Predicting Rare Event Failures in Space Hardware using Common Event Data

TOM SEITZ
WALDEN UNIVERSITY

Source: YouTube: Is the Butterfly Effect Real?
What is a Rare Event Failure?

- A low probability failure that causes significant damage to operational effectiveness
- For the purpose of this presentation we will assume that a rare event failure:
  - Is the result of some previous fault or combination of faults in the system that is not obvious prior to launch
  - Can be found and is not “bad luck” or “random”

Source: http://forum.kerbalspaceprogram.com
Expensive Parts lead to Expensive Reliability Paradoxes

- Space products cost more
  - Budget pressure creates desire to utilize less expensive commercial parts
  - So we wind up trying to control quality with TESTING and modeling

- **Testing is great, but ...**
  1. Part and device testing consumes some of the life of the product, making it less reliable
  2. And we often test similar components to failure, which teaches us stuff about the components we broke, not necessarily the one we actually use
  3. The actual reliability of an individual part or assembly may have more to do with the interactivity of the parts than the quality of the parts themselves.

After we break the part, we can tell you how strong it would have been
We have data...So we build models

- We test lots of parts, and hope that the part we put in the space hardware behave like the family of parts we tested.
- We overlay assumptions about distributions and failure statistics and hope we chose well.
- We use mathematics to model with the intent to predict the future.

But our models also suffer from our perspectives, assumptions and biases.
The problem with predictions ....

- “It’s tough to make predictions, especially about the future.”
  — Yogi Berra

- “The best way to predict the future is to create it.”
  - Peter Drucker
How you use data depends on where you are on the evolutionary scale

“Data isn’t as important as gumption and skill”

“Test in the Quality”

“Use Statistical Methods”

“Model”

“Leverage Data Analytics”

Source: https://www.reddit.com/r/batman

Note: Analytical processes evolve faster than the space community’s willingness to change/Adapt to the new processes
This Presentation....

- Will review several methods by which to use common event data to predict rare event problems in space hardware
- Will describe novel methods in which data analytics can supplement our knowledge of system-level performance
- Will be approximately 45 minutes long, presented by an engineer devoid of public speaking skills...
  - Please do not operate heavy machinery within 30 minutes of viewing this presentation.
  - If you feel yourself getting sleepy, please fall away from your neighbors.
Starting Simple

...HARVESTING PROCESS DATA

Source: enterprisefeatures.com
The Purpose of Testing...

1. Screening:
   ▶ Determine the minimum nominal characteristics of a component or system in a way that does not damage the system

2. Associative Testing to Failure:
   ▶ Test a sample of parts or systems to destruction to determine their limits, and then associate that behavior to the parts or systems used in space.

Testing gives us data that we can use to model performance
Rare Events Failures in Space Hardware

...Because the Maytag repair person won’t travel that far on a service call.

- We buy really expensive parts, with the presumption that they will be very reliable.
- And they are….until they aren’t.

- Space rated hardware typically is built to higher standards, and avoid known bad materials
  - Outgassing
  - Dendritic Growth
  - Radiation hardened
  - Etc.

When you care enough to buy the very best.
To prevent rare events, we must first reduce variation

‘WHILE WE WERE TRYING TO FIGURE OUT HOW TO MAKE GOOD PARTS, THEY WERE FIGURING OUT HOW TO MAKE EVERY PART THE SAME’

Former Ford executive talking about how they lost market to the Japanese
IEEE 1413 Guide for Reliability Analysis

- Suggests that there is no single “best way” to perform analysis
- Rare events are tied to the intended use of the hardware
- The key is documentation so that the strengths and weaknesses of the methods used may be understood

Source: Elerath, & Pecht (2012). IEEE 1413
Variation: Common and Special Causes

- Common Causes
  - The process has a stable and repeatable distribution over time.
    - “In a state of statistical control”
    - “In statistical control”
    - “in control”

- Special (Assignable) Causes
  - Special Causes make the overall process distribution change
    - “The process output is not stable over time”
    - “Affect the process output in unpredictable ways”
    - “Can be beneficial or detrimental to the process output”
Rare Event Special Cause

- Basic SPC can detect events in the frequency range of 3/1000

- Pros:
  - Typically utilizes data already available in production
  - Can be tied together to provide system view of quality
  - A powerful indicator of variance-induced failure

- Cons:
  - Requires stable process
  - Largely dependent on distributions used
  - Difficult to assess failures at occurrence rates <10^{-3}

Source: Alibaba.com
A Normal Curve allows us to predict likelihood of occurrence

- 99.9999998% of parts/assys within 6 stdev of the mean
- 99.73% of parts/assys within 3 stdev of the mean
- 68.26% of parts/assys within 1 stdev of the mean

The Vertical Scale is the cumulative probability of occurrence

This “predictive” capability is useful for modeling
SPC data informs rare events

- Straight Probability – For a stable and controlled product and process, the probability a data point will be outside the control limits is $P = 0.003$

- Cumulative Effects – the data from an SPC chart can also be used to model Bayesian relationships between data and rare event occurrence, scaling geometrically with increased number of parts and accumulated knowledge of rare events in a system (This will be discussed in a later section)

Testing forms the first line of defense and SPC allows test data to be harvested to make predictions
Signals in the data – Special Causes

Test 1  One point beyond zone A
A
B
C
C
B
A

Test 2  Nine points in a row on same side of center line
A
B
C
C
B
A

Test 3  Six points in a row, all increasing or decreasing
A
B
C
C
B
A

Test 4  Fourteen points in a row, alternating up and down
A
B
C
C
B
A

Test 5  Two out of three points in a row in zone A (one side of center line)
A
B
C
C
B
A

Test 6  Four out of five points in zone B or beyond (one side of center line)
A
B
C
C
B
A

Test 7  Fifteen points in a row in zone C (both sides of center line)
A
B
C
C
B
A

Test 8  Eight points in a row beyond zone C (both sides of center line)
A
B
C
C
B
A

Basic SPC provides clues in performance, often long before a system failure occurs
Example: Satellite MFG

- A Military satellite from the early 2000’s demonstrated intermittent failures in system level qualification
  - The problem was traced to a subcontractor assembly, whose test yields were found to be 100%
    - The subcontractor performed substantial testing, but had no formal SPC process, and treated each test independently (Pass or Fail)
  - To achieve 100% test yield, the subcontractor hand “tweaked” the assembly until it passed.
  - When plotted over time, several SPC inconsistencies were noted. This led to the eventual resolution of a soldering problem on a component.
  - Yields at the system level jumped to 100%

The first rule in rare event prevention is to utilize the information you already have
A special case of SPC charts analyze the time between rare events to determine whether such events are stable or unpredictable.

- A T-chart shows the amount of time between rare events (i.e. Time between failures).
- A G-chart measures the count of attributes between rare events (days without accident, test counts between failures).

T-Charts and G-Charts are so-called rare event SPC charts that evaluate the time between problems.

- Not as useful for space hardware due to the limited number of data points available.
- Can be useful to indicate stability of terrestrial processes used to build space hardware.

Source: Cleary, B. (2010, November 29).
These kind of charts are useful to determine if the predictability of rare events is changing, but not necessarily to predict the next event.
Special Case:

Time Series of variables with downstream success and failures

- Data that shows periodicity can be evaluated using Fourier Transformations, discerning failures from primary frequency differences (Deshpande, 2012).

Blue line: Passing vibration signature
Red line: Failing vibration signature

Linking periodicity differences in upstream tests with downstream failures can be a powerful method to prevent rare event failures.
Increasing the gain on Rare Events

- **Exponentially Weighted Moving Average (EWMA)**
  - Used to detect sustained shifts in the process mean.
  - Typically useful for processes that have high yields, and conventional individuals or averages consistently are in control.
  - EWMA tends to be robust to non-normal data, and accounts for Poisson Data, more commonly associated with rare events.

- **Cumulative Sum (CuSum)**
  - Similar to the EWMA. This method uses the current and recent past process data to detect small to moderate shifts in the process mean or variability.
  - Equally weights past and present data.

These techniques are particularly well-suited to highly reliable systems like space hardware.
Individual data SPC tells a valuable story, but does not necessarily indicate system level performance.
Notice a certain periodicity in the system behavior?
The interactive effect of components and assembly test data can be combined to form multivariate EWMA, amping up the power of troubleshooting rare events.
Section two: Inviting Analytics to the Party
Simple Analytics using Heat Mapping
Simple Visual Analytic: Heat Mapping

- Looks at the frequency by which parts or assemblies are mentioned in quality and engineering documentation.
- Presumes that more traffic indicates more likely to have problems.
- Technique is not statistical.
- But when used in space hardware, it can point to where the REAL issues may be lurking that drive reliability.

In the same way circuit bottlenecks show heat in cell phones, quality and engineering traffic can indicate quality issues on space hardware.

Source: androidcentral.com
Heat Mapping of Space Hardware

- With older systems, or those that undergo major redesign, pure statistics are challenging
  - Heat Map technique used
  - Includes counts of EVERY indication of mention of part or assembly in the quality management system
  - Numerical counts are not significant, only their relativity.
- Shows “heat” using color coding (red is hot, green is cold)
  - Each part number gets its own heat level
    - Quality documents include: Drawing Changes, Quality Notices, CAR’s, SCAR’s, email exchanges, etc.
- Product structures are broken down:
  - By major component
  - By Subassembly and Assembly

Heat Mapping indicates that problems occur where frequency is highest.
Adding Fidelity with Network Analyses

- Heat Mapping is useful
  But can be improved using network diagrams
- By capturing the severity of the quality data and the relative distance of parts from a datum, the heatmap can be expanded to show network heat, further indicating risk areas

Networks diagrams contextualize the HeatMap information
Bayesian Analytics
Another way to harvest common event data to predict rare event occurrences is to employ Bayesian rules:

\[ P( A | B ) = P( B | A ) \times P( A ) / P( B ) \]

\[ P( A | B ) \] read as “probability of A given B”, indicates a conditional probability. In this case, how likely is A if B happens.

To put it into context for rare event modeling, we can ask how likely is a rare event to happen, given that a common event (or events) has occurred.
“Brute Force” Bayesian Analysis is relatively easy...

- We first must understand what factors we “know” are important to the long-term reliability of the space hardware in question.
  - We can find this through multivariate EWMA, or through the identification of principle components in historical data.
- Minitab® statistical software does this:
- Python also does this:
- But for “Brute Force”, we mostly guess (intelligently)

Source: dudeiwantthat.com
Next, we build a spreadsheet from which to learn....

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SubAssy A</td>
<td>Test Yield</td>
<td>Rolled yield</td>
<td>0.667380406</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
</tr>
<tr>
<td>1.1</td>
<td>VGA</td>
<td>Signal Quality</td>
<td>&gt;99% STN ratio=pass</td>
<td>0.56399727</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
</tr>
<tr>
<td>1.11</td>
<td>RCVR</td>
<td>STN Value</td>
<td>Network Analyzer #4 Program Vehemen</td>
<td>0.983127332</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
</tr>
<tr>
<td>1.12</td>
<td>PWR</td>
<td>VSWR</td>
<td>Insertion Loss test</td>
<td>0.430179265</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
</tr>
<tr>
<td>1.2</td>
<td>Case</td>
<td>Visual Insp.</td>
<td>Workmanship Defects</td>
<td>0.05609208</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
</tr>
<tr>
<td>2</td>
<td>System</td>
<td>Test Yield</td>
<td>System Test</td>
<td>0.93972313</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
</tr>
<tr>
<td>2.1</td>
<td>SubAssy A</td>
<td>Test Yield</td>
<td>from line 1</td>
<td>0.185996869</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
</tr>
<tr>
<td>2.2</td>
<td>SubAssy B</td>
<td>Test Yield</td>
<td>External Suppl. Test</td>
<td>0.419800132</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
</tr>
<tr>
<td>2.3</td>
<td>SubAssy C</td>
<td>Test Yield</td>
<td>External Suppl. Test</td>
<td>0.024232388</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
</tr>
</tbody>
</table>

Agrregated Score

| P(Downstream Fault|cum data) | 0.353027624 | 0.360693645 | 0.368294469 | 0.376344756 | 0.380189412 | 0.385997145 | 0.387699394 | 0.392684334 | 0.396971449 |

Individual Elements of the system

Test Values

Bayesian effect for that element on overall Reliability

Bayesian effect prediction for overall reliability, given the performance of the individual elements

Data for individual Systems, used to update and improve future predictions
Rare Event Modeling in the Near Future
Bayesian Belief Networks

- A probabilistic model representing relationships between nodes and outcomes
- Each node has a set of possible states and is treated (for modeling) as a random variable
  - Rectifier: [DC Output, On, Off, etc.]
  - Edges of the network (lines connecting nodes) represent relations between features
  - Direction of the edges indicate causality
- All possible states of the nodes are gathered into a single conditional probability table
- When properly structured, computer software (i.e. Bayesnet, Bnlearn, Uninet, etc.) evaluates common data against rare event outcomes

Techniques like this promise to help us predict reliability
Rectifier Example

- A simple rectifier consisting of resistors and capacitors.
- Tested as simply “pass” or fail at the capacitor and resistor level.
- The completed rectifier undergoes ESS testing and is attributed passing or failing.
- 1,000 data elements for each passing and failing test are fed into a Bayesian belief network using the Java software tool SamIam (http://reasoning.cs.ucla.edu/samiam).
- Analysis of the disparate data shows previously unexplored impacts.

In this example capacitor quality is more impactful to rectifier quality.
Regressive Neural Networks

- The use of Regressive Neural Networks show promise in predicting reliability. (Xu, Wang, Liu, Guo, & Liu, 2016).
- Evaluates factors like common cause data with outcomes in an interconnected “neuron” structure and review timeline data, making the algorithm capable of learning.

- It consists of an input layer $i$, an output layer $y$, a hidden layer $h$ that connects to the other information in time, plus the corresponding weight matrices. Input to the network in time $t$ is vector $i(t)$ that represents potential failure drivers. $h(t)$ denotes the output of the hidden layer in time and maintains a history of variable attributes.
- The recurrent connections $R$ between $h(t-1)$ and $h(t)$ can propagate sequential signals, where the vector $h(t-1)$ represents the values in the hidden layer computed from the previous step.
- The activation values of the hidden and output layers are computed as:
  - Sequential Health = $h(t) = f(U(t) + Rh(t-1))$
  - Health degree probability = $y(t) = g(Vh(t))$

Fig. 1. RNN training process
Unfolding step is set to 3 in this figure.
References:


