ABSTRACT

Title of dissertation: EVALUATION-FOCUSED RELIABILITY TEST PROGRAM PLANNING METHODOLOGY

Robert N. Tamburello, Doctor of Philosophy, 2013

Dissertation directed by: Associate Professor Jeffrey W. Herrmann
Department of Mechanical Engineering

In practice, various ad hoc approaches for designing reliability test programs have been observed. Many of these approaches rely on previously established rules of thumb for which the underlying rationale is indefensible. As a consequence, those who use such approaches are unlikely to maintain a firm resource commitment for the conduct of reliability test program activities. Furthermore, it is difficult to ascertain the impact that budgetary cuts will have on the adequacy of the reliability test program with any degree of accuracy.

The contributions of this research are as follows. This dissertation presents a novel 7-step planning process to aid practitioners in designing adequate reliability test programs. This planning process serves as a tool to systematically identify, quantify, and mitigate evaluation risks subject to resource constraints. By performing the 7 steps associated with this planning process, practitioners will be able to logically justify reliability test program requirements and more effectively articulate the significance of

evaluation risks associated with a particular reliability test program design. Additionally, it is a straightforward process to assess the impact of a reduction in reliability test program resources.

This planning process includes a step for assessing the level of risk associated with key aspects of the reliability test program. One such consideration that is of paramount importance is the adequacy of the test configuration of the system. Hence, we present a simulation-based approach for assessing the adequacy of the test configuration of a complex system-of-systems. For the purpose of demonstration, an application of this approach to air defense systems is included; however, the approach is valid for any type of system.

As well, this dissertation presents an evaluation risk assessment process for reliability test programs—adapted from the traditional failure mode and effects analysis (FMEA) process. This process can be applied to any reliability test program, irrespective of the manner in which the plan was formulated. Just as a FMEA facilitates the identification of potential weaknesses in a system architecture, this evaluation risk assessment process is designed to surface reliability test program weaknesses and gauge the potential impact of each weakness to the system reliability evaluation.

EVALUATION-FOCUSED RELIABILITY TEST PROGRAM PLANNING METHODOLOGY

by

Robert N. Tamburello

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park, in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2013

Advisory Committee:

Associate Professor Jeffrey Herrmann, Chair
Associate Professor Mark Austin, Dean's Representative
Dr. Michael Cushing, Special Member
Professor Mohammad Modarres
Professor Ali Mosleh

DEDICATION

Completion of this body of work could not have been possible without the enduring encouragement and indefatigable support of my family, especially, my loving wife, Michelle.

ACKNOWLEDGEMENTS

The author would like to extend his appreciation to his PhD advisor and committee for their time, support, and assessment of his research:

- Associate Professor Jeffrey Herrmann, Committee Chair and Advisor
- Associate Professor Mark Austin, Dean's Representative
- Dr. Michael Cushing, Special Member
- Professor Mohammad Modarres
- Professor Ali Mosleh
- Professor Bilal Ayyub

The author would like to recognize the critical support over the years by his leadership at the US Army Evaluation Center:

- Dr. Michael Cushing
- Ms. Jane Krolewski
- Mr. Richard Sullivan
- Mr. Stephen Conley

Additionally, the author would like to express his appreciation to the following colleagues for their counsel throughout the evolution of this body of work:

- Dr. J. Brian Hall
- Mr. Angelo Christino
- Mr. Martin Wayne

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

ACAT	Acquisition Category
ALT	Accelerated Life Testing
AST	ATEC System Team
ATEC	Army Test and Evaluation Command
CDD	Capability Development Document
CM	Cruise Missile
DfR	Design for Reliability
DoD	Department of Defense
DOT&E	Director of Operational Test and Evaluation
DT	Developmental Testing
FDSC	Failure Definitions and Scoring Criteria
FEF	Fix Effectiveness Factor
FM	Failure Mode
FMEA	Failure Modes and Effects Analysis
GPS	Global Positioning System
HALT	Highly Accelerated Life Testing
HWIL	Hardware in the Loop
IOT	Initial Operational Test
IPT	Integrated Product (or Process) Team
IT	Information Technology
LUT	Limited User Test
MS	Management Strategy
M&S	Modeling and Simulation
OMS/MP	Operational Mode Summary / Mission Profile
OT	Operational Testing
PM	Program Manager
PoF	Physics of Failure
RAM	Reliability, Availability, and Maintainability
RBD	Reliability Block Diagram
RG	Reliability Growth
RGPC	Reliability Growth Planning Curve
RQT	Reliability Qualification Test
RTP	Reliability Test Program
SoS	System of Systems
T&E	Test and Evaluation
TIR	Test Incident Report
UAV	Unmanned Aerial Vehicle
VV&A	Validation, Verification, and Accreditation

1 INTRODUCTION

1.1 Purpose of this Dissertation

Evaluation risk is defined as a measure of the residual uncertainty associated with the characterization of a given system's expected behavior in its intended operational environment. This dissertation will present risk management techniques to identify, assess, and mitigate —subject to resource constraints—the evaluation risks associated with a given system's reliability test program. Reliability test planners can use the guidance presented herein to logically justify reliability test program activities and more effectively articulate the significance of evaluation risks associated with a particular reliability test program design to funding decision authorities. If reliability test program resources are reduced, these techniques are a means to assess the impact that such a reduction in reliability test program resources will have on the adequacy of the system reliability evaluation for Major Defense Acquisition Programs (MDAPs[1]). In this context, adequacy is defined as the condition achieved when the evaluation risk associated with a given reliability test program plan is within the acceptable tolerance threshold of the decision authority.

[1] An acquisition program that is designated by the Under Secretary of Defense (Acquisition, Technology, and Logistics) (USD(AT&L)) as an MDAP, or estimated by the USD(AT&L) to require an eventual total expenditure for Research, Development, Test and Evaluation (RDT&E) of more than $365 million in Fiscal Year (FY) 2000 constant dollars or, for procurement, of more than $2.19 billion in FY 2000 constant dollars [46].

1.2 Overview of Dissertation and Its Contributions

Chapter 1: Introduction

The purpose of Chapter 1 is to provide the reader with a synopsis of the focus of this dissertation as well as the organizational structure and material contained herein.

Chapter 2: Background and Motivation

In Chapter 2, we discuss current practices with respect to reliability test program planning and reliability evaluation for Army systems, and we highlight heretofore unresolved challenges. Most notably, it is the case that no logically-defensible approach to address the adequacy of a given reliability test program currently exists. Based on the diminishing amount of resources available for the conduct of reliability test programs, there is a keen interest in designing and executing efficient reliability test programs. However, it is critical to first recognize that there is a tipping point at which a further reduction in resources would preclude an adequate evaluation of system reliability. The research goal was to develop innovative techniques that can serve to identify the tipping point as an aid in the planning process during which time there is a fierce competition for a firm commitment of resources. To that end, 5 minimum requirements for reliability test programs are introduced. These 5 minimum requirements will be used to establish an evaluation risk assessment process for reliability test programs in Chapter 4 as well as a planning process for the design of reliability test programs in Chapter 6.

Chapter 3: Literature Review

In Chapter 3, we document the findings from our review of relevant work associated with the design of adequate reliability test programs. The material in this chapter is organized according to the 5 minimum requirements for reliability test programs outlined in Chapter 2.

Chapter 4: Evaluation Risk Assessment Process for Reliability Test Programs

As discussed in Chapter 2, a range of flaws has been observed in reliability test programs for military systems. Therefore, it is desirable to formulate a structured approach to designing reliability test programs that enables the identification, assessment, and mitigation of such flaws. In order to formulate an appropriate process for designing reliability test programs, a systematic examination of actual reliability test programs must be performed to document these flaws (or "failure modes").

Chapter 4 presents a novel evaluation risk assessment process for reliability test programs. This process, which was adapted from the traditional failure mode and effects analysis (FMEA) process, can be applied to any reliability test program, irrespective of the manner in which the plan was formulated. Just as a FMEA facilitates the identification of potential weaknesses in a system architecture, our evaluation risk assessment process is designed to surface reliability test program weaknesses and gauge the potential impact of each weakness to the system reliability evaluation. As well, the reliability test program evaluation risk assessment process includes steps to postulate alternatives, develop a monitoring strategy, and devise contingency actions.

The 5 minimum requirements for reliability test programs introduced in Chapter 2 are used to develop reliability evaluation risk indicators to be considered during the application of the evaluation risk assessment process. In Chapter 5, this evaluation risk assessment process is applied to the reliability test programs for 10 Major Defense Acquisition Programs.

Chapter 5: Findings from the Application of the Evaluation Risk Assessment Process to 10 Major Defense Acquisition Programs

Based on the application of the evaluation risk assessment process defined in Chapter 4 to 10 Major Defense Acquisition Programs, a summary of poor design practices for reliability test programs is presented. The systems considered form representative cross-section of the types of systems in the US Department of Defense's portfolio, specifically—1 networked communications system-of-systems, 1 system comprised of networked sensors, 3 wheeled combat vehicles, 1 wheeled tactical vehicle, 1 tracked combat vehicle, 1 dismounted battle-space awareness system, and 1 missile defense system. Recommendations regarding risk mitigation strategies, contingency plans, and alternate approaches are included. The findings discussed in Chapter 5 are used to inform the development of the reliability test program planning process in Chapter 6.

Chapter 6: Proposed Process for Reliability Test Program Planning

Bearing in mind (i) the 5 minimum requirements for reliability test programs specified in Chapter 2 and (ii) the findings contained in Chapter 5, Chapter 6 presents a novel 7-step planning process to for designing adequate reliability test programs for

military and commercial systems. This tailorable planning process serves as a tool to systematically identify, assess, and mitigate evaluation risk subject to resource constraints (e.g., amount of funding, time available for test conduct, quantity of test assets). As the considerations of the evaluation risk assessment process defined in Chapter 4 are also, necessarily, important considerations in the design of RTPs, the steps of the planning process presented in Chapter 6 reflect these same considerations.

By performing the 7 steps associated with this planning process, it is possible to logically justify reliability test program requirements and more effectively articulate the significance of evaluation risks associated with a particular reliability test program design. Additionally, it is a straightforward process to assess the impact of a reduction in reliability test program resources and determine whether or not there is a need to reformulate the reliability test program. This planning process was used to generate lower risk reliability test programs for two real systems in order to demonstrate its usefulness.

Chapter 7: A Simulation-Based Risk Assessment Methodology to Determine the Evaluation Adequacy of System-Of-Systems Operational Test Configurations

The reliability test program planning process from Chapter 6 includes steps to establish and assess the level of risk associated with essential evaluation risk areas (planning steps 4 and 6, respectively). Chapter 7 presents a simulation-based method for assessing the risk associated with a particular essential evaluation risk area that may be established during the reliability test program planning process—the disparity that exists between the field/production configuration of the system-of-systems and the test configuration of the system-of-systems. Per the Defense Acquisition Guidebook [46], a

system-of-systems is defined as “a set or arrangement of systems that results from independent systems integrated into a larger system that delivers unique capabilities.”

Given practical reliability test program resource constraints, it is rare that the full field configuration of a system-of-systems can be exercised during an operational test event. As a consequence, an incomplete configuration of the system-of-systems is exercised, and those test results are used to make an assessment of the full field configuration. However, as we consider various (incomplete) operational test configurations for a given system-of-systems, it is critical to acknowledge that there is a threshold at which a further reduction in the scale and scope of the test configuration will yield results that are not representative of how the full field configuration will behave. Conceptually, only test configurations that lie above such a threshold are deemed to be adequate, from a system-of-systems reliability evaluation standpoint.

The simulation-based approach presented in this chapter can be employed to assess the adequacy of a given test configuration for any type of military or commercial system-of-systems. As discussed above, this simulation-based assessment of the adequacy of the planned system-of-systems test configuration is a supporting activity for step 6 of the reliability test program planning process from Chapter 6. To illustrate how this simulation-based approach can be used to aid in the identification of the best alternative from among a group of potential operational test configuration alternatives, Chapter 7 concludes with an example application.

Chapter 8: Summary and Conclusions

In Chapter 8, we summarize the findings of this dissertation research. Specifically, we concentrate on insights in the following 4 major areas: (1) the reliability test program evaluation risk assessment process described in Chapter 4, (2) findings from examination of reliability test programs for 10 Major Defense Acquisition Programs documented in Chapter 5, (3) the reliability test program planning process defined in Chapter 6, and (4) the simulation-based approach for assessing the evaluation adequacy of system-of-systems operational test configurations provided in Chapter 7. We revisit the topic of applicability for the techniques presented in Chapters 4 through 7 along with implementation recommendations for practitioners. Finally, we discuss opportunities for future work to extend the contributions in this dissertation.

2 BACKGROUND AND MOTIVATION

2.1 Limitations with the Current Reliability Test Program Planning Paradigm

In practice, we have observed various ad hoc approaches for designing reliability test programs (RTPs). Many of these approaches rely on previously established rules of thumb for which the underlying rationale is indefensible. As a consequence, it is unlikely to maintain a firm resource commitment from program managers for the conduct of RTP activities. Furthermore, it has proven difficult to ascertain the impact that budgetary cuts will have on the adequacy of the RTP with any degree of accuracy.

For systems that have at least one reliability requirement, Army evaluators collaborate with the system's program management office to plan and execute a series of developmental and operational test events in order to sufficiently characterize the reliability behavior of the system. Such a characterization should not be limited to a single value such as a point estimate of the mean time between system abort (MTBSA); it must include a comprehensive list of all observed reliability failure modes as well as the conditions under which those failures occurred during testing. The motivation for conducting a robust reliability test and evaluation program is to inform the final decision regarding whether the demonstrated system reliability will enable or inhibit mission accomplishment. As well, operation and support costs associated with the fleet of systems should be estimated to assess affordability over 20 or 30 years of planned service in the field.

2.2 Shortfalls in the Reliability of Army Systems

Between 2004 and 2007, the success rate associated with the demonstration of US Army materiel system reliability requirements was approximately 26% [4]. In December of 2007, the Office of the Assistant Secretary of the Army for Acquisition, Logistics, and Technology (ASA[ALT]) issued the first Army-level reliability policy, aimed at improving the success rate for the demonstration of Army materiel system reliability requirements [1]. From 2008 through 2011, the success rate improved to approximately 37%, and if we consider only results from 2010 through 2011, the success rate was approximately 53% for the Army [4]. Across all of the Services (Army, Navy, and Air Force) in the US Department of Defense (DoD), 54% of the systems evaluated in 2012 met their reliability thresholds [6].

Despite the observed increase in the reliability demonstration success rate, the majority of system developers are still failing to conduct effective design-for-reliability activities prior to the start of the formal reliability growth program [4]. As a result, programs are encountering many more failure modes than can be effectively addressed and mitigated prior to the reliability demonstration event [4]. Therefore, in 2011, ASA(ALT) issued a reliability policy update [3] augmented with early detection mechanisms (engineering-based reviews and reliability assessments) for significant departures from system reliability growth program plans. The intent of the additional elements of the 2011 Army reliability policy is to identify issues early enough in the reliability growth program to have the potential to make course corrections before it is too late to have an impact.

2.3 System Reliability Requirements

Irrespective of the particular RTP planning approach adopted, the initial activity should be to ensure a shared interpretation of the system's reliability requirement(s) among all stakeholders. Reliability requirements for an Army system are specified in the system's associated Capability Development Document (CDD). Given the system reliability requirements, the test and evaluation community must achieve consensus regarding the interpretation of the requirements in order to devise a sufficient RTP. Within the CDD, a system-level reliability requirement is typically specified as a lower bound on the probability of completing a mission within a certain duration. For example, a requirement might be written as "the system shall have at least a 90% probability of completing a 24-hour mission." Often, an associated reliability metric, such as the mean time between system abort (MTBSA) is derived from the CDD requirement, and that serves as the focus of the reliability test and evaluation program in such cases. Further, per Department of the Army Pamphlets 70-3 [7] and 73-1 [8], the system reliability requirement must be demonstrated with high statistical confidence (typically 80% is the level adopted).

In the context of the system evaluation, the level of statistical confidence refers to the one-sided lower confidence bound on the adopted system-level reliability metric (such as MTBSA). Although a system's reliability evaluator will use the adopted metric to assess reliability subsequent to each test event in the system's overall test and evaluation program, the stipulation regarding confidence primarily applies to the failure data captured during the single test event designated as the reliability demonstration test (generally referred to as the initial operational test). The selection of a particular level of

statistical confidence relative to the system reliability requirement is motivated by the government's desire to temper the risk of accepting a system for which the inherent design does not meet the reliability requirement. Such a risk is traditionally referred to as the consumer's risk. Similarly, a reasonable RTP plan will balance the producer's risk, i.e., the risk that the government does not accept a system that actually meets the reliability requirement. Given the current reliability test and evaluation paradigm—a single reliability demonstration event of finite length—it is possible for either risk to become the reality due to random chance. After all, it is possible that the observed reliability behavior of the system during the demonstration test will not be representative of its true tendency in the field. This possibility is the primary motivation to demand the demonstration of system reliability requirements with high statistical confidence.

Along with the aforementioned RTP planning considerations, it is pivotal to conduct analyses to identify potential unreliability drivers prior to the commencement of the RTP. We wish to develop intuition as to where we should anticipate the bulk of the failure intensity to be concentrated. For analysis purposes, one may decompose the system in terms of its subsystems or functions, and make an effort to map failure modes to each aspect. The insights derived from such activities can enable the planners to construct an RTP that includes particular events designed to exercise the system in such a way as to induce important failure modes and facilitate a representative characterization of the behavior of the system.

2.4 Areas of Uncertainty Encountered in the Evaluation of System Reliability

Despite the concerted effort to investigate and quantify the reliability behavior of systems, it is clear that uncertainty regarding the true underlying nature of the system will persist. Further, the uncertainty is exacerbated by the fact that, for most complex systems-of-systems (SoS), it is infeasible to test the full configuration that the Army intends to field. In the event that a representative configuration of the SoS is indeed available for testing, it is often the case that the duration of testing is insufficient to demonstrate the reliability requirement with statistical confidence. Hence, it is pivotal to the Army that the plan for evaluating the system under test hedge against the introduction of unmitigated uncertainty and improve our knowledge of the system.

As a direct result of the language used to specify system reliability requirements, historically, the evaluation of system reliability has emphasized the estimation of metrics such as MTBSA. Inherently, there is aleatory uncertainty associated with the estimation of MTBSA or other similarly defined system reliability metrics via finite developmental and/or operational testing activities. There is also epistemic uncertainty associated with the underlying behavior of system reliability; although, the standard assumption is that reliability failures occur according to a homogeneous Poisson process. Depending on the system design, this may not be a good model to use to characterize system reliability. The validity of such an assumption should, in practice, be assessed given a reliability block diagram of the system along with system-level failure data. In reality, even if the assumption of a homogeneous Poisson process is not correct, resource constraints may preclude the acquisition of sufficient evidence to reject that model.

Notwithstanding the emphasis on estimation of system reliability metrics such as MTBSA, it is incumbent on system reliability evaluators to capture and analyze additional pertinent information. In 2 reports [9, 10] published by a National Academy of Sciences panel regarding the operational test and evaluation of the Stryker family of systems, the panel recommended that evaluators should (1) track system failure modes and maintenance information across developmental and operational testing and (2) assess system reliability by specific failure modes as well as across failure modes instead of assigning a particular exponential model for all failures. Additionally, the panel asserted that system reliability behavior should be assessed within the context of environmental and operational conditions [10]. After all, a holistic evaluation of system reliability (as a component of operational suitability) should explicitly address

- system failure modes and their estimated recurrence rates
- the unreliability drivers (based on recurrence rate)
- the major subsystems that are associated with the bulk of the failure intensity
- the cost to repair the system (in terms of maintenance man-hours and parts),
- the down-time associated with each failure mode (including diagnostic time, active maintenance time, administrative and logistic delay time, etc.), and
- special tools or training required for maintainers.

There may be other areas of interest, depending on the given system. The ability to address each area at a high level of fidelity is inexorably linked to the availability of reliability test program resources. Intuitively, as the availability of resources increases, it is possible to reduce the epistemic uncertainty in our characterization of system reliability; yet, it is impossible to completely mitigate aleatory uncertainty [23].

2.5 Minimum Requirements for the Characterization of System Reliability

Because test and evaluation resources are limited, it is impractical to set forth to design risk-free RTPs. Instead, the goal of practitioners should be to identify essential evaluation risk areas associated with each RTP and mitigate those evaluation risks to the extent that available resources will support. As the goal is to characterize the reliability behavior of each system, we assert that it is imperative to address 5 major areas during the design of an RTP. Hereafter, we will refer to these 5 areas as our minimum requirements for RTPs. Below, we briefly discuss the 5 minimum requirements along with the reasoning behind the inclusion of each requirement.

Requirement 1: Evaluate the Reliability of the Field Configuration of the System

Although this requirement may seem obvious, a single, clear definition of the field configuration of the system cannot always be specified. For some complex systems-of-systems (SoS), there may be such a large number of possible configurations that the field configuration definition established by the user only represents one potential realization of the SoS. From the perspective of the reliability evaluator, this is not an ideal situation; however, the RTP is constructed to evaluate the configuration that the user specifies in the system requirements document. Where appropriate, it may be desirable to identify multiple field configurations that will be evaluated.

Requirement 2: Test all Elements of the System

The purpose of this requirement is to ensure that all distinct elements of the system are actually tested under operationally realistic conditions. For example, we may

have an SoS with a number of identical, redundant systems (active parallel, standby, etc.). We want to ensure that each type of system will be physically present during the RTP. Moreover, in order to confirm that the redundancy actually works as intended (e.g., a user can switch from a failed terminal to an operational terminal to perform essential mission tasks) multiple test assets of each type must be present during the reliability demonstration event. As well, all software must be exercised. No elements of the SoS should be purely simulation-based. Moreover, the use of system surrogates (similar systems) should be considered as a last resort because using system surrogates introduces additional uncertainty in the characterization of system reliability behavior that may not be well-understood.

Requirement 3: Exercise all Anticipated Unreliability Drivers

The system developer should perform engineering-based analyses early in the RTP planning process in order to identify anticipated system unreliability drivers—the vital few failure modes that constitute the largest proportion of the system failure intensity. More precisely, the system developer should strive to determine which failure modes should have the greatest contribution to system unreliability in the actual operational environment, not in the lab or test chamber. Such an effort is essential, and it would ensure that appropriate test events (and test durations) are incorporated into the RTP to sufficiently exercise the anticipated unreliability drivers. For the Army, the design of a given system's RTP must conform to the Operational Mode Summary/Mission Profile (OMS/MP), which is a document generated by the Army's Training and Doctrine

Command (TRADOC). The OMS/MP specifies TRADOC's assumptions regarding the system's missions, annual usage rate, load cycles, and operating environment.

