig. 3 are used for Method 3.

The three methods are summarized in Table I with respect to the observed parameter estimation range and the prediction range—observed $(i \leq t)$ and future $(i > t)$—where T is the upper limit of the prediction range.

As developed in [17], [18], the log of the generalized likelihood function that is applicable to the three methods is given by

$$\log L = X_t[\log X_t - 1 - \log(1-\exp(-\beta t))]$$

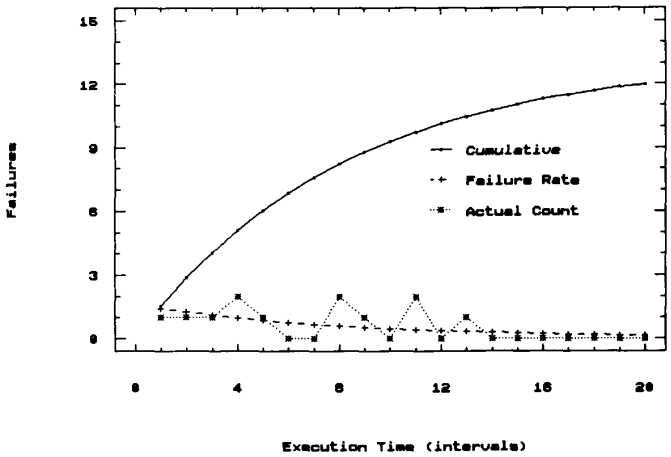


Fig. 3. Method 3: Predicted failures and failure rate, and actual failure count, for $s = 8$.

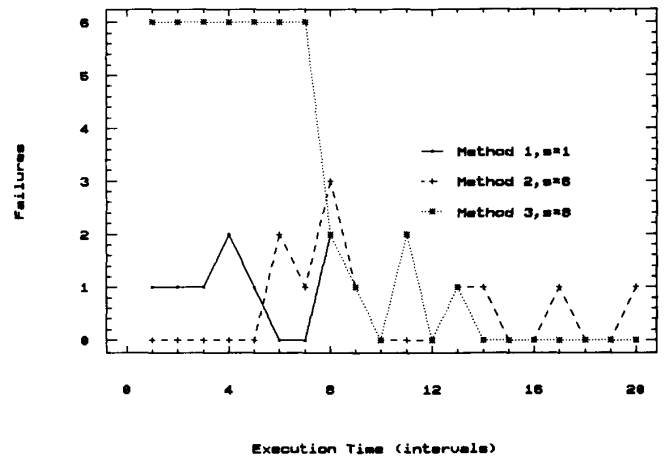


Fig. 4. Treatment of failure counts for parameter estimation.

$$+ X_{s-1}[\log(1 - \exp(-\beta(s-1)))] + X_{s,t}[\log(1 - \exp(-\beta))] - \beta \sum_{k=0}^{t-s} (s+k-1)x_{s+k} \quad (7)$$

where X_t is the *actual* cumulative failure count and the *predicted* count in $1, t$ is given by

$$F_t = (\alpha/\beta)[1 - \exp(-\beta t)] \quad (8)$$

where X_{s-1} is the *actual* cumulative failure count and the *predicted* count in $1, s-1$ is given by

$$F_{s-1} = (\alpha/\beta)[1 - \exp(-\beta(s-1))] \quad (9)$$

where $X_{s,t}$ is the *actual* cumulative failure count and the *predicted* count in s, t is given by

$$F_{s,t} = (\alpha/\beta)[1 - \exp(-\beta(t-s+1))] \quad (10)$$

where x_{s+k} are *actual* failure counts in $s, s+1, \cdots, s+k, \cdots, t$ and the *predicted* counts are given by

$$f_{s+k} = (\alpha/\beta)[\exp(-\beta(s+k-1)) - \exp(-\beta(s+k))] \quad (11)$$

and α and β are parameter estimates.

