$\mathcal{I}^2\mathcal{SDS}$

The Institute for Integrating Statistics in Decision Sciences

Technical Report TR-2012-11

June 24, 2012

Using Model-Based Systems Engineering as a Framework for Improving Test and Evaluation Activities

Eileen A. Bjorkman
Department of Engineering Management and Systems Engineering
The George Washington University

Thomas A. Mazzuchi
Department of Engineering Management and Systems Engineering
The George Washington University

Shahram Sarkani
Department of Engineering Management and Systems Engineering
The George Washington University

Using Model-Based Systems Engineering as a Framework for Improving Test and Evaluation Activities

Eileen A. Bjorkman, Shahram Sarkani, Thomas A. Mazzuchi

Department of Engineering Management and Systems Engineering, The George Washington University, Washington, D.C. 20052

[bjorkman@gwmail.gwu.edu, sarkani@gwu.edu, mazzu@gwu.edu]

Abstract

Model-based systems engineering (MBSE) approaches are based on a paradigm shift from document-centric engineering to model-based engineering. Although MBSE methods are intended to apply across the entire system life cycle, one area that has not received much attention to date is the role of test and evaluation. Test and evaluation activities provide information that reduces the uncertainty about system performance, effectiveness, and suitability. This uncertainty reduction becomes particularly important within the context of defense systems, which can cost billions of dollars. This paper describes a methodology that uses an MBSE framework and Monte Carlo simulation to define uncertainty reduction goals for test planners to use in developing test strategies and detailed test designs for evaluating technical performance parameters. As tests are completed, physical models can be updated with test data and additional analyses conducted with combat models to determine if the system meets user requirements. The methodology is demonstrated through a simple case study involving a series of tests to predict the landing performance of an aircraft.

1. Introduction

Similar to the way computer-aided design and manufacturing (CAD/CAM) techniques have revolutionized the design and manufacturing of complex components, model-based systems engineering (MBSE) approaches have the potential to vastly improve the systems engineering of modern, complex systems characterized by the integration of hardware, software, and human interaction. MBSE is "the formalized application of modeling to support systems requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases" [Crisp, 2007:15]. In particular, MBSE uses digital models to describe and represent all aspects of a system; the digital models largely replace current documents such as system specifications and drawings. Potential benefits from using MBSE approaches include enhanced communication among developers and stakeholders; reduced development risk due to continuous evaluation of requirements and design verification; improved system quality due to rigorous requirements traceability and testability; increased productivity due to the ability to quickly evaluate the impact of changing requirements; and enhanced knowledge transfer due to the standardization of design information [Friedenthal, et al., 2009]. These benefits are similar to the benefits realized by CAD/CAM techniques. For example, CAD techniques alone are capable of producing higher-quality drawings than hand-produced drawings, but unless coupled with CAM techniques, those drawings still must be interpreted by the technicians who build the final product. Producing a digital model of a part using a standard language allows automated setup and manufacturing that is less error-prone and more precise. Similarly, one can produce drawings of a system model, but the digital model produced by MBSE will be less error-prone and more precise.

In general, MBSE can improve test processes in several ways. First, enhanced communication can help test planners to better understand the system they are testing. Second, improved requirements definition and enhanced requirements traceability and testability can help test planners by providing clear test objectives with measurable outcomes. Clear test objectives make it easier to design tests that provide the right information that can be related directly back to the original system requirements. Finally, MBSE can help to define an optimum test program by determining the information that is needed in each step of the design and test processes.

This research proposes that MBSE approaches can also help determine desired uncertainty goals for information to be gained from testing and to track that information and uncertainty reduction as testing proceeds. Uncertainty reduction can then become an explicit test objective and can be used as a measure of test merit for decision makers to compare test options during test planning. Appropriate test options can be selected based on the estimated level of uncertainty reduction and the available resources (e.g., funding and schedule) instead of relying on subjective test planning techniques.

Section 2 provides a brief review of the existing literature regarding MBSE, current test planning processes used by the US Department of Defense (DoD), and technical uncertainty quantification. Section 3 discusses how testing and uncertainty reduction relates to the MBSE paradigm. Section 4 describes the proposed methodology for using technical performance measures and Monte Carlo simulation to determine uncertainty reduction goals for a test or series of tests. Section 5 applies the methodology to a case study to evaluate the landing performance for a notional aircraft. Section 6 provides conclusions and suggestions for future research.

2. Literature Review

2.1 Model-Based Systems Engineering

A model is "a physical, mathematical, or otherwise logical representation of a system, entity, phenomenon, or process" [DoD, 2010:187]. Although models have been used by engineers and scientists for hundreds of years, the concept of model-based systems engineering was first formalized by Wymore [1993]. This text provides a rigorous mathematical and formal treatment of MBSE and introduces the concept of testable system and component representations; these representations are models tested throughout the design process until the real system is available for testing. Ultimately the system test requirements must be applied to the real system. However, the text provides no practical guidance as to how to actually develop the system test requirements during the design process and how to relate them to actual testing.

With MBSE, the output of the systems engineering design process is not a set of documents but instead a system model, and the focus of subsequent systems engineering activities is on updating and refining the model as the system design matures [Friedenthal, et al., 2009]. The system model is a coherent set of models that together include all the information required to build, test, and maintain the system; it can also be integrated with more detailed engineering-level models and simulations to analyze and evaluate dynamic system performance [Friedenthal, et al., 2009]. Model-based metrics are used throughout the development process to monitor all aspects of the system design; in particular, various properties such as performance and reliability can be evaluated at any point in the process [Friedenthal, et al., 2009]. Figure 1, adapted from Friedenthal, et al. [2009] depicts the overall MBSE concept, with the system engineering model in the center. Karban, et al. [2008] point out that the current SysML parametric diagrams depicted in Figure 1 cannot be evaluated directly and must be augmented with other models for such activities as developing error budgets and evaluating the system.

In general, using model based approaches to design systems tends to move verification and validation activities (including test and evaluation) to an earlier point in the systems engineering process than occurs with traditional approaches [Piaszczyk, 2011]. MBSE can also be used with DoD Architecture Framework (DoDAF) products to generate requirements, beginning with top-level operational views and ending with detailed physical requirements

[Piaszczyk, 2011]. Although Piaszczyk [2011] mentions that DoDAF views can support testing and verification activities, no details are provided on how this might be done.

