ACCOUNTING FOR POINT ESTIMATE UNCERTAINTY IN SPACE SYSTEMS RELIABILITY & RISK ANALYSIS

Paul J Collier[(1)], Bruce C Reistle[(2)]

[(1)]NASA – Johnson Space Center, 2101 NASA Pwy, Houston, TX 77058, USA. Email:paul.j.collier@nasa.gov
[(2)] NASA – Johnson Space Center, 2101 NASA Pwy, Houston, TX 77058, USA. Email: bruce.c.reistle@nasa.gov

1. Introduction

Understanding and accounting for uncertainty in risk analysis is a critical step in the management and communication of risk in engineered systems. The component and system-level analysis to determine the probability of a negative outcome and its consequence is often quantified by a point estimate.

Many Program and Enterprise decisions involving technical concerns and issues rely on reliability engineering activities to produce quantified risk analysis to inform the decision making process. At NASA, it is common to use a Probabilistic Risk Analysis (PRA) to inform the overall risk to Loss of Mission or Loss of Crew that involves integration across all spacecraft subsystem fault trees to produce an overall probability of mission failure. The point estimate is an estimate of this overall probability and is an immediate result of a fault tree model. It is the result of a model where the probability of each event is taken to be equal to its mean. The value provides an approximation of the overall mean without running any uncertainty calculations (e.g., no sampling). Using only the point estimate can lead to a false sense of precision and the point estimate may not match the resulting mean when uncertainty is taken into consideration.

This paper will explore five conditions that can cause the PRA model mean to diverge from the point estimate and will provide engineers and managers insight into the importance of understanding uncertainty in the elements of PRA models.

2. Condition One - Lack of Convergence

A lack of convergence can cause a disparity between the point estimate and the mean. Consider a single event X that is lognormally distributed with mean 0.10 and error factor 5.0. The point estimate for this event is 0.10. Clearly if there are enough samples then the sample mean will converge to 0.10. Table 1 below shows the percent error with respect to the sample size for random samples of event X.

Samples	Mean	Percent Error
20	0.1215	21.5%
100	0.0917	8.3%
1,000	0.0970	3.0%
10,000	0.0986	1.4%
20,000	0.0990	1.0%

Table 1 - Percent Error With Respect to Sample Size

The disparity between the point estimate and the mean due to lack of convergence can be avoided by simply running more replications of the model. Reference [1] describes strategies to ensure convergence within given specifications. Generally speaking, convergence is achieved when the confidence interval about the mean is sufficiently small.

3. Condition Two - Correlated Events in AND-Gates

It is generally considered good practice to use correlation classes for similar items in the same functional area. In a fault tree, a correlation class is a group of similar events that not only have the same failure distribution but also use the exact same sampled value for each replication. That is, if X_1 and X_2 are correlated, then in each replication if X_1 has a sampled value of *x* then X_2 is also assigned a value of *x*.

Consider an AND-Gate with two correlated events. Suppose events X_1 and X_2 are in the same correlation class X, where X is lognormally distributed with mean 0.10 and error factor 5.0 (the resulting variance is 0.016).

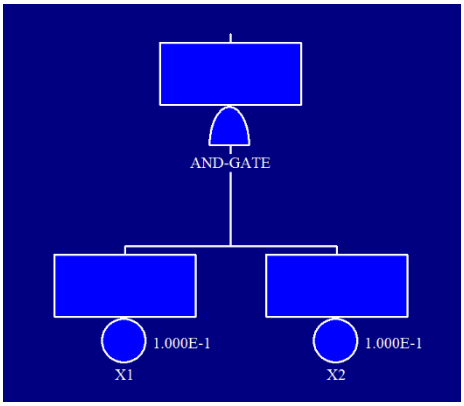

Figure 1- AND-Gate with Two Correlated Events

For each replication a value *x* is sampled, and the result for the AND-Gate is x^2. So, the expected value of the AND-Gate is the expected value of X^2, or $E(X^2)$. From Reference [2]:

$$Var(X) = E(X^2) - [E(X)]^2 \qquad \text{Eq. 1}$$

Solving for $E(X^2)$ yields:

$$E(X^2) = [E(X)]^2 + Var(X) \qquad \text{Eq. 2}$$

So, the expected value of the AND-Gate is $E(X^2) = 0.10^2 + 0.016 = 0.026$. Running 20,000 replications of the AND-Gate yields the same result, 0.026. The point estimate for the AND-Gate is $[E(X)]^2 = 0.10^2 = 0.01$.

It is evident from Eq. 2 that $E(X^2) > [E(X)]^2$ since $Var(X) > 0$. So correlated AND-Gates will return a value that is always larger than the point estimate. This effect will be more pronounced when the error factor is large (hence the variance is large) and when there are several items in the AND-Gate.

It should be noted that if the events are not correlated, that is events X_1 and X_2 have the same distribution but are sampled independently, then the expected value of the AND-Gate is equal to the point estimate.

4. Condition Three – Truncated Events

Consider an event X that is lognormally distributed with a mean of 0.5 and an error factor of 5.0. Clearly this is a large mean and there is a high probability of sampling values greater than 1.0. This is a problem when the samples represent probabilities which are constrained to be between zero and one. In a situation like this, the sampling software will typically either remove samples greater than one, or truncate the distribution, effectively cutting the tail off the distribution and renormalizing it. The two methods of truncation appear to be different but are in fact mathematically equivalent.

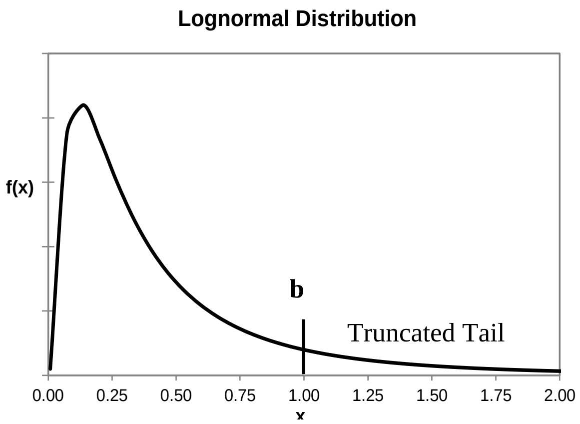

Figure 2 - Lognormal Distribution with b = 1.0

In this example, setting the truncation point, *b*, at $b = 1.0$ results in the right 24% of the distribution being removed.

Two ways to truncate a distribution are the rejection method and the truncation method.

With the rejection method, sample from X and reject values that are greater than 1.0. Doing this yields a mean of 0.33, which is noticeably lower than the untruncated mean of 0.50. This is because the largest 24% of the values are rejected.

In Excel, to obtain a sample, x, from a lognormal distribution that is right-truncated at location $b > 0$ (Reference [1]):

$$x = \mathrm{LOGNORM.INV}(\mathrm{LOGNORM.DIST}(b, \mu, \sigma, 1) \cdot U, \mu, \sigma)$$
Eq. 3

where $\mathrm{LOGNORM.INV}$ is the inverse of the lognormal distribution, $\mathrm{LOGNORM.DIST}$ is the cumulative lognormal distribution, μ is the log-mean, σ is the log-standard deviation, and U is a random number between zero and one.

Truncating the distribution also yields a mean of 0.33. This is not surprising since this method is mathematically equivalent to the rejection method.

Right-truncating a distribution will result in a lower mean than the non-truncated distribution. However, in general it is not good practice to model probabilities with a distribution that will require frequent truncation.

5. Condition Four – Quotients of Events

Consider random variables *X* and *Y*. In general, the expected value of the quotient is not equal to the quotient of the expected values, that is:

$$E\left(\frac{X}{Y}\right) \neq \frac{E(X)}{E(Y)}$$
Eq. 4

An approximation for the expected value of the quotient is (Reference [3]):

$$E\left(\frac{X}{Y}\right) \approx \frac{E(X)}{E(Y)}\left(1 + \frac{Var(Y)}{\left[E(Y)\right]^2}\right)$$
Eq. 5

Suppose events X and Y are both lognormally distributed with mean 0.10 and error factor 5.0 (the resulting variance is 0.016). Using Eq. 5:

$$E\left(\frac{X}{Y}\right) \approx \frac{0.10}{0.10}\left(1+\frac{0.016}{0.10^2}\right) = 2.60$$

Running 20,000 samples of X and Y and finding the quotient yields a mean of 2.60, as expected.

The expected value of the quotient is always greater than the quotient of the expect values. Quotients are rarely used in fault trees (perhaps as compound events) but if they are used it is clear from the example above that the results could be problematic.

6. Condition Five – Long Mission Times

Fault tree models typically sample a failure rate, $\hat{\lambda}$, and then calculate the failure probability based on some time t. Usually λ is lognormally distributed and after sampling the failure rate $\hat{\lambda}$, the failure probability P_f is calculated using the exponential distribution:

$$P_f = 1 - e^{-\hat{\lambda}t}$$
Eq. 6

Consider a lognormal distribution with mean and standard deviation both equal to $1.0E-6$ (units in hours) and suppose the operating time is 100 hours. The point estimate is:

$$P_f = 1 - e^{-1.0E-6 \cdot 5{,}000} = 1.0E-4$$
Eq. 7

Simulating 100,000 values of $\hat{\lambda}$ the corresponding values of P_f results in a mean of $1.0E-4$.

Keeping everything the same except for changing the operating time to 1,500,000 hours gives a point estimate of mean of $9.5E-2$ and a sample mean of $8.9E-2$.

Figure 3 shows the relationship between the point estimate and the mean over increasing operating time.

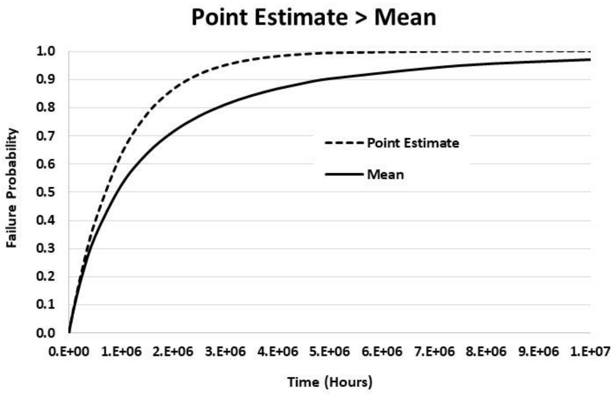

Figure 3 Relationship Between Point Estimate and Mean over Time (hours)

Notice that they both start out similarly at $t = 0$ and then reunite when t gets large. This is because both methods start at zero and coverage at 1.0, albeit at different rates.

7. Conclusion

The point estimate used to quantity the probability for overall mission failure is a key component of Space Systems and Mission risk analysis such as PRA models. Given the use of the PRA results by critical decision makers in Space Programs, it is important to understand the conditions at which the point estimate can deviate from the mean. It is good practice to identity the components of the point estimate model that directly lead to deviations to quantify the confidence in the overall PRA. When making decisions based on PRA models, decision makers should consider both the resulting failure probability and confidence in that probability. As such, when Reliability Engineers and Risk Analysts present results from a point estimate model, there should be inclusion of the five conditions described in this paper and their influence on the confidence of the final result.

APPENDIX A - REFERENCES

[1] Law, A., and Kelton, W. D. Simulation Modeling and Analysis, McGraw-Hill Inc., 1991.

[2] Hogg, R., and Tanis, E. Probability and Statistical Inference, Pearson Prentice Hall, 2010.

[3] CRC Press. OR Handbook, CRC Press LLC, 2001.

APPENDIX B – AUTHOR BIOGRAPHIES

Paul Collier *received a B.S. in Computer Science from Louisiana State University – Shreveport, Master's Certificate in Systems Engineering from California Institute of Technology, and a M.S. in Space Systems Engineering from Johns Hopkins University. He has supported human spaceflight Programs and Projects for over 20 years at NASA's Johnson Space Center in Houston, TX through various roles in Engineering, SE&I, Safety & Mission Assurance, and Cross-Program Integration. Paul is currently the Orion Program Functional Area Manager for mechanical, pyrotechnic, and landing & recovery systems in the Crew & Service Module Office.*

Bruce Reistle received a B.S and M.S in Mathematics from Virginia Tech University and a Master of Operations Research from North Carolina State University. He worked for Intel as a facility design analyst and then SAIC as a reliability analyst. Later he joined NASA's Johnson Space Center in Houston, TX as a Safety & Mission Assurance Analysis Branch where he has served for over 15 years as the Data Lead. In addition to data analysis, Bruce enjoys building ad-hoc Monte Carlo models and spreadsheet-based analysis tools.