In Method 1 [17], [18] we use all the failure counts in $1, t$. Thus in (7) $X_{s-1} = 0$, $X_{s,t} \equiv X_t$, and (7) becomes

$$\log L = \log X_t - 1 - \log[1 - \exp(-\beta t)] + \log[1 - \exp(-\beta)] - \beta \sum_{k=0}^{t-1} k x_{1+k}/X_t. \quad (12)$$

III. Optimal Selection of Failure Data Using Method 2

Only Method 2 and its special case $s = 1$, corresponding to Method 1, are covered in this paper. As stated, Method 2 disregards failure counts for intervals $1, \cdots, s-1$ (except for Method 1, where $s = 1$). In this section we evaluate Method 2 with respect to four criteria, each of which is designed to

TABLE II
OPTIMAL STARTING INTERVAL

	Method 2		
	Parameter Estimation Range s^*		
Criterion	Module 1	Module 2	Module 3
1	8	6	5
2	11	8	13
3	12	2	7
4	11	7	10
	Prediction Range s^{**}		
4	11	6	4
MRE	11	6	4

TABLE III
ANALYSIS OF OPTIMAL STARTING INTERVAL

	Method 2			
	Prediction Range *MRE* for s^*			
Criterion	Module 1	Module 2	Module 3	Average
1	0.0048	0.041	0.036	0.027
2	0.000095	0.057	0.16	0.072
3	0.0019	0.202	0.046	0.083
4	0.000095	0.063	0.083	0.049
All data	0.038, $s = 1$	0.202, $s = 2$	0.096, $s = 1$	0.112

TABLE IV
TIME TO NEXT FAILURE
(Current Time: $t = 20$)

	Method 2			
Module	s^* (*MSE*)	Predicted (Using s^*) (Intervals)	Predicted (All Data) (Intervals)	Actual (Intervals)
2	6	7.72	2.03 ($s = 2$)	8
3	2	3.07	2.99 ($s = 1$)	4

identify s^*. In all examples, α and β are estimated in the range $t = 1$–20 and failure count predictions are made in the range $T = 21$–30, where an interval is 30 days of continuous execution of the Space Shuttle software. We show plots for each criterion for Module 1 and summarize results for Modules 1, 2, and 3 in Tables II–IV. The observed failure data are shown in the Appendix. Fortunately for the U.S. space program and the astronauts, the failures are sparse! Despite the sparsity of failures the model, with the aid of data aging, can predict quite accurately, as will be seen.

A. Criterion 1: $s^* = s$ where log $L(\alpha, \beta, s)$ is Maximum

This criterion is based on the following novel concept: If the model is a good representation of the observed failure counts, then it should be possible to substitute *predicted* failure counts for the corresponding *actual* failure counts in the likelihood function so that we can maximize it with respect to s (i.e., we treat s as a third parameter). We refer to this type of likelihood function as one which uses model functions to distinguish it from the usual case of using data vectors in the likelihood function. If we represent the original, generalized likelihood function (7), which is not differentiable in s, as follows:

$$l(\alpha, \beta, s, t, X_{s-1}, X_{s,t}, X_t, X_{s+k}) \tag{13}$$

then the idea is to substitute the corresponding *predictor* functions for the *actual* failure counts to form a new likelihood function as follows:

$$L(\alpha, \beta, s, t, F_{s-1}, F_{s,t}, F_t, f_{s+k}). \tag{14}$$

The maximum value of this function can be identified and used as a criterion for selecting s^*. What is desired is a procedure for obtaining the best estimates of α, β, and s that will maximize the likelihood function simultaneously in one step. Unfortunately, we are forced to do iterative optimization because a value of s must be selected to estimate α from $(\partial \log l/\partial \alpha)_s = 0$ and β from $(\partial \log l/\partial \beta)_s = 0$ for $1 \le s \le t$. Once these parameters are estimated, s can be estimated from the **maximum** value of $\log L(\alpha, \beta, s)$ and the optimal triple (α, β, s) can be selected.

Since in Method 2 [17], [18] we ignore failure counts in $1, s-1, x_1, x_2, \cdots, x_{s-1} = 0 \Rightarrow X_{s-1} = 0$ and $X_t = X_{s,t}$, and t in (7) must be replaced by $t - (s-1) = t - s + 1$ and $s + k - 1$ must be replaced by $s + k - 1 - (s-1) = k$. Therefore, (7) becomes

$$\log L = \log X_{s,t} - 1 - \log[1 - \exp(-\beta(t - s + 1))] + \log[1 - \exp(-\beta)] - \beta \sum_{k=0}^{t-s} k x_{s+k}/X_{s,t}. \tag{15}$$

If $s = 1$ in (15), we obtain (12). Thus for $s = 1$, Method 2 is equal to Method 1.

Now, after substituting (10) for $X_{s,t}$ and (11) for x_{s+k} in (15), respectively, and deriving an expression for the summation term in (15), we obtain

$$\log L = \log[(\alpha/\beta)(1 - \exp(-\beta))] - 1 - \frac{\beta[\exp(-\beta s) - \exp(-\beta t)][(t-s)(1 - \exp(-\beta)) + 1]]}{1 - \exp(-\beta(t - s + 1))}. \tag{16}$$

Equations (15), the likelihood function using data vectors, and (16), the likelihood function using model functions, are plotted in Fig. 5 for Module 1 of the Space Shuttle software. Both equations are shown to see whether (16) has the same pattern as (15); it does. The plots have negative values because the likelihood function is the log of a product of probability density functions. The maximum (least negative) values of (16) occur at $s^* = 8$, 6, and 5 for Modules 1, 2, and 3, respectively (see Table II). Equation (16) is too complex to solve for $\partial \log L/\partial s = 0$ directly in order to obtain s^*. Therefore, s^* is obtained from the maximum value of (16).

B. Criterion 2: $s^* = s$ where $|\partial \log L/\partial \beta(\beta, s)|$ is Minimum

This criterion is based on a second novel concept related to the concept of Criterion 1: If the model is a good representation of the data, and considering the fact that β was estimated from

$$\partial \log l/\partial \beta(\beta, s, t, X_{s,t}, x_{s+k}) = 0$$

for a given value of s, a "good" value of s is where

$$|\partial \log L/\partial \beta(\beta, s, t, F_{s,t}, f_{s+k})|$$

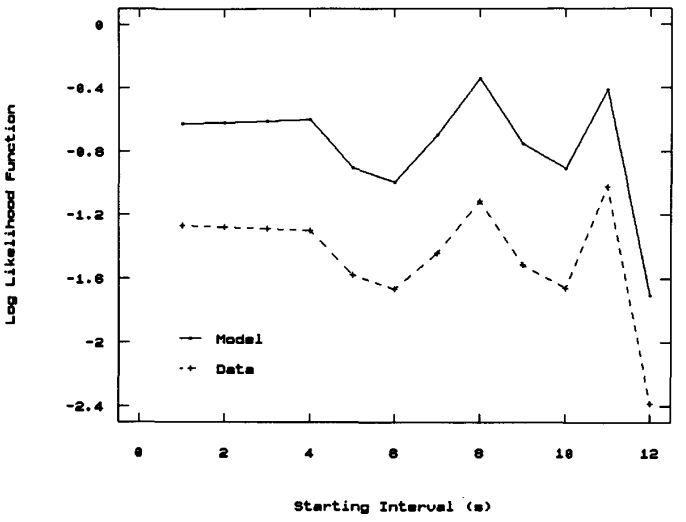


Fig. 5. Method 2, Criterion 1, Module 1.

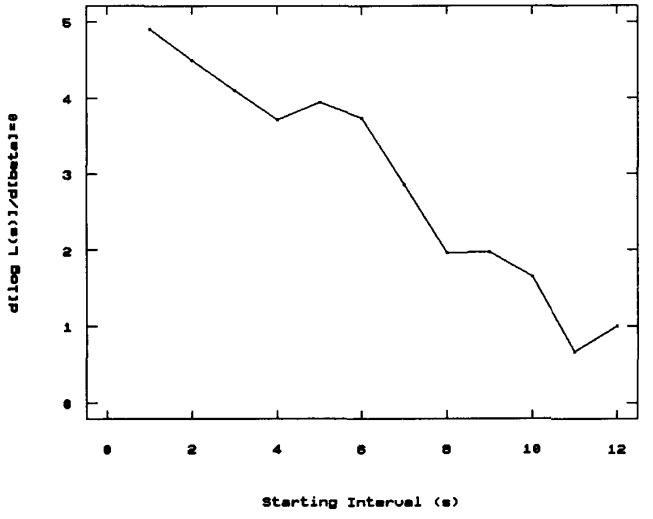


Fig. 6. Method 2, Criterion 2, Module 1.

is minimum (i.e., a direct solution of $\partial \log L/\partial\beta = 0$ for s^* cannot be obtained because of the complexity of the equation). As in the case of Criterion 1, Criterion 2 is based on using model functions, but uses a derivative of the likelihood function. Criterion 2 is inherently inferior to Criterion 1 because it only optimizes with respect to α and β while the latter optimizes with respect to α, β, and s. In both cases β is optimized over the range of s, $1 \leq s \leq t$. The advantage of Criterion 2 is that (17) is a simpler function than (16) to evaluate.

Now, after substituting (10) for $X_{s,t}$ and (11) for x_{s+k} in (3) and deriving an expression for the summation term in (3), we obtain

$$\frac{\partial \log L}{\partial\beta} = \frac{1}{\exp(\beta) - 1} - \frac{t - s + 1}{\exp(\beta(t - s + 1)) - 1} - \frac{[\exp(-\beta s) - \exp(-\beta t)][(t - s)(1 - \exp(-\beta)) + 1]}{1 - \exp(-\beta(t - s + 1))} = 0. \quad (17)$$

Equation (17) is plotted in Fig. 6 for Module 1. The minimum values of (17) occur at $s^* = 11$, 8, and 13 for Modules 1, 2, and 3, respectively (see Table II).

C. Criterion 3: Weighted Least Squares: $s^* = s$ where (18) is Minimum

In the original model [17], [18], the method of weighted least squares (WLS = mean weighted squared difference between predicted and actual *interval* failure counts x_i) was used to choose the optimal value of s (the one yielding the minimum value of WLS) for prediction purposes. Weighted least

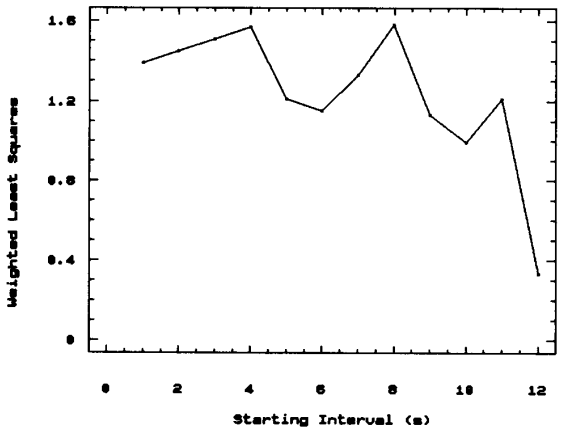


Fig. 7. Method 2, Criterion 3, Module 1.

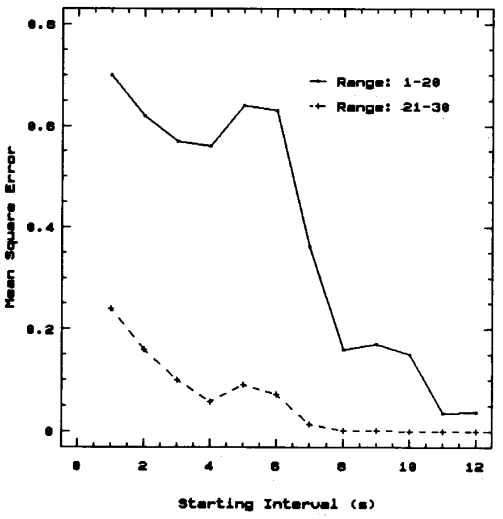


Fig. 8. Method 2, Criterion 4, Module 1: Parameter estimation range (1–20) and prediction range (21–30).

squares was used because the model assumed an NHPP and exponentially decreasing failure rate with increasing s which implies a decreasing variance between expected and actual failure counts over time. Since an assumption of the method of least squares is constant variance, the squared deviations are weighted by the appropriate factor to maintain constant variance over time (i.e., weights are inversely proportional to variance) [4], [5]. In the original model, WLS was computed by using all the failure counts. Equation (18) at the bottom of the page generalizes WLS to allow for *not* using all the failure counts. The original model WLS is obtained from (18) by letting $s = 1$. Although WLS seemed to be a reasonable criterion at the time because it comports with the assumptions of the model, experience suggests that other criteria should be evaluated with the objectives of providing values of s that result in better predictions and a reduction in the computation required to apply the criteria.

Equation (18) is plotted in Fig. 7 for Module 1. The minimum values of (18) occur at $s^* = 12$, 2, and 7 for Modules 1, 2, and 3, respectively (see Table II).

D. Criterion 4: Mean Square Error: $s^* = s$ where (19) or (20) is Minimum

The MSE_F [2] computes the mean of the sum of the squared differences between model predictions and actual cumulative failure counts $X_{s,i}$ in the range $s \leq i \leq t$, where

$$X_{s,i} = X_i - X_{s-1}.$$

$$MSE_F = \frac{\sum_{i=s}^{t}[\alpha/\beta(1 - \exp(-\beta(i - s + 1))) - X_{s,i}]^2}{t - s + 1}. \tag{19}$$

The MSE_T criterion for time to next failure(s) is similarly defined and is given by (20) at the bottom of the page, where T_{fi} is the actual time to the next F_i failure(s), i is the current time, and $X_{s,i}$ failures have been observed between s and i.

The rationale of the fourth and last criterion is to minimize the sum of the variance and the square of the bias of predicted failure count or time to failure [10]. Equation (19) is plotted in Fig. 8 for Module 1 for both parameter estimation and prediction (*future*) ranges. The minimum values of (19) occur at $s^* = 11$, 7, and 10 for Modules 1, 2, and 3, respectively (see Table II). The minimum values of (20) occur at $s^* = 6$ and 2 for Modules 2 and 3, respectively (see Table IV). It was not possible to satisfy the condition indicated in (20) for Module 1; therefore, no result is shown.

IV. Evaluation of Criteria

A. Cumulative Failures

Now we evaluate how good a job the four criteria did in identifying s^{**} with respect to the prediction of cumulative

$$WLS = \frac{\sum_{i=s}^{t} \exp(\beta(i - s + 1))[\alpha/\beta(\exp(-\beta(i - s + 1)))(\exp(\beta) - 1) - x_i]^2}{t - s + 1}. \tag{18}$$

$$MSE_T = \frac{\sum_{i=s}^{t}[[\log[\alpha/(\alpha - \beta(F_i + X_{s,i}))]]/\beta - (i - s + 1)] - T_{fi}]^2}{t - s + 1}, \quad \text{for} \quad \alpha > \beta(F_i + X_{s,i}) \tag{20}$$

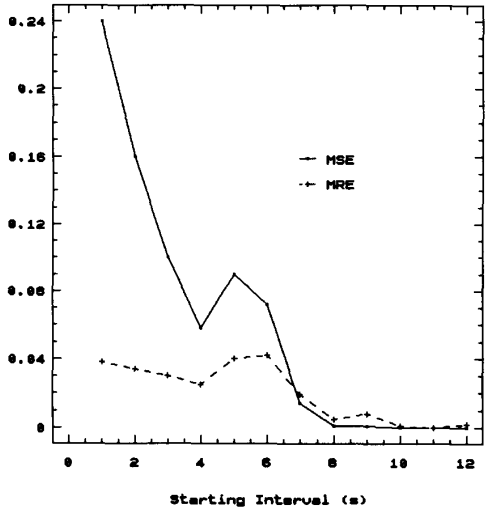


Fig. 9. Method 2, MSE_F and MRE, Module 1: Prediction range (21–30).

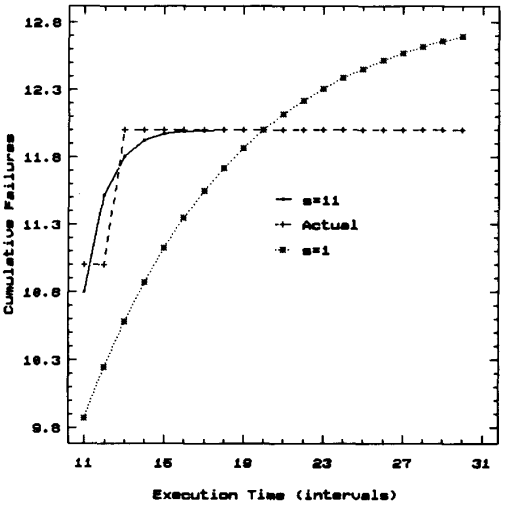


Fig. 10. Method 2, Module 1: Predicted and actual cumulative failures.

failures using (21)

$$F(T) = (\alpha/\beta)[1 - \exp(-\beta((T - s + 1)))] + X_{s-1}. \quad (21)$$

We calculate goodness of fit for 10 intervals into the future and compare predicted cumulative failures (21) with actual cumulative failures for those intervals [14]. Both MSE_F (19) and mean relative error (MRE) are plotted, as a function of s, for the *prediction* range $T = 21$–30 for Module 1 in Fig. 9 in order to identify $s^{**} = 11$ (value of s where MSE and MRE are minimum). Mean relative error [13], [19] is given by

$$MRE = \sum_i (|X_i - F_i|/X_i)/N \quad (22)$$

for N intervals.

Equation (22) is used to provide a measure of prediction accuracy that is independent of any of the criteria being evaluated. Table II summarizes the results obtained for s^* for the four criteria as applied to the three modules in the *parameter estimation range* $t = 1$–20. These results are compared with those obtained for s^{**} in the *prediction range* $T = 21$–30 by using the MSE and MRE criteria, emphasizing the latter since it is an independent criterion. Table III shows the MRE values in the *prediction range* for each criterion's s^* for the three modules, and the averages for the three modules. Also shown in Table III are the MRE values for using all the data and their average ($s = 2$ is the starting interval when all the failure data are used for Module 2 because parameter estimates could not be obtained for $s = 1$). The most significant finding derived from these tables is that *all* criteria produced better predictions than using all the data for the three modules. Secondly, the tables show that both Criterion 2 and Criterion 4 produced s^{**} ($s^* = 11$) for Module 1, Criterion 1 produced s^{**} ($s^* = 6$) for Module 2, and none of the criteria produced s^{**} for Module 3, although Criterion 1 came close ($s^* = 5$). Thirdly, Criterion 1 was the best on the basis of average MRE, with Criterion 4 making a respectable showing.

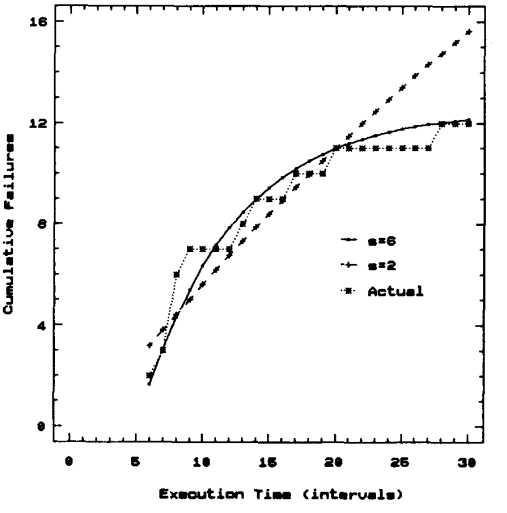


Fig. 11. Method 2, Module 2: Predicted and actual cumulative failures.

In order to compare cumulative failure predictions that use s^* with those that use $s = 1, 2$ and compare both to the actual cumulative failures, we show Figs. 10–12 for Modules 1–3, respectively. All three figures show much better predictions using s^* as opposed to using all the data, with the latter exhibiting overshoot. In fact, in the case of Fig. 10 (Module 1) only four failures out of a total of thirteen produced a much more accurate prediction than the one obtained by using all the data!

B. Time to Next Failure

A summary of time to next failure results is shown in Table IV, where the predictions were obtained using (23) [2] and s^* was obtained using (20) for Modules 2 and 3

$$T_f(t) = [(\log[\alpha/(\alpha - \beta(F_t + X_{s,t}))])/\beta] - (t - s + 1), \quad \text{for } \alpha > \beta(F_t + X_{s,t}) \quad (23)$$

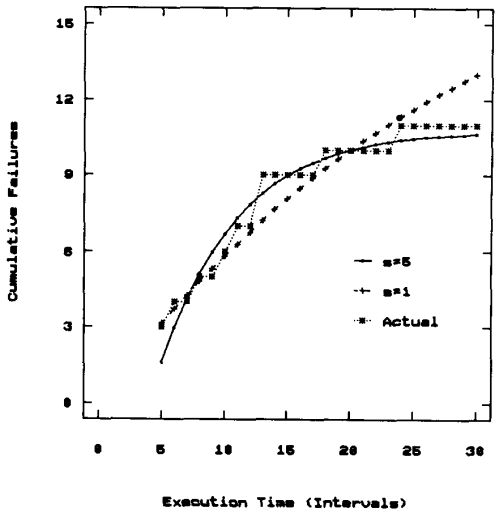


Fig. 12. Method 2, Module 3: Predicted and actual cumulative failures.

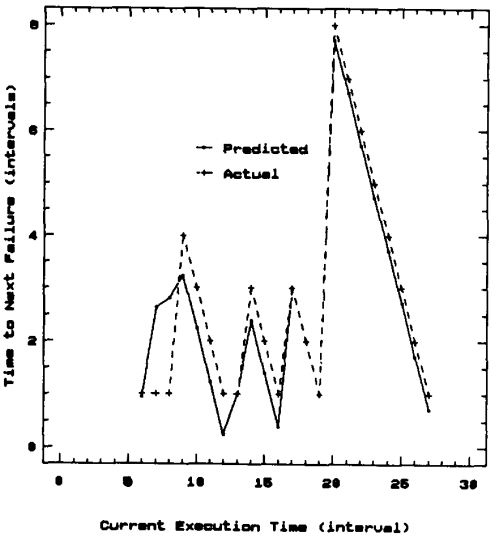


Fig. 13. Method 2, Module 2: Predicted and actual time to next failure.

where the current time is t, and we predict the time for F_t failures (one or more) to occur, and $X_{s,t}$ failures have been observed between s and t.

Again it is seen that using s^* produces better predictions than using all the failure data. An example of prediction accuracy obtained by using $s^* = 6$ for Module 2 is shown in Fig. 13 where (23) is plotted for various current execution times and contrasted with actual time to next failure.

C. Mean Relative Error

Potentially, MRE itself could be used as a criterion for selecting s^* in the *parameter estimation range* because of its accuracy in identifying $s^{**} = 11$, 6, and 4 for Modules 1, 2, and 3, respectively, in the *prediction range* (see Table II); its average error is 0.020, smaller than any average error in Table III of criteria that were evaluated. However, there is a problem in using MRE for this purpose and that is the possibility of $X_i = 0$ for some interval i (see the Appendix which lists zero failures for early intervals of Modules 2 and 3).

V. Conclusions

Our four research objectives were stated in the *Introduction*. The results we obtained with respect to achieving these objectives are as follows. 1) Rather than finding a single criterion that was consistently superior, we found that all criteria did better than using all the failure data, whether the prediction was cumulative failures or time to next failure. We found that Criterion 1 (maximum likelihood estimation using model functions) had the minimum average error. However, Criterion 4 (MSE_F), with the second lowest average error, did a respectable job, and is easier to compute than Criterion 1. It also has the advantage of providing predictions in the *parameter estimation range* as a by-product of computing MSE_F so that the model user can see the fit with the data in this range; the same statement applies to MSE_T. 2) We were unable to solve for s^* directly because the expressions for Criterion 1 and Criterion 2 are too complex. 3) Although we could not find a criterion that could be evaluated in a single step, a program could be written to evaluate a criterion, say MSE_F and MSE_T, find s^*, and select the best model for the user. 4) We demonstrated significant improvements in failure prediction, both cumulative failures and time to next failure by using data aging. Although the results from three modules may not be considered definitive, we note that there were fourteen cases (four criteria applied to three modules for cumulative failure prediction and one criterion applied to two modules for time to failure prediction) in which data aging produced superior predictions and none in which this was not the case. This result is significant for increasing the accuracy of software reliability predictions for the Space Shuttle. Since the other Space Shuttle modules have failure count distributions over execution time that are similar to the ones analyzed, we assume data aging is applicable in general to the Space Shuttle software. Our results suggest that other software reliability models could benefit from using data aging.

The next stage of our research will involve the use of other failure data sets to determine whether data aging is applicable to a different environment. In addition we will analyze the four criteria relative to the use of Method 3.

Appendix
Observed Failure Counts
(Interval = 30 days execution time)

Interval	Module 1	Module 2	Module 3
1	1	0	0
2	1	0	0
3	1	0	0
4	2	0	0
5	1	0	3
6	0	2	1
7	0	1	0
8	2	3	1

9	1	1	0
10	0	0	1
11	2	0	1
12	0	0	0
13	1	1	2
14	0	1	0
15	0	0	0
16	0	0	0
17	0	1	0
18	0	0	1
19	0	0	0
20	0	1	0
21	0	0	0
22	0	0	0
23	0	0	0
24	0	0	1
25	0	0	0
26	0	0	0
27	0	0	0
28	0	1	0
29	0	0	0
30	0	0	0
31–63	0		
64	1		
31–43		0	
44		1	
31–45			0
46			1
47–58			0
59			1
60–65			0
66			1
Total	13	13	14

DISCLAIMER

The analysis of experimental results of the intermediate software failure data in this paper should not be construed as a prediction of the final Space Shuttle software reliability. Rather, the Space Shuttle data are used as real project examples for the purposes of developing, enhancing, and validating software reliability models.

ACKNOWLEDGMENT

The author wishes to acknowledge the support provided for this project by Dr. W. Farr, Naval Surface Warfare Center; T. Keller, IBM Corporation; and R. Paul, U.S. Army Operational Test and Evaluation Command. The author also acknowledges the mathematical support provided by P. Schneidewind.

REFERENCES

[1] A. A. Abdel-Ghaly, P. Y. Chan, and B. Littlewood, "Evaluation of competing software reliability predictions," *IEEE Trans. Software Eng.*, vol. SE-12, no. 9, pp. 950–967, Sept. 1986.

[2] Recommended Practice for Software Reliability, R-013-1992, American National Standards Institute/American Institute of Aeronautics and Astronautics, 370 L'Enfant Promenade, SW, Washington, DC 20024, 1993.

[3] R. G. Brown, *Smoothing, Forecasting and Prediction of Discrete Time Series.* Englewood Cliffs, NJ: Prentice-Hall, 1963.

[4] C. Daniel and F. S. Wood, *Fitting Equations to Data.* New York: Wiley-Interscience, 1971.

[5] N. R. Draper and H. Smith, *Applied Regression Analysis.* New York: Wiley, 1966.

[6] W. H. Farr, "A survey of software reliability modeling and estimation," Naval Surface Weapons Center, Tech. Rep. NSWC TR 82-171, Sept. 1983.

[7] W. H. Farr and O. D. Smith, "Statistical modeling and estimation of reliability functions for software (SMERFS) users guide," Naval Surface Weapons Center, Tech. Rep. NAVSWC TR-84-373, rev. 2, Mar. 1991.

[8] ——, "Statistical modeling and estimation of reliability functions for software (SMERFS) library access guide," Naval Surface Weapons Center, Tech. Rep. NAVSWC TR-84-371, rev. 2, Mar. 1991.

[9] A. L. Goel, "Software reliability models: Assumptions, limitations, and applicability," *IEEE Trans. Software Eng.*, vol. SE-11, no. 12, pp. 1411–1423, Dec. 1985.

[10] G. M. Jenkins and D. G. Watts, *Spectral Analysis and its Applications.* New York: Holden-Day, 1968.

[11] K. Kanoun, M. R. Bastos Martini, and J. Moreira de Souza, "A method for software reliability and analysis and prediction application to the TROPICO-R switching system," *IEEE Trans. Software Eng.*, vol. 17, no. 4, pp. 334–344, Apr. 1991.

[12] T. M. Khoshgoftarr, A. S. Pandya, and H. B. More, "A neural network approach for predicting software development faults," in *Proc. 3rd Int. Symp. on Software Reliability Engineering.* New York: IEEE Computer Society Press, Oct. 1992, pp. 83–89.

[13] T. M. Khoshgoftarr, J. C. Munson, B. B. Bhattacharya, and G. D. Richardson, "Predictive modeling techniques of software quality from software measures," *IEEE Trans. Software Eng.*, vol. 18, no. 11, pp. 979–987, Nov. 1992.

[14] B. Littlewood, "Theories of software reliability: How good are they and how can they be improved," *IEEE Trans. Software Eng.*, vol. SE-6, no. 5, pp. 489–500, Sept. 1980.

[15] M. Lu, S. Brocklehurst, and B. Littlewood, "Combination of predictions obtained from different software reliability growth models," in *Proc. 10th Annu. Software Reliability Symp.*, Denver, CO, June 1992, pp. 24–33.

[16] M. R. Bastros Martini, K. Kanoun, and J. Moreira de Souza, "Software reliability evaluation of the TROPICO-R switching system," *IEEE Trans. Reliab.*, vol. 39, no. 3, pp. 369–379, Aug. 1990.

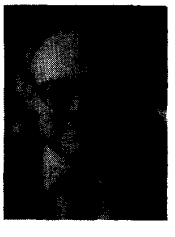
[17] N. F. Schneidewind, "Analysis of error processes in computer software," in *Proc. Int. Conf. Reliable Software*, Apr. 21–23, 1975, pp. 337–346.

[18] ——, "Analysis of error processes in computer software," *ACM Sigplan Notices*, vol. 10, no. 6, 1975.

[19] ——, "Methodology for validating software metrics," *IEEE Trans. Software Eng.*, vol. 18, no. 5, pp. 410–422, May 1992.

[20] N. F. Schneidewind and T. W. Keller, "Application of reliability models to the Space Shuttle," *IEEE Software*, pp. 28–33, July 1992.

[21] M. Xie and M. Zhao, "The Schneidewind software reliability model revisited," in *Proc. Int. Symp. Software Reliability Engineering.* New York: IEEE Computer Society Press, Oct. 1992, pp. 184–192.



Norman F. Schneidewind (A'54–M'59–M'72–SM'77–F'93) is Professor of Information Sciences at the Naval Postgraduate School, Monterey, CA, where he teaches and performs research in software engineering and computer networks. He is also the director of laboratories in his department. He is the developer of the Schneidewind software reliability model which is used by IBM-Houston to assist in the prediction of software reliability of the NASA Space Shuttle. This model is one of the models recommended by the American Institute of Aeronautics and Astronautics and the American National Standards Institute "Recommended Practice for Software Reliability."

Dr. Schneidewind was elected Fellow of the IEEE for "contributions to software measurement models in reliability and metrics, and for leadership in advancing the field of software maintenance." He was awarded a certificate for outstanding research achievements in 1992 by the Naval Postgraduate School.