Kraft [2010] discusses a proposed framework for developing a "campaign of tests" based on MBSE concepts; the framework integrates modeling, simulation, ground testing, and flight testing with the goal to reduce late discovery of defects. Although he discusses a "variance reduction strategy" as tests progress from wind tunnels to flight testing, his focus is on reducing late defect discovery versus uncertainty reduction traced back to user requirements.

Ramos, et al. [2012] provide a good overview of MBSE and summarize various MBSE approaches in use today by describing the development approach (e.g., Vee or spiral), the flow of major tasks, the modeling language used (e.g., SysML or UML), and the software tools used. Although this is a good description of MBSE concepts, the focus is primarily on the left side of the "Vee" with no discussion of verification and validation activities; however, they do discuss the need to identify MBSE best practices, which could presumably include those activities.

Table I summarizes the MBSE-related concepts found in the literature review, along with a comparison to the CAD/CAM analogy. In general, MBSE requires a method for executing the systems engineering process, a set of system views, a language for developing the views, and tools that are compatible with the process, views and language used. This table is not intended to be exhaustive, but to provide some examples of how the various MBSE concepts can be implemented in practice. For example, a US designer of a defense system might choose to use the traditional Vee process and build the various models using SysML to create whatever DoDAF views are needed to fully document the system using the Enterprise Architect tool. A designer in the UK might do the same thing, but use MoDAF views and the CATIA tool.

Despite the fact that MBSE approaches are intended to apply to the entire life cycle of a system, a search of recent MBSE literature found that most applications currently focus on developing unambiguous requirements, consistent and coherent system designs, requirements traceability, and initial system analysis, with little focus on evaluation of the "as built" system. For example, Foster [2011] discussed the application of MBSE to a heavy lift launch vehicle system and found the MBSE approach improved communication among the geographically distributed design team and provided "unbroken traceability" through increasing levels of detailed designs; however, there was no explicit mention of test or validation activities. Although Bernard [2012] briefly mentions using MBSE models to verify and assess architectures for complex avionics systems, the focus of the paper is on fully integrating requirements engineering into an MBSE approach by transforming ambiguous natural language requirements into "well-formed" requirements. Lopes, et al. [2011] applied MBSE methodologies to the design of a "Smart Grid" energy enterprise; their focus was on understanding the Smart Grid from a system of systems perspective, with no discussion on verification, validation and testing. Van Ruijven [2012] develops information models for the process, physical, and work breakdown aspects of complex capital facilities such as ships and infrastructure; additional models can be further developed for all systems engineering domains (e.g., verification and test) but are not illustrated. Nottage and Corns [2012] briefly mention incorporating verification and test documentation into the model of a satellite, but do not elaborate further on how such documentation is developed or used during the system life cycle. Russell [2012] discusses using different MBSE views to depict requirements traceability and assist with making decisions regarding design options during the design process; although he mentions using test cases to trace and verify requirements, he does not discuss post-design validation and test activities.

Cornford, et al. [2006] bring together management aspects of a system (such as cost and schedule) and the system design using an MBSE framework to make cost, schedule, and performance trades. Although they discuss how desired capabilities and standards (e.g., components and processes) can influence design and programmatic decisions (e.g., how much testing to do), their focus is on initial tradespace decisions versus detailed test planning. Soyler and Sala-Diakanda [2010] used MBSE to document complex and interconnected disaster management system designs to evaluate them for emergent behaviors due to interactions, but there is no mention of testing actual systems. Spangelo, et al. [2012] describe using an MBSE approach to develop models of a nanosateliite that can be used by design teams to optimize nanosatellite designs and evaluate them within a mission context using external models. Measures of effectiveness are included in the system model, but there is no discussion on using these measures for test planning. Karban, et al. [2008] discuss using SysML to define dependencies between requirements and tests, and they demonstrate the use of test cases for requirements traceability, but do not discuss using this information for detailed test planning.

Two instances were found where MBSE approaches were used to develop systems and infrastructure used to support testing. Song, et al. [2007] describe using MBSE to develop an information management system for tracking the performance tests of an express train system, and McVittie, et al. [2012] used an MBSE approach to design the systems of systems infrastructure to be used for testing a space vehicle. However, neither of these approaches was integrated with an MBSE approach used for designing the system itself.

2.2 Current US DoD Test Planning Processes

There are two general approaches to planning the testing of systems. The first approach assumes that models of the system attributes of interest are available, and that we are conducting the test primarily to collect additional information to validate or improve those models and use them to make predictions about the overall system performance. In this case, according to Schrader, et al. [1993], we are "model using". This type of test is often used for those aspects of a system which have well-defined physical processes; e.g., aircraft and missile performance testing typically rely heavily on this sort of testing. In the second approach, no detailed model is available for the desired system attributes, and the purpose of the test is to collect data that can be used to build some sort of empirical model. In this case, we are "model building" [Schrader, et al., 1993]. This type of test is common during operational testing, and is also useful when phenomenon are not well understood; e.g., integrated mission systems testing may use "model building" since it is difficult to develop a detailed model of all possible system interactions.

Regardless of whether a test planner is "model using" or "model building", many current US DoD test planning processes rely on subject matter experts who use judgment and experience from testing similar systems. Explicit statements of uncertainty and statistical techniques are not routinely used for test planning or for estimating system performance [Cohen, et al., 1998]. However, in the last two decades, statistical approaches have been increasingly advocated for providing additional test discipline [Gilmore, 2010]. As a result, statistical design of experiments (DOE) and other traditional approaches, such as hypothesis testing, are now common [Kidman, et al., 2011]. However, these techniques will not work for all types of testing [Deaconu and Coleman, 2000]. To overcome some of these problems, many non-traditional approaches have been tried, including Bayesian techniques [Dezfuli, et al., 2009] and Monte Carlo simulation [Hurwitz, et al., 2011], as well as a variety of operations research techniques [Clarke and Gardner, 1995]. However, these techniques are applied primarily to increase test efficiency, with uncertainty reduction being a secondary goal, if it is a goal at all.

2.3 Uncertainty Quantification

There are many types of uncertainty associated with large defense acquisition programs; e.g., technical (performance, effectiveness, and suitability related to user needs), stakeholder, political, operational, event, safety, cost, and schedule. The focus of our research is on technical uncertainty, since that is the primary type of uncertainty directly reduced by a test (although test results can of course indirectly reduce other types of uncertainty).

