

## Optimal Selection of Failure Data for Predicting Failure Counts

Norman F. Schneidewind  
Code AS/Ss  
Naval Postgraduate School  
Monterey, CA 93943

### 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. While there are techniques that involve using trends in failure data or model predictions for selecting reliability data, one software reliability model that has a built-in method for optimally selecting a subset of the failure data for parameter estimation is the Schneidewind Non-Homogeneous Poisson Process (NHPP) software reliability model. We did not find use of this idea in the other models we surveyed. 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. Our research has identified the mean square error as a good criterion for selecting the starting interval of the failure data. In this paper we apply the criterion to select the optimal starting interval. We show that significantly improved reliability predictions can be obtained by using a subset of the failure data, based on applying the criterion, and using the Space Shuttle On-Board software as an example.*

**Keywords:** NHPP software reliability model, optimal selection of failure data, Space Shuttle.

### 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. 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 [5]. Therefore it may be possible to obtain more accurate predictions of future failures by excluding or giving lower weight to the earlier failure counts. Techniques such as a six month moving average is a practice recommended by Bellcore when the reliability of switching equipment is considered [3, 4]. This technique is also discussed in [6] with respect to reliability and availability prediction, where an eleven time unit moving average is used. Smoothing techniques, such

as computing failure intensity based on  $k$  consecutive failure points, are also used [14]. While these techniques involve using trends in failure data or model predictions for selecting reliability data, one software reliability model that has a *built-in* method for optimally selecting a subset of the failure data for parameter estimation is the Schneidewind Non-Homogeneous Poisson Process (NHPP) software reliability model [2, 15, 17, 18]. We did not find use of this idea in the many other models we examined in various papers and reports that contain surveys of models [1, 2, 7, 8, 9, 10, 13]. 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).

### Importance of Research

The importance of this research is that significant improvements were obtained in the accuracy of predicting failure count and time to next failure (due to space limitations, only the failure count analysis is presented in this paper) by *not using all the observed failure data*, where appropriate, as we will illustrate in the examples. Also of significance is the identification of a criterion for determining "where appropriate"; that is, the method for determining the optimal value of  $s$ ,  $s^*$ , where "optimal" is defined as the value of  $s$  that produces the most accurate predictions. This research was conducted on the Schneidewind model and the criterion was applied to the Space Shuttle On-Board flight software. Since this model is used to assist IBM-Houston in making software reliability predictions for the Space Shuttle software, we were motivated to find a generic method for optimal failure data selection and to apply this method to obtain the most accurate predictions possible for the Space Shuttle [2, 16]. 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.

The purpose of this paper is to demonstrate the effectiveness of the Mean Square Error (MSE) criterion, which we identified as providing a good balance between achieving prediction accuracy and computational efficiency of the four criteria which we developed and analyzed in our research [17], for selecting  $s^*$ . We demonstrate that, when conditions warrant,  $s^* > 1$  can

produce more accurate failure predictions than  $s=1$  for the Space Shuttle software.

Before discussing the criterion for selecting  $s^*$ , we provide an overview of the Schneidewind model parameter estimation in order to establish the rationale for data aging. As a by-product of this analysis we show that, for certain modules, dramatic improvements can be made in prediction accuracy by *not using all the failure counts*. We close with conclusions about the utility of the data aging approach and the criterion to use for data aging; we also indicate our future research efforts.

### Overview of Schneidewind Model Parameter Estimation

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

#### **a. Parameter estimation: Method 1**

Use all of the failure counts from interval 1 through  $t$  ( $1 \leq s \leq t$ ). 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. Equations (1) and (2) are used to estimate  $\beta$  and  $\alpha$ , respectively [7, 8, 9, 15].

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

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

where  $x_{k+1}$  are failure counts in  $1, 2, \dots, k+1, \dots, t$  and  $X_t$  is the 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 historical failure counts from  $s$  through  $t$  are representative of the future failure process. Equations (3) and (4) are used to estimate  $\beta$  and  $\alpha$ , respectively [7, 8, 9, 15].

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

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

where  $x_{s+k}$  are failure counts in  $s, s+1, \dots, s+k, \dots, t$  and  $X_{s,t}$  is the cumulative failure count in  $s, t$ . We note that Method 2 is equivalent to Method 1 for  $s=1$  (i.e., (1) and (2) are obtained by substituting  $s=1$  in (3) and (4), respectively).

#### **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. This method is intermediate to Method 1, which uses all the data, and Method 2, which discards "old" data. Equations (5) and (6) are used to estimate  $\beta$  and  $\alpha$ , respectively [7, 8, 9, 15].

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

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

where  $X_{s-1}$  is the cumulative failure count in  $1, s-1$ . We note that Method 3 is equivalent to Method 1 for  $s=2$  (i.e., (1) is obtained by substituting  $s=2$  in (5)).

The three methods are summarized in Table 1 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.

### Optimal Selection of Failure Data Using Method 2

As stated, Method 2 disregards failure counts for intervals  $1, \dots, s-1$ , where  $1 \leq s \leq t$ . In this section we apply Method 2 with respect to the MSE criterion. In all examples  $\alpha$  and  $\beta$  are estimated in the range  $i=1-20$  and failure count predictions are made in the range  $i=21-30$ , where an interval is 30 days of continuous execution of the Space Shuttle software. The observed failure data are shown in the Appendix.

#### **Failure Prediction**

Once  $\alpha$  and  $\beta$  have been estimated for a given  $s$ , using one of the three methods, various predictions can be made. However, since we want to find  $(\alpha^*, \beta^*, s^*)$ , the computational procedure is to first use the MSE criterion, which is described in the next section, to find the optimal triple and then use it in the prediction equation. The predicted cumulative number of failures is given by (7) for Method 2. This equation is derived from (4), where  $F_i - X_{s-1}$  replaces  $X_{s,t}$ , reflecting the fact that  $X_{s,t}$  only accounts for failures in the range  $s, t$ . Failures in the range  $1, s-1$ , which are accounted for by  $X_{s-1}$ , must be included in (7).

$$F_i(s) = (\alpha/\beta)[1 - \exp(-\beta(i-s+1))] + X_{s-1} \quad (7)$$

The equation for Methods 1 and 3 is obtained by setting  $s=1$ , using the values of  $\alpha^*$ ,  $\beta^*$  obtained from the respective parameter estimation methods, and setting  $X_{s-1}=0$ .

### Mean Square Error Criterion

The Mean Square criterion for cumulative failures is to minimize (8).

$$MSE(s) = \frac{\sum_{t=s}^T [\alpha/\beta (1 - \exp(-\beta(i-s+1))) - (X_t - X_{s-1})]^2}{t-s+1} \quad (8)$$

The rationale of MSE (mean squared difference between predicted and actual cumulative failure counts  $X_t - X_{s-1}$  in the range  $s, t$ ) is to minimize the sum of the variance and the square of the bias of the predicted failure count [11]. However, since a substantial amount of computation could be involved in computing MSE for all values of  $s$ , we adopt a modified rule of using the value of  $s$  where MSE starts to increase after initially decreasing from  $s=1$ ; we call this value  $s'$  to distinguish it from  $s^*$  the value of  $s$  where MSE is minimum. This approach results in less computation and minimum discarding of "old" data. Our experience indicates that  $s'$  will provide accurate predictions and much better ones than  $s=1$  in those cases where  $s=1$  is not optimal. Furthermore, we recognize that there is no assurance that  $s^*$  computed in the *parameter estimation range* will necessarily result in the minimum MSE or most accurate prediction in the *prediction range*. In fact, as will be seen, in some cases our heuristic produces better predictions than that obtained with  $s^*$ . The MSE for Methods 1 and 3 is obtained by making the adjustments described in the previous section. Equation (8) is plotted in Figure 1 for Module 1 of the Space Shuttle software for both the *parameter estimation range* and the *prediction range*. For the former, we use (8) to compute the MSE in the range where failures have been *observed* (see Table 1) in order to select  $s'$  or  $s^*$  prior to predicting future failures. For the latter, we modify (8) to use summation limits of  $t+1$  to  $T$  and to use a denominator of  $T-t$ , where  $T$  is the upper limit of the prediction range (see Table 1). This is done *after* predictions are made to see whether the MSE computed in the prediction range is the same as that computed in the parameter estimation range. This figure shows  $s'=4$  and  $s^*=11$  in both curves.

In order to provide a measure of prediction accuracy that is independent of the MSE criterion, we compute the mean relative error (MRE) for the *prediction range*, which is given by (9) [12]

$$MRE = \sum_i (|X_i - F_i| / X_i) / (T-t). \quad (9)$$

This result is shown in Figure 2, where MRE and MSE (repeated from Figure 1) are plotted for Module 1. For MRE, we again have  $s'=4$  and  $s^*=11$ . Now, we compare  $F_i(4)$  with  $F_i(1)$  in Figure 3 and see that  $s'=4$  provides a better prediction than  $s=1$ , with the latter showing too much overshoot.

These procedures are repeated for Module 2 in Figures 4, 5

and 6 and for Module 3 in Figures 7, 8 and 9. The prediction curves in Figures 3, 6 and 9, which all show better prediction accuracy for  $s'$  as compared to  $s=1$  ( $s=2$  is used for comparison for Module 2 because estimates of  $\alpha$  and  $\beta$  could not be obtained for  $s=1$ ), dramatize the importance of using data aging, where appropriate. The analysis of the starting interval is summarized in Table 2. When  $s'$  is obtained from MSE for the *parameter estimation range*, it produces the "best"  $s$  for Module 2 (MRE(6) differs from MRE(7) by only .03) and Module 3, as determined by the MSE and MRE for the *prediction range*, and provides better predictions than using all the data for all three modules. As the execution of the software continues for  $T > 30$ , the described procedures would be repeated with  $t=30$  (i.e., new upper limit of *parameter estimation range*).

### Summary, Conclusions and Future Research

We found that MSE does a good job of identifying  $s'$ ; it has no dependence on model assumptions and it minimizes the sum of the variance and the square of the bias of the predicted failure count. We noted that once MSE reaches a minimum, as a function of  $s$ , and starts to increase, the computation can be terminated at that point because  $s'$  provides a good (better than  $s=1$ ) prediction, although not necessarily the best prediction. Since the future failure process may not mirror the past, no criterion can produce the best prediction in all cases. What we can accomplish is to produce *better* predictions than would be the case in using all the data. This we have demonstrated with the examples. Since the other Space Shuttle modules have failure count distributions over execution time that are similar to the ones analyzed, we believe 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 Jet Propulsion Laboratory planetary mission data and Shuttle mission ground control data from the Johnson Space Center to determine whether data aging is applicable to different environments. In addition we will analyze the MSE criterion relative to the use of Method 3.

### 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 is used as real project examples for the purpose of developing, enhancing and validating software reliability models.

### Acknowledgements

We wish to acknowledge the support provided for this project by Dr. William Farr, Naval Surface Warfare Center; and Mr. Ted Keller and Mr. David Hamilton, International Business Machines Corporation.

## References

- [1] A. A. Abdel-Ghaly, et al., "Evaluation of Competing Software Reliability Predictions", IEEE Transactions on Software Engineering, Vol. SE-12, No. 9, September 1986, pp. 950-967.
- [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, February 23, 1993.
- [3] Network Switching Element Outage Performance Monitoring Procedures, SR-TSY-000963, Bellcore, Issue 1, April 1989.
- [4] Reliability and Quality Measurements for Telecommunications Systems (RQMS), TR-TSY-000929, Bellcore, Issue 1, June 1990.
- [5] Robert Goodell Brown, Smoothing, Forecasting and Prediction of Discrete Time Series, Prentice-Hall, Inc., 1963.
- [6] R. Cramp, M. A. Vouk, and V. V. Jones, "On Operational Availability of a Large Software-Based Telecommunications System", Proceedings Third International Symposium on Software Reliability Engineering, IEEE Computer Society, pp. 358-366, 1992.
- [7] William H. Farr, A survey of Software Reliability Modeling and Estimation, NSWC TR 82-171, Naval Surface Weapons Center, September 1983.
- [8] William H. Farr and Oliver D. Smith, Statistical Modeling and Estimation of Reliability Functions for Software (SMERFS) Users Guide, NAVSWC TR-84-373, Revision 2, Naval Surface Weapons Center, March 1991.
- [9] William H. Farr and Oliver D. Smith, Statistical Modeling and Estimation of Reliability Functions for Software (SMERFS) Library Access Guide, NAVSWC TR-84-371.
- [10] Amrit L. Goel, "Software Reliability Models: Assumptions, Limitations, and Applicability, IEEE Transactions on Software Engineering, Vol. SE-11, No. 12, December 1985, pp. 1411-1423.
- [11] Gwilym M. Jenkins and Donald G. Watts, Spectral Analysis and its Applications, Holden-Day, 1968.
- [12] Taghi M. Khoshgoftarr, et al., "Predictive Modeling Techniques of Software Quality from Software Measures", Transactions on Software Engineering, Vol. 18, No. 11, November 1992, pp. 979-987.
- [13] B. Littlewood, "Theories of Software Reliability: How Good Are They and How Can They Be Improved", Transactions on Software Engineering, Vol. SE-6, No. 5, September 1980, pp. 489-500.
- [14] John D. Musa, Anthony Iannino, and Kazuhira Okumoto, Software Reliability (Measurement, Prediction, Application), McGraw-Hill Book Company, 1987.
- [15] Norman F. Schneidewind, "Analysis of Error Processes in Computer Software", Proceedings of the International Conference on Reliable Software, IEEE Computer Society, 21-23 April 1975, pp. 337-346 and ACM Sigplan Notices, Volume 10, Number 6, 1975.
- [16] Norman F. Schneidewind and T.W. Keller, "Application of Reliability Models to the Space Shuttle", IEEE Software, July 1992 pp. 28-33.
- [17] Norman F. Schneidewind, "Software Reliability Model with Optimal Selection of Failure Data", IEEE Transactions on Software Engineering, November 1993 (to be published).
- [18] M. Xie and M. Zhao, "The Schneidewind Software Reliability Model Revisited", Third International Symposium on Software Reliability Engineering, IEEE Computer Society Press, October 1992, pp. 184-192.

APPENDIX

Observed Failure Counts

(Interval = 30 days execution time)

<u>Interval</u>	<u>Module 1</u>	<u>Module 2</u>	<u>Module 3</u>
-----------------	-----------------	-----------------	-----------------

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

---

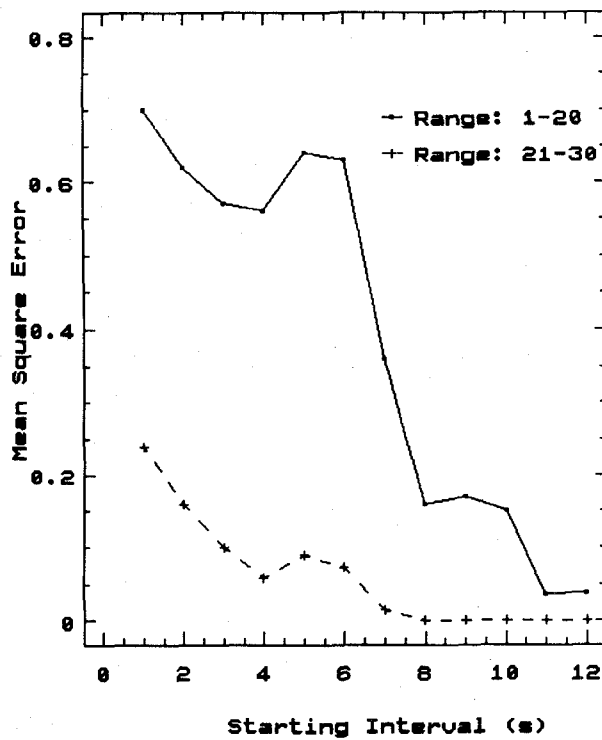
Total	13	13	14
-------	----	----	----

**Table 1**  
**Parameter and Prediction Ranges**

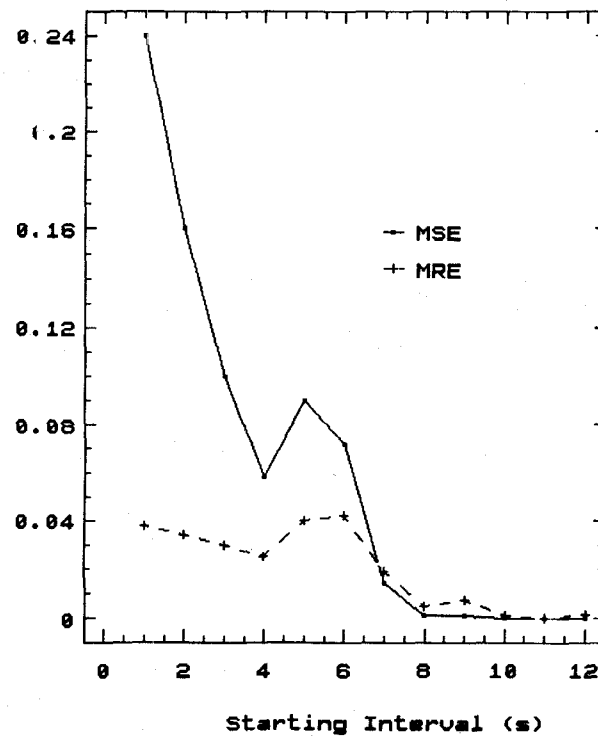
Method	Parameter Range (Observed)	Prediction Range (Observed)	Prediction Range (Future)
1	$s=1$	$1 \leq i \leq t$	$t < i \leq T$
2	$1 \leq s \leq t$	$s \leq i \leq t$	$t < i \leq T$
3	$2 \leq s \leq t$	$1 \leq i \leq t$	$t < i \leq T$

**Table 2**  
**Analysis of Starting Interval**

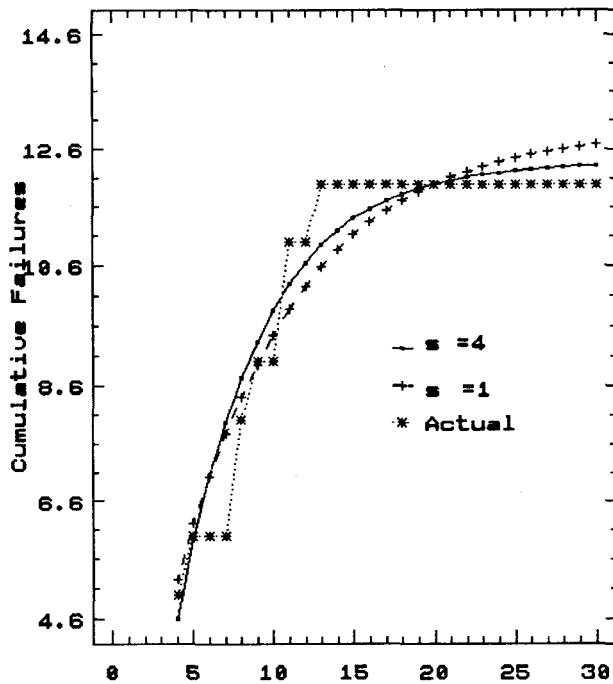
Module	MSE: Parameter Estimation Range	MSE: Prediction Range	MRE: Prediction Range
1	$s'=4, \text{MSE}=.56; s^*=11, \text{MSE}=.035$	$s'=4, \text{MSE}=.058; s^*=11, \text{MSE}=8\text{E-}8$	$s'=4, \text{MRE}=.025; s^*=11, \text{MRE}=9\text{E-}5$
2	$s'=7, \text{MSE}=.56; s^*=7, \text{MSE}=.56$	$s'=6, \text{MSE}=.31; s^*=6, \text{MSE}=.31$	$s'=6, \text{MRE}=.041; s^*=6, \text{MRE}=.041$
3	$s'=4, \text{MSE}=.32; s^*=10, \text{MSE}=.15$	$s'=4, \text{MSE}=.067; s^*=4, \text{MSE}=.067$	$s'=4, \text{MRE}=.020; s^*=4, \text{MRE}=.020$



**Fig. 1. Module 1. Prediction:21-30**  
**Parameter Estimation:1-20**

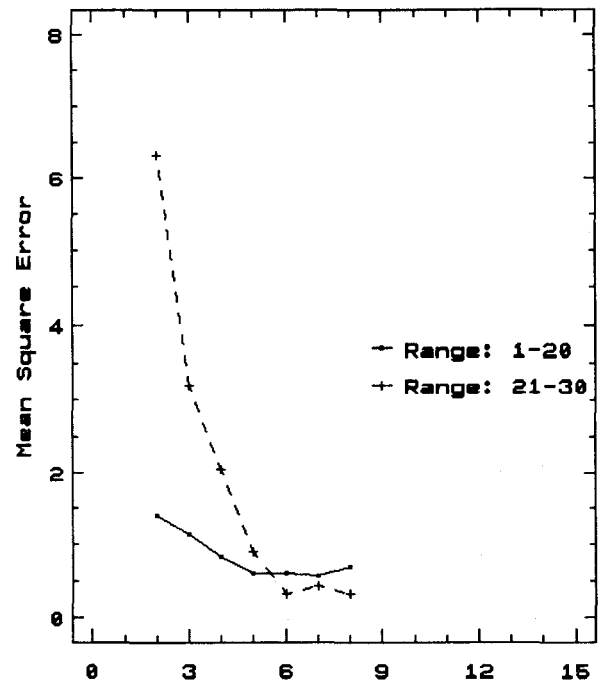


**Fig. 2. Module 1. MSE & MRE**  
**Prediction Range:21-30**



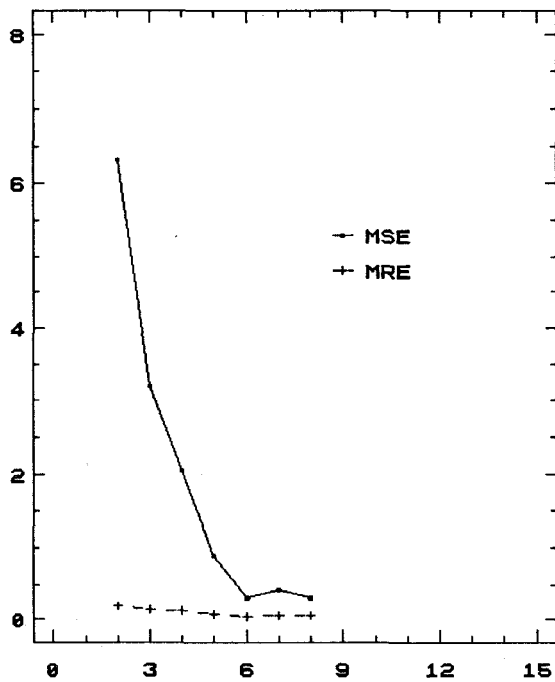
Execution Time (Intervals)

Fig. 3. Module 1. Predicted and Actual Cumulative Failures



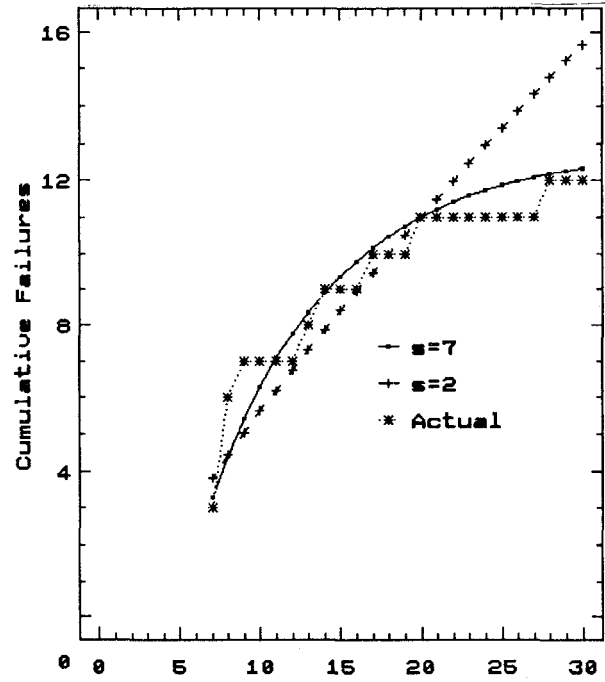
Starting Interval (s)

Fig. 4. Module 2. Prediction: 21-30  
Parameter Estimation: 1-20



Starting Interval (s)

Fig. 5. Module 2. MSE & MRE  
Prediction Range: 21-30



Execution Time (Intervals)

Fig. 6. Module 2. Predicted and Actual Cumulative Failures

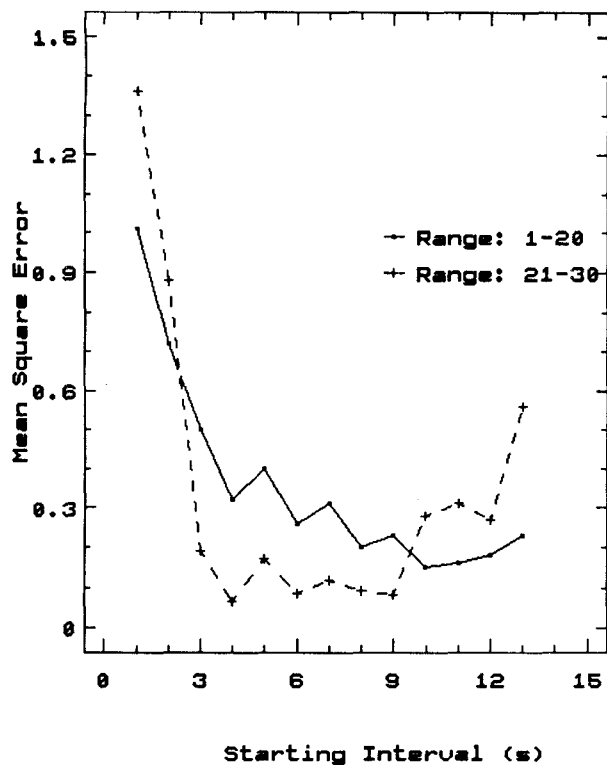


Fig. 7. Module 3. Prediction:21-30  
Parameter Estimation:1-20

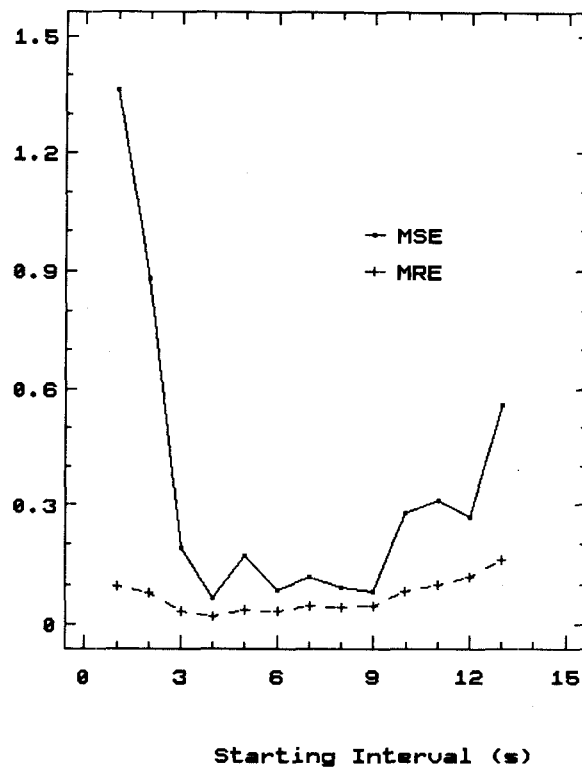


Fig. 8. Module 3. MSE & MRE  
Prediction Range 21-30

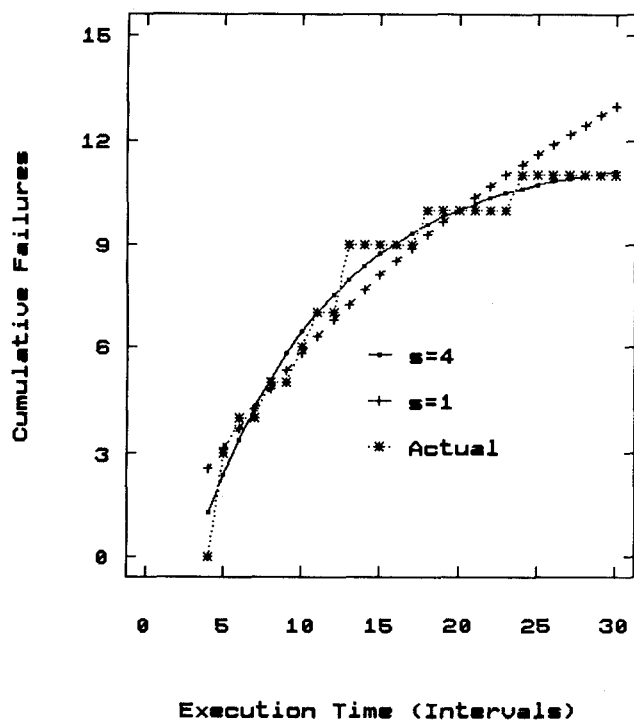


Fig. 9. Module 3. Predicted  
and Actual Cumulative Failures