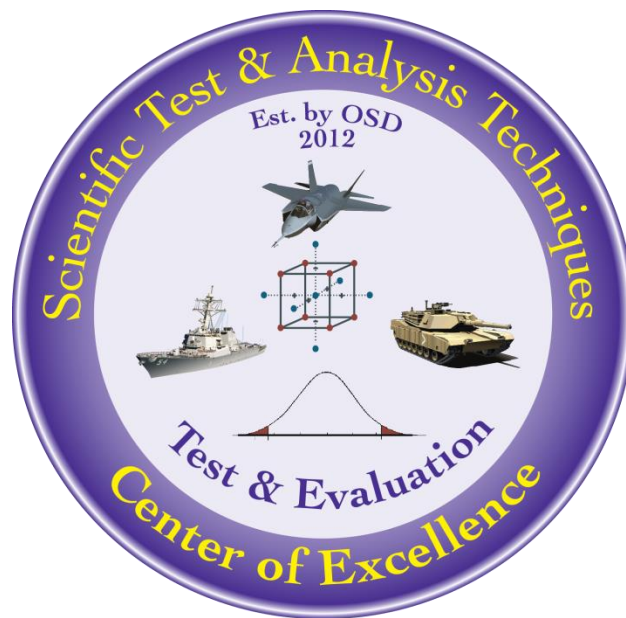# Bayesian Reliability for Complex Systems

Authored by: Sarah Burke, PhD
Michael Harman
31 August 2019

The goal of the STAT COE is to assist in developing rigorous, defensible test strategies to more effectively quantify and characterize system performance and provide information that reduces risk. This and other COE products are available at www.afit.edu/STAT.

Table of Contents

Executive Summary

Defense system complexity, combined with limited and/or costly resources, presents a challenge to obtaining a meaningful system-level reliability analysis. Configuration changes throughout development and limited operational test durations conspire to limit reliability estimates and widen confidence intervals using traditional methods, reducing information to inform decisions. Additionally, leadership will require interim and recurring reliability estimates before the full system is ever operated, a challenge given the cited issues. The Bayesian approach is a well-known method to provide a way to address multiple data sources, changing configurations, and the need for accurate, recurring estimates of system reliability. Bayesian methods are always an option for calculating reliability estimates, but they are especially useful when data collection is limited. Bayesian inference more completely informs the true variability of the reliability estimate at all levels (including the full system) even before the system has been constructed and tested. The use of modeling, simulation, and hardware-in-the-loop testing creates multiple data sources captured at different times and venues. Frequentist methods do not provide a suitable way to combine data from these various tests or configurations in a mathematically rigorous and meaningful way. This paper discusses the implementation of a Bayesian reliability estimation process for requirements of a complex system expressed as a proportion (between 0 and 1).

Keywords: Bayes, reliability, test planning, likelihood, prior, posterior, credible interval, STAT

Introduction and Motivation

Defense system complexity combined with limited and/or costly resources presents a challenge to obtaining a meaningful system-level reliability analysis. Developmental testing (DT) will generate a significant amount of component and sub-system information, but these are likely to undergo configuration changes to correct deficiencies along the way. Additionally, leadership will require recurring reliability estimates even before the full system is operated. As a part of the Scientific Test and Analysis Techniques (STAT) toolset, the Bayesian method provides a way to address multiple data sources, changing configurations, and the need for accurate, recurring estimates of system reliability.

Bayesian methods are always an option for calculating reliability estimates, but they are especially useful when data collection is limited. Frequentist reliability estimation methods rely on point estimates of individual components to populate a model. However, before the complete system is actually built, only a reliability point estimate is calculated along with confidence bounds. These confidence bounds rely on large sample sizes to obtain an estimate that is of practical use (i.e., the width of the interval is sufficiently small), which may be an issue early in testing. Bayesian inference more completely informs the true variability of the reliability estimate at all levels (including the full system) even before the system has been constructed and tested.

The sequential nature of DT likely includes changing configurations and testing of components and sub-systems long before the final system is delivered. The use of modeling, simulation, and hardware-in-the-loop testing creates multiple data sources captured at different times and venues. Frequentist methods do not provide a suitable way to combine data from these various tests or configurations in a mathematically rigorous and meaningful way, as frequentist analysis assumes testing has been done under the same conditions.

Finally, operational testing (OT) typically generates a relatively small volume of reliability data, compared to DT, which negatively impacts the calculation of a meaningful estimate. The resulting confidence bounds may be ambiguously wide and uninformative, even when the point estimate meets the requirement.

This paper discusses the implementation of a Bayesian reliability estimation process for requirements of a complex system expressed as a proportion (between 0 and 1).

Bayesian Method Overview

Bayesian inference is a method of statistical inference that combines previous testing and physical theory with limited additional test data (Meeker and Escobar, 1998). Bayesian analysis mathematically combines priors and statistical distributions of expected performance (i.e., our belief on the performance before the data is observed) with new test results (likelihood) to generate an output distribution (posterior) of the performance parameter. The posterior distribution represents updated knowledge from additional testing about the parameter. **Error! Reference source not found.** shows the components required to perform Bayesian analysis.
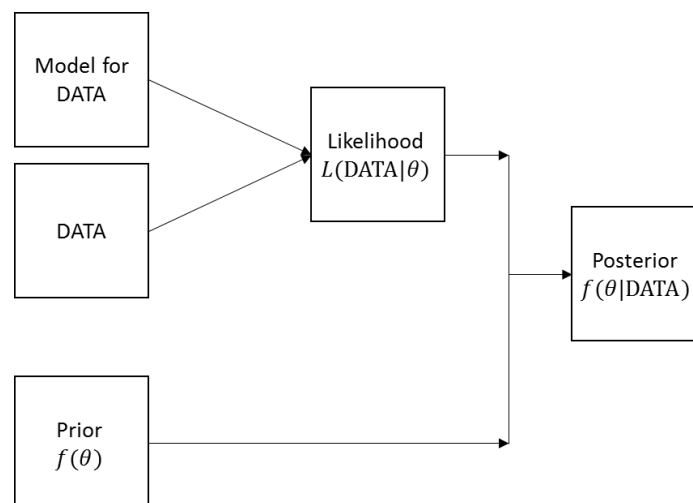

Figure 1: Bayesian Analysis (Adapted from Meeker and Escobar, 1998)

Implementation of a Bayesian Approach

Parameters in a frequentist model are assumed to be unknown constants (i.e., they're fixed, unknown values). Bayesian inference assumes the parameters in a model are also unknown, but are random variables and follow a distribution. The method is based on conditional probability where the parameters in a model are treated as random variables. The goal of Bayesian inference is to estimate the unknown parameter(s) using observed data. The major benefit is that Bayesian inference outputs a distribution of the parameter estimate instead of merely a point estimate. In addition, a Bayesian credibility interval provides the estimated range of the parameter value. This is in contrast to a frequentist confidence interval which simply computes bounds on the point estimate of the parameter.

Equation 1 shows Bayes' conditional probability equation.

$$P(\theta|D) = \frac{P(D|\theta)P(\theta)}{P(D)} \tag{1}$$

- P(D|θ) (Likelihood): Function of the data set (D) *given* a parameter (θ)
- P(θ) (Prior): Probability distribution of the parameter (θ) (represents our belief on the parameter before the dataset D is observed)
- P(θ|D) (Posterior): Probability distribution of the parameter (θ) *given* a data set (D)
- P(D) (Evidence): Probability distribution of the data set (D). This is a normalizing constant that ensures the cumulative posterior distribution sums to 1.

We must specify the population distribution (which informs the likelihood function $P(D|\theta)$) and prior distribution $P(\theta)$. Simple systems (single components) may employ the properties of the chosen distributions to determine a closed-form solution of $P(\theta|D)$ (called conjugate priors), whereas more realistic and complex systems will require Markov Chain Monte Carlo (MCMC) software methods to estimate the posterior distribution $P(\theta|D)$. The example in the next section will cover this specifically, and software code is provided in the appendices or can be obtained by emailing COE@AFIT.edu.

Likelihood

The likelihood function is the joint probability distribution of the collected sample; i.e., the probability of the observed data, given the parameters. The parameters are the only unknowns in the likelihood. The likelihood function is determined by specifying a population probability distribution. This distribution should be based on system knowledge while leveraging the features of common statistical distributions. The population distribution should resemble the nature of the data being collected, for example, pass/fail events or failure times, two common reliability metrics. Modeling failure times using the Weibull distribution is a common application.

Prior

Information for a prior can come from two primary sources: 1) past data or 2) expert opinion. Past test data may come from (in preferential order): current system test data, previous test data from a similar

system, previous component test data, current system simulation, and analysis. Analysis of past data can provide an appropriate distribution and reasonable values of the parameters. When test data is available, we can use defined specifications to determine the number of units that passed or failed in the test. For example, any unit that was within specifications would be a "pass", and any outside of the specifications would be classified a "fail." If no test data is available, but there are defined specifications for the component, we could tie a distribution's quantiles to the specification values. Similarly, physics or scientific principles may provide reasonably assumed values of the prior distribution. Historical data can be used to estimate the parameters in the prior distribution using methods such as maximum likelihood.

Method Applicability

Bayesian analysis does not apply to or replace standard reliability tasks like Failure Modes Effects and Criticality Analysis (FMECA) and root cause analysis. Bayesian analysis can be, and should be, used to estimate subsystem or component reliability as well as total system reliability. The method contained in the later example employs the system reliability block diagram (typically provided via the system engineering and/or reliability team). The program team (typically a STAT or Reliability working group) will need to determine the lowest level of component tracking desired for program use.

The overall process entails the use of initial prior information (expectations) regarding the components and the gathering of actual reliability data through testing. As testing proceeds, the last posterior becomes the prior for the next step. This process naturally includes any configuration changes into the posterior estimates. The program office and contractor must come to an agreement on how and which test article configurations will be utilized as priors in the reliability assessment. In addition to a final estimate of system reliability, interim estimates at all levels of the system, which are not readily available using a frequentist approach, can be generated throughout the test process.

Practical System Reliability Example

Components can be assessed for reliability as an individual unit, often using traditional reliability analysis methods. To quantify the reliability of a system, we can leverage the system structure, often depicted in a reliability block diagram (RBD). RBDs are part of the systems engineering process and are typically delivered by the contractor. Figure 2 shows two examples of simple systems. Figure 2a shows a three-component system where the components are in series. If any one of these components fails, then the whole system fails. Figure 2b shows a three-component system where the components are in parallel. The system fails only if all of the components fail.
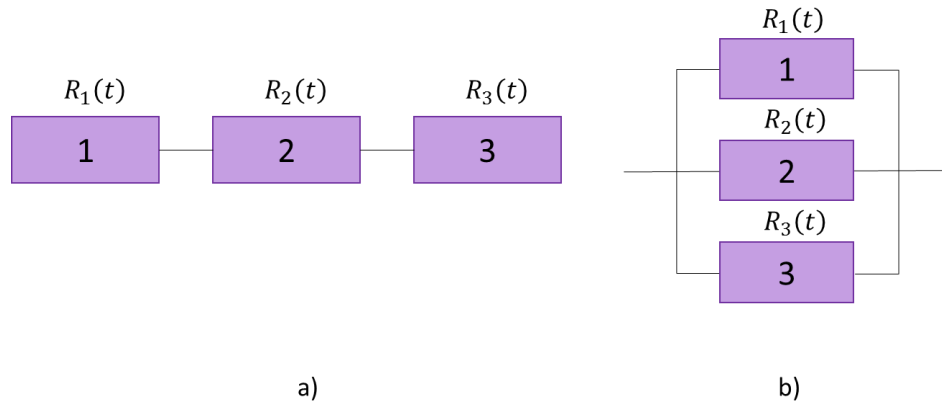
Figure 2: Reliability block diagram for a) a series system b) a parallel system

Assuming the components are independent, we can estimate the system reliability using the following formulas for a system with k compononents in series or parallel, respectively:

$$R_S(t) = R_1(t)R_2(t)\dots R_k(t)$$
$$R_S(t) = 1 - \big(1 - R_1(t)\big)\big(1 - R_2(t)\big)\dots\big(1 - R_k(t)\big)$$

These formulas can be applied for complex systems by breaking down a system into its sub-systems and components. They provide a single estimate for reliability. By utilizing the Bayesian approach and the structure of the system, we can estimate the distribution of the system reliability. The information necessary for this approach is to: 1) determine the sources of information for the component reliability assessment; 2) determine the component prior distributions; 3) collect component test data; and 4) collect system test data.

In the following subsections, we demonstrate this process for a notional three-component series system as shown in Figure 2a.

Determine Sources of Information for Components
Reliability block diagrams for a complex system can be as detailed as desired. The level of detail will depend on the amount of data available at each level of the system.

Choice of Population Distribution (Likelihood)
The population distribution should reflect the nature of the response variable. Failure times are often well modeled with exponential, Weibull, or Gamma distributions. Binary (i.e., Pass/Fail) outcomes can be modeled using a Bernoulli/binomial distribution. Count data may be modeled with the Poisson distribution. Continuous data restricted between 0 and 1 can employ a Beta distribution. Choose a distribution that make sense for the specific data. For example, if the data must be nonnegative (e.g., failure times), choose a distribution that is nonnegative. The parameters associated with the probability distribution used are further defined by the priors.

Choice of Prior

The prior defines the distribution for the likelihood parameters. For example, the population distribution may be modelled using the binomial distribution (described in the previous sub-section) which has parameters n and p. Parameter n (sample size) is comprised of whole numbers and could be modeled using a Poisson prior. Parameter p is a proportion (between 0.0 and 1.0) and could employ a Beta prior. For the cases where the parameter varies from 0 to infinity, the gamma distribution with shape > 2 and the lognormal distribution are feasible choices. This is the significant difference with a Bayesian method; the priors are not fixed values but distributions themselves.

Once the level of detail for the system has been agreed upon, the test team must then agree on what sources of data will be used to determine the prior distributions for the components. This information will be used as the sources for the prior distribution of each component. Potential sources of data in preferential order (most to least informative) to inform the prior distributions are:

- Actual component reliability testing
- Actual component qualification testing
- Similar system performance
- Specification information
- Subject matter expert input

Expert opinion can be used when no other prior test data is available. Statisticians typically develop prior information from experts by eliciting the general shape of the parameter distribution of interest (symmetrical, skewed, etc.) and typical values for given quantiles; e.g., 80% of the data resides between X and Y values. This information can be used to calculate parameters for a specific distribution.

If there is no prior information available from either past data or expert opinion, a vague prior, an approximately constant prior over the range of the parameter, can be used. When using expert elicitation, "wishful thinking" must not be used to determine the priors; vague priors are preferred when there is limited prior information (Meeker and Escobar, 1998).

Since the top-level reliability metric (in this paper's example) is expressed as a proportion, all reliability data in the Bayesian model must also be expressed as a proportion. However, since the prior data may come from a continuous measure (e.g., power, volts, time) we must transform them into a probability distribution that exists between 0 and 1. If there is a specification associated with the component, binary data can be generated based on how often the component met specifications. For example, if the component response voltage must remain below X volts to pass the test and 90% of the data is below X volts, we could determine the particular distribution whose 90th percentile is at X volts. This topic will not be discussed further here but can be accomplished in several ways and is described in Anderson-Cook et al. (2007). Anderson Cook et al. (2007) also describes how to combine information from multiple experts into one prior. The method for mapping data between domains should follow established methods and be accomplished through mutual discussion and agreement between the government

program office and contractor. Sensitivity analysis should be performed to assess the reliance of the reliability estimates on the choice of the prior assumptions. The final Bayesian model should be fairly robust to the choice of priors.

Determination of Example Component Priors

In our notional example, component 1 is a newer component and does not have previous test data available, so the test team will rely on subject matter expertise to provide information on the prior for this component. Component 2 has test data available for analysis, and the team will use this information to determine the prior. Component 3 does not have test data available, but it is a replacement for the legacy component, which does have reliability information available. This similar component will be used to determine the prior for component 3.

We assume each component reliability (population distribution) can be characterized by a binary response during component testing (i.e., the component passes or fails). Each component, therefore, has a reliability defined by the binomial distribution:

$$R_i(t) = \binom{n_i}{c_i} p_i^{c_i} (1 - p_i)^{n_i - c_i}$$

where n_i is the sample size for component i testing; c_i is the number of passes observed in component i testing; p_i is the reliability for the ith component. Each component has an associated prior for the p_i. The p_i are proportions and range from 0 to 1. The parameter n is considered a fixed, known value in this example. The only unknown prior is the proportion p_i.

For the three-component system, therefore, there are three priors to determine. Using the sources of information agreed upon by the test team, the priors p_i for each component can be defined. The goal is to describe the reliability of the component as a distribution instead of a single value. Note this difference between the likelihood function, which reflects the nature of the reliability data itself, and the prior, which serves to describe the expected distribution of the underlying parameters.

For component 1, there was no test data available; however, the system experts believe that the reliability of this component is on average 0.9 with a standard deviation of 0.1. This can be translated to a beta distribution with parameters 7.2 and 0.8. For component 2, past data provided an estimate of the distribution for the reliability. The parameters of the beta distribution were estimated using maximum likelihood of the observed test data. For component 3, the legacy system had an average reliability of 0.85 with a standard deviation of 0.1. This translates to a beta distribution with parameters 10 and 1.76. Figure 3 shows the prior distributions for each component in this example. Compared to components 1 and 2, component 3 has worse reliability.
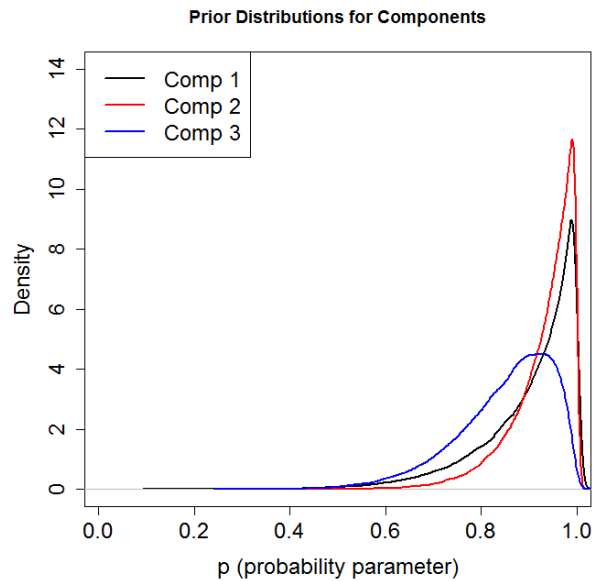
Prior Distributions for Components

Figure 3: Prior Distributions for components in three-component system

With the system structure and priors identified, we can generate a pre-test reliability prediction of the system.

Collect Component Test Data

As component/sub-system data is collected, posterior distributions for the components are estimated. These posteriors can then be used as new priors of the components. This will generate new component posteriors after any correction of deficiencies/configuration changes during component development. Using new component posteriors as priors for the next phase of testing correctly incorporates the information due to a configuration change into the following posterior analysis. This method also allows you to better inform the priors of the components with actual test data of the components. This is the beauty of Bayesian inference – as more data is collected, the data will outweigh the effect of the prior in the analysis. Finally, this approach supports the test-analyze-fix-test approach to DT. We would continue to iterate, updating the priors at the component level as component-level test data is executed.
As the component data is collected, we feed this data into the Bayesian reliability model. Using this information, we can estimate the system reliability before system-level testing has even occurred.

In the notional example, Figure 4 shows the prior and posterior distribution for the system reliability. This model incorporates component testing where component 1 had 10 tests with 2 failures, component 2 had 12 tests with 1 failure, and component 3 had 6 tests with 1 failure. Table 1 shows the mean, standard deviation, and credible interval of system reliability without component test data (prior) and with the test data (posterior).
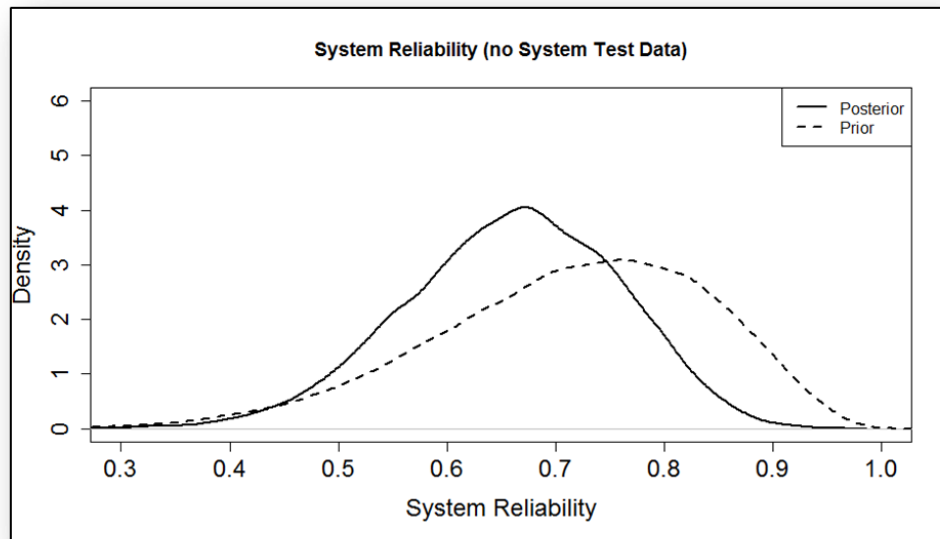
Figure 4: Posterior distribution with no system test data

Table 1: Prior and Posterior Summary Metrics (No System Test Data)

	Prior	Posterior
Mean	0.714	0.660
SD	0.127	0.100
0.025 quantile	0.435	0.449
0.975 quantile	0.920	0.837

With the incorporated component data, the system reliability estimate drops from 0.71 to 0.66. The posterior distribution also provides a 95% credibility interval of [0.45, 0.84]. This is a very wide interval; however, no system-level data has been incorporated into the model yet.

Collect System-Level Test Data

As testing of the system continues, system test data can be added into the reliability model. We use the component posteriors to feed into a model with system-level test data. In this example, 8 system-level tests are executed with 1 failure. Figure 5 shows the prior and posterior distribution of the system reliability incorporating the 8 system-level tests. Table 2 shows the summary metrics for the prior and posterior distributions.
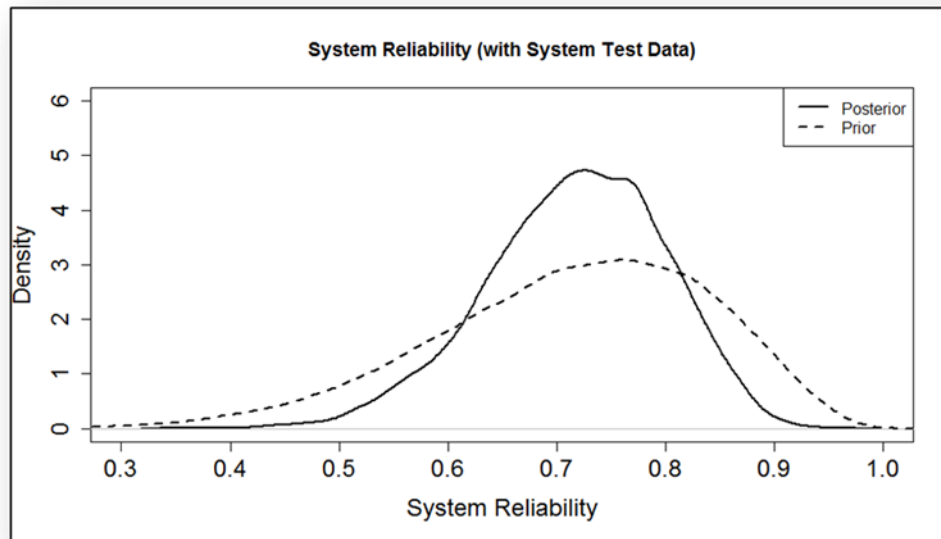
Figure 5: Prior and posterior distribution of notional system with system-level data

Table 2: Prior and Posterior Summary Metrics (With System Test Data)

	Prior	Posterior
Mean	0.714	0.717
SD	0.127	0.082
0.025 quantile	0.435	0.543
0.975 quantile	0.920	0.861

The system reliability estimate is now 0.72 with a 95% credible interval of [0.54, 0.86], as shown in Table 2. Note that incorporating the system-level data in particular provides a smaller estimate for the variance of the system reliability (the tails of the posterior distribution are narrower than the tails of the prior distribution). Similar to before, system-level test data will be counted only for the current phase of testing, since previous flight test data will have been incorporated into the posteriors.

Suppose now 24 system-level tests are completed with 1 failure. Figure 6 shows the prior and posterior distribution of the notional system. Note how the posterior distribution reflects a much higher reliability of the system, with a tighter distribution of the reliability. As more data is collected, the data will overwhelm the influence of the prior on the estimate for the system reliability.
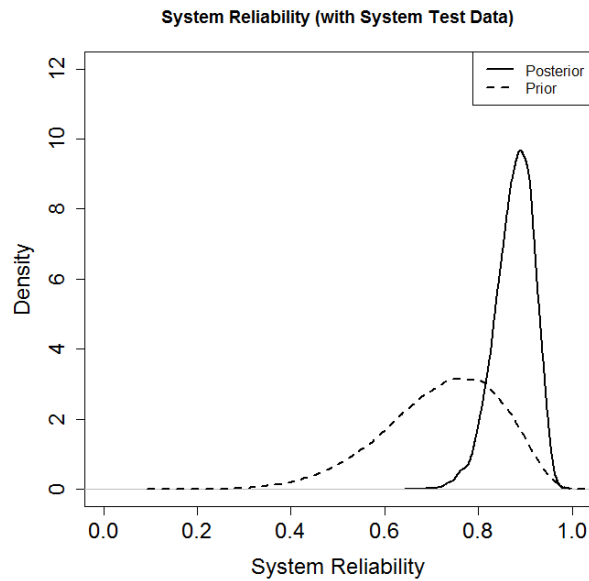
Figure 6: Prior and posterior distribution of notional system with 24 system-level tests

Required Planning Actions

In short, executing a Bayesian reliability estimation process requires a reliability block diagram (RBD), priors, test data, and code to execute the Markov Chain Monte Carlo (MCMC) simulations.

The team must develop an RBD and determine to what level reliability data will be tracked. Two critical questions are

- How far down into the system (level of detail) **can we obtain** component reliability information?
- How far down **do we want** to track it?

The RBD must be sufficiently detailed to accommodate the depth of reliability information that is desired to be tracked. Conversely, the RBD may be detailed beyond the interest of the program. In either case, the team will need to source "prior" information and test data to the level that is both acceptable to the program and compatible with the RBD.

The RBD also serves to inform the expected data collection events. If data is needed, then an event should be planned to collect it. The required data sources for "priors" need to be identified and discussed so they realistically inform the data. In cases where the prior or test data is not expressed as a proportion (e.g., a failure time or stress value), the team needs to identify and agree on a method to convert these data into probability distributions that lie between 0 and 1. Finally, the team needs to develop a formal plan to manage the RBD, reliability code/model, files, data, and output.

Applicability to Reliability Growth

Recurring reliability estimates can be recorded and plotted to track reliability growth over time. Initially, the estimates will be based on component reliability posteriors derived independently and combined mathematically. As testing progresses, the reliability estimates will become more realistic, as they will reflect the larger subsystems, until the final reliability estimates contain full system test results. The recurring process is

- Update the reliability estimate after any testing completes or priors are updated
- Record the most likely (ML) reliability value (peak) and lower bound (LB) values
- Track growth of ML and LB values throughout testing to plot trends

Data Use for Operational Testing

The ability to continually update the reliability estimate throughout testing is useful to inform trends and provide rationale to transition to operational testing (OT). In particular, the final set of DT component (or system) posteriors can be provided as priors to be used with OT data. Also, since no configuration changes are expected during OT, the full set of OT data can be used with the DT-generated component priors to generate a more complete and informed operational reliability estimate. Typically, OT would draw conclusions based on OT data alone. This can be an issue when OT is limited in time or scope, generating relatively small data sets with potentially wide (uninformative or vague) confidence bounds when using a frequentist approach. The application of DT priors and OT data more accurately reveals true system reliability expectations and should be more refined (tighter and more specific) in its estimate of the credibility intervals.

Conclusion

Frequentist reliability methods are useful when we expect a large number of tests. In DOD testing, system-level testing is often limited, leaving limited analysis if we rely on frequentist methods. Bayesian inference can be utilized, particularly with small sample sizes, by utilizing component level data and the system-level structure. Code to implement the example described in this paper is available in the appendix. As the system complexity increases, we recommend consulting with a STAT Expert to ensure the analysis is valid. For more information on the code and methodology, consult the STAT COE at COE@afit.edu.

References

Anderson-Cook, C. M., Graves, T., Hamada, M., Hengartner, N., Johnson, V. E., Reese, C. S., & Wilson, A. G. "Bayesian stockpile reliability methodology for complex systems." *Military Operations Research*, vol. 12, no. 2, 2007, pp. 25-37.

Anderson-Cook, C. M., Graves, T., Hengartner, N., Klamann, R., Wiedlea, A.C.K., Wilson, A. G., Anderson, G., Lopez, G. (2008), "Reliability Modeling using Both System Test and Quality Assurance Data." *Military Operations Research*, vol. 13, no. 3, 2008. pp. 5-18.

Meeker, W. Q., & Escobar, L. A. *Statistical methods for reliability data*. John Wiley & Sons, Inc., 1998.

Appendix A: Creating Temp Input

You can consolidate many of the key concepts introduced in this best practice to include in the reliability section of the TEMP for your program. The specifics you choose to include and level of detail will depend on how mature your system is in the acquisition cycle. You will want to ensure you include information on the reliability block diagram of the system and the data sources to be used to generate the priors. The proper execution of this effort requires a significant amount of coordination to ensure the data is continually managed and properly analyzed. Early in the acquisition cycle the description of these processes may be of high significance to TEMP reviewers looking to understand the program's plans and expectations. This information can be conveyed in a table similar to Table 3.

Table 3: Example table of data sources for priors

Subsystem	Component	Data Source
Subsystem 1	A	Ground test
	B	Component XYZ test data
	C	Expert opinion
	D	Model and simulation
Subsystem 2	A	
	B	
	C	
	D	

As information on the specifics of the design and data sources become available, the TEMP and related planning documents can be updated to reflect the new information. The STAT COE can support these efforts and can be contacted at COE@afit.edu.

Appendix B: R Code

To execute the MCMC, you must install both R and jags (just another gibbs sampler). R calls jags to perform the MCMC runs.

Jags model: SystemReliabilitySeries3Comps.txt

```
model{
# feed in component 1 data into likelihood
for(i in 1:n1){
    c1[i] ~dbern(p1)
 }
# feed in component 2 data into likelihood
 for(i in 1:n2){
    c2[i] ~dbern(p2)
 }
# feed in component 3 data into likelihood
 for(i in 1:n3){
    c3[i] ~dbern(p3)
 }
# feed in system-level data into likelihood
   for(i in 1:nS){
    cS[i] ~dbern(pS)
 }
# Define prior distributions for the three components
 p1 ~ dbeta(7.2,0.8)
 p2 ~ dbeta(22.864,1.083)
 p3 ~ dbeta(9.988,1.76)
 #p3 ~ dunif(0,1) #Example 3 (vague prior for one component)
# Define system structure; this example is a 3 component series system, so reliabilities multiply. Can
change this formula depending on the form of the system and leveraging RBD.
pS = p1*p2*p3
}
```

R code:

```
# Observed Data for component & System Tests (if there are system tests)
C1 = c(rep(1,20),rep(0,1)) # number successes and failures for Component 1
C2 = c(rep(1,15),rep(0,1)) # number successes for component 2
C3 = c(rep(1,25),rep(0,1)) # number for component 3
CS = c(rep(1,0),rep(0,0)) # number for system tests [Example 1; no system test data]
CS = c(rep(1,7),rep(0,1)) # Example 2 (have system test data; 8 tests with 1 failure)
```

```
CS = c(rep(1,23),rep(0,1)) # Example 3 (have system test data; 24 tests with 1 failure)

#Define sample sizes for component data and system test data
n1 = length(C1)
n2 = length(C2)
n3 = length(C3)
nS = length(CS)

#Sets initial values for MCMC
inits <- list(p1 = 0.5, p2 = 0.5, p3 = 0.5)

#Store data needed for JAGS in list
#jags.dat <- list(c1 = C1, c2 = C2, c3 = C3, n1 = n1, n2 = n2, n3 = n3)
jags.dat <- list(c1 = C1, c2 = C2, c3 = C3, cS = CS, n1 = n1, n2 = n2, n3 = n3, nS = nS)

#Generate JAGS model
jags <- jags.model('SystemReliabilitySeries3Comps.txt', inits = inits, n.chains = 1, data = jags.dat)

#Update starts sampler at a value n.iter into the chain (burn-in)
update(jags,n.iter = 5000)

#Obtain draws from the MCMC algorithm
sim.draws <- coda.samples(jags,thin = 5, variable.names = c('p1', 'p2', 'p3','pS'), n.iter = 50000)

#Graphical, numerical summaries (mean, median, 0.025 & 0.975 quantiles) and estimate of joint
posterior (p1, p2, p3)

summary(sim.draws)

#Compare prior vs Posterior of Component Reliability [If no test data, curves should be very similar]
plot(density(sim.draws[[1]][,1]), ylim = c(0,16), xlim = c(0.5, 1), lwd = 2, main = "Component Reliability
Posterior vs Prior",
    xlab = "p (probability parameter)", cex.axis = 1.5, cex = 1.5, cex.lab=1.5)
lines(density(p1Prior),lty=2, lwd = 2)
legend("topleft",lty = c(1,2), lwd = c(2,2), col = c("black","black"), c("Comp 1 Post","Comp 1 Prior"))
lines(density(sim.draws[[1]][,2]),  ylim = c(0,16), lwd = 2, col = "red")
lines(density(p2Prior),lty=2, col = "red", lwd = 2)
legend("topleft", lty = rep(c(1,2),2), lwd = rep(2,4), col = c("black", "black", "red", "red"),
    c("Comp 1 Post", "Comp 1 Prior", "Comp 2 Post", "Comp 2 Prior"))
lines(density(sim.draws[[1]][,3]), ylim = c(0,6), col = "blue", lwd = 2)
lines(density(p3Prior),lty=2, lwd = 2, col = "blue")
```

```
legend("topleft", lty = rep(c(1,2),3), lwd = rep(2,6), col = c("black", "black", "red", "red", 'blue', "blue"),
    c("Comp 1 Post", "Comp 1 Prior", "Comp 2 Post", "Comp 2 Prior", "Comp 3 Post", "Comp 3 Prior"))

#Compare prior vs Posterior of System Reliability
RSPost = sim.draws[[1]][,1]*sim.draws[[1]][,2]*sim.draws[[1]][,3]
RSPrior = p1Prior*p2Prior*p3Prior

plot(density(sim.draws[[1]][,4]), ylim = c(0,8), xlim = c(0, 1), lwd = 2, lty = 1,
    main = "System Reliability (with System Test Data)", xlab = "System Reliability", cex.axis = 1.5,
    cex.lab = 1.5)
lines(density(RSPrior),lty=2, lwd = 2)
legend("topright", lty = c(1,2), lwd = c(2,2), c("Posterior", "Prior"))
#lines(density(RSPost), col="red") #A spot check (Should be the same line as that in first line

#Draws from posterior to get empirical distribution of system reliability
#(system defined for 3 components in series)

summary(RSPost)
sd(RSPost)
mean(RSPost)

summary(RSPrior)
sd(RSPrior)
mean(RSPrior
```