A good overview on uncertainty is available in Morgan and Henrion [1990]. Although primarily written for policy analysis, this book discusses general sources of uncertainty (e.g., statistical variation, linguistic imprecision, and approximations), methods for dealing with experts to elicit subjective judgments, and an excellent compendium of ways to represent uncertainty using probability distributions and graphical techniques. Morgan and Henrion [1990] also provide three reasons for discussing the uncertainty surrounding an issue: (1) an explicit uncertainty discussion forces planners to think more carefully about their issue, (2) forcing experts to estimate their own uncertainty helps to determine how much uncertainty really exists, and (3) understanding the uncertainty associated with a previous analysis helps us to decide whether or not we should use that analysis as a starting point for our own problem.

The literature on uncertainty quantification describes three general categories of uncertainty: aleatory uncertainty, epistemic uncertainty [Parry, 1996], and ambiguity [Schrader, at al., 1993]. However, since engineers usually use the terms random error or precision for aleatory uncertainty and systematic error or bias for epistemic uncertainty [Coleman and Steele, 2009; BIPM, 2008b], we will use the terms random and systematic. Random uncertainty is the uncertainty that always exists with a specific system or phenomenon and can often be reduced, but never eliminated completely. For example, in firing a weapon at a target, it is never possible to hit the exact center of the target repeatedly due to minor variations in the overall aiming and firing process. This random uncertainty will manifest itself by target holes that are randomly spaced around the target; the spacing of these target holes can be modeled using a probability distribution. On the other hand, systematic uncertainty is uncertainty that can in theory be eliminated by learning more about a system or phenomenon. In our weapon firing example, if the sight of our weapon is misaligned, our holes will tend to appear more in one part of the target than the other (still randomly distributed due to the random uncertainty); this additional bias error is not usually modeled using a probability distribution. In theory, we can eliminate this systematic uncertainty by realigning the sight or aiming our weapon to account for it; however, in practice, we are likely to always have some systematic uncertainty remain. Ambiguity is often treated as a separate form of uncertainty in the literature; however, since ambiguity is also related to lack of system knowledge, we will treat it as systematic uncertainty for this research.

If the uncertainty of a test is to be estimated, the sources of that uncertainty must be understood and techniques must exist for quantifying the uncertainty. Sources of technical uncertainty include input and output variables [Wendelberger, 2010] and uncertainties related to modeling, such as model parameter estimates and model structure selection when more than one model is available [Burnham and Anderson, 2004]. A wide variety of statistical and subjective techniques for characterizing and estimating uncertainty are available [BIPM, 2008a; Burnham and Anderson, 2004; Coleman and Steele, 2009; Parry, 1996]. The wide variety of techniques found in the literature indicates there is not a single "best" technique for characterizing and quantifying uncertainty; the approach selected must be the best one for the problem at hand.

2.4 Literature Review Summary

Most of the current MBSE literature is focused on describing the overall vision for MBSE and shortfalls that must be overcome to make MBSE a reality. Although there is some literature on MBSE applications, it is primarily focused on the front end of the design process; to date, there has been little insight on integrating verification, validation, and test processes into the MBSE framework. Current test planning processes focus primarily on test efficiency at the expense of recognizing the need for an effectiveness-based metric for test planning. Uncertainty quantification and reduction, which can be conducted throughout the system life cycle using the MBSE framework, is proposed as a metric for verification, validation, and test activities.

3. Using Test Results to Enhance MBSE Paradigm

3.1 Relationship Between MBSE and Test and Evaluation

The use of modeling and simulation during test and evaluation is, of course, nothing new. The concept of "model-test-model" has existed for decades, but in practice has been executed unevenly across most large defense programs. Even on programs with robust simulation and test activities, there is often very little feedback between the two activities. For example, simulation products produced early in the acquisition process are often discarded once a new acquisition phase begins. In addition, many simulations are viewed as simply risk-reduction activities prior to testing and are not subsequently updated with test data. One of the major reasons for this is that there is no coherent framework that ties simulation and testing together. The MBSE approach could be the framework needed. In particular, we believe that MBSE can be used to improve the definition of test requirements, develop better test plans, streamline the test results feedback loop, and even help to define optimum test strategies for programs with significant concurrence. Each of these is discussed below.

MBSE can help define test requirements early in the design process, and to refine those test requirements as increasing levels of detail are added to a system design. For example, DoDAF uses a series of views to describe different system aspects. The Operational View (OV) depicts top-level activities and interactions among the major nodes in the overall system. In the MBSE paradigm, the OV is developed first and is then used to derive operational requirements. Three layers of Systems Views (SVs) are then derived from the OV: functional, system, and physical. The SVs are then used to derive the requirements for each system layer. Once the functional views are built, functional performance requirements can be derived for each function in the system. These functional performance requirements begin to provide the basis for a test program to verify that those requirements have been met. Although it may not be possible to specify exactly how something will be tested at this point, initial uncertainty analysis can begin to take place. An overall uncertainty budget can be allocated to each individual functional performance requirement based on how variation in each functional requirement impacts the overall system performance and effectiveness. Although this type of analysis could be conducted without an MBSE approach, in the MBSE approach the uncertainty measure can more easily become an additional performance requirement and tracked as such. These functional performance uncertainty requirements can then be used as a starting point for deriving uncertainty requirements for the detailed system level; the system level uncertainty requirements then become the starting point for the physical level uncertainty requirements. Test planners can often begin to establish test requirements based on the functional requirements. For example, if a functional requirement must be tested to a particular level of uncertainty, higher fidelity instrumentation may be needed. Thus, test planners can begin development or procurement activities to meet those requirements. Once the physical implementation of the system is known in detail, detailed test requirements and procedures can be developed based on the uncertainty requirements associated with the technical performance measures (physical and system level parameters for developmental testing and the functional level parameters for operational testing). Technical performance measures will be discussed in more detail in Section 3.2.1.

Another test activity that could benefit from MBSE is the development of detailed test plans. The current test planning process relies heavily on technical experts who interpret operational requirements documents and higher-level test planning documents to develop detailed test objectives that are then translated into detailed test designs. In addition to being error-prone, the process also requires the test planners to have the most current versions of the documents. An MBSE approach would ensure that test requirements and their associated uncertainty reduction goals would be maintained in the digital system model that all personnel would have easy access to. In addition, this model can be easily updated once actual test results are available so that test planners always have the most recent test results available to them.