Requirement 4: Evaluate the Reliability of the System Correctly

Due to the ever-increasing scale and complexity of systems, the evaluation methodology employed must be tailored for each system. For example, it may be necessary to build-up SoS reliability estimates from system-level data when testing the full field configuration is infeasible. In such a case, it is critically important to exercise an appropriate test configuration in order to identify failure modes such as those due to scale, integration, and interaction among constituent systems of the SoS.

Requirement 5: Identify and Manage the Evaluation Risks of the RTP

Clearly, the objective behind conducting an RTP is to evaluate system reliability. Therefore, it is desirable to assess the evaluation risks associated with an RTP plan before implementing it. If the evaluation risks are unacceptable, the plan must be revised to mitigate such risks—provided sufficient resources are available to do so. In addition, if any of the planning assumptions are determined to be invalid during the course of the RTP, a follow-on risk assessment that addresses the new information should be performed and the results should be shared with all stakeholders.

2.6 RTP Efficiency versus RTP Adequacy

Now that the key system reliability policies discussed in Section 2.2 are in effect, the attention has shifted to T&E efficiency due to the reduction in available resources

across the DoD. The objective is to identify methods to streamline T&E activities and promote innovation—not to compromise the adequacy of system reliability evaluations. However, it is critical to first recognize that there exists the tipping point at which a further reduction in resources would preclude an adequate evaluation of system reliability. Much to the dismay of many members of the community of practice, there is no simple closed-form solution to this problem. Instead, RTP adequacy must be assessed via more complex analytical activities.

The primary motivation underpinning this dissertation is to develop techniques that can serve to identify the RTP evaluation adequacy tipping point during the RTP planning process. After all, it is during the planning process that a fierce competition for a firm commitment of resources erupts, and practitioners need to construct logical and compelling arguments in order to secure critical resources. It is important for all stakeholders to realize that an RTP can be deemed as efficient only if the RTP is adequate because efficiency implies that all evaluation requirements are satisfied using the minimum amount of resources. In Chapter 6, we directly address the subject of RTP adequacy as the foundation of the RTP planning process. In Chapter 7, we delve further into a simulation-based method to assess the evaluation adequacy of test configurations for complex systems-of-systems.

3 LITERATURE REVIEW

3.1 Overview

In Chapter 2, we discussed the challenges facing the reliability T&E community, and we presented a set of minimum RTP requirements to address those challenges. We will now discuss relevant methodology, tools, and guidance found in the existing literature . Given our discussion in the previous chapter, we have organized our literature review in terms of the 5 minimum requirements for RTPs. For the most part, Requirements 1 and 2 are straightforward, and we submit that these requirements do not demand further embellishment here. In contrast, Requirements 3, 4, and 5 are non-trivial. More importantly, there is no established standard guidance for the fulfillment of Requirements 4 and 5. Therefore, our literature review will only address methodology, tools, and guidance applicable to Requirements 3, 4, and 5.

3.2 Identify all Anticipated System Unreliability Drivers

Although the associated minimum requirement previously appears in Chapter 2 as "Exercise all Anticipated Unreliability Drivers," we must first understand how to identify system unreliability drivers—generally before system-level testing has been conducted. First applied by nuclear and defense industries in the 1940s, and later formalized by NASA in the 1960s, the traditional failure mode effects and criticality analysis (FMECA) methodology [24] has been widely employed to identify system failure modes. The long-standing FMECA guides utilized by practitioners, Military Standard 1629A [11] and Military Standard 785B [28] were both cancelled in 1998, yet they continue to be

consulted. The Society of Automotive Engineers (SAE) established 2 new standards for FMECAs: (1) SAE-J1739 [26], which is intended for automotive systems, and (2) SAE-ARP-5580 [27], which is intended for non-automotive systems.

For the majority of Army acquisition programs, system developers are contractually obligated to generate and deliver a FMECA report to the government, but the FMECA report does not automatically translate into an improvement in the reliability of the system. One problem with this interpretation of how to perform a FMECA is that it is inherently task-based; however, it should be a continuous process. Fortunately, the updated SAE-ARP-5580 promotes this notion of a FMECA as a process, not a one-time deliverable [25].

In practice, the system developer's team responsible for the synthesis of the FMECA may not interact with the system engineering team [4]. As a consequence, the potential insights gained from the FMECA process are not shared. Furthermore, guidance regarding FMECAs contained in Military Standard 785B features the activity of postulating all potential failure modes for the system [28]. Yet, this is not truly practical, nor is it a useful activity. The goal is not to postulate thousands of potential system failure modes; rather, the goal is to conduct up-front analyses to identify those potential failure modes that will provide the greatest "contribution" to the initial failure intensity of the system under operationally-realistic loads and stresses.

The "hard part" of the FMECA effort concerns the identification of operationally important failure modes—the vital few as opposed to the trivial many. There are a number of reasonable approaches to employ in the pursuit of identifying the potential unreliability drivers of a given system. In 2008, the Information Technology Association

of America (ITAA) released GEIA-STD-0009: Reliability Program Standard for Systems Design, Development, and Manufacturing [12]. Approved by the American National Standards Institute (ANSI) and written by members from DoD, industry, and academia, GEIA-STD-0009—along with its companion handbook [13]—offers practitioners the following guidance regarding the failure mode identification process:

- Contract for closed-loop, continuous-improvement effort to identify and mitigate failure modes likely to occur under operationally-realistic loads and stresses
- In advance of system/subsystem testing, apply techniques such as:
  - Engineering- and physics-based failure mechanism modeling
  - Accelerated and low-level testing of components and assemblies
  - MANPRINT analytical methods (for failure modes that may be charged to operators, maintainers, or software)
  - Lean Six Sigma methods (for failure modes that may be induced by manufacturing variation or errors)
  - Execution of system/subsystem-level reliability growth testing to surface and mitigate the modeling-resistant failure modes that remain

As well, commercial software tools such as Raptor (developed and licensed by ARINC) and BlockSim (developed and licensed by ReliaSoft) have become popular among certain organizations within the community of practice. These and other software-based tools may be employed as a means to aid in the identification of subsystems that will have the greatest impact on system-level reliability. Murphy et al. [14] assert that the underlying Raptor simulation methodology consistently yields more accurate results, in terms of the identification of key subsystems that impact system-level reliability, than traditional

ranking methods [14]. However, for the example discussed, the Raptor ranking of subsystems is not compared to actual test results or field data for the V-22 Osprey tilt-rotor aircraft; hence, the authors do not provide compelling evidence that their assertions regarding accuracy are valid. This assertion of enhanced accuracy requires further study.

For complex systems-of-systems, the utilization of software-based modeling and simulation (M&S) tools can be highly beneficial, provided that the underlying models have gone through a formal validation, verification, and accreditation (VV&A) process. Army policy mandates that the VV&A process must be completed for before any M&S tools may be used in the evaluation of a system. For commercial (i.e., proprietary) software-based tools, it may not be possible for the government to perform an independent VV&A. Thus, the employment of commercial products may be at the user's own risk.

Alternatively, Yadav et al. [15, 16] offer another systematic approach to identify weak links within the system. By executing this approach, one decomposes the system into three dimensions: physical, functional, and temporal. Intermediate outputs of the methodology include two- and three-dimensional matrices that cleanly delineate relationships between subsystems/components, functions, and failure mechanisms. By inspection, one can quickly identify those subsystems/components, functions, and failure mechanisms that contribute to the largest portions of the initial system-level failure intensity. The discussion by Yadav et al. [15, 16] is based on the assumption that the system is a series-only configuration of subsystems; therefore, we would need to modify the approach for additional system configurations to be more generally applicable.

3.3 Evaluate the Reliability of the System Correctly

Recall from Chapter 2 that, for the majority of complex Army systems, it is unlikely that we will be able to exercise the full-up field configuration of the system during the reliability demonstration event. However, under these circumstances, it is likely that we will have at least a single system of each type from the SoS under test during the reliability demonstration event. As a notional example, let us say that our SoS is an air defense battery. The reliability block diagram associated with the notional SoS air defense battery is depicted in Figure 1.

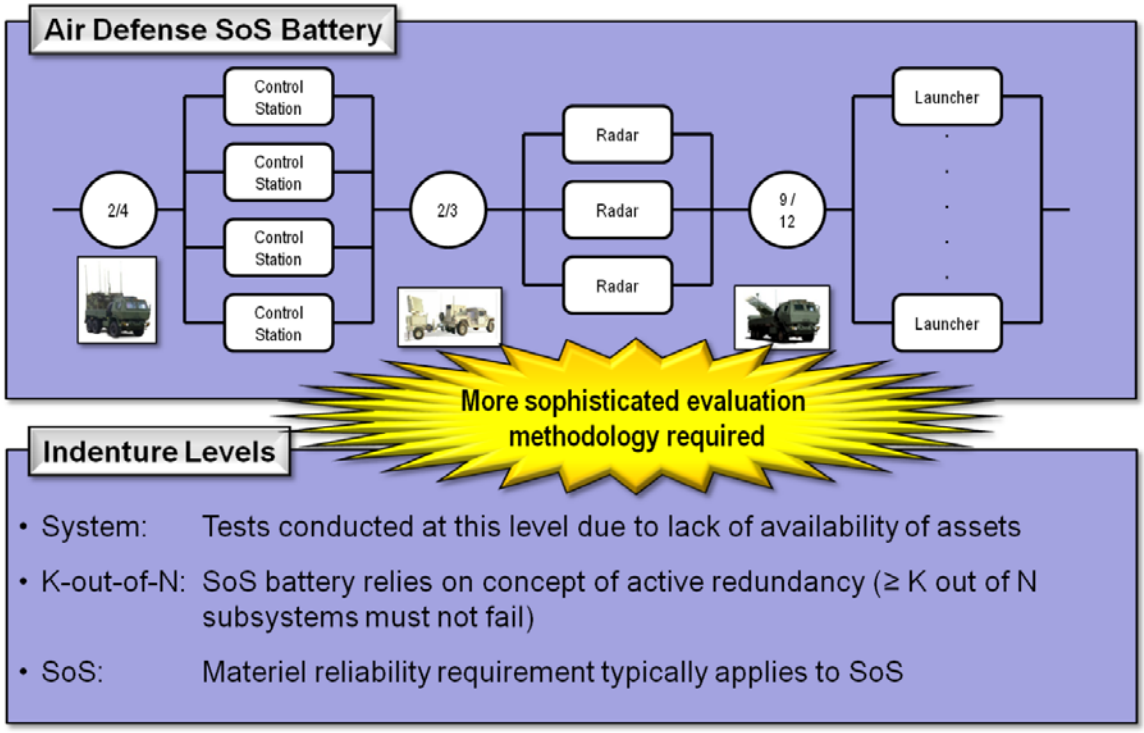


Figure 1. Notional Air Defense System-of-Systems Reliability Block Diagram

Without loss of generality, let us say that during the SoS reliability demonstration event, we will have 2 control stations, 2 radars, and 2 launchers. By inspection of the SoS reliability block diagram in Figure 1, it is easy to see that we will not have the opportunity to exercise the full SoS during the reliability demonstration event. Despite

the apparent disparity, we still have an SoS-level reliability requirement, and we must use the information obtained during the operational test in which the "2-2-2" SoS configuration is exercised. Under these circumstances, we must use the system-level data to build-up our SoS-level reliability estimate. Nelson and Hall [17] recommend using the system-level reliability estimates to build-up the SoS-level reliability estimate. The basic steps in the approach by Nelson and Hall are as follows.

1. Estimate the failure recurrence rate for each type of system on test (from our example, the 3 types are: control station, radar, and launcher).
2. Take the failure recurrence rate estimate for each type of system and apply it to all systems within each k-out-of-n structure (all systems of the same type are assumed to behave identically). Use the standard reliability block diagram method to develop the reliability expression for each k-out-of-n structure. For computational convenience, Nelson and Hall recommend the use of the beta cumulative distribution function to calculate the k-out-of-n structure reliability estimate based on k, n, and the estimated failure recurrence rate for the type of system in the structure (e.g., launcher).
3. As each k-out-of-n structure is in series, the SoS-level reliability expression is simply the product of the k-out-of-n structures.
4. Compute the lower one-sided likelihood ratio limit on the SoS-level reliability. Based on work by Jeng and Meeker [45], the likelihood ratio limit has been found to provide a result for which the true confidence is expected to be closer to the desired level of confidence than the traditional one-sided lower confidence bound.

For systems consisting of multiple non-repairable components in configured series, one may wish to consider using the Lloyd-Lipow (Lindstrom-Madden) methodology for developing system-level reliability estimates [18]. Effectively, the Lloyd-Lipow method randomly combines test results for individual system components in order to obtain a system-level reliability estimate. One may think of the result as a vector of length n, where n is equal to the number of components in the system. Each element in the reliability vector will either be 0 (component did not fail during its associated test) or 1 (component did fail during its associated test). As the methodology applies for series-only systems, a single occurrence of the value of 1 in the reliability vector is interpreted as a system-level failure. For series-parallel systems, the Maximus method can be used to generate point estimates and confidence bounds on system-level reliability [19]. Coit also provides a method for calculating confidence intervals for systems with active redundancy by using component-level failure data [34], and no assumption regarding the time-to-failure distributions for the components is required.

In certain cases, it may be appropriate and desirable to pool failure data from multiple test events from a given RTP. There are various reasons one may wish to follow such an approach, e.g., to reduce the overall length of the RTP, or improve the robustness of the characterization of the reliability behavior of a given system. Practitioners may wish to combine subsystem- or system-level failure data from a mix of developmental and operational test events in order to generate a system-level reliability estimate. In [10], a National Academy of Sciences (NAS) panel suggests that, in addition to using information from developmental and operational testing, it may be constructive to leverage information from training exercises, other less controlled uses of the system, and

information previously acquired on similar systems with similar components. Further, the NAS panel cautions that underlying model assumptions must be dutifully confirmed, and that the process by which information is combined (potentially involving subjective judgment) must be completely transparent to all stakeholders [10].

There are numerous approaches discussed in the literature that follow a Bayesian framework for combining failure data from disparate sources. Martz and Waller provide a comprehensive treatment of the subject of Bayesian reliability analysis techniques in [31]. One particular approach is provided by Reese et al. [20], in which the authors discuss the application of a hierarchical Bayesian reliability model for an anti-aircraft missile system. The strengths of the Bayesian approach discussed in [20] include the capability to include diverse sources of information (failure data and subject matter expert opinion) from different test events at different levels of fidelity (component, subsystem, system, etc.).

Arguably, the most important consideration associated with the application of Bayesian methods is the selection process for prior distributions. Wayne and Modarres [29] have developed a method for the generation of prior distributions based on the premise of maximum entropy—a subject previously discussed by E.T. Jaynes [21, 22]. Essentially, the concept of maximum entropy is related to Claude Shannon's theory of information [30], and the goal of the approach, in the context of generating prior distributions, is to give no more credit than is due to each source of information. I.e., the outcome is a minimally-biased result for the system-level reliability estimate [29].

Despite the well-known analytical robustness and flexibility offered by Bayesian-based methods, acceptance of such methods for the purpose of system reliability

evaluation has not been widely achieved within the DoD. Typically, the major area of concern with Bayesian-based methods is the appropriateness of leveraging data from sources other than the system's reliability demonstration event. In a 2013 memorandum [33], the Director of Operational Test and Evaluation (DOT&E), Dr. J. Michael Gilmore, asserted that he is "amenable to the potential of using Bayesian methodologies as well the potential assimilation of OT and relevant DT reliability data but only after observation of the subject tests and confirmatory analyses have been performed to verify it makes sense do to so." In general, the evaluation of a given system's operational reliability must be based on observed performance under operationally realistic conditions. According to a paper by Hall et al. [32] from DOT&E, such conditions include, but are by no means limited to

- who the system operators are (e.g., Soldiers, contractors, or government testers),
- who the maintainers are (e.g., Soldiers, Field Support Representatives, government maintainers, or a combination thereof),
- what the environmental conditions of the tests may be (e.g., blowing dust, blowing sand, solar loading, rain, dense fog, snow, ambient temperature, relative humidity, day/night, etc.), and
- what operational aspects that the tests may or may not include (e.g., operational tempo, force-on-force missions with a credible opposing force, mission durations, mission types to be executed, electronic warfare, information assurance, threat computer network operations).

If a data source does not conform to the above conditions that are fitting for the system under evaluation, it is extremely unlikely that combining such data will be deemed

appropriate by the DoD oversight community. As it is quite rare, in practice, that events other than the operational test (reliability demonstration event) will be conducted in a manner consistent with the actual operational test, it is uncommon for reliability evaluators in the DoD to apply Bayesian-based methods to analyze system reliability. This lack of homogeneity in testing can be mitigated by designing additional system test events to mirror the conditions of the operational test. Yet, until supporting evidence exists indicating that it is appropriate to leverage system reliability information from alternative sources, it is preferred to conservatively plan for the evaluation of system reliability to be based solely on the stand-alone results from the single reliability demonstration event.

3.4 Identify and Manage the Evaluation Risks of the RTP

Based on the search for existing tools and techniques in the literature, it is apparent that no documented approach currently exists that is geared toward assessing the evaluation risk associated with a given RTP. The appendix of this dissertation includes a questionnaire that we administered to reliability evaluators for 10 Major defense Acquisition Programs (MDAPs). Subsequent to the receipt of the responses to the questionnaire, follow-on interviews were conducted. As anticipated, the only evaluation risks considered during planning of the RTP were the risks of committing type I and type II statistical errors based on the outcome of the reliability demonstration event—also referred to as the initial operational test.

In 2004, the NAS published findings [10] subsequent to a thorough examination of the planned initial operational test for Stryker, which is a wheeled combat vehicle

program. In the report [10], the NAS panel asserts that the qualitative, non-statistical aspects of an operational test, such as in the case of Stryker, are substantially more important than the statistical aspects, such as consumer and producer risks. Further, the NAS panel adds that this inadequacy is not particular to the Stryker program; the panel asserts that it is likely the case for the majority of MDAPs [10]. Unfortunately, based on the responses to our questionnaire and targeted interviews, the message from the NAS panel has not yet had the desired impact with respect to reliability test program planning.

Therefore, in Chapter 4, we present a process that may be employed in order to assess the nature and severity of risks associated with reliability test program adequacy. Also in Chapter 4, we present several examples of potential reliability evaluation risk indicators along with suggested assessment rationale. In Chapter 5, we will discuss lessons learned through the application of the RTP evaluation risk assessment process to 10 MDAPs. We note that continued applications of the evaluation risk assessment process to additional acquisition programs would certainly yield further insights regarding potential evaluation risk indicator categories. Ultimately, we could build a comprehensive "library" of RTP weaknesses or risk categories to share between DoD and industry partners. We could further classify each evaluation risk category by the type of system for which we expect to potentially be affected—some categories may be universal, whereas, other categories may only apply to particular types of systems.

4 EVALUATION RISK ASSESSMENT PROCESS FOR RELIABILITY TEST PROGRAMS

4.1 The Purpose of Reliability Test Programs

A reliability test program (RTP) is designed to investigate and quantify the reliability characteristics of a given system from a mission-based perspective. More precisely, it provides information about the nature and recurrence rate of system failures that can be expected in the operational environment and evaluates this behavior against user-defined requirements. However, due to budget and schedule constraints, it is typically infeasible to conduct an exhaustive RTP. Evaluation risk is the risk that the reliability evaluation will be insufficient to characterize the reliability behavior of a given system. A system RTP must be designed carefully by assessing and mitigating evaluation risk. This chapter discusses the minimum requirements for an RTP, lists the associated evaluation risks, and presents a framework for assessing these evaluation risks in the context of US Army test and evaluation activities. This evaluation risk assessment approach can be easily applied to any type of military or commercial system.

4.2 Review of the Minimum Requirements for Reliability Test Programs

- Requirement 1: Evaluate the Reliability of the Field Configuration of the System
- Requirement 2: Test all Elements of the System
- Requirement 3: Exercise all Anticipated Unreliability Drivers
- Requirement 4: Evaluate the Reliability of the System Correctly
- Requirement 5: Identify and Manage the Evaluation Risks of the RTP

4.3 Reliability Evaluation Risk Indicators

The reliability evaluation risks describe the ways in which an RTP plan may fail to meet the 5 minimum requirements listed in Section 4.2. Table 1 lists important general risk indicators (or risk areas) as well as the rationale for assessing the severity of each risk. This list can be easily tailored to the type of system under evaluation. Below, we elaborate on the 7 evaluation risk indicators from Table 1.

Table 1. Reliability Evaluation Risk Indicators and Assessment Rationale

Risk Indicator Category	Low Risk	Medium Risk	High Risk
1. Potential to surface unreliability drivers	Full spectrum of OMS/MP conditions included in test plan	OMS/MP conditions expected to be encountered most frequently included in test plan	Small subset of OMS/MP conditions are included in the test plan
2. Potential to observe impact of system reliability on mission accomplishment	Robust operational testing planned with users; includes full range of missions	Operational testing is planned that includes users, and mission vignettes consist of subset of missions representing majority of anticipated roles/functions	Little or no operational testing is planned; only a small fraction of mission roles are included; minimal interaction with representative users
3. Potential to document impact of system reliability on user experience	User surveys developed targeting complete range of system functionality; surveys will be administered immediately following missions	User surveys developed targeting subset of system functionality; surveys will be administered immediately following missions	No user surveys will be administered during the RTP
4. Likelihood of detecting system-to-system variation in reliability of 25%	> 80%	60%-80%	< 60%
5. Likelihood of detecting a 25% drop in reliability from DT to OT environment	> 80%	60%-80%	< 60%
6. Planned proportion of the fielding configuration to be tested during the initial operational test (IOT) –the demonstration event	Full fielding configuration will be under test during the initial operational test	Only a subset of the intended fielding configuration will be under test during the IOT; data from alternative events during the RTP will be leveraged for system-level evaluation	Only a subset of the intended fielding configuration will be under test, but system-level evaluation will not leverage alternate events from the RTP in the evaluation
7. Planned data sources	Test Incident Reports (TIRs) and Instrumented Data	TIRs Only	Survey responses only, or no formal data collection process or products

1. Potential to surface unreliability drivers

Through various techniques (some of which are discussed in Chapter 3), it is possible to identify anticipated unreliability drivers for a given system. However, such an identification process is conducted prior to the start of the RTP. Therefore, the initial list of anticipated unreliability drivers may not precisely align with those that will be observed during the RTP. In order to mitigate the risk associated with not observing a

prominent unreliability driver during the RTP, it is important to include a range of operational mode summary/mission profile (OMS/MP) conditions in the RTP. From a practical standpoint, it is unlikely that every aspect of the OMS/MP for the system will be included in the RTP. However, as we include more elements from the OMS/MP in the RTP, we reduce the risk of not observing reliability issues that will have an appreciable adverse impact on the reliability of the system. This risk is related to Requirement 1.

2. Potential to observe impact of system reliability on mission accomplishment

There are many aspects included in the OMS/MP for a given system—terrain, climate, types of missions, mission duration, etc. If the system reliability evaluator is to build a comprehensive characterization of the impact of the reliability of the system under test on mission accomplishment, it is critical to exercise the system in a mission-based context. Operational testing events combine representative users, the operational environment (or a close approximation), and mission vignettes in order to collect some evidence of how the system will perform in a mission-based context. This risk is related to Requirements 3 and 5.

3. Potential to document impact of system reliability on user experience

Although user feedback is qualitative in nature, the insights gained can prove to be very influential at program decision milestones. For example, the Army may decide to buy and field 50,000 vehicles as-is. Alternately, before fielding the system, or the Army may require the program manager to address certain issues that degrade the user experience. In some cases, changes to the scheduled maintenance or “reset” may be

sufficient to improve the user experience. For a software-intensive system that tends to crash, perhaps the training manual for the system could be developed further to improve the user's understanding of what to expect and how to handle known issues. Some use the term "graceful degradation" to refer to cases in which a reliability failure occurs (not necessarily a system abort, possibly a lesser class of failure), but the user experience is only modestly degraded in the process. This risk is related to Requirement 5.

4. Likelihood of detecting a 25% system-to-system variation in reliability

This risk is also related to Requirement 5. When we use the term "system-to-system variation," we are generally referring to one of two cases. In the first case, we have multiple "identical" systems, but at least one system behaves differently (say, from a reliability standpoint) than the others. In the second case, we have distinct variants or configurations within a family-of-systems. It is not uncommon to observe peculiarities among individual test articles in either case during the RTP. Clearly, it is a desirable property for a given RTP design to offer a reasonably high likelihood that practitioners could identify a statistically-significant variation among test articles.

The approach that we propose (though, certainly not the only approach) is to stochastically simulate the failure behavior for each variant, taking into account the number of test articles and test time per article (may be in terms of hours, miles, cycles, etc.). We set the "baseline" reliability of one of the variants, and we then set the reliability for the other variant lower than the first variant. Hence, we assume that, say, the true reliability of the first variant is 1,000 mean miles between operational mission failure (MMBOMF), and the true reliability of the second variant is 900 MMBOMF (i.e.,

10% lower than the first variant). Next, we use a statistical test recommended by Nelson [35] to determine whether or not there is evidence, at a specified level of significance, that the reliability (more precisely, the Poisson failure recurrence rate) of each variant is not the same. We repeat this simulation, say, 1,000 times and we calculate the proportion of times that the test successfully detects the difference in the true (as established for the simulation by the analyst) failure recurrence rate between the two variants. The result is a theoretical (and purely mathematical) estimate of the likelihood that we will be able to statistically detect a difference in the reliability of the two vehicle variants.

The "success" proportion (number of times that the statistical comparison test described above successfully detects the difference in the failure recurrence rates of the two vehicle variants) that we obtain via stochastic simulation is not necessarily intended as an absolute measure; rather, it may be used to compare different test options (quantity of test vehicles, miles per vehicle, etc.). Applied consistently across RTPs for different systems (not just the vehicle from our hypothetical example here), we now have a standard approach for assessing the likelihood that a given RTP will enable us to detect system-to-system variation.

A criticism of this approach, perhaps, is that it is purely mathematical and does not address, in any manner, differences in actual failure modes for each vehicle variant. However, it would seem reasonable to expect that, from an evaluation standpoint, we would be able to identify differences in the failure modes between variants (i.e., existence of failure modes on one variant only, or different failure recurrence rates for particular failure modes), provided that we plan to accumulate a sufficient amount of mileage on each variant. Here, the term "sufficient" refers to the notion that we have planned to

accumulate enough mileage per variant in order to have the potential to observe the presence of various unique failure modes. As part of the process to develop a reliability growth planning curve, we explicitly consider the expected number of new failure modes during each test event in the RTP. Given that we already explicitly consider the capability to surface an adequate quantity of failure modes, we only consider the potential to detect differences in the failure recurrence rates among distinct variants.

One final note for this evaluation risk indicator category—while we do recommend using a 25% system-to-system variation in the failure recurrence rates of the test articles, in practice, multiple differences should be considered (such as 10%, 25%, and 50%) in order to get a relative notion of the risk associated with different courses of action. As a general rule, the likelihood of detecting a statistically-significant variation in system-to-system reliability is directly related to the amount of test time per test article.

5. Likelihood of detecting a 25% drop in reliability between DT and OT environments

This risk, like the previous one, is also related to Requirement 5. Historically, we have observed the trend that the reliability estimates for a given system (independent of commodity area) are appreciably lower during operational testing (OT) than during developmental testing (DT). In particular, we have observed as much as an 83% drop in system reliability between DT and OT. For reliability evaluators (and Army decision-makers), it is highly important to determine whether or not the observed drop is a true drop, or if it is instead simply an artifact of chance. Therefore, as in evaluation risk indicator category 4 above, we want to consider our ability to detect a drop in system-level reliability as we transition from a DT environment to an OT environment.

We propose to apply the same approach that we described for gauging the likelihood of detecting system-to-system variation. Here, we are no longer comparing the two vehicle variants; instead, we are comparing DT and OT test results. Thus, our stochastic simulation would be constructed assuming a particular DT system-level reliability and an OT system-level reliability that is 25% less than that of the DT system-level reliability. Applying the same procedure discussed for evaluation risk indicator category 4, we determine the proportion of time we would expect to successfully detect a drop (for a specified level of statistical significance) in the true reliability of the system.

Our decision to adopt 25% as the standard drop to consider in this evaluation risk indicator category is somewhat arbitrary, in that even a 1% reduction in the reliability of the system implies that ownership costs would, potentially, be higher, and the likelihood of mission success would be lower. However, it makes sense to apply some standard for practitioners and decision-makers, and a 25% drop is easily recognizable to decision-makers as significant. As in our discussion of system-to-system variation in reliability (evaluation risk indicator category 4), we could certainly consider other drops (e.g., 10% or 50%) along with the standard 25% drop. In practice, decision-makers may indicate a particular preference.

6. Planned proportion of the fielding configuration to be tested during the IOT

The field configuration for a given system may include multiple distinct variants (sometimes referred to as configuration items), or may be a system-of-systems (such as an air defense system, including a collection of radars, ground control stations, launcher units, and munitions). In many cases, not all portions of a given system may be available

during the reliability demonstration test (typically referred to as the initial operational test [IOT]). Clearly, the system evaluation will be more robust if all components of the system will be exercised as intended under operational conditions during the IOT. From an evaluation standpoint, if not all components of the system are available, then there will necessarily be gaps in our knowledge and characterization of the system. This risk is related to Requirements 1, 2, and 5.

In the event that certain components will not be exercised during the IOT, it may be possible to obtain data for those components from other events during the RTP (for example, a developmental test completed prior to the IOT). A Bayesian hierarchical approach similar to that discussed by Reese [36] could be adapted to incorporating data from test events other than the IOT along with the actual results from the IOT. In addition, it may be a viable option to represent the missing components of the system during the IOT through the use of real-time simulations. Such an approach will only be acceptable upon completion of validation, verification, and accreditation (VV&A) of such models.

7. Planned data sources

Test Incident Reports (TIRs): The test center personnel capture details regarding notable incidents that occur during each test event. Typically, information included in a TIR may consist of attributes such as mileage, operating hours, location, and actions taken (e.g., diagnostics or repairs). TIRs serve as the basis for calculating the reliability tracking estimates for the system under test. During system testing, it may be necessary to modify the original TIR data collection plan in order to capture all of the relevant

information regarding a particular incident. In the event that this situation occurs, the reliability evaluator can work with the test center personnel to ensure the key data are captured for the remainder of the test (or during future test events).

Instrumented Data: Multiple options exist for capturing quantitative (objective) system data via on-board instrumentation. For example, attributes such as engine speed, road speed, throttle position, orientation (roll/pitch/yaw), ambient temperature, engine temperature, and message completion rate can be recorded. When the potential exists to incorporate instrumentation in the data collection plan, it is possible to perform in-depth analyses, exploring circumstances surrounding each incident. As well, if the same instrumentation package is used during DT and OT, we gain the potential to identify variations in reliability behavior due to test environment, operators, etc.

User Surveys: Developing reliability tracking estimates and cataloguing failure modes provides a rich amount of insight into the suitability of a given system, but feedback from actual users can be illuminating with respect to the military utility of the system. Along with TIRs and instrumented data user surveys offer the potential to round-out the holistic perspective of system reliability, thus reducing the risk that the evaluation will be insufficient to characterize the reliability behavior of the system. This evaluation risk indicator category is related to Requirements 4 and 5.

4.4 Framework for Army RTP Plan Evaluation Risk Assessment

The evaluation risk assessment procedure is similar to a traditional Failure Mode and Effects Analysis (FMEA). This FMEA-like procedure yields a synopsis of identified RTP weaknesses along with the potential impact of each weakness with respect to the

reliability evaluation. In Chapter 5, we apply this risk assessment framework in the examination of the RTP details for 10 systems that are Major Defense Acquisition Programs (MDAPs). For these 10 systems, we identified certain weaknesses—issues that would potentially erode the adequacy of the system-level reliability evaluation—along with potential alternatives to reduce evaluation risks related to these weaknesses. The 10 systems chosen form a representative cross-section from the various types of commodity areas. Hence, we expect the insights, or "lessons learned," that we document regarding RTPs to have wide applicability within the Army as well as the other Services within the Department of Defense (DoD) and industry. Although we employed this approach after each RTP was either in-progress or complete, the same approach could be used during the planning process. In fact, we assert that this approach should be practiced during the RTP planning process in order to assess risks and assist program managers in the allocation of resources in the overarching test program.

Our approach for analyzing RTPs is designed to achieve the following four overarching objectives.

- Document the RTP details.
- Analyze the RTP details in order to identify potential weaknesses (or "failure modes").
- Assess the impact of each weakness on the outcome of the RTP.
- Formulate corrective actions that we expect to reduce the evaluation risk.

Resolving potential weaknesses in an RTP plan requires support from Army leadership. Because reliability is one of many concerns, Army leadership may be willing to accept greater evaluation risk due to budget and schedule constraints. The Army Center for

Reliability Growth is continuously monitoring the outcomes from testing in order to determine which aspects of RTPs are working and which aspects are failing with respect to system reliability evaluations.

For example, the current Army reliability test and evaluation paradigm is to conduct a single demonstration event (typically referred to as the IOT), employing the system under operationally realistic conditions with Soldiers as operators. In many cases, the system configuration that will be exercised during the demonstration event may not be precisely equivalent to the intended fielding configuration of the system. In such cases, stakeholders (i.e., the program management office, the user community, and the evaluator) must agree during the planning stage that the test configuration is sufficiently representative of the fielding configuration of the system (from a reliability standpoint). Depending on the design of the actual system, this may not be a reasonable assumption to make when planning the RTP, and it would constitute a weakness in the RTP plan. For certain types of systems, the impact of such a weakness could be a significant distortion of the true system reliability behavior. As a consequence, the Army may have to absorb increased ownership costs and adapt to higher than anticipated system down-times over the life of the system fleet.

To mitigate this risk, the Army could plan to complement the IOT failure data with system failure data from an event other than the IOT. Such an event could potentially include the portion of the system that will not be available during the IOT, and the event may already be part of the RTP for the system. As we discussed in Chapter 3, pooling system-level failure data from the IOT along with failure data from another RTP event is feasible only if the stakeholders agree on the appropriate criteria. Even if the

established criteria for pooling the failure data are met, certain system failures stemming from integration or interaction may remain unobserved.

4.5 Traditional FMEA Steps

Per MIL-STD-1629A [11], a FMEA has the following 8 steps:

a. Define the system to be analyzed.

b. Construct block diagrams.

c. Identify all potential item and interface failure modes and define their effect on the immediate function or item, on the system, and on the mission to be performed.

d. Evaluate each failure mode and assign severity classification category (catastrophic, critical, marginal, or minor).

e. Identify failure detection methods and compensating provisions for each failure mode.

f. Identify corrective design or other actions required to eliminate the failure or control the risk.

g. Identify effects of corrective actions or other system attributes.

h. Document the analysis and summarize the problems which could not be corrected by design and identify the special controls which are necessary to reduce failure risk.

A criticality analysis can be considered an extension to steps a-h.

4.6 RTP Evaluation Risk Assessment Process

Our proposed evaluation risk assessment process has the following 7 steps, which closely follow the 8 steps of a FMEA:

a. *Define the RTP*.

The purpose of this initial step is to clearly lay out all pertinent aspects of a given RTP. As the program master schedule is not driven solely by the RTP events, any RTP planning should include coordination with all stakeholders. At a minimum, the RTP should include the following information:

- the nature/type of RTP events
- the duration of each RTP event in terms of system operating time (hours, miles, cycles, rounds, etc.) and calendar time
- the number of items on test and the configuration of each item
- the test conditions (such as test location, Soldier involvement, and time of year)
- the reliability growth planning curve (RGPC) for the program (see appendix section A.3 for additional background on RGPCs)

It is the policy of the DoD for all of the MDAPs to develop a viable reliability growth plan [2]. The RGPC is a management tool used to aid stakeholders in the formulation of a feasible reliability growth plan, and it serves as a kind of reliability block diagram (RBD) for the RTP. An RBD is a graphical representation of the relationships between all elements of a given system. We can scrutinize the RGPC similar to the manner in which we would intend to scrutinize the RBD in the traditional FMEA process. As with the RBD, the RGPC depicts the relationships and dependencies between various events in a

RTP. RGPC development is the responsibility of the acquisition program management office; however, the reliability evaluator is obligated to participate in the development of the RGPC. For additional background on RGPCs, we refer readers to the AMSAA Reliability Growth Guide [5] or Military Handbook 189C [37].

b. *Identify potential weaknesses in the RTP with respect to evaluation adequacy*.

Table 1 lists reliability evaluation risk indicators that can be used to begin generating a list of potential weaknesses associated with a given RTP. The elements in Table 1 are by no means exhaustive, and the evaluators should consider other ways in which the RTP may fail to evaluate reliability adequately.

c. *Identify monitoring/detection methods and contingency plans*.

How will we know if a given RTP weakness has led to a "failure?" For example, if an initial RTP planning assumption is that the reliability characteristics of two variants within a family-of-systems are effectively the same (sometimes referred to as the "commonality" assumption), we can periodically re-examine this assumption once we have additional failure data from system-level testing. The monitoring frequency may be driven by program decision points, time delays to cut-in design changes, etc.

d. *Develop RTP planning alternatives that mitigate, in whole or in part, the potential weaknesses identified in step b.*

If we revisit the "commonality" example from step c, we could develop an RTP that does not assume commonality across each distinct system variant. Of course, this

adjustment to the RTP would likely result in higher test costs and, possibly, additional calendar time to execute the RTP. Potentially, we could add more shifts per week in the test program to stay on schedule, but we may have to absorb additional costs associated with more test articles and test time to characterize the reliability behavior of each distinct variant.

e. *State the perceived impact associated with the implementation of planning alternatives developed in step d.*

As alluded to in step d, the implementation of planning alternatives may result in increased test costs, but the benefit would be the potential improvement to the robustness of the reliability evaluation, i.e., reduced uncertainty in the characterization of system reliability behavior.

f. *Document the analysis and summarize the problems which could not be corrected by changing the RTP design and identify the special controls which are necessary to reduce failure risk.*

g. *Provide a recommendation to decision makers based on the results from steps a through f of the RTP evaluation risk assessment process.*

4.7 Concluding Remarks

In this chapter, we presented minimum requirements for RTPs, a list of reliability evaluation risks, and an evaluation risk assessment process for RTPs. This approach can be employed to scrutinize any system RTP plan and may be employed to compare

multiple RTP design alternatives for the same system. Ultimately, in a resource-constrained atmosphere, reducing the uncertainty associated with the characterization of system reliability will need to be balanced with other competing evaluation priorities. In Chapter 5, we will discuss the findings derived from our application of the RTP evaluation risk assessment framework to the RTPs for 10 Major Defense Acquisition Programs.

5 FINDINGS FROM THE APPLICATION OF THE RELIABILITY TEST PROGRAM EVALUATION RISK ASSESSMENT PROCESS TO 10 MAJOR DEFENSE ACQUISITION PROGRAMS

5.1 Army Evaluation of System Effectiveness, Suitability, and Survivability

Before the decision to field an Army system can be made, the Army Test and Evaluation Command (ATEC), serving as the Army's independent evaluator of materiel systems, plans and conducts developmental and operational testing to provide essential information to acquisition decision authorities. The driving force behind conducting a rigorous test and evaluation program is to gauge whether or not a given system will satisfy the essential mission capabilities required by commanders in the field. In particular, ATEC system evaluations focus on three major areas—effectiveness, suitability (which includes reliability), and survivability. From Army Regulation 73-1 [38], we have the following definitions for each evaluation area:

- Effectiveness: The overall degree of mission accomplishment of a system when used by representative personnel in the expected (or planned) environment. Some examples of environment are: natural, electronic, threat, and so forth for operational employment of the system considering organization, doctrine, tactics, survivability, vulnerability, and threat (including countermeasures; initial nuclear weapons effects; nuclear, biological, and chemical contamination threats).
- Suitability: The degree to which a system can be satisfactorily placed in field use with consideration given to availability, compatibility, transportability, interoperability, reliability, wartime usage rates, maintainability, safety, human

factors, manpower supportability, logistic supportability, and training requirements.

- Survivability: The capability of a system and crew to avoid or withstand manmade hostile environments without suffering an abortive impairment of its ability to accomplish its designated mission.

5.2 Evaluation Risk is Inescapable

In a given year, more than 500 Army systems are in various phases of testing [39]. Given the fierce competition for resources and the urgent need to meet the Army's operational requirements, it is impractical to design RTPs that will completely eliminate the uncertainty in characterizing a system's behavior in the actual operational environment. Yet, it is possible for practitioners to assess the evaluation risks associated with a given RTP, and, if the evaluation risks are unacceptable, revise the RTP to mitigate the risks.

In Chapter 4, we presented a process for performing an evaluation risk assessment of any system's RTP. This chapter discusses the application of this evaluation risk assessment process to the RTPs for 10 systems that are Major Defense Acquisition Programs (MDAPs) and reviews the findings of this study. Performing this process resulted in the identification and analysis of weaknesses in the RTPs examined and produced ideas for mitigating the evaluation risks, or countermeasures. In the remainder of this chapter, the evaluation risk assessment process steps are reviewed and along with reliability evaluation risk indicators introduced in Chapter 4. As well, we discuss the RTP weaknesses that we identified for the 10 MDAPs along with the mitigation activities

suggested in each case. Presenting these examples not only illustrates the usefulness of this evaluation risk assessment process but also provides test and evaluation practitioners with ideas for improving the RTPs that they design.

5.3 Review of the RTP Evaluation Risk Assessment Process

As stated in Chapter 4, our approach for analyzing Army RTPs is designed to achieve the following 4 objectives:

- Document the RTP details.
- Analyze the RTP details in order to identify potential weaknesses (or "failure modes").
- Assess the impact of each weakness on the outcome of the RTP.
- Formulate corrective actions that we expect to reduce the evaluation risk.

Resolving potential weaknesses in a RTP plan requires support from Army leadership. Because reliability is one of many concerns, Army leadership may be willing to accept greater evaluation risk due to budget and schedule constraints. The Army Center for Reliability Growth is continuously monitoring the outcomes from testing to determine which aspects of RTPs are working and which aspects are failing with respect to system reliability evaluations. The RTP evaluation risk assessment process has the following 7 steps:

a. Define the RTP.
b. Identify potential weaknesses in the RTP with respect to evaluation adequacy.
c. Identify monitoring/detection methods and contingency plans.

d. Develop RTP planning alternatives that mitigate, in whole or in part, the potential weaknesses identified in step b.
e. State the perceived impact associated with the implementation of planning alternatives developed in step d.
f. Document the analysis and summarize the problems which could not be corrected by changing the RTP design and identify the special controls which are necessary to reduce failure risk.
g. Provide a recommendation to decision makers based on the results from steps a through f of the RTP risk assessment process.

5.4 Reliability Evaluation Risk Indicators Revisited

The potential weaknesses in a given RTP are the evaluation risks. Based on our experience with RTP planning and execution, we identified 7 risk indicators in Chapter 4—these are briefly reviewed here. Additional details regarding the underlying rationale for each evaluation risk indicator category can be found in Chapter 4.

1. Potential to surface reliability drivers

All of the OMS/MP conditions should be included in the RTP. The evaluation risk is greater if a small subset of these conditions is included.

2. Potential to observe impact of system reliability on mission accomplishment

The RTP should include robust operational testing with users and a full range of missions. The evaluation risk is greater if little or no operational testing is planned, a

small number of missions are included, or representative users do not participate in planned TP events.

3. Potential to document impact of system reliability on user experience

The RTP should include user surveys that cover all of the system's functions and are administered immediately following the missions. The evaluation risk is greater if no user surveys are administered.

4. Likelihood of detecting system-to-system variation in reliability

A common RTP planning assumption is that each test article (system) will have roughly the same reliability characteristics, given the same test exposure. The evaluation risk is greater if the likelihood of detecting statistically-significant differences in reliability of test articles is low.

5. Likelihood of detecting drop in reliability between DT and OT environments

During system IOTs, ATEC has documented as much as an 83% degradation in system-level reliability when compared to the DT portion of a system's RTP. The evaluation risk is greater for this risk indicator category if the likelihood of detecting statistically-significant drop in system-level reliability is low.

6. Proportion of the fielding configuration that will be tested during the IOT

The RTP should include a full fielding configuration in the IOT. The evaluation risk is greater if a subset of the intended fielding configuration will be tested and the system-level evaluation will not leverage the results from alternate events.

7. Planned data sources

The RTP should include both instrumented and non-instrumented (e.g. TIRs and surveys) data. The evaluation risk is greater if the only data sources are survey responses from system operators.

5.5 Description of 10 Systems Examined

In order for the findings of this chapter to be releasable to the general public, we have not used the actual system names here; instead, we distinguish between each by using the type of system. Specifically, we examined RTPs for the following 10 systems:

Missile Defense System

Designed to operate as a family-of-systems that will network, via a Joint[2] enabling network and system of systems common operating environment, existing systems, systems already under development, and new systems under development into a system-of-systems (SoS) architecture to meet the needs of the Joint forces. Unlike other acquisition programs that focus primarily on one system or vehicle platform, this missile defense program focus is on systems integration, common battle command, Joint

[2] The term Joint is used to denote that a given acquisition program is shared by more than one Service, e.g., the Army and the Navy.

enabling network, logistics and training to ensure operational requirements—such as combat identification, fratricide prevention, survivability, lethality, transportability, and maneuverability—are achieved.

Networked Communications System-of-Systems

This system will be the integrating communications network for the Army, optimized for offensive and Joint operations. It will be a framework which will utilize common standards and protocols for Army info-spheres and interface with and/or replace equipment in the legacy and interim forces. This networked communications system will serve as the Army's high-speed and high-capacity backbone communications network. It will be focused on moving information in a manner that supports commanders, staffs, functional units, and capabilities-based formations—all mobile, agile, lethal, sustainable, and deployable. This system will provide Army units with communication capability at-the-halt as well as on-the-move.

Networked System of Sensors

This system will serve as the next generation network-centric multiple intelligence enclave ground station architecture, providing broad access to ground- air- and space-based sensors and platforms for the Army. This system is designed to provide the Army with actionable intelligence, surveillance, and reconnaissance data from the battlefield, and it establishes the core framework for a worldwide distributed, network-centric, system-of-systems architecture that conducts collaborative intelligence operations and production. Additionally, this system will provide real-time tasking, enhanced data

access and fusion, and dissemination of information and products. It is designed to be dynamically re-configurable and rapidly deployable with a reduced footprint and total ownership cost.

Dismounted Battle-space Awareness System

This program integrates multiple Soldier systems and components and leverages emerging technologies to provide overmatching operational capabilities to all ground combatant Soldiers, their attachments and small units. These capabilities include increased command and control, situational awareness, embedded training, lethality, mobility, survivability, and sustainability. The objective is to meet the needs of all Soldiers who conduct ground close combat. This effort includes determining the optimal distribution of operational capabilities across teams and squads to maximize small unit mission performance.

Tracked Combat Vehicle

This initiative is, effectively an upgrade to an existing tracked combat vehicle system from the Army's inventory. Changes to the system are extensive and include: a modified engine and transmission, a re-designed hull structure, updated electronics, a new suspension, electronic rammer and gun drives, changes to the crew and driver's compartments, and changes to the support structure of the turret. Other changes may include vehicle health management and survivability improvements.

Unmanned Aircraft System

This system is designed to provide a responsive, agile, and flexible capability to perform reconnaissance, surveillance, and target acquisition as well as serve as a communications relay. Major subsystems of this unmanned aircraft system include: unmanned aircraft equipped with and electro-optical/infrared/laser designator payloads, ground control stations, data relays; and a satellite communication element. Additionally, all aircraft will be weapons capable, and the system will have sufficient mission equipment to support wartime operations. This system will be fully integrated into the combined arms air-ground team—conducting operations day and night, in adverse weather, in open, close, complex, and all other terrain conditions throughout the battle-space.

Wheeled Combat Vehicle #1

This system includes integral weapons designed to provide rapid and lethal direct fires to the supported assaulting infantry. The primary weapon is designed to defeat bunkers and create openings in reinforced concrete walls through which infantry can pass to accomplish their missions. It is also required to defeat armor up to that of a T-62 tank.

Wheeled Combat Vehicle #2

This system is designed to provide stationary and on-the-move “detect to warn” capabilities for nuclear/radiological and chemical hazards and “detect to treat” for biological hazards. This is by its ability to detect, and identify chemical, biological and radiological hazards. It provides the capability for early warning to units of potential

contamination, reports the location of hazards, marks areas of contamination, locates and marks clean bypass routes, and collects and transports samples of radiological, biological, and chemical material/vapors for later analysis.

Wheeled Combat Vehicle #3

This program is intended to demonstrate the effectiveness of chassis modifications (vehicle structural geometries and armor recipe) to an existing wheeled combat vehicle family-of-systems to improve survivability.

Wheeled Tactical Vehicle

This system is designed to provide protected, sustained, networked light tactical mobility to enhance the effectiveness of ground combat and supporting forces. This capability is required across full spectrum operations and under all weather and terrain conditions facing Joint Forces to enable the effects of operations at the tactical, operational, and strategic levels of war. The family of vehicles consists of several variants, including companion trailers that carry the specific variant payloads.

5.6 Selection Criteria for the 10 Examined Systems

Our selection of the 10 systems was based on 3 factors. First, we wanted to apply our evaluation risk assessment process to a broad cross-section of the types of systems that undergo testing and evaluation for the Army. By inspection, our list of 10 systems consists of a combination of manned, unmanned, air, ground, combat, combat service support, weapon, and sensor systems. Second, we wanted to examine systems that are in

different phases of the Defense Acquisition Management System, i.e., Technology Development, Engineering and Manufacturing Development, Production and Deployment, Operations and Support [40]. Third, all 10 of the systems that we examined are designated as Acquisition Category (ACAT) I systems. Per Department of Defense (DoD) Instruction 5000.02, ACAT I systems are those for which the Office of the Under Secretary of Defense for Acquisition, Technology, and Logistics has estimated the expenditure for research, development, test and evaluation (RDT&E) will exceed $365 million in fiscal year 2000 constant dollars, or will exceed $2.19 billion in fiscal year 2000 constant dollars [40]. ACAT I systems represent the greatest level of investment by the DoD.

Out of the 10 systems, only 1 is still in the RTP planning stage. One system has established its RTP plan and will begin the RTP in 2013. Of the 10 systems considered, 3 are nearing completion of their respective RTPs. The remaining 5 systems have already completed their RTP events. Given the various stages of RTP completion for the 10 systems, we were not only able to analyze the RTPs using our evaluation risk assessment process—we were also able to examine outcomes from ongoing and/or completed RTPs.

In order to perform the 7 steps in our evaluation risk assessment process, we needed to acquire key information and documents for each of the 10 systems. Thus, we deemed it essential to first contact each system's reliability evaluator. We generated and delivered a detailed questionnaire to each evaluator consisting of 21 items (see appendix section A.4). We requested that they compose written responses for each item in the questionnaire. Once we received the responses from the evaluators for all 10 systems of interest, we scheduled face-to-face meetings to discuss the information that they

provided. During the follow-on meetings, we also obtained key documents such as RTP plans, system requirements specifications, OMS/MPs, reliability growth planning curves, official test reports containing estimates for system reliability metrics, etc. These documents served as the inputs to our evaluation risk assessment process.

5.7 Poor Design Practices Identified during the Examination of 10 DoD RTPs and Recommended Countermeasures

Through our examination of the RTP details for the 10 systems in our study, we identified several poor design practices for RTPs—issues that could potentially erode the adequacy of the system-level reliability evaluation. We summarize these poor design practices below along with potential countermeasures to reduce evaluation risks related to these poor design practices. The following poor design practices, which were found in these 10 RTPs, increase evaluation risk.

Commonality Assumption

When the program manager assumes that all variants of a system will exhibit the same reliability behavior (i.e., have the same failure modes and failure recurrence rates), the RTP may include little or no testing of certain variants. However, if there are significant differences in the reliability characteristics of the variants, this assumption is not valid. Furthermore, if there is minimal testing of a variant planned in the RTP, we may not be able to develop an accurate characterization of the family of variants.

This weakness can be detected by establishing "checkpoints" based on accumulation of test time. If failure modes or recurrence rates are inconsistent, the commonality assumption is likely to be invalid. If the existing RTP plan already includes

one or more test assets for each variant, an appropriate contingency plan would be to increase the testing per variant. Planners must ensure there is enough test time allocated to obtain estimates with the desired accuracy (and statistical power). This weakness can be mitigated by not assuming commonality. Instead the RTP should be designed to balance testing across all variants. Such an alternative may require additional test assets, test range time, or test personnel.

Minimum Test Configuration

For a large system-of-systems (SoS), it may be infeasible to exercise the full field configuration during the demonstration event (or at any point during the course of the RTP). In this case, some subset of the SoS will be exercised; however, this minimum test configuration may not behave in a manner that is comparable to the full configuration of the SoS. Considering only the minimum test configuration may obscure interactions due to scale, integration issues, and similar concerns. As well, if the success of the system hinges upon an assumption that redundancy works as envisioned, the test regime should be robust enough to confirm that this is true under operationally-realistic conditions.

This weakness can be detected by conducting a modeling and simulation based validity analysis. For example, during the RTP planning stage, a computer-based model should be constructed to compare the actual system and the minimum test configuration. A range of scenarios should be included and the results compared. An appropriate contingency plan would require augmenting testing of the physical system with virtual components to simulate the loads, stresses, and interaction effects.

One approach for mitigating this risk is to design the RTP so that all portions of the SoS will be exercised (either in a single demonstration event, or over the course of multiple RTP events), or the user should re-consider the testability of the system reliability requirement(s). The evaluation risk would be mitigated substantially if the full field configuration of the SoS is exercised, but this alternative may be infeasible depending on the type of SoS.

"Bootstrap" Calculation of System-of-System (SoS) Level Reliability using only System-Level Failure Data

Although the term "bootstrap" has been used by the community in other contexts, we are referring to generating a SoS-level reliability estimate using only system-level failure data. As with the minimum test configuration approach, it may be impossible to obtain a SoS-level reliability estimate that includes failure modes due to integration and/or interaction of systems within the SoS. It really depends on how the system will be exercised. For example, will systems be tested independently, or will multiple systems from the SoS be tested cooperatively (and what proportion of the total number of permutations will be tested)?

This potential RTP weakness can be detected by performing a periodic review of failure modes surfaced and their root causes. An appropriate contingency action would be to revise the RTP plan to include opportunities to observe integration issues (multiple permutations of systems from the SoS on test in each event). Planners should design the RTP so that full configuration will be exercised (at least during the demonstration event), or user community should re-consider reliability evaluation expectations. If this

contingency action is performed, the evaluation risk would be mitigated substantially, but this alternative may be infeasible—from a resource standpoint—depending on the type of SoS.

Operational Mode Summary/Mission Profile (OMS/MP) Test Slice

This refers to the case wherein only a subset of the operational conditions will be included in the RTP. For example, a vehicle may have a "worldwide" OMS/MP; yet, the RTP does not include cold regions testing. Another example would be the case in which only a subset of missions specified in the OMS/MP are included in the operational testing activities for the system. Such cases can lead to an inaccurate evaluation of fleet reliability.

This potential weakness can be detected by consultation of the RTP plan—it should be clear whether or not certain missions, environments, or other aspects of the OMS/MP have been omitted. An appropriate contingency plan would be to augment the RTP with events tailored to exercise the system under remaining conditions/missions if deemed necessary. Alternately, it may be appropriate to develop a validated simulation to examine the missing aspects of the system OMS/MP. The most straightforward method to mitigate this weakness is to design the RTP to include full-spectrum of OMS/MP conditions/missions. Such an approach would enable a comprehensive evaluation of the system.

Insufficient Test Time to Surface Key Failure Modes

Due to resource constraints (e.g., calendar time, facilities, and personnel), the planned test time per asset in the RTP is less than the time in which a key failure mode has been observed in the past—also known as the time to first occurrence. Hence, we would not expect to observe such a failure mode during the RTP (though it is possible that such a failure would occur), and this will increase the uncertainty associated with the findings of our system reliability evaluation.

If data are available on similar systems, periodically compare observed failure modes to the list of those anticipated. An appropriate contingency plan would be to be prepared to increase system time on test. In order to gauge the amount of additional testing, consider anticipated (based on historical data) failure recurrence rates to gauge additional time needed. An alternative approach would be to size the RTP to surface postulated unreliability drivers. The potential benefit to this approach is an improved characterization of the system reliability behavior; however, such an approach may be cost-prohibitive. Employment of modeling and simulation tools such as hardware-in-the-loop facilities or purely computer-based analytical methods can also reduce uncertainty.

Low Likelihood of Detecting System-to-System Variation in Reliability

One test assumption often made is that all test articles are identical. If the RTP does not allow for a sufficient amount of test time per test article, the likelihood of detecting a variation among the individual test articles may be undesirably low. This weakness can be detected by performing up-front analysis of planned number of test assets and time on test per asset (power of the test).

Mitigating this weakness can be accomplished by re-configuring the RTP to include an appropriate composition of test assets and test time per asset to achieve the desired likelihood threshold for detection, i.e., increase power of test. This approach can be expected to reduce risk, but random variation may still obscure the truth. In the event that this deficiency remains unresolved beyond the RTP planning stage and is identified during RTP execution, an appropriate (and premeditated) contingency plan would be to protect sufficient funding to augment testing of assets. It may be possible to leverage test articles intended exclusively for performance or survivability testing; thus, the additional cost and time incurred would be due to exercising the test assets and collecting the data.

Low Likelihood of Detecting Drop in Reliability between DT and OT

Typically, the time on test during the developmental portion of the RTP is appreciably greater than the length of the operational reliability demonstration event. Thus, the analysis of the test results could detect only very large differences in system reliability going from the developmental test environment to the operational test environment. However, even a 5% difference in the reliability of the system can lead to a significant impact to ownership costs [4]. The guidance regarding the detection, contingency planning, and RTP planning alternative is essentially the same as in the system-to-system variation case, except for the fact that we are comparing results between DT and OT instead of system-to-system.

Technology Moving Faster than RTP

For certain types of systems (such as radios, information technology systems, and smart phones), the technology matures faster than the acquisition program can. New releases of software and hardware typically occur between the end of the RTP and the time of fielding. Thus, testing the exact version of the system that actually makes it to the field is virtually impossible.

In order to detect the presence of this issue, it is advisable to conduct a historical industry trend analysis for associated technology. Include requirement in contract for vendor to supply plans, specifications, and data for the next generation of the technology as soon as it becomes available. As a contingency, test the next generation (e.g., smart phone or radio) prior to deployment to verify reliability of new design, and monitor system configuration changes from test event to test event.

An alternative approach would be to design the RTP to include a reliability verification event in conjunction with fielding of the first unit equipped; or conduct a Forward Operational Assessment (FOA) in the actual operational environment, with actual users. Such an approach would result in an updated indication of the reliability behavior of the fielded system; can inform design, quality, or production changes.

No Instrumented Data Collection Activities Planned

Without an on-board instrumentation package, the reliability evaluator will have a limited amount of information available from which to develop insights. For a vehicle, system parameters such as engine speed, road speed, orientation (roll, pitch, and yaw), ambient external temperature, GPS location, and many more may be captured for near-

real-time or post-test analysis. Such information can be highly valuable during failure mode root cause analysis.

It is possible to detect this issue if there is an inability to answer questions regarding system behavior to desired level of fidelity (such as a comparison between a vehicle's engine speed and the road speed at the time just prior to a failure). As well, there may be an inability to perform root cause analyses, and formulate effective corrective actions. An appropriate contingency plan would add on-board/in-the-loop instrumentation package to capture key system attributes and develop context for each failure observed during testing. An alternative approach for the RTP would be to devise a feasible and affordable plan to instrument all (or some representative subset) of the test assets to capture key system attributes. Improved capability to investigate the circumstances surrounding observed failures; will certainly increase T&E costs (instrumentation, maintenance, data harvesting, data storage, and data analysis).

5.8 Concluding Remarks

In this chapter, we discussed the application of our evaluation risk assessment process to RTPs for 10 Major Defense Acquisition Programs. This process is based on the technique of failure mode and effects analysis (FMEA) and can be utilized for any type of system RTP, whether the system is intended for military or commercial use. Performing this process identified and analyzed poor design practices for the RTPs examined and resulted in the development of countermeasures to mitigate the evaluation risks linked to those RTPs.

These examples and results discussed here not only illustrate the usefulness of this evaluation risk assessment process but also provide test and evaluation practitioners with ideas for improving the RTPs that they design. This evaluation risk assessment process can be employed to scrutinize any system RTP plan and to compare multiple RTP designs for the same system. Ultimately, in a resource-constrained atmosphere, the desire to reduce the uncertainty associated with the characterization of system reliability must be balanced with other competing evaluation priorities.

The process of evaluating these 10 RTPs and documenting the poor design practices contributes to the process of learning from failure in the test and evaluation community. It is essential to learn from failures in RTP design because, at the least, repeating such mistakes results in inefficiency. In more extreme cases, continuing to embrace poor RTP design practices may result in inadequacy—but program managers may not even perceive the potential threat. As Hatamura asserts in [48], overt acknowledgement and documentation of failures do not tend to be organizational practices that are embraced. However, the value of identifying the poor design practices for RTPs described in Section 5.7 is only truly realized at the moment these insights are promoted to the level of organizational knowledge and made accessible to the workforce. Due to the stigma associated with broadcasting failures, or poor practices, Hatamura argues that insights regarding failures tend to remain within the immediate group that commits the error and are not archived within an organization's central knowledge repository.

Although Hatamura's discussion in [48] explores causes of failure in the design of systems, the same logic can be applied to the design of RTPs. As Hatamura explains, the process of learning from failure includes the following steps:

- description of the failure (phenomenon, cause, countermeasures generalization)
- recording the details of the failure (archived and accessible)
- transmission of the details of the failure (education, knowledge exchange)
- learning (analysis and classification/structure)
- experience (heightened awareness and improved readiness).

Thus, including the insights in this dissertation regarding the poor design practices for RTPs associated with a diverse collection of Major Defense Acquisition Programs begins the learning process by describing the failures and possible countermeasures, recording this information, and publishing this information in this dissertation and elsewhere.

6 PROPOSED PROCESS FOR RELIABILITY TEST PROGRAM PLANNING

6.1 Overview

Due to the reduction in available resources across the US Department of Defense, there is a renewed emphasis on operational efficiency, particularly in the conduct of reliability test and evaluation (T&E) programs for materiel systems. The intent is to identify methods to streamline T&E activities and promote innovation—not to compromise the adequacy of system reliability evaluations. However, it is imperative to recognize that there is, indeed, a tipping point at which a further reduction in T&E activities will preclude the capability to perform an adequate system reliability evaluation.

We define *adequacy* as the condition achieved when the level of evaluation risk associated with a given T&E program plan is acceptable to the decision authority. In recent years, we have observed a wide range of ad hoc approaches for designing reliability test programs (RTPs). Many of these approaches rely heavily on previously established rules of thumb for which the underlying rationale is indefensible. As a consequence, it is difficult to assess the adequacy of such RTPs. Furthermore, it is equally challenging to ascertain the impact that budgetary cuts—resulting in a reduction in reliability T&E activities—will have on the adequacy of the RTP.

In this chapter, we present a novel 7-step planning process for designing feasible RTPs that are both efficient and adequate. This planning process serves as a tool to systematically identify, assess, and mitigate evaluation risk subject to resource constraints. By performing the 7 steps associated with this planning process, practitioners

will be able to logically justify RTP requirements and more effectively articulate the nature and significance of evaluation risks associated with a particular RTP design. Additionally, it is a straightforward process to assess the impact of a reduction in RTP resources (amount of funding, time available, quantity of test assets, etc.) and determine whether or not there is a need to reformulate the RTP.

6.2 Definitions

In order to properly orient the reader, we now provide definitions for 3 key terms related to the reliability test program (RTP) planning process described herein.

Evaluation Risk

A measure of the residual uncertainty associated with the characterization of a given system's expected behavior in its intended operational environment.

Essential Evaluation Risk Area

A designated area of concern assessed as having the potential to significantly impact the overall evaluation of system-level reliability.

Adequacy

The condition achieved when the evaluation risk associated with a given RTP plan is within the acceptable tolerance threshold of the decision authority.

6.3 The Spirit of the RTP Planning Process

As we discussed earlier, widespread variability currently exists with respect to the level of analytical rigor applied during the RTP design process. Concordantly, it has proven challenging for practitioners to assess RTP adequacy as well as defend RTP requirements. The spirit and intent of this planning process is to manage evaluation risk and support the following activities:

- document the rationale underpinning the RTP design,
- communicate the RTP plan to the decision authority,
- justify all elements of a given RTP design (number and type of events, quantity of test assets, duration of testing, test environment, etc.),
- assess overall RTP adequacy, and
- determine the impact of T&E resource reductions on the adequacy of an established RTP plan.

Section 6.4 describes the 7 steps of the proposed RTP planning process. This planning process is not intended to be unforgivingly rigid; it can be easily modified or extended. Furthermore, this process is suitable for military or commercial systems that will undergo a formal RTP.

6.4 The 7 RTP Planning Process Steps

Step 1: Clarify and refine the system reliability requirement(s)

Although this step may appear to be straightforward, one must ensure that the interpretation of each user-specified reliability requirement matches the user's intent. At this point in the planning process, it may be necessary to modify the system's reliability

requirement(s). Typically, the best approach to achieve this objective is to establish and maintain an ongoing dialogue with the user's representative throughout the planning process.

For an Army system, reliability requirements depend on 3 documents, namely, the Capability Development Document (CDD), the Operational Mode Summary/Mission Profile (OMS/MP), and the Failure Definitions and Scoring Criteria (FDSC). The CDD includes the threshold value for each reliability parameter of interest. The OMS/MP describes the user's expectations regarding the types of missions for which the system will be employed as well as the environment within which the system will operate. In order to properly evaluate a given system's capabilities in relation to the CDD requirements, the FDSC serves as a guide for the system evaluator to assess the impact of reliability-related incidents observed during testing. Taken together, the CDD, OMS/MP, and FDSC serve as the 3 pillars that define system reliability requirements. It is also worth noting that all 3 of these documents are subject to change throughout the system acquisition management process.

Step 2: Define the field configuration of the system

It is vital to clearly define the system configuration that the materiel developer intends to field. Without such information, it is impossible to design an RTP that will promote an accurate characterization of the actual system reliability in the field. However, it is necessary to first determine the level of fidelity required to sufficiently define the field configuration of the system. For the purpose of designing the system's RTP, it is unlikely that practitioners would need to go down to the component level.

There are various reasons that prevent practitioners from testing the exact system configuration that will be fielded. By defining the field configuration of the system, it is possible to identify any deviations associated with the system test configuration and assess the potential impact to the system reliability evaluation. RTP planning process step 4 concerns the establishment of RTP essential evaluation risk areas, and Table 2 includes an example related to the disparity between the test configuration and the field configuration of the system. The methodology utilized to evaluate system reliability should explicitly account for limitations relating to testing the field configuration.

Table 2. Examples of Essential Evaluation Risk Areas

Essential Evaluