Statistical Analysis of Field Data for Repairable Systems

David C. Trindade
Swami Nathan

David C. Trindade, Ph.D.
Swami Nathan, Ph.D.
Sun Microsystems, Inc.
6005 Assisi Court
San Jose, CA 95138 USA
Internet (e-mail): david.trindade@sun.com; swami.nathan@sun.com

SUMMARY & PURPOSE

The purpose of the tutorial is to present simple graphical methods for analyzing the reliability of repairable systems. Many talks and papers on repairable systems analysis deal primarily with complex parametric modeling methods. Because of their highly esoteric nature, such approaches rarely gain wide acceptance into the reliability monitoring practices of a company. This tutorial will present techniques based on non-parametric methods which have been successfully used within Sun Microsystems to transform the way reliability of repairable systems is analyzed and communicated to management and customers. Upon completion of this tutorial, attendees should be able to analyze a large dataset of repairable systems, identify trends in the rates of failures, identify outliers, causes of failures and present this information using a series of simple plots that can be understood by management, customers and field support engineers alike.

David C. Trindade, Ph.D.

Dr. David Trindade is a Distinguished Engineer at Sun Microsystems. Formerly he was a Senior Fellow at AMD. His fields of expertise include reliability, statistical analysis, and modeling of components, systems, and software; and applied statistics, especially design of experiments (DOE) and statistical process control (SPC). He is co-author (with Dr. Paul Tobias) of the book Applied Reliability, 2nd ed., published in 1995. He has a BS in Physics, an MS in Statistics, an MS in Material Sciences and Semiconductor Physics, and a Ph.D. in Mechanical Engineering and Statistics. He has been an adjunct lecturer at the University of Vermont and Santa Clara University.

Swami Nathan

Dr. Swami Nathan is a senior staff engineer at Sun Microsystems. His field of interest is on field data analysis, statistical analysis and reliability/availability modeling of complex systems. He received his B.Tech from Indian Institute of Technology, and M.S. and Ph.D in reliability engineering from the University of Maryland, College Park. He has authored over twenty papers in peer reviewed journals and international conferences and holds 2 patents.

Table of Contents

1. Introduction....1
2. Dangers of MTBF....1
3. Parametric Methods....2
4. Mean Cumulative Function....4
5. Calendar Time Analysis....5
6. Failure Cause Plots....7
7. MCF Comparisons....7
8. MCF Extensions....7
9. Conclusions....9
10. References....10
11. Tutorials/Visuals....11

1. INTRODUCTION

A repairable system, as the name implies, is a system which can be restored to operating condition in the event of a failure. The restoration involves any manual or automated action that falls short of replacing the entire system. Common examples of repairable systems include computer servers, network routers, printers, automobiles, locomotives, etc. Although repairable systems exist in all walks of life, the techniques for analyzing repairable systems are not as prevalent as those for non-repairable systems. This situation often leads to incorrect analysis techniques due to confusion between the hazard rate and rate of occurrence of failures [1,2].

The techniques for repairable systems found in the literature are primarily parametric methods, requiring a certain degree of statistical knowledge on the part of the practitioner. The difficulty of communicating techniques, such as testing distributional assumptions, to management renders them impractical for widespread usage within an organization.

Recently analysis of repairable systems based on non-parametric methods are becoming increasingly popular due to their simplicity as well as ability to handle more than just *counts of recurrent events* [3,4,5,6,7]. This tutorial provides a simple yet powerful approach for performing reliability analysis of repairable systems using non-parametric methods. Innovative reliability plotting methods are explored for the identification of trends, discerning deeper issues relating to failure modes, assessing effects of changes and comparing across platforms, vintages, environments etc. These approaches have been applied with great success to datacenter systems (both hardware and software), and the tutorial is based on courses and training sessions given to sales, support services, management, and engineering personnel within Sun Microsystems™. These techniques can be easily applied within a spreadsheet environment such as StarOffice™ or Excel™ by anybody and demands only a very rudimentary knowledge of statistics. Interesting examples and case studies from actual analysis of computer servers at customer datacenters will be provided for all concepts.

1.1 Notation and Acronyms

MTBF	mean time between failure
MCF	mean cumulative function
CTF	calendar time function
RR	recurrence rate
ROCOF	rate of occurrence of failures
HPP	homogeneous Poisson process
NHPP	non-homogeneous Poisson process

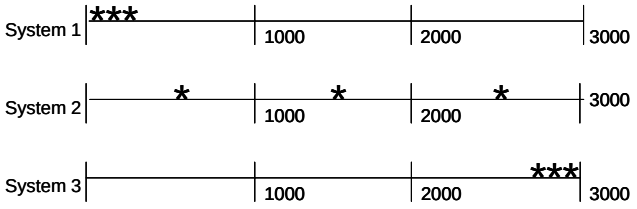
2. DANGERS OF MTBF

The most common metric used to represent the reliability of repairable systems is an MTBF, which is calculated by adding all the operating hours of all the systems and dividing by the number of failures. The popularity of the MTBF metric is due to its simplicity and its ability to cater to the one number syndrome. MTBFs are often stated by equipment manufacturers with imprecise definitions of a failure most often in fine print. MTBF hides information by not accounting for any trends in the arrival of failures and treating machines of all ages as coming from the same population.

There are several assumptions involved in stating an MTBF. Firstly, it is assumed that the failures of a repairable system follow a renewal process, i.e., all failure times come from a single population distribution. A further assumption is that the times between events are independent and exponentially distributed with a constant rate of occurrence of events, and consequently, we have a homogeneous Poisson process (HPP). The validity of a HPP is rarely checked in reality. As a result, strict reliance on the MTBF without full understanding of the consequences can result in missing developing trends and drawing erroneous conclusions.

Figure 1. MTBF hides information

In Figure 1, we have three systems that have operated for



3000 hours, with each experiencing three failures. Thus, all three systems have the same MTBF of 1000 hours. However, System 1 had three early failures and none thereafter. System 2 had a failure in each 1000 hour interval while System 3 had three late failures. The behaviour of the three systems are dramatically different and yet they have the same MTBF! Clearly there is a need for better reliability metrics that account for trends in the failure data.

2.1 The "Failure Rate" Confusion

Well intentioned practitioners often invert the MTBF and quote a failure rate. However the term failure rate has become a confusing term in the literature [1,2]. Often engineers analyze data from repairable systems using methods for the analysis of data from non-repairable systems. Let us consider the failure of a computer due to a central processing unit or CPU. The computer is a repairable system while the CPU is a non-repairable component. When we have a dataset of times of failures of the computer due to the CPU, analysts often take the times between failures and treat them as times to failures of CPUs. The implication is that the times to failures of individual CPUs arise from the same distribution, i.e., they are independent and identically distributed. Consequently, the sequence of times to failures is neglected. This assumption is valid only if the distribution of the times to first failure is identical to the distribution of the times to second failure, and so on ad infinitum. If for example, the cooling fan inside the computer is degrading, then the times to successive CPU failures will start getting shorter, violating the iid assumption.

Usher[2] shows an interesting case study where the times between failures are treated as lifetimes from the same distribution to fit a Weibull distribution with a decreasing

hazard rate, while a simple cumulative plot shows that the rate of occurrence of failures is actually increasing! The hazard rate is a property of a time to failure while ROCOF is a property of a sequence of times to failures i.e., order of occurrence of failures matters.

Despite Ascher's passionate arguments more then twenty years ago [14], the term failure rate continues to be arbitrarily used in the industry and sometimes in academia to describe both a hazard rate of a lifetime distribution of a non-repairable system and a rate of occurrence of failures of a sequences of failure times of a repairable system. This lack of distinction can lead to poor analysis choices even by well intentioned individuals.

3. PARAMETRIC METHODS

One of the common parametric approaches to modeling repairable systems reliability typically assumes that failures occur according to a non-homogeneous Poisson process with an intensity function. One of the popular intensity functions is the power law Poisson process [8,9] which has an intensity function of the form

$$u(t) = \lambda\beta t^{\beta-1} \qquad \lambda, \beta > 0 \tag{1}$$

The probability that a system experiences n failures in t hours has the following expression

$$P(N(t) = n) = \frac{(\lambda t^{\beta}) e^{-\lambda t^{\beta}}}{n!} \tag{2}$$

To estimate the two parameters in the model one can use maximum likelihood estimation. The equations for the parameter estimates are given in [8,9]

$$\hat{\lambda} = \frac{\sum_{q=1}^{K} N_q}{\sum_{q=1}^{K} T_q^{\hat{\beta}} - S_q^{\hat{\beta}}}$$

$$\hat{\beta} = \frac{\sum_{q=1}^{K} N_q}{\hat{\lambda}\sum_{q=1}^{K}(T_q^{\hat{\beta}} \ln T_q - S_q^{\hat{\beta}} \ln S_q) - \sum_{q=1}^{K}\sum_{i=1}^{N_q} \ln X_{iq}} \tag{3}$$

where we have *K* systems, *S* and *T* are start and end times of observation accounting for censoring, *Nq* is the number of failures on the *q*th system and *Xiq* is the age of the *q*th system at the *i*th failure.