MBSE also holds the promise of streamlining the test results feedback loop and improving the chances that the right system was built by feeding back test results into operational analysis models as early as possible to ensure that the system is meeting user requirements in addition to complying with specifications. Data collected from both developmental and operational tests can be used to update detailed engineering models and higher level system models; additional analysis with combat models can then be conducted to ensure that the correct system has, in fact, been built. The verification and validation process then becomes model-driven, instead of focusing primarily on interpretation of test results.

Finally, since many defense programs have significant concurrency associated with them (i.e., developmental testing often begins before the system is completely designed and built) the ideas on uncertainty and concurrent development presented by Loch and Terwiesch [1998] may have some application. An MBSE approach could be used to determine the ideal rate of uncertainty reduction and test programs could be designed to achieve that uncertainty reduction rate. Of course, it may also be discovered that the required rate of uncertainty reduction is not achievable with available test resources; decision makers could then use that information to determine whether to accept the risk, enhance the test program, or to reduce the concurrency.

Of course, all of the activities described in the previous paragraphs can be conducted without using an MBSE approach. The difference in the MBSE approach is that the system description and the various requirements (including the uncertainty requirements) become part of a system model that is captured using a standard modeling language, such as SysML or UML. As noted from the literature review, this system model is then no longer a document, but is a living digital model that can be more easily maintained and updated when a design changes or new information becomes available. For example, if the test planners determine that it is not possible with existing technology to economically measure a parameter within the desired level of uncertainty, a program manager might have three choices. One choice would be to invest in technology to improve the ability of the test to reach the desired level of uncertainty. Another choice would be to relax the uncertainty requirement for that parameter. However, before that could be done, it might be necessary to tighten the uncertainty requirement for another parameter. A third choice might be to increase the required parameter performance so the uncertainty has less impact. The MBSE approach makes it easy to conduct these tradeoff decisions and update the system model to maintain requirements consistency. This also means test planners will always be confident that they are conducting their detailed test planning based on the most current test requirements that include uncertainty as an explicit test parameter.

3.2 Using MBSE as an Uncertainty Reduction Framework for Test and Evaluation

As discussed, we believe MBSE approaches can use a system model, along with operational combat analysis models, to establish uncertainty budgets for various system measures. These uncertainty budgets can then be used to establish uncertainty goals for one or more test events.

3.2.1 Technical Performance Measures

Technical performance measurement is described as "the set of measurement activities used to provide the supplier and/or acquirer insight into progress in the definition and development of the technical solution, ongoing assessment of the associated risks and issues, and the likelihood of meeting the critical objectives of the acquirer" [Roedler and Jones, 2005:6]. From this description, we see that technical performance measures (TPMs): (1) have a time element associated with them ("progress" and "ongoing assessment"), (2) are focused on technical risk impacts on meeting critical objectives, and (3) have a probability ("likelihood") associated with them. These concepts are illustrated in Figure 2. Figure 2a is adapted from Roedler and Jones [2005] and is the typical way TPMs are depicted. A threshold value based on the TPM mean is given, along with a planned profile to reach (and possibly exceed) the threshold value; tolerance bands that decrease with time are also depicted. However, in reality, these tolerance bands do not decrease in a continuous fashion as shown in Figure 2a; instead, they usually decrease in discontinuous jumps as various uncertainty-reducing activities are applied, as depicted in Figure 2b. Figure 2b shows three notional uncertainty-reducing activities: the final design selected from the design process, a lab test, and a flight test. Each of these activities results in a discontinuous narrowing of the tolerance bands; uncertainty is not reduced further until the next uncertainty-reducing activity. Although the continuous tolerance bands in Figure 2a may be useful as a starting point, we believe that the discontinuous bands depicted in Figure 2b provide more information regarding the expected outcomes of uncertainty-reduction activities.

However, even Figure 2b can be improved. The tolerance bands as drawn do not include any explicit depiction of probability (likelihood) associated with them. As depicted in Figure 2, all outcomes on either side of the planned profiles within the tolerance bands are equally likely (a uniform probability distribution) when this may not be the case. Streit, et al. [2008] define a hierarchy of visualization approaches that can apply here. The first layer is an absolute value or an estimate that is presented with no uncertainty; the second layer is an estimate that is presented with bounds that have no probability attached to them; and the third layer is an estimate that is depicted with" bounds" (versus bands) that have a probability, membership, or belief associated with them [Streit, et al., 2008]. In Figures 2a and 2b, the first layer is represented by the planned profile line with no associated tolerances and the second layer is represented by the tolerance bands. Figure 3 suggests a way of incorporating the third layer, by making the probabilities associated with the tolerance bounds explicit. In this notional example, the uncertainty is represented as a uniform distribution through the lab test, and then becoming a normal distribution, so the likeliest values are closer to the planned profile.

One last suggestion we make regarding TPMs is that the threshold value should not always be based on the mean parameter value. Many times, an upper or lower percentile value may be more appropriate. For example, in evaluating small, high-precision air-to-surface weapons, the 90th percentile value of the impact error is often more important than the mean value. The small weapon reduces the damage incurred outside the immediate target area, but the weapon can still cause unintended damage if it misses the target. The 90th percentile value is the radius at which 90% of the weapons are expected to impact, so it provides a more conservative estimate (much larger potential damage radius) than the mean when the consequences of unintended damage are high (e.g., when targeting a building near a school or place of worship).

3.2.2 Linking TPM Uncertainty to MBSE

The overall MBSE process to develop explicit uncertainty reduction goals for various test events is depicted in Figure 4, which is derived from Figure 1 (some elements have been deleted for clarity). First, in the upper left corner of Figure 4, requirements and derived requirements should be stated in terms of uncertainties whenever possible to facilitate determining if a requirement has been met. If a requirement is stated in absolute terms, even one failure or shortfall during a test can mean that the requirement has not been met; this is usually not what the system user had in mind [Shaikh and Moore, 1997]. Using a statistical definition of a requirement helps define the test for determining if the requirement has been met (e.g., by allowing an appropriate sample size to be calculated) [Shaikh and Moore, 1997].

Second, analytic models and simulations (upper right in Figure 4), with input from postulated system designs, should be used to allocate initial uncertainty budgets for technical performance measures to meet overall requirements (lower left in Figure 4). An uncertainty budget is defined as a "statement of a measurement uncertainty, of the components of that measurement uncertainty, and of their calculation and contribution" [BIPM, 2008c:27]. Once the individual uncertainty budget components are understood, the design engineers can work with the test engineers to establish reasonable uncertainty reduction goals for specific test events (TPM chart in the middle of Figure 4). These uncertainty reduction goals, along with the parameter requirements, then form the basis for determining sample sizes, required instrumentation accuracy, test point placement, test conditions, and other planning considerations (Test Methods and Models box in Figure 4).

Once a test is conducted, a new estimate of a technical parameter will be available, along with a new estimate of the uncertainty bounds (including the probability distribution). This information can be used to update the overall system model, conduct additional analytic studies if required, and notify decision makers if there are potential risks.

4. Methodology for Estimating and Using Uncertainty Reduction Goals

4.1 Estimating Uncertainties

The process for estimating uncertainty reduction goals uses procedures described in the Guide to the Expression of Uncertainty Measurement (GUM) [BIPM, 2008b] and a supplement on Monte Carlo simulation [BIPM, 2008c]. The procedures are not repeated in detail here, but involve using either a Taylor series expansion of the measurement model (equation) or a Monte Carlo simulation of the measurement model to examine the sensitivity that individual components of the measurement have on the resulting measurement. The discussion here is focused on the random uncertainty associated with a particular model form; however, the techniques to incorporate systematic uncertainties are easily accomplished and documented in the references.

Although some measurements are taken directly, it is more common for a measurement Y to be derived through a relationship to several input variables (the measurement model):

$$Y = f(X_1, X_2, \ldots, X_N) \qquad (1)$$

The estimated standard deviation of Y can be found by estimating the standard deviation of each of the inputs u_i and combining them. For the case of independent, uncorrelated inputs, the equation for combining the individual uncertainties is:

$$u_c^2(y) = \sum_{i=1}^{N} \left(\frac{\partial f}{\partial x_i}\right)^2 u^2(x_i) \qquad (2)$$

The GUM includes additional equations for combining uncertainties in the case of correlated inputs. The Taylor series expansion method (also called the GUF, for GUM Framework) works well when the measurement model is relatively simple, but for more complicated models, the Monte Carlo method (MCM) may be more easily applied. One caution in using the MCM method is that it may result in poor estimates of individual uncertainty components in the presence of non-linearities; however, this is easily detected by determining if the root sum square of the individual component uncertainty contributions is equal to the square root of the uncertainty when all components are varied together [Solaguren-Beascoa Fernández, 2011].

4.2 Using Uncertainty Reduction Goals for Test Planning

4.2.1 Using MCM to Estimate Uncertainties

Our methodology uses MCM analysis to estimate the initial uncertainty of a particular TPM based on the uncertainties associated with the components of the TPM measurement model. MCM is a simple and intuitive technique for analyzing the impact of the uncertainty of a parameter on a system-level performance, effectiveness, or suitability measure; however, any suitable uncertainty estimation technique could be used.

We assume that MBSE methods have already been applied to the requirements analysis and system decomposition depicted in Figure 4 and that the goal TPM uncertainty has been determined as part of these processes. Using this goal uncertainty, a series of test events can be planned for reducing the various uncertainty components that will result in the goal uncertainty. The steps are summarized as follows:

Step 1. Define the measurement model(s) (e.g., determine a physics-based model that relates the input variables to the TPM).

Step 2. Determine the initial uncertainties associated with each input variable (this will be discussed further below).

Step 3. Conduct a MCM using the initial uncertainties determined from step 2 using: (a) the individual uncertainties alone to determine the impact of each individual uncertainty on the TPM and (b) all uncertainties together to determine the combined impact of each individual uncertainty on the TPM. If the root sum square from (a) is not close to the square root of (b), then investigate for potential non-linear interactions to take into account during test planning.

Step 4. Determine appropriate activities to reduce uncertainty from the initial uncertainty to the desired uncertainty; for example, simulations, laboratory tests, or flight tests, as appropriate. Based on available instrumentation, environmental factors, and analysis techniques, determine appropriate uncertainty reduction goals for each uncertainty reduction activity.

Step 5. Use uncertainty reduction goals for detailed test planning.

Step 6. After each test is completed, determine new TPM estimates and adjust future test activities if necessary to meet uncertainty reduction goals.

Step 7. When final tests are completed, use the final TPM estimate (including uncertainty) to update operations analysis models and determine if the system is operationally effective and suitable, as appropriate.

4.2.2 Estimating Initial Uncertainties

Initial uncertainties can be estimated in many ways; for example, subject matter experts or previous test data. We believe the uncertainties associated with many TPMs can be represented using either uniform or normal probability distributions, although any probability distribution can be used (e.g., triangular). Given the state of knowledge associated with most variables early in the design process, the uniform distribution may provide the best uncertainty estimate, as it results in a higher level of uncertainty than other distributions for approximately the same range of values. Once more information is available in the form of actual test data, it may be reasonable in some cases to assume that the uncertainty takes on a normal (or other) distribution; however, this will not always be the case. For example, conservative or risk-averse estimates may want to always use uniform distributions.

Figure 5 depicts a normal random variable with mean 0 and standard deviation of 1 compared to a uniform distribution from -3 to 3 with mean 0 (standard deviation of 1.732). Nearly 99.9% of the normal distribution is captured from -3 to 3, so the two distributions contain nearly the same set of possible values for the variable. However, the normal distribution is much more concentrated about the mean.

We also note that Figure 5 is not meant to imply that all uncertainties should be represented as two-sided or with equal sides if two sides are used. Many uncertainties may have restricted values (e.g., for variables that cannot go below a certain range) or may have uncertainties that are much greater on one side of the mean than the other. Fortunately, with modern statistical and simulation packages, unequal or one-sided uncertainties are easy to model.

5. Landing Roll Distance Case Study

We now present a case study that applies the proposed methodology to a single TPM, an aircraft landing roll distance. This TPM was selected for several reasons: it is usually an important aircraft performance measure, it demonstrates all of the concepts previously discussed regarding TPMs, and open-source data regarding landing distance were available in the literature. For this case study, we assume that a military aircraft is being developed that must operate from a set of runways in remote locations around the world. The takeoff and landing distances are critical to the operation of the aircraft, as the runways in remote locations may be relatively short. The landing roll consists of the phase from runway touchdown until the aircraft comes to a complete stop (with brakes applied). The landing roll distance is used, along with expected takeoff distance, to determine the runway length required for safe aircraft operations.

5.1 Overall MBSE Approach

The overall MBSE approach involves using a tool to capture an aircraft system-level model that is then further decomposed to reach the analysis of the landing roll distance. The models depicted here are simplifications of the models that would be used in a real application, but they are representative of the type of information that would be captured in the overall system model. In addition, since the focus of our research is test and evaluation, this example assumes that the up-front work on overall system requirements and initial design has been completed.

Figure 6 depicts the overall aircraft requirements, with the runway length requirement of 8000 feet (for clarity, this is the only requirement shown). Figure 7 shows the air vehicle domain, broken down into the air vehicle, the physical environment, one or two pilots, and the baggage to be carried. Several requirements are captured here. The maximum weight of the aircraft (the design goal) is 10800 lbs, along with the current uncertainty (+15%, since design weights tend to increase instead of decrease). The current estimate of the aircraft reference area (S) is also shown (170 ft^2). And finally, the total distance (SgTotal) allocated to the takeoff roll and landing roll together is depicted (7000 ft). We assume the operations analysis has determined that the landing roll distance (sea level, no wind, no slope runway conditions) should have a 99^{th} percentile value of less than 4000 feet. This allows the aircraft to accelerate to takeoff speed (assumed 3000 feet allocated for takeoff), abort the takeoff, and stop within the 4000 feet allocated for the landing roll, leaving 1000 feet as an additional margin for error. Other blocks capture runway and atmosphere information that impact the landing distance.

Figure 8 depicts the top-level hierarchy of the air vehicle structure. For our purposes, we will focus on the aerodynamic characteristics of the airframe, the engine thrust, and the braking capability of the main landing gear.

5.1 Using MCM to Estimate Uncertainties

Step 1 – Measurement Model

The landing roll distance is a function of the runway elevation, runway slope, air temperature, aircraft touchdown speed, weight, flap configuration (which impacts the lift and drag), thrust, and the braking friction coefficient. To simplify the calculations, we will also only examine one braking level (heavy) and one flap setting (100 percent, or full flaps). The model used is:

$$S_G = -\frac{W}{g\rho S(C_D - \mu C_L)} ln\left[\frac{T - \mu W}{T - \mu W - \frac{1}{2}\rho V_{td}^2 S(C_D - \mu C_L)}\right], \tag{3}$$

where

W = aircraft weight at touchdown, in pounds

V_{td} = touchdown airspeed, in ft/sec

g = gravity constant = 32.17405 ft/sec^2

T = idle thrust = 100 lbs, from prior testing

ρ = sea level density = 23.77 x 10^{-4} slug/ft^3

S = reference wing area = 170 ft^2

C_D = coefficient of drag, dimensionless, depends upon flap setting

C_L = coefficient of lift, dimensionless, depends upon flap setting

μ = braking friction coefficient, dimensionless, depends upon braking level

Step 2 -- Initial Uncertainty Estimates

Table II summarizes the distributions used in the uncertainty analysis (random uncertainty only).

Step 3 -- Uncertainty Contributions to Total Uncertainty

Table III presents the results of the Monte Carlo simulation (500000 runs). The square root of the sum of the squares of the uncertainties of the individual components is very close (within about 2 percent) to the square root of the overall uncertainty, so no further initial analysis is needed.

Step 4 -- Uncertainty Reduction Event Planning

From the Monte Carlo simulation runs, the initial 99th percentile landing distance is 4821 feet, which is well above the desired value of 4000 feet. From Table II, the two largest contributors to uncertainty are the touchdown speed and the braking coefficient; the drag coefficient also has somewhat of an impact. Touchdown speed variability is an inherent part of the landing process, so uncertainty reduction events should focus on reducing the uncertainty estimates surrounding the braking and drag coefficients. The designers and test planners note that wind tunnel testing is already planned to better estimate the coefficients of lift and drag; these estimates are expected to be very accurate, with a normal distribution having a standard deviation within 1% of the mean. Since this testing is already scheduled, it becomes the first uncertainty reduction event. The design engineers believe that a laboratory test can be conducted on the brakes which will reduce the standard deviation of the braking coefficient to about 20% of the mean value (still with a uniform distribution). Based on existing instrumentation, the test planners believe the best uncertainty reduction they can obtain for the braking coefficients after live landing tests is a standard deviation of about 7% of the mean values (normal distribution). Figure 9 captures the analysis information, including current parameters, initial uncertainties, and goal uncertainties.

The results of Monte Carlo simulation (500000 runs) of each uncertainty reduction event are summarized in Figure 10 and Table IV. These values are also used to develop the TPM chart in Figure 11. Of particular note is that the 99th percentile value after the landing test, at 4167 feet, is somewhat above the goal of 4000 feet. These values are updated in the parametrics part of the systems engineering model and used by the operations analysts to determine the impact on overall system effectiveness. The results of the analysis indicate that the aircraft should still be able to operate from about 97% of the required airfields. At this point, the program manager and the user could decide to make design modifications; however, for the purposes of our use case we will assume they decide to instead closely monitor subsequent test results.

Steps 5-7 -- Actual Test Results

The estimates for each parameter and associated uncertainties after the actual tests are shown in Table V, and the results of Monte Carlo simulations after each test with the updated parameter/uncertainty estimates are shown in Table VI. Of particular note is that after the wind tunnel test the uncertainty has actually increased! What happened? Inspection of Table V shows that the drag coefficient mean decreased significantly, from 0.12 to 0.097, accounting for the uncertainty increase. The original analysis did not include systematic uncertainties, so there was no accounting for this bias error found in the test. This result points out the need for planners to consider both random and systematic errors during uncertainty analysis, but also shows that tests do not always result in a decrease in uncertainty. This can be due to a poorly executed test, poor initial assumptions, or an incomplete uncertainty model, among other things. If uncertainty goes up after a test, the test must be carefully examined to determine if there was a problem with the test; if a problem is found, the test should be conducted again if feasible. If the test team can determine that the test was designed and executed properly, then the most recent test results can simply be used to establish a new uncertainty baseline. This may also require adjusting other uncertainty reduction activities. The new parameter and uncertainty values can be fed back into the MBSE framework to examine the impact of the new information.

In our simple example, we assume as a result of additional analysis, a new uncertainty goal of 5% is now set for the landing tests. The program manager agrees to a cost increase to install better instrumentation to achieve the new uncertainty goal. After the lab test the uncertainty and 99th percentile values decrease, and the landing test (with an actual uncertainty of 5%) confirms that the 99th percentile value is well within the desired 4000 feet.

6. Discussion and Future Work

Although MBSE approaches have much promise for improving existing systems engineering processes, to date not much attention has been paid regarding the role of test and evaluation. We presented a methodology that uses a coherent set of MBSE models to define TPM requirements in terms of probabilities and to establish uncertainty reduction goals for the test process when possible. As test events conclude, models are updated and used to make new predictions and possibly revise the uncertainty reduction goals. Test planners and program managers can use the uncertainty reduction goals to make test design decisions regarding instrumentation, number of test points, and so on to achieve the desired uncertainty reduction goals.

The modified TPM tracking charts we present provide a more realistic visual representation of the actual uncertainty reduction process versus traditional TPM charts. The use of Monte Carlo simulations makes the uncertainty predictions (means and probability distributions) easy to obtain and visualize. We also emphasize the need to use the appropriate statistical value for the TPM; many times the mean is not the appropriate threshold value.

The research here also demonstrates the need to better account for systematic uncertainties in the uncertainty budgeting and test planning processes. As we saw, the coefficient of drag did not have a large impact on the overall landing distance uncertainty when only the random uncertainty was considered; however, the systematic error discovered during the notional wind tunnel test led to an unexpected increase in uncertainty.

Additional work is needed to determine how well the concepts proposed here extend to other types of testing. We considered only uncertainty reduction as it pertains to TPM variance reduction, but variance reduction is not always the goal of a test. Sometimes we are simply trying to characterize a system, or to determine if one system has an impact on another, and so on. Additional work is also needed to evaluate the scalability of the concepts. The technique was relatively easy to apply to one TPM for one test, but it may become cost prohibitive to apply the technique to dozens of parameters and dozens of test events across a complex program.

As a last note, The Object Management Group recently released the "UML Testing Profile," a "test modeling language" extension to the UML [OMG, 2012]. Although primarily intended for software testing, the profile includes concepts such as Test Architecture, Test Behavior, Test Data, and Time Concepts that could be extended to the testing any type of system. Future research should also consider incorporating and extending the UML testing profile into MBSE methods and tools.

References

Y. Bernard, Requirements management within a full model-based engineering approach, Systems Engineering, 15(2) (2012), 119-139.

Bureau of International des Poids et Mesures (BIPM), Evaluation of measurement data -- Supplement 1 to the Guide to the expression of uncertainty in measurement -- Propagation of distributions using a Monte Carlo method, Sèvres, France, 2008.

BIPM, Guide to the expression of uncertainty in measurement, Sèvres, France, 2008.

BIPM, International vocabulary of metrology -- basic and general concepts and associated terms (VIM), 2008, Sèvres, France, 2008.

K. P. Burnham and D. R. Anderson, Mutlimodel inference: Understanding AIC and BIC in model selection, Sociological Methods and Research, 33(2) (2004), 261-304.

L. W. Clarke and L. S. Gardner, How much testing is enough? An iterative, mathematical programming approach, International Test and Evaluation Association (ITEA) Journal 16(3) (1995), 26-30.

M. L. Cohen, J. E. Rolph and D. L. Steffey, Statistics, testing and defense acquisition: New approaches and methodological improvements, National Research Council, Washington, D.C., 1998.

H. W. Coleman and W. G. Steele, Experimentation, validation, and uncertainty analysis for engineers, A. John Wiley & Sons, Inc., Hoboken, New Jersey, 2009.

S. L. Cornford, M. S. Feather, V. A. Heron and J. S. Jenkins, Fusing quantitative requirements analysis with model-based systems engineering, 14th IEEE International Requirements Engineering Conference (RE'06), Minneapolis, Minnesota, 2006.

H. E. Crisp, Systems engineering vision 2020, Seattle, Washington, 2007.

S. Deaconu and H. W. Coleman, Limitations of statistical design of experiments approaches in engineering testing, Journal of Fluids Engineering 122(2) (2000), 254-259.

H. Dezfuli, D. Kelly, C. Smith, K. Vedros and W. Galyean, Bayesian inference for NASA probabilistic risk and reliability analysis, NASA, Washington, DC, 2009.

DoD, DoD Modeling and simulation (M&S) glossary, 2010.

M. A. Foster, Developing a heavy lift launch vehicle systems architecture using a model based systems engineering approach, AIAA Space 2011 Conference and Exposition, Long Beach, California, 2011.

S. Friedenthal, A. Moore and R. Steiner, A practical guide to SysML: The systems modeling language, Morgan Kaufmann Publishers, Burlington, MA, 2009.

J. M. Gilmore, Rigor and objectivity in T&E, ITEA Journal 31(2) (2010), 161-164.

A. Hurwitz, W. Kitto, T. Remund and J. Brownlow, Estimation of location error for targeting using parametric, Monte Carlo, and Bayesian techniques, ITEA Journal 32(3) (2011), 322-332.

R. Karban, M. Zamparelli, B. Bauvir, B. Koehler, L. Noethe and A. Balestra, Exploring model based engineering for large telescopes: Getting started with descriptive models, Proceedings of Society of Photo-Optical Instrumentation Engineers (SPIE) Astronomical Telescopes and Instrumentation 2008, p70171I, Marseille, France.

D. Kidman, C. Stevens, C. Moulder, W. Kitto, J. Brownlow and T. Remund, Developing statistically defensible propulsion system test and evaluation techniques, ASME Turbo Expo 2012: Power for Land, Sea, and Air, 2011, Copenhagen, Denmark.

C. H. Loch and C. Terwiesch, Communication and uncertainty in concurrent engineering, Management Science 44(8) (1998), 1032-1048.

A. J. Lopes, R. Lezama and R. Pineda, Model based systems engineering for smart grids as systems of systems, Complex Adaptive Systems 2011, Conference, Chicago, Illinois, Procedia Computer Science 6 (2011), 441-450.

T. I. McVittie, O. V. Sindiy and K. A. Simpson, Model-based system engineering of the Orion Flight Test 1 end-to-end information system, 2012 IEEE Aerospace Conference, Big Sky, Montana, 2012.

M. G. Morgan and M. Henrion, Uncertainty: A guide to dealing with uncertainty in quantitative risk and policy analysis, Cambridge University Press, New York, 1990.

D. Nottage and S. Corns, Application of model-based systems engineering on a university satellite design team, New Challenges in Systems Engineering and Architecting Conference on Systems Engineering Research 2012, St Louis, Missouri, Procedia Computer Science 8 (2012), 207-213.

Object Management Group (OMG), UML testing profile, 2012.

G. W. Parry, The characterization of uncertainty in probabilistic risk assessments of complex systems, Reliability Engineering & System Safety 54(2-3) (1996), 119-126.

C. Piaszczyk, Model based systems engineering with Department of Defense Architectural Framework, Systems Engineering 14(3) (2011), 305-326.

A. L. Ramos, J. V. Ferreira and J. Barceló, Model-based systems engineering: An emerging approach for modern systems, Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on 42(1) (2012), 101-111.

G. J. Roedler and C. Jones, Technical measurement, International Council on Systems Engineering Practical Software Syst Measurement, 2005.

M. Russell, Using MBSE to enhance system design decision making, New Challenges in Systems Engineering and Architecting Conference on Systems Engineering Research 2012, St Louis, Missouri, Procedia Computer Science 8 (2012), 188-193.

S. Schrader, W. M. Riggs and R. P. Smith, Choice over uncertainty and ambiguity in technical problem solving, Journal of Engineering and Technology Management 10(1-2) (1993), 73-99.

M. A. Shaikh and W. H. Moore, II, Statistical interpretation of absolute system performance requirements, Aerospace and Electronic Systems, IEEE Transactions on 33(2) (1997), 626-631.

M. Solaguren-Beascoa Fernández, On the use of the 'uncertainty budget' to detect dominant terms in the evaluation of measurement uncertainty, Accreditation and Quality Assurance: Journal for Quality, Comparability and Reliability in Chemical Measurement 16(2) (2011), 83-88.

Y. Song, S. Han and S. Lee, Construct digital management system of the performance test and design standard of the TTX using model-based systems engineering approach, LINDI 2007, International Symposium on Logistics and Informatics, Wildau, Germany, 2007, 93-97.

A. Soyler and S. Sala-Diakanda, A model-based systems engineering approach to capturing disaster management systems, 4th Annual IEEE Systems Conference, San Diego, California, 2010.

S. C. Spangelo, D. Kaslow, C. Delp, B. Cole, L. Anderson, E. Fosse, B. S. Gilbert, L. Hartman, T. Kahn and J. Cutler, Applying model based systems engineering (MBSE) to a standard CubeSat, 2012 IEEE Aerospace Conference, Big Sky, Montana, 2012.

A. Streit, P. Binh and R. Brown, A spreadsheet approach to facilitate visualization of uncertainty in information, Visualization and Computer Graphics, IEEE Transactions on 14(1) (2008), 61-72.

L. C. van Ruijven, Ontology and model-based systems engineering, New Challenges in Systems Engineering and Architecting Conference on Systems Engineering Research 2012, St Louis, Missouri, Procedia Computer Science 8 (2012), 194-200.

J. R. Wendelberger, Uncertainty in designed experiments, Quality Engineering 22(2) (2010), 88-102.

A. W. Wymore, Model-based systems engineering, CRC Press, Inc., Boca Raton, Florida, 1993.

Table I. Summary of MBSE Concepts

Paradigm	Example Methods	Example Views	Example Languages	Example Tools
MBSE to create system model	Vee OOSEM	DoDAF MoDAF FEAF	UML SysML FFBD IDEF0	Enterprise Architect CORE CATIA
Creation of manufacturing drawings	Hand-drawn CAD/CAM	Plan & elevation Isometric 3-D	Drafting standards CAD languages	Drafting kit CATIA

Table II. Initial Uncertainties of Components

Component (Parameter)	Initial Value	Initial Standard Deviation	Initial Distribution
C_L	0.30 (mean)	+/-10% of mean	Uniform
C_D	0.12 (mean)	+/-20% of mean	Uniform
μ_{heavy}	0.20 (mean)	+/-30% of mean	Uniform
W	10800 lbs (minimum)	+15% of minimum	Uniform
V_{td}	128 knots (mean)	5 knots	Normal
T	100 lbs	10 lbs	Normal

Table III. Component Uncertainty Contributions to Total Uncertainty (Monte Carlo Simulation Results)

Component (u_i)	Uncertainty Contribution (ft)
Weight (W)	8.9
Thrust (F)	14.5
C_D	87.2
C_L	21.7
V_{td}	233.3
μ_{heavy}	500.6
All	573.0
$\sqrt{\sum u_i^2}$	559.8

Table IV. Pre-Test Predictions of Uncertainty Reductions

Test Events	Landing Distance Standard Deviation (ft)	1st Percentile Landing Distance (ft)	99th Percentile Landing Distance (ft)	Mean Landing Distance (ft)
Initial	573.0	2457	4821	3457
Wind Tunnel	564.4	2466	4768	3452
Lab	407.4	2615	4370	3411
Landing	309.7	2721	4167	3391

Table V. Component Uncertainties After Test Events

Component (Parameter)	Post-Wind Tunnel Mean Value, Standard Deviation, and Distribution	Post-Lab Test Mean Value, Standard Deviation, and Distribution	Post-Landing Test Mean Value, Standard Deviation, and Distribution
C_L	0.2828, 1%, N	0.2828, 1%, N	0.2828, 1%, N
C_D	0.097, 1%, N	0.097, 1%, N	0.097, 1%, N
μ_{heavy}	0.20, +/-30%, U	0.20, +/-20%, U	0.222, 5%, N
W	10800 lbs, +15%, U	10800 lbs, +15%, U	10800 lbs, +15%, U
V_{td}	128 knots, 5 knots, N	128 knots, 5 knots, N	128 knots, 5 knots, N
T	100 lbs, 10 lbs, N	100 lbs, 10 lbs, N	100 lbs, 10 lbs, N

Key to Distributions: N = Normal, U = Uniform

Table VI. Post-Test Predictions of Landing Roll

Test Events	Landing Distance Standard Deviation (ft)	1st Percentile Landing Distance (ft)	99th Percentile Landing Distance (ft)	Mean Landing Distance (ft)
Initial	573.0	2457	4821	3457
Wind Tunnel	607.2	2529	5004	3582
Lab	438.2	2685	4575	3537
Landing	248.7	2643	3798	3201