These equations cannot be solved analytically and require an iterative procedure or special software. Crow [8] also provides methods for confidence interval estimation and a test statistic for testing the adequacy of the power law assumption.

Further extensions of renewal process techniques known as Generalized Renewal Process were proposed by Kijima[10,11]. Kijima models removed several of the assumptions regarding the state of the machine after repair present in earlier models. However, because of the complexity of the renewal equation closed form solutions are not possible and numerical solutions can be quite tedious. A Monte Carlo simulation based approach for the Kijima formulation was developed in [12]. Mettas and Zhao [13] present a general likelihood function formulation for estimating the parameters of the general renewal process in the case of single and multiple repairable systems. They also provide confidence bounds based on Fisher information matrix.

Despite the abundance of literature on the subject, parametric approaches are computationally intensive and not intuitive to the average person who performs data analysis to support his/her particular customer. Special solution techniques are required along with due diligence in justifying distributional assumptions (rarely done in practice).

Non parametric approaches based on MCFs are far simpler, understandable by lay persons and customers, and are easily implemented in a spreadsheet. The next sections cover the methodology.

4. MEAN CUMULATIVE FUNCTION

4.1Cumulative Plot

Given a set of failure times for a repairable system, the simplest graph that can be constructed is a cumulative plot. The cumulative plot is a plot of the number of failures versus the age of the system. This plot can be constructed for all

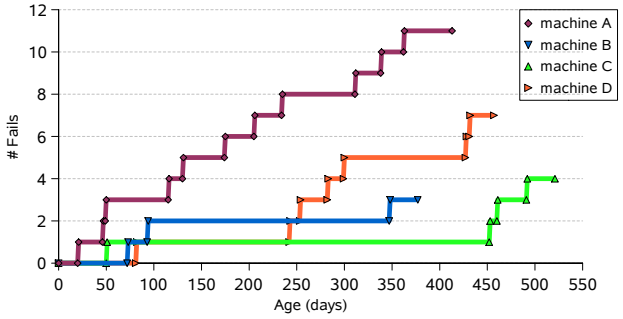
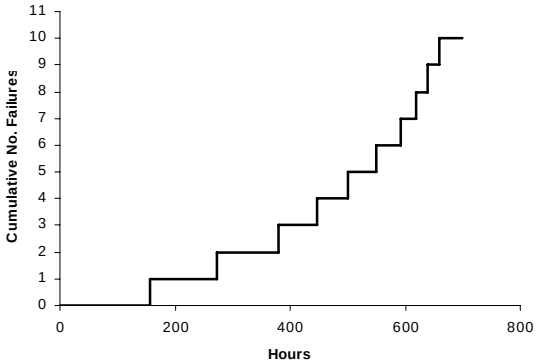


Figure 2. Cumulative plots for a group of four machines

failures, outages, system failures due to specific failure modes etc. A cumulative plot can be constructed for just 1 machine or for a group (all) machines in a population. Figure 2 shows an example cumulative plot. There are four machines in the population. W
various failure eve
evolution of failures
one failure at 50 day
After about 450 days
the next 100 days of



Although a cumulative plot looks quite simple it is of great importance because of its ability to reveal trends. Figures 3, 4, and 5 show three different cumulative plots.

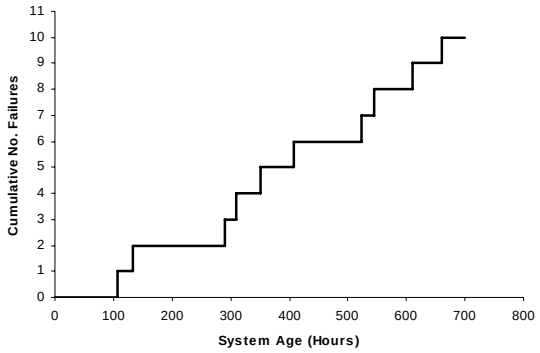


Figure 3. Cumulative plot for a stable system.

.

The shape of the cumulative plot can provide ready clues as

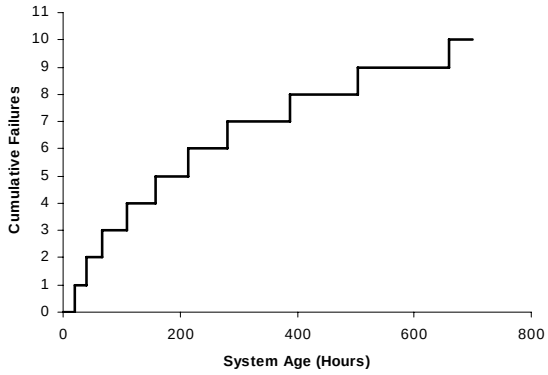


Figure 4. Cumulative plot for an improving system

to whether the system is improving, worsening, or stable. An improving system has the times between failures lengthening with age (takes longer to get to the next failure) while a worsening system has times between failures shortening with age (takes less time to get to the next failure).

It is to be noted that all three plots show a system with 10 failures in 700 hours, i.e., MTBF of 70 hours. Despite having identical MTBFs, the behaviors of the three systems are dramatically different.

4.2 Mean Cumulative Function versus Age

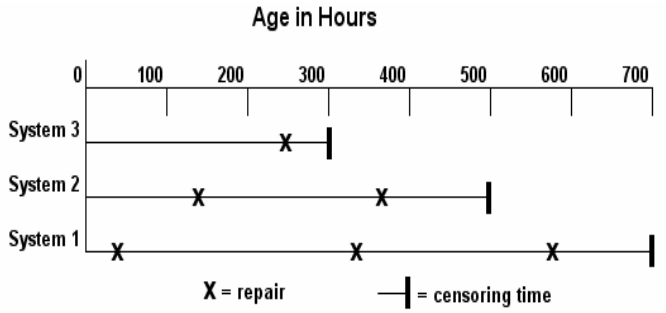
If there are many machines in the population it would be fairly tedious to construct cumulative plots for each individual

Figure 5. Cumulative plot for a worsening system

machine. A useful construct would be to plot the average behavior of these numerous machines. This is accomplished by calculating the Mean Cumulative Function (MCF).

The MCF is constructed incrementally at each failure event by considering the number of machines at risk at that point in time. The number of machines at risk depends on the how many machines are contributing information. Information can be obscured by the presence of censoring and truncation. Right censoring occurs when information is not available beyond a certain age, e.g., a machine that is 100 days old cannot contribute information to the reliability at 200 days, and hence is not a *machine at risk* when calculating the average at 200 days. Similarly information may be obscured at earlier ages if for example a machine is installed on Jan 1 2004 and service contract was initiated on Jan 1 2005. In this case there is no failure information available during the 1st year of operation. Therefore, this machine cannot contribute any information before 365 days of age but will factor into the calculation only after 365 days. One could also have interval or window censoring that is dealt with extensively in [15]. The MCF accounts for gaps in information by appropriately normalizing by the number of machines at risk.

The example below illustrates a step by step calculation of the MCF for three systems.



Time (Hrs)	*Number of Systems at Risk*	*Failures per Machine*	*MCF*
33	3	1/3	1/3
135	3	1/3	2/3
247	3	1/3	1
300	3		1
318	2	1/2	1.5
368	2	1/2	2
500	2		2
582	1	1/1	3
700	1	1/1	3

Figure 6. Step by step calculation of the MCF

The ages of the systems at failure and censoring are first sorted by magnitude. The column of times in the table above shows the evolution of events by age. At age 33, system 1 had a failure, and since three machines operated beyond 33 hours, the fails/machine is 1/3 and the MCF is 1/3. The MCF aggregates the fails/machine at all points in time where failures happen. At 135 hours, system 2 has a failure and there are still three machines at risk in the population. Therefore the fails/machine is 1/3, and the MCF aggregate of the fails/machine at points of failure is now 2/3. Similarly at 247 hours the MCF jumps to 3/3 due to a failure of System 3. At 300 hours, system 3 drops out of the calculation and the

number of machines at risk becomes two. System 3 drops out not because it is removed (in this case) but simply because it is not old enough to contribute information beyond its current age. At 318 hours, system 1 has a failure and the fails/machine is now 1/2 since we have only two machines in the population that are contributing information. The MCF now becomes 3/3+1/2 and so on. This fairly straightforward procedure can be easily implemented in a spreadsheet.

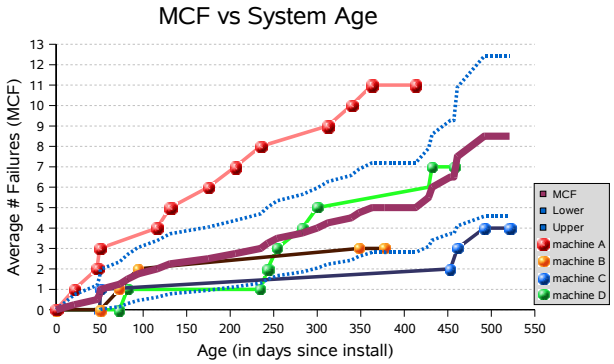


Figure 7. MCF and confidence intervals for the population in Figure 2.

Figure 7 shows the MCF for the population of machines shown in figure 2. The MCF represents the average number of failures experienced by this population as a function of age. If a new machine enters the population, the MCF represents its expected behavior.

Confidence intervals can be provided for the MCF. Nelson[3,7] provides several procedures for point-wise confidence bounds.

4.3 Identifying anomalous machines

In computer systems installed in datacenters, often a small number of misbehaving machines tend to obscure the behavior of the population at large. When the sample sizes are not too large, the simple confidence bounds can serve to graphically point out machines that have been having an excessively high number of failures compared to the average. Although it is not a statistically correct test of an outlier, overlaying the cumulative plots of individual machines with the MCF and confidence bounds tend to visually point to problem machines. Support engineers can easily identify these problem machines and propose remediation measures to the customer. More rigorous approaches for identifying these anomalous machines has been the subject of recent research. Glosup [16] proposes an approach for comparing the MCF with N machines with the MCF for (*N*-1) machines and arrive at a test statistic for determining if the omitted machine had a significant influence on the MCF. Heavlin[17] proposed a powerful alternate approach based on 2X2 contingency tables and the application of Cochran Mantel Hanzel statistic to identify anomalous machines.

4.4 Recurrence Rate vs Age

Since the MCF is the cumulative average number of failures versus time one can take the slope of the MCF curve to obtain a rate of occurrence of events as a function of time. This slope is called the recurrence rate to avoid confusion with terms like failure rate [7].

The recurrence rate can be calculated by a simple numerical differentiation procedure i.e., estimate the slope of the curve numerically. This can be easily implemented in a spreadsheet using the SLOPE(Y_1:Y_n, X_1:X_n) function where MCF is the Y axis and time is the X axis. One can take 5 or 7 adjacent points and calculate the slope of that section of the curve by a simple ruler method and plot the slope value at the midpoint. The degree of smoothing is controlled by the number of points used in the slope calculation [18]. The rate tends to amplify sharp changes in curvature in the MCF. If the MCF rises quickly, it can be seen by a sharp spike in the recurrence rate, and similarly, if the MCF is linear the recurrence rate is a flat line. When the recurrence rate is a constant, it may be a reasonable assumption to conclude that the data follows a HPP, allowing for the use of metrics such as MTBF to describe the reliability of the population.

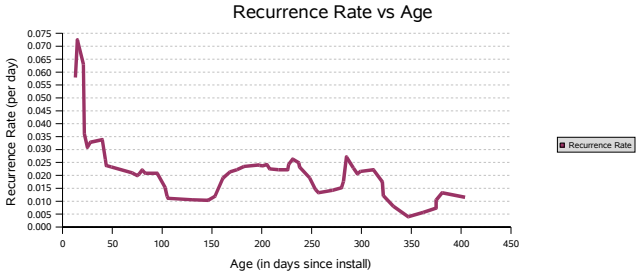
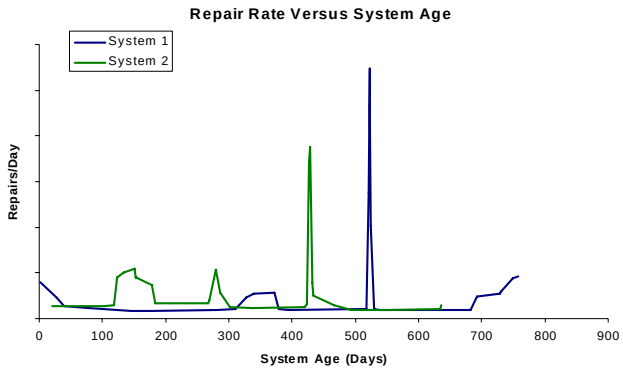


Figure 8. Example of Recurrence Rate vs Age

One can see from Figure 8 that the recurrence rate is quite high initially and drops sharply after around 50 days. Beyond



50 days the recurrence rate is fairly stable and keeps fluctuating around a fairly constant value. If the cause of failures were primarily hardware, then this would indicate potential early life failures, and one would resort to more burn-in or pre release testing. In this example, the causes of the failures were more software and configuration type issues. This problem was identified as *learning curve issues* with systems administrators. When new software products are released, there is always a learning process to figure out the correct configuration procedures, setting up the correct

directory paths, network links and so on. These activities are highly prone to human error because of lack of knowledge, improper documentation, and installation procedures. Making the installation procedure simpler and providing better training to the systems administrators resolved this issue in future installs of the product.

5. CALENDAR TIME ANALYSIS

Most reliability literature focuses on analyzing reliability as a function of the age of the system. In the case of advanced computing and networking equipment installed in datacenters, the systems undergo changes on a routine basis. There are software patches, upgrades, new applications, hardware upgrades to faster processors, larger memory, physical relocation of systems, etc. This situation can be quite different from other repairable systems like automobiles where the product configuration is fairly stable since production. Cars may undergo changes in the physical operating environment, but rarely do we see upgrades to a bigger transmission.

In datacenter systems many of the effects will not be age dependent but are a result of operating procedures that change the configuration and operational environment. These changes are typically applied to a population of machines in then datacenter and the machines can all be of different ages. It will be difficult to catch changes if the analysis is done as a function of the only of the age of the machine, but some effects will be quite evident when the events are viewed in calendar time [6]. This possibility is illustrated in Figures 9a and 9b.

Figure 9a shows the recurrence rate versus age for two systems i.e., the slopes of their cumulative plots. One can see that System 1 had a spike in the rate around 450 days while system 2 had a spike in the rate around 550 days. When looked at purely from an age perspective one can easily conclude that they were two independent spikes related only to that particular system. However in figure 9b the recurrence rate versus date shows that the two spikes coincide on the same date. This indicates clearly that we are not dealing with an age related phenomenon but an external event related to calendar time. In this case it was found that a new operating systems patch was installed on both machines at the same time, and shortly thereafter, there was an increase in the rate of failures. By plotting the date as a function of calendar time one can easily separate the age related phenomenon from the date related phenomenon.

Figure 9a. Recurrence Rates for two systems versus system age

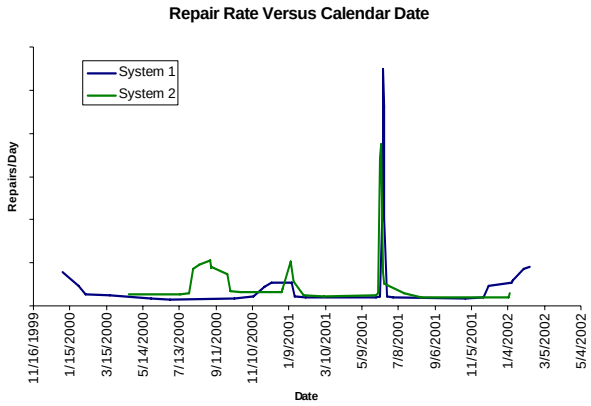


Figure 9b. Recurrence Rates for 2 systems versus Date

In order to analyze the data in calendar time, one can perform an analogous procedure by calculating the cumulative average number of fails per machine at various dates of failure. This result is called the Calendar Time Function (CTF). We begin with the date on which the first machine was installed and calculate the number of machines at risk at various dates on which events occurred. As more machines are installed, the number of machines at risk keeps increasing until the current date. The population will decrease if machines are physically removed from the datacenter at particular dates. This consideration is contrary to the machines at risk as a function of age where the number of machines will be a maximum at early ages and will start decreasing as machines are no longer old enough to contribute information. The calculation is identical to the table shown in Figure 5 except that we have calendar dates instead of age. The recurrence rate versus date is extremely important in practical applications because support engineers and customers can more easily correlate spikes or trends with specific events in the datacenter. The calculation of the recurrence rate versus date is identical to the procedure outlined for recurrence rate versus age. The SLOPE function in spreadsheets automatically converts dates into days elapsed and can calculate a numerical slope. This routine is an extremely useful and versatile function in spreadsheets. Figure 9b shows an example of a recurrence rate versus date.

6. FAILURE CAUSE PLOTS

The common approach to representing failure cause information is a Pareto chart or simple bar chart as shown in Figure 10.

One can conclude that Cause A is the highest ranking cause while causes B,C and D are all equal contributors, while Cause E is the lowest ranked cause in terms of counts. However, one can see that the Pareto chart has no time element, i.e., one cannot tell which causes are currently threats and which have been remediated. Yet this chart is one of the most popular representations in the industry. One can plot the failure causes as a function of time (age or calendar) to ascertain various

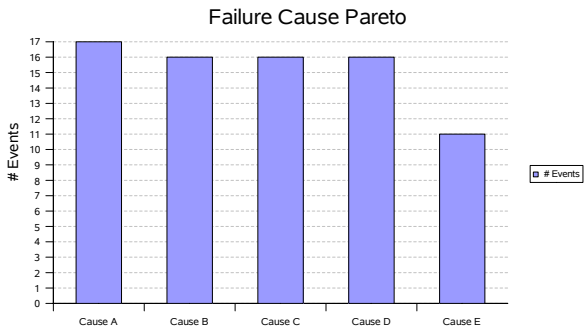


Figure 10 Example Pareto chart showing failure causes.

hypotheses. Figure 11 shows the same plot as a function of calendar time, and it is quite revealing.

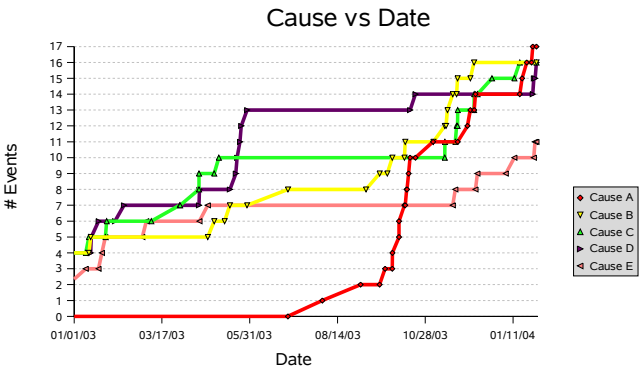


Figure 11 Failure causes versus date.

One can see that even though Cause A is only slightly higher than the other causes in Figure 10, its effect is dramatic when viewed in calendar time. It was non-existent for a while but became prevalent around September, with an extremely increasing trend. Even though Figure 10 showed that causes B,C and D were all equal contributors their contributions in time are clearly not equivalent. Cause E was shown as the lowest ranked cause but we can see in Figure 11 that even though it has been dormant for a long time, there have been a rash of cause E events in very recent times, a situation that needs to be addressed immediately.

In Figure 11 the causes are plotted simply as counts. One can definitely plot MCFs for each of the causes and normalize them by the machines at risk. One can easily imagine an MCF of all events with MCFs for individual causes plotted along with it to show the contribution of each cause to the overall MCF at various points in time.

7. MCF COMPARISONS

7.1 Comparison by Location, Vintage or Application

Often the attention is on comparing populations of machines. Customers are interested in comparing a population of machines in Datacenter X with their machines in Datacenter Y to see if there are differences in operating procedures. Engineers might be interested in comparing machines running high performance technical computing with machines running online transaction processing to see if the effect of applications is something that needs to be considered in designs. Manufacturing might be interested in comparing machines manufactured in a particular year with machines manufactured in the following year to see if there are tangible improvements in reliability. Typically people compare MTBFs of random subgroups and try to conclude if there are differences. This approach is not a correct because of inherent flaws in the MTBF metric. The MCF by virtue of being time dependent and normalization by the number of machines at risk facilitates meaningful comparisons of two or more

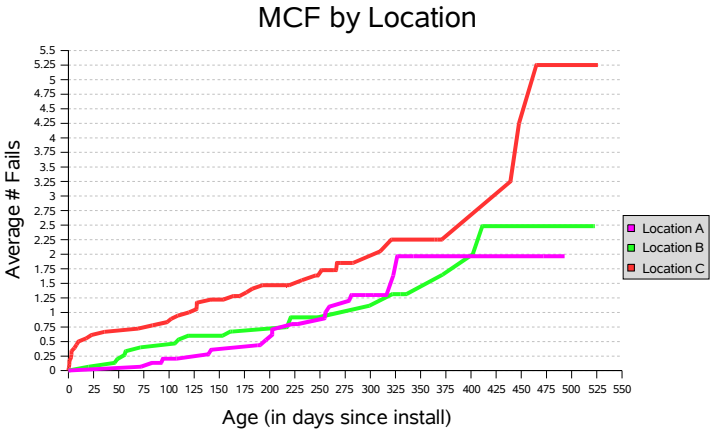


Figure 12. Comparison by Location.

populations. Figure 12 compares populations of machines belonging to the same customer but located in different datacenters.

One can see that Location C has an MCF that has been consistently higher than the other locations. The difference between the locations starts to become visually apparent after about 300 days. Investigation into the procedures at Location C revealed that personnel were not following correct procedures for avoiding electrostatic discharge while handling memory modules. This was rectified by policy and procedural changes and the reliability at this location improved. One can see that flattening of the MCF towards the end becoming parallel with the MCFs for the other locations, i.e., same slope or recurrence rate. Nelson provides procedures for assessing statistically significant difference between two MCFs [3].

7.2 Comparing recurrence rates by vintage to handle left censoring/truncation

Often times there are gaps in data collection, i.e., machines may have been installed since 1999 and data collection begins in a window starting only after 2003 because that was when service contracts were initiated. Due to the amount of missing information in the earlier ages it would be difficult to compare MCFs because we don't know how many failures have occurred before we started collecting data. One may apply statistical regression models on the MCF in the measurement window to estimate failure counts at the beginning of the window, and thereby create adjusted MCF curves as shown in Figure 13.

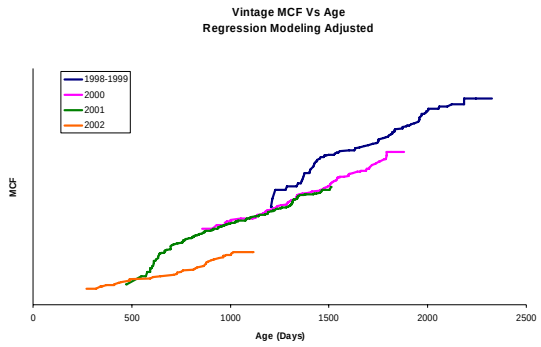


Figure 13. Adjusted MCF curve to Account for Window Truncation

Alternatively, under this situation it would be advantageous to compare recurrence rates as a function of time instead of expected number of failures. This idea is shown in Figure 14. Machines manufactured in year XXXX appear to have the highest rate of failures. Year WWWW had small spikes in the rate due to clustering of failures but otherwise has enjoyed long periods of low rates of failure. There appears to be no difference among years VVVV, YYYY and ZZZZ.

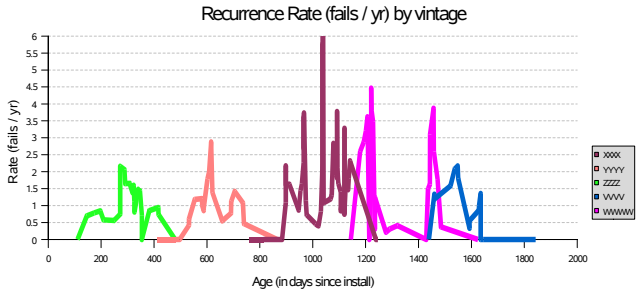


Figure 14. Comparison of recurrence rates vs age by vintage

There does not appear to be a statistically rigorous procedure to assess significant difference between two recurrence rate curves. Visual interpretation has proved to be sufficient in practical experience.

8. MCF EXTENSIONS

All parametric methods apply primarily to "counts" data i.e., they provide an estimate of the expected number of events as they are generalizations of counting processes. However, the MCF is far more flexible than just counts data. It can be used in availability analysis by accumulating average downtime instead of just average number of outage events. MCFs can be used to track service cost per machine in the form of Mean Cumulative Cost Function. It can be used to track any continuous cumulative history (in addition to counts) such as energy output from plants, amount of radiation dosage in astronauts etc. In this section we show two such applications that are quite useful for computer systems, namely downtime for availability and service cost.

8.1 Mean Cumulative Downtime Function

Availability is of paramount importance to computing and networking organizations because of the enormous costs of downtime to business. Such customers often require service level agreements on the amount of downtime they can expect per year and the vendor has to pay a penalty for exceeding the agreed upon guarantees. For such situations it is useful to plot the cumulative downtime for individual machines and get a cumulative average downtime per machine as a function of time. The calculation would proceed identical to Figure 6 except that the integer counts of failure are replaced by the actual downtime due to the event. Since availability is a function of both the number of outage events and the duration of outage events, one needs to plot the Mean Cumulative Downtime Function as well as the MCF based on just outage events. Sometimes the cumulative downtime may be small but the number of outage events may be excessive because of the amount of failure analysis overhead that goes into understanding the outage. Contracts are often drawn on both the number of outage events as well as the amount of downtime. Figure 15 shows an example mean cumulative downtime function.

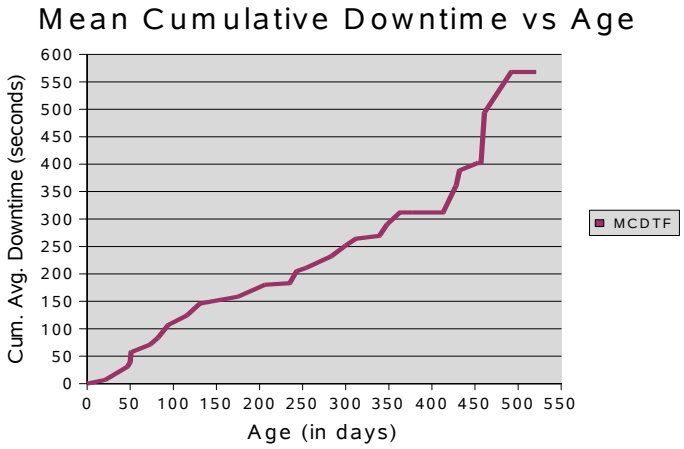


Figure 15. Mean cumulative downtime function

8.2 Mean Cumulative Cost Function

This application is quite similar to the downtime analysis mentioned in the previous section. This cost analysis could be performed by the vendor on service costs to understand one's cost structure, warranty program, pricing of subscription programs for support services, etc. The cost function could also be created by the customer to track the impact of failures on the business. The notion of downtime and the outage events can be combined to just one plot by looking at cost. The cost would be lost revenue due to loss of functionality plus all administrative costs. So in the situation of lots of outages with small amounts of downtime, the administrative costs will become noticeable. Again the calculation of the mean cumulative cost function would be similar to one of calculating an MCF for failure events except costs are used instead of counts of failures. These mean cumulative cost and downtime functions enjoy all the properties of an MCF in

terms of being efficient non parametric estimators and identifying trends in the quantity of interest.

9. CONCLUSIONS

This tutorial addressed the dangers of using summary statistics like MTBF and the important distinction between analyzing the data as a non-repairable or repairable system.

The analysis of repairable systems does not have to be difficult. Simple graphical techniques can provide excellent estimates of the expected number of failures without resorting to solving complex equations or justifying distributional assumptions. MCFs as a function of calendar time can provide important clues to non age related effects for certain classes of repairable systems. MCFs and recurrence rates are quite versatile because of their extensions to downtime and cost while parametric methods mostly handle counts type data. The approaches outlined in this tutorial have been successfully implemented at Sun Microsystems and also have found ready acceptance among people of varied backgrounds, from support technicians and executive management to statisticians and reliability engineers.

10. REFERENCES

1. H. Ascher, "A set of number is NOT a data-set", *IEEE Transactions on Reliability,* Vol 48 No. 2 pp 135-140, June 1999.
2. J. Usher, " Case Study : Reliability models and misconceptions", *Quality Engineering*, **6**(2), pp 261-271, 1993
3. W. Nelson, *Recurrence Events Data Analysis for Product Repairs, Disease Recurrences and Other Applications*, ASA-SIAM Series in Statistics and Applied Probability, 2003.
4. P.A. Tobias, D. C. Trindade, *Applied Reliability*, 2nd ed., Chapman and Hall/CRC, 1995.
5. W.Q. Meeker, L.A. Escobar, *Statistical Methods for Reliability Data*, Wiley Interscience, 1998.
6. D. C. Trindade, Swami Nathan, Simple Plots for Monitoring the Field Reliability of Repairable Systems, *Proceedings of the Annual Reliability and Maintainability Symposium (RAMS)*, Alexandria, Virginia, 2005.
7. W. Nelson, "Graphical Analysis of Recurrent Events Data", *Joint Statistical Meeting, JSM'05*, Minneapolis, Aug 2005.
8. L. H. Crow, Reliability Analysis of Complex Repairable Systems in *Reliability and Biometry,* ed. By F. Proschan and R.J. Serfling, pp. 379-410, 1974, Philadelphia, SIAM.
9. L.H. Crow, Evaluating the Reliability of Repairable Systems, *Proceedings of the Annual Reliability and Maintainability Symposium (RAMS)*, pp 275-279, 1990.
10. M. Kijima, Some Results for Repairable Systems with General Repair, *Journal of Applied Probability*, Vol. 26, pp 89-102, 1989.
11. Kijima, M. and Sumita, N. "A useful generalization of renewal theory: counting process governed by non-negative Markovian increments." *Journal of Applied Probability,* 23, 71–88, 1986.
12. Kaminskiy, M. and Krivtsov, V. "A Monte Carlo approach to repairable system reliability analysis." *Probabilistic Safety Assessment and Managemen*t, New York: Springer; p. 1063–1068, 1998.
13. A. Mettas, W. Zhao, "Modeling and Analysis of Repairable Systems with General Repair", *Proceedings of the Annual Reliability and Maintainability Symposium(RAMS)*, 2005.
14. H. Ascher, H. Feingold, *Repairable Systems Reliability: Modeling, Inference, Misconceptions and Their Causes*, 1984; Marcel Dekker.
15. J. Zuo, W. Meeker, H. Wu, "Analysis of Window-Observation Recurrence Data", *Joint Statistical Meeting, JSM'05*, Minneapolis, Aug 2005.
16. J. Glosup, "Detecting Multiple Populations within a Collection of Repairable Systems", *Joint Statistical Meeting*, Toronto, 2004.
17. W. Heavlin, Identification of Anomalous machines using CMH statistic, *Sun Microsystems Internal Report*
18. Trindade, D.C., "An APL Program to Numerically Differentiate Data", IBM TR Report 19.0361, January 12, 1975

Statistical Analysis of Field Data for Repairable Systems

David Trindade, Ph.D.
Swami Nathan, Ph.D.
Sun Microsystems Inc.

Introduction

- Reliability is a key concern for customers
  - What is the reliability ?
  - What should be the reliability ?
- MTBF (Mean Time Between Failures) is the typical metric used for communicating reliability.
- MTBFs imply many assumptions and are prone to misinterpretation.

Introduction

- Customers want to know more than MTBFs.
  - What are the causes of downtime ?
  - What can we expect going forward ?
- Pareto, stacked bar, pie, and other static charts are often used to convey analysis results.
- Such charts can mislead by hiding important effects related to time.

Outline

- MTBF Limitations
- Parametric Methods
- Repairable Systems Analysis for Age and Calendar Time
- Time Dependent Cause Plotting
- Case Studies
- Downtime and service cost plots
- Summary

MTBF

- Summary statistics such as MTBF are commonly used to report reliability.
- All failures are combined and all hours on systems are treated equally.

MTBF - Assumptions

- A ***single*** MTBF value characterizes all systems of a particular type over all time periods.
- There is ***no aging*** effect, that is, aged systems fail at the same rate as new systems.
- Events occur at a ***constant*** rate, i.e., there is no trend.

MTBF Implications

- Repairable Systems
  - Occurrence order of times between failures is ignored
  - Repairs restore system to good as new
- Renewal Process
  - Times between failures are independent and identically distributed
  - Single distribution of times between failures

MTBF Calculations

- Calculations are easy since we don't require the system ages, only the accumulated hours on systems over some time period.
- We also don't consider the ages at which failures occur or the actual times between failures. We are concerned only with the number of failures during the time period.

Modeling Assumption...

MTBF modeling applies a further assumption:

- Times ***between*** failures are assumed independent and identically distributed according to a single ***exponential*** distribution with failure (hazard) rate λ.
- Referred to as a **Homogeneous Poisson Process** (HPP).
- System qualifications and reported summary statistics are typically based on the HPP model.

Collection of HPP Systems

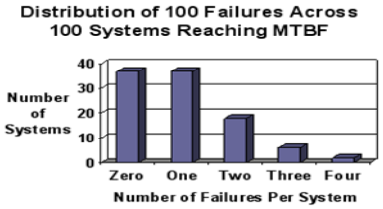
Consider a group of 100 identical HPP systems with same MTBF = 1,000 hours.

When all systems reach 1,000 hours, the expected number of failures is 1 per system or 100 total.

How are the failures actually distributed across the 100 systems ?

MTBF Implications for HPP

For a 100 repairable systems, by the time they all reach the MTBF, on the average there will be 100 failures.

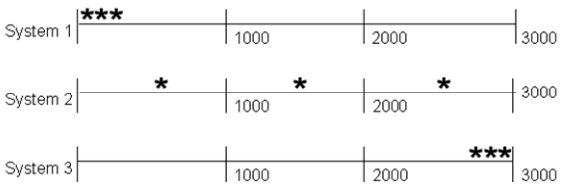


#Sys(# Fails/Sys)
37(0)
37(1)
18(2)
6(3)
2(4)

Customers can ***perceive reliability problems*** by focusing on machines with multiple failures.

MTBF Hides Information

Example with 3 failures in 3000 hours...



MTBFs are the same, implying equal reliability.

Dangers of Extrapolating to an MTBF

- During the years 1996-1998, the average annual death rate in the US for children ages 5-14 was ***20.8 per 100,000*** resident population.
- The average failure rate is thus 0.02%/yr
- The MTBF is 4800 years!!

What Customers Do Not Appreciate Hearing

- Overall your MTBF is OK.
- Your MTBF is within specification.
- The expected MTBF is around XXXX hours.

What Customer May Be Experiencing

- Recent failures after periods of calm.
- Systems that appear to have more failures than others.
- Repeated attempts to fix failures followed by additional failures.
- Frequent scheduled maintenance actions.

MTBF Assumptions: When is it OK ?

- Repairable Systems
- Renewal Process ("as good as new")
  - Times between failures are independently and identically distributed
  - Single distribution of times between failures
- Exponential Distribution Appropriate
  - Constant hazard rate (time independent)
  - No trend (constant recurrence rate RR or ROCOF)
- Homogeneous Poisson Process

MTBF-Inadequate Reliability Measure

- Valid only for a constant RR (HPP).
- Treats all system hours and all failures as equivalent and ignores age effects.
- Data is rarely checked for validity of HPP.
- Customers misinterpret MTBF usage.
- MTBFs are often quoted with imprecise definition of failure (outage or not).
- We need a better and more accurate approach to measure reliability.

Failure Rate Confusion

- The term failure rate is often used to describe the behaviour of both repairable and non-repairable systems.
- Arbitrary usage can be quite misleading because failure rates have different meanings in different situations.
- Using incorrect analysis techniques can lead to contradictory results.

Non-repairable system

- A system that is discarded upon failure.
- The lifetime is a random variable described by a ***single*** time to failure.
- For a group of systems, the lifetimes are assumed to be *independent and identically distributed, i.e., from the same population.*
- "failure rate" is the ***hazard rate*** of a lifetime distribution and is a property of the time to failure.

Repairable System

- A system that is restored to operating condition by any means short of replacing the entire system.
- Lifetime of the system is the age of the system or total hours of operation.
- The random variables of interest are the *times between failures* and the *number of failures at a particular age*.
- "failure rate" is the ***rate of occurrence*** of failures and is a property of a ***sequence*** of failure times.

ROCOF vs Hazard Rate

- ROCOF (rate of occurrence of failures) is the probability that a failure (not necessarily the first) occurs in a small time interval.
- Hazard rate is the conditional probability that a component fails in a small time interval given that it has survived from time zero until the beginning of the time interval.
- ROCOF is the *absolute* rate at which a system failures occur
- Hazard rate is the *relative* rate of failure of a component that has *survived until time T.*

Example Data

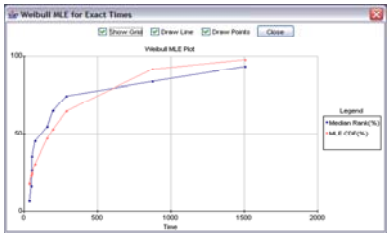
Failure Number	Failure Times	Time Between Failures
1	876	876
2	2382	1506
3	2576	194
4	2863	287
5	2912	49
6	2964	52
7	3120	156
8	3195	75
9	3249	54
10	3284	35

- The data describes the times to failure of a server due to single component X.
- Component X is replaced at every failure time.
- The times between failures are how long each new component X lived in the system.

Non-Repairable Systems Approach

- The times between failures are treated as the lifetimes of component X.
- The lifetimes can be sorted by magnitude.
- One can do distributional fitting on these ordered times to failure.
- There is no difference between component X being replaced ten times within a system compared to ten components being placed on a lifetest.
- Both testing methods assumedly provide equivalent data, and order of failures does not matter.

Weibull Fitting



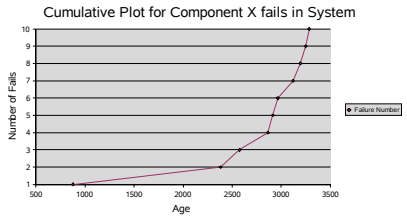
MLE Estimates :

Characteristic Life = 277.0 hours

Shape Parameter = 0.78

Shape parameter < 1 → decreasing hazard rate

Repairable Systems Approach



- A plot of number of fails versus age.
- Failures appear to be happening faster and faster as the system ages.
- Rate of occurrence of failures ROCOF or recurrence rate (RR) is actually increasing!

What is really happening ?

- Times between failures are not independent and identically distributed, i.e, time to first failure distribution is not the same as time between first and second failure.
- Order of occurrence of failures is important because components are within a repairable system.
- A degrading fan resulting in poor cooling of component X caused the increased rate of failures.

Summary

- Hazard rates apply to non-repairable systems.
- ROCOF or recurrence rates apply to repairable systems.
- The generic term "failure rate" can cause confusion because hazard rate and recurrence rates are conceptually different.
- For a repairable system, the recurrence rate can be increasing even if the hazard rate of the replacement component is decreasing or vice versa.

Parametric Methods

- Typically assume a failure intensity.
- Estimate the parameters from the data.
- Plug it into a NHPP model.
- Estimate expected number of failures versus time.
- Fancier models such as Generalized Renewal Process can be used for capturing imperfect repair and repair effectiveness.
- These methods are extremely powerful and rigorous, supported with lots of literature.

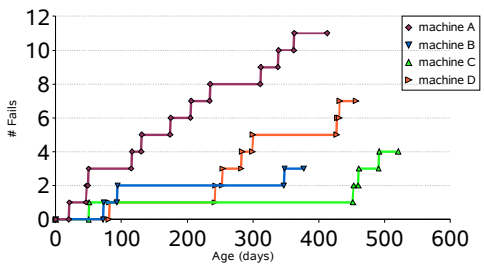
Parametric Methods

- Assumptions are rarely verified even by well intentioned practitioners.
- Can handle only "counts" data, not useful for tracking downtime and service costs.
- Too complex for communicating with management and customers.
- Some customers tend to think "*information is hidden with statistical cleverness*".
- Showing a maximum likelihood equation is not the fastest way to gain credibility with customers.

Non-Parametric Methods

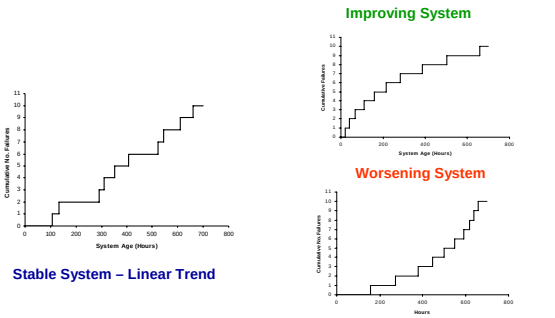
- Based on simple graphical techniques.
- Can handle "counts" as well as continuous data types.
- Does not require distributional assumptions or need solutions for complex MLEs.
- Novice practitioners relate more easily to the non-parametric approach compared to complex modeling with various assumptions.

Cumulative Plot



Cumulative plot shows failure history as number of failures (repairs) versus age of each machine.

Cumulative Plots Reveal Trends

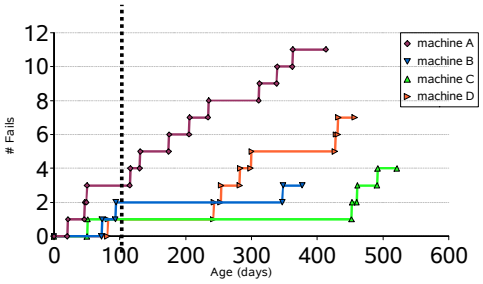


All three plots have ten fails in 700 hours leading to an MTBF of 70 hours. Clearly these behaviors are different

Mean Cumulative Function (MCF)

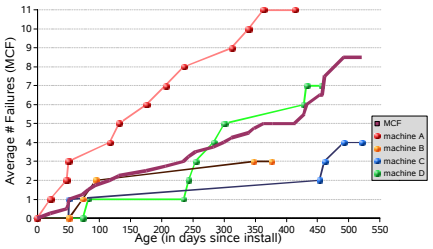
- MCF represents the average behavior of the cumulative plots across a group of systems at risk at any point in time.
- MCF can be viewed as a vertical slice across the individual cumulative plots at a time point, prior to any censoring.
- MCF is the average number of failures of a group of systems at a particular age.

MCF from Cumulative Plots



Prior to right censoring, average at each vertical time slice is the MCF. We show slice at 100 hours on four cumulative plots.

MCF and Cumulative Plots

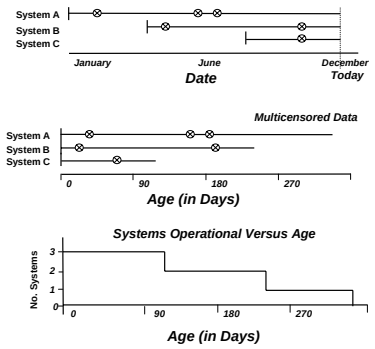


Note: Steps replaced with connecting lines

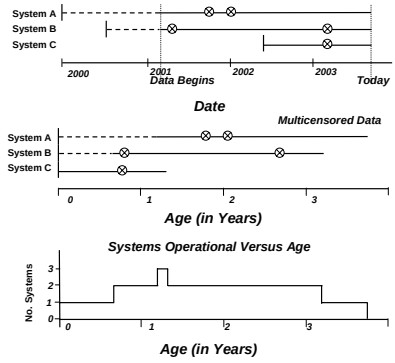
Multicensoring Issues

- Systems installed at different times during the year will have different ages (***multicensored data***).
- ***Right*** *censored data* has no failure information beyond a specific system age; e.g., if a machine is 100 days old, it contributes no information regarding reliability beyond that point.
- ***Left*** *censored data* has no information before a specific date; e.g., data collection begins on Jan 2004 and no failure history is available.
- MCF accounts for the number of systems at risk at any age or date.

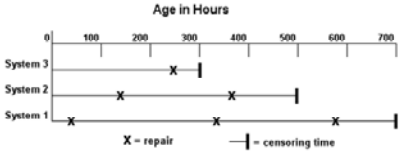
Right Censored Data



Left Censored Data



MCF Calculation (Three Systems)



Time (Hrs)	Number of Systems at Risk	Failures per Machine	MCF
33	3	1/3	1/3
135	3	1/3	2/3
247	3	1/3	1
300	3		1
318	2	1/2	1.5
368	2	1/2	2
500	2		2
582	1	1/1	3
700	1	1/1	3

Confidence bounds can be computed for MCF.

MCF Data Template

- History on every machine, including systems without failures.
- Note install, begin, failure, and end event dates.

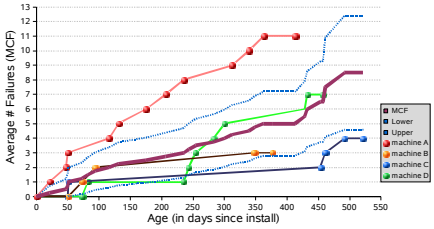
HostName	SN	Platform	Location	Install date	Event	Event Date	Event Age from install (days)	Case Id	Sys Impact	Type	Code	Cause
speedy1	555H5555	E3500	SALEM	8/11/2000	Install	8/11/2000	0					
speedy1	555H5555	E3500	SALEM	8/11/2000	begin	8/12/2000	1					
speedy1	555H5555	E3500	SALEM	8/11/2000	Failure	8/23/2001	378	55559945	I	HW	MEM	Memory Error
speedy1	555H5555	E3500	SALEM	8/11/2000	Failure	12/18/2001	495	55559864	I	SC	SC	System Config - improper sys. config
speedy1	555H5555	E3500	SALEM	8/11/2000	Failure	3/1/2002	568	55559999	N	HW	PS	Power Supply/converter
speedy1	555H5555	E3500	SALEM	8/11/2000	end	8/12/2002	731					
speedy2	666F6666	E3500	SALEM	12/8/1997	Install	12/8/1997	0					
speedy2	666F6666	E3500	SALEM	12/8/1997	begin	8/12/2000	978					
speedy2	666F6666	E3500	SALEM	12/8/1997	end	8/12/2002	1708					

MCF Calculation with Confidence Intervals

Confidence limits based on Nelson's book.

Age (i)	Event	Number at Risk (ni)	mi=1/ni	MCFi=MCFi-1 +mi	ci=mi/ni	V(MCFi)=v=v+vi-1+ci	Lower Limit	Upper Limit
0	Begin	4		0		0	0	0
21	Fail	4	0.25	0.25	0.06	0.06	-0.24	0.74
47	Fail	4	0.25	0.5	0.06	0.13	-0.19	1.19
50	Fail	4	0.25	0.75	0.06	0.19	-0.1	1.6
51	Fail	4	0.25	1	0.06	0.25	0.02	1.98
73	Fail	4	0.25	1.25	0.06	0.31	0.15	2.35
82	Fail	4	0.25	1.5	0.06	0.38	0.3	2.7
94	Fail	4	0.25	1.75	0.06	0.44	0.45	3.05
116	Fail	4	0.25	2	0.06	0.5	0.61	3.39
131	Fail	4	0.25	2.25	0.06	0.56	0.78	3.72
175	Fail	4	0.25	2.5	0.06	0.63	0.95	4.05
206	Fail	4	0.25	2.75	0.06	0.69	1.12	4.38
235	Fail	4	0.25	3	0.06	0.75	1.3	4.7
243	Fail	4	0.25	3.25	0.06	0.81	1.48	5.02
254	Fail	4	0.25	3.5	0.06	0.88	1.67	5.33
283	Fail	4	0.25	3.75	0.06	0.94	1.85	5.65
300	Fail	4	0.25	4	0.06	1	2.04	5.96
312	Fail	4	0.25	4.25	0.06	1.06	2.23	6.27
339	Fail	4	0.25	4.5	0.06	1.13	2.42	6.58
348	Fail	4	0.25	4.75	0.06	1.19	2.61	6.89
363	Fail	4	0.25	5	0.06	1.25	2.81	7.19
377	End	3		5		1.25	2.81	7.19
413	End	2		5		1.25	2.81	7.19
428	Fail	2	0.5	5.5	0.25	1.5	3.1	7.9

MCF, CI, Cumulative Plots



- In one year of operation this population has five fails/machine with an upper bound of seven.
- Machine A has higher than average failures at all ages (***anomalous system***).

Detecting anomalous machines

Naïve Confidence Intervals (from Nelson)

- If machine cumulative function is well above the upper bound, it is deemed anomalous.
- Method works well on small sample sizes (eyeball approach).
- Graphically focuses attention on machines with high failures (overall or in small time windows).
- Alternative approaches are more accurate for detecting anomalous systems.

Glosup's Method

- One machine is removed from the population and the MCF is computed.
- MCF with *N* machines is compared with MCF with (*N*-1) machines for all combinations.
- Anomalous machine is based on the differences among these MCF combinations.

Heavlin's Method

- Based on Cochran-Mantel-Hanzel statistic for 2X2 contingency tables.
- Powerful but more computation involved.
- Scripts such as JMP or S-plus can be used to perform these calculations efficiently.

Recurrence Rate

- A key characteristic of the MCF is the recurrence rate determined by the MCF slope at any point in time.
- The local slope represents the rate at which failures are occurring.
- Local slope is estimated by fitting a line to a group of points in a "window".
- Degree of smoothing is the number of points used in estimating the tangent to the MCF.
- SLOPE(Y,X) function in spreadsheets can be used to obtain recurrence rates easily.

Recurrence Rate: Step by Step

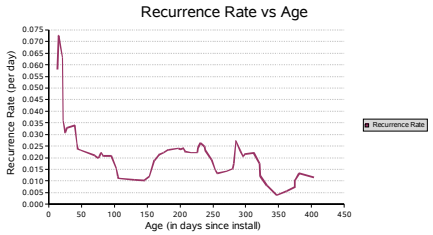
Age (i)	Event	Number at Risk (ni)	mi=1/ni	MCF=MCF-1 +mi	Recurrence Rate =Slope(MCF vs Age)
0	Begin	1			
0	Begin	2		0	
0	Begin	3		0	4.35
0	Begin	4		0	3.92
21	Fail	4	0.25	0.25	4.76
47	Fail	4	0.25	0.5	5.86
50	Fail	4	0.25	0.75	7.21
51	Fail	4	0.25	1	8.48
73	Fail	4	0.25	1.25	7.29
82	Fail	4	0.25	1.5	5.9
94	Fail	4	0.25	1.75	5.94
116	Fail	4	0.25	2	3.85
131	Fail	4	0.25	2.25	3.13
175	Fail	4	0.25	2.5	2.87
206	Fail	4	0.25	2.75	3.06
235	Fail	4	0.25	3	4.34
243	Fail	4	0.25	3.25	5.02
254	Fail	4	0.25	3.5	5.11
283	Fail	4	0.25	3.75	4.85
300	Fail	4	0.25	4	4.49
312	Fail	4	0.25	4.25	5.29
339	Fail	4	0.25	4.5	5.5
348	Fail	4	0.25	4.75	4.66
363	Fail	4	0.25	5	2.11
377	End	3		5	2.45
413	End	2		5	4
428	Fail	2	0.5	5.5	7.28

5 point slope

Recurrence Rate Vs Age

Recurrence rate peak at age 450 caused by single system.

Recurrence Rate Vs Age Example



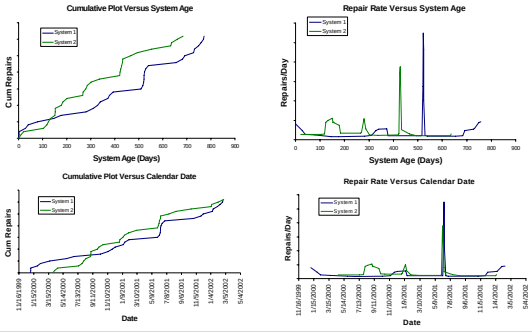
- Rate of failures is decreasing with age.
- "Peaks" or "spikes" are related to multiple fails in short time periods indicating possible misdiagnosis or repeated attempts at repairs.

Calendar Time Analysis

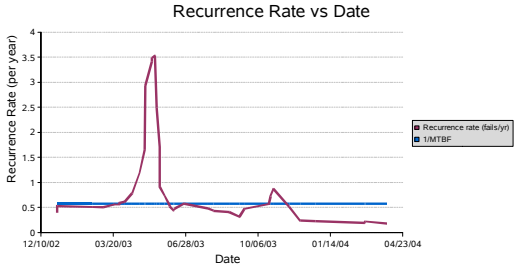
- Applications of patches, upgrades to faster processors, changing of operational policy of systems administrators, physical relocation, etc., can affect a population of machines which are at several ages on a single date or calendar window.
- These effects are captured by MCF and recurrence rates versus ***calendar time*** instead of ***age***.
- SLOPE(Y,X) in spreadsheets handles both age (numbers) or dates. Dates are automatically converted to days elapsed.

Calendar Time Analysis

Simulated data: Two systems. One installed 1/1/2001 and second 4/1/2001. Software upgrades on both systems on 6/1/2001.

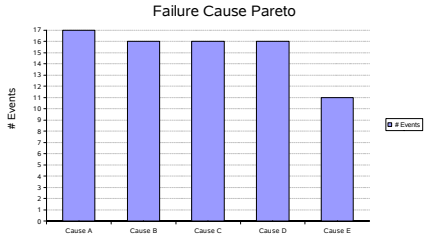


Recurrence Rate Vs Date Example



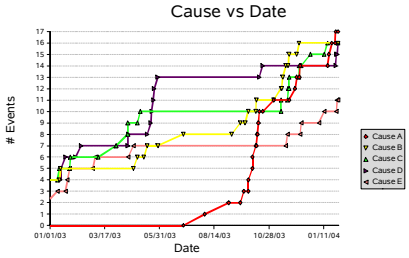
- Sharp rate increase in Apr-Jun followed by decreasing rate.
- Spike related to multiple fails on a single machine in a short time period.

Failure Cause Pareto



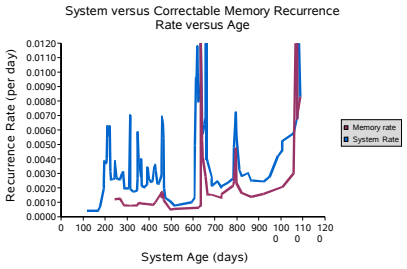
- Pareto charts are static.
- Plot does not show which causes have been remediated and which ones are current issues.

Failure Cause Plots



- Failure cause can be plotted against age or date.
- Cause D was remediated in '03 while Cause A is a more recent concern.
- Plot conveys time evolution of causes.

Failure Cause Plots and RR

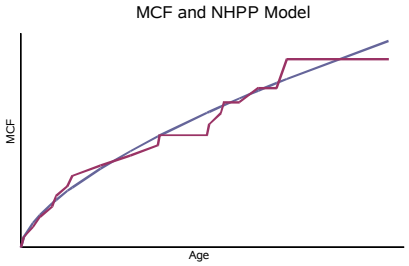


- System Recurrence Rate can be plotted against failure cause recurrence rate
- Spikes in system RR coincide with memory RR after 500 days.

MCF Comparisons

- MCF versus system age can be used to compare various (sub) populations despite multicensoring.
  - Machines at different customer sites.
  - Machines belonging to the same customer but located at different datacenters.
  - Machines of different vintages, e.g., manufactured in 2003 versus 2004.
  - Machines performing different functions, e.g., production versus development.
- Case studies will illustrate these uses.

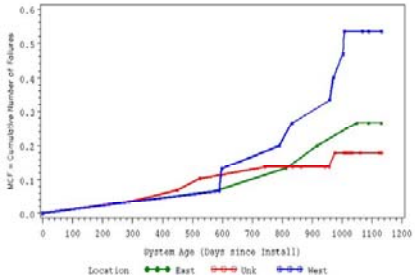
NHPP Fitting



- Example of fitting a power law model to an MCF
- The parametric fit to a non parametric MCF can be used for prediction and extrapolation.

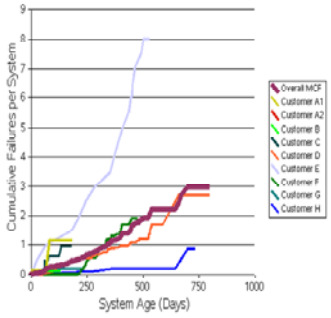
Case Study 1: Comparing Locations

Customer with east coast versus west coast data centers. Air conditioning issues caused west coast problems.

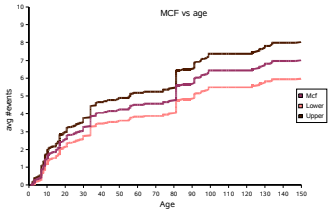


Case Study 2: Comparing Industry Segment

Comparison across customers, common platform.

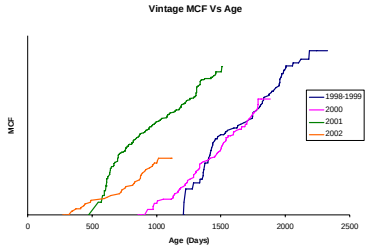


Case Study 3: Software Events



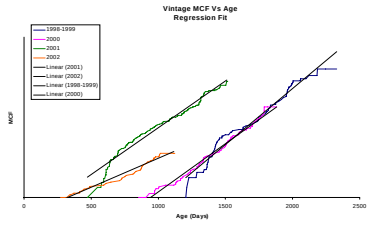
- Case of improving MCF for failures to failover.
- The rate in the first 45 days was higher than subsequent rates, hinting at learning curve/configuration type issues.
- Better installation procedures, training, and documentation significantly decreased the rate in the next version.

MCF Vintage Comparisons: Analysis Window and Left Truncation



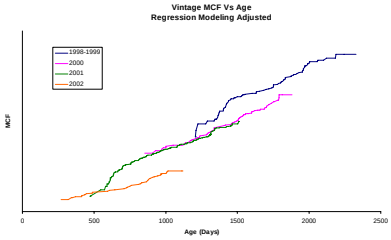
- Analysis window was 700 days.
- Different ages because of vintage mix.
- Individual MCFs start at different ages.

MCF Vintage Comparisons in Time Window: Regression Modeling



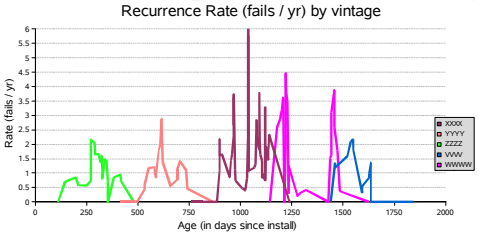
- Unknown starting point of MCF at left censoring ages makes comparing MCFs non trivial.
- HPP model gives better fit over NHPP, implying linear regression suitable.

MCF Vintage Comparisons in Time Window: Regression Modeling Adjusted



- Each vintage adjusted by intercept (extrapolation back to time zero).
- Adjusted curves represent best estimate of MCF without left truncation.

Case Study 4: Comparing Vintages Using RR

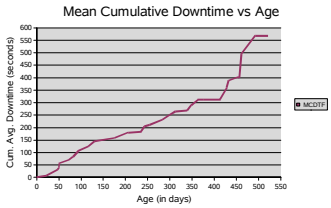


- Comparing RR of different vintages can be revealing for handling severely left censored/truncated data.
- XXXX has the highest RR. WWWW had some clustering of failures but has had long failure free periods.
- No difference among YYYY, VVVV and ZZZZ.

MCF Extensions

- MCF can be used for non "counts" data.
- Continuous cumulative histories can be tracked by cumulative plots.
- ***Availability***: Plot cumulative ***downtime*** versus age and calendar and aggregate by Mean Cumulative Downtime Function.
- These approaches can be useful for service level guarantees, determining penalties, etc.

Mean Cumulative Downtime



- Useful information for writing contracts for service level agreements, e.g., mean downtime per year increasing in the second year of operation might cause maintenance issues.
- This plot should be viewed in conjunction with a MCF of outages to look at both frequency of outages and length of outages versus time.

Mean Cumulative Costs

- Vendors can track service costs by machine and plot cumulative average cost per machine versus time.
- Result can be broken down into logistics costs, spare parts costs, etc., analogous to failure cause plots.
- Customers can plot cumulative lost revenue due to downtime and root cause investigations.
- Both plots are useful for pricing service contracts, setting penalties for not meeting service levels, etc.

Summary

- MTBF can have serious limitations for measuring reliability.
- Measuring and monitoring repairable system reliability can be greatly facilitated by the use of simple but very powerful graphical techniques.
- These methods are very revealing, easily understood by engineers and managers, and can be used to drive quality and reliability improvements.

Summary

- MCFs can be used to analyze also other data responses such as cost and downtime.
- Calendar time analysis can be quite helpful in identifying non age related issues, which are quite common in computer systems.
- Multicensored data can be easily analyzed using these techniques.
- Time dependent failure cause plots are simple and more useful than static Pareto charts.

Benefits

- These procedures can be applied very effectively for monitoring the reliability of equipment in the field.
- These methods allow engineers and service teams to quickly identify trends, anomalous systems, unusual behavior, effects of hardware and software changes, maintenance practices, and installation actions.
- These approaches have been readily accepted by people with varied statistics backgrounds from technicians and management to statisticians and reliability engineers.

Presenter's Biographical Sketch

Dr. David Trindade is a Distinguished Engineer at Sun Microsystems. Formerly he was a Senior Fellow at AMD. His fields of expertise include reliability, statistical analysis, and modeling of components, systems, and software, applied statistics, especially design of experiments (DOE), and statistical process control (SPC). He is co-author (with Dr. Paul Tobias) of the book *Applied Reliability*, 2nd ed., published in 1995. He has authored many papers and presented at many international conferences. He has a BS in physics, an MS in material sciences and semiconductor physics, an MS in statistics , and a Ph.D. in mechanical engineering and statistics. He has been an adjunct lecturer at the University of Vermont and Santa Clara University.

e-mail: david.trindade@sun.com

Dr. Swami Nathan is a Senior Staff Engineer in Sun Microsystems. His field of interest is field data analysis, statistical analysis and reliability/availability modeling of complex systems. He received his B.Tech from Indian Institute of Technology, and M.S. and Ph.D. in reliability engineering from the University of Maryland, College Park. He has authored over twenty papers and presented at international conferences and holds two patents.

email: swami.nathan@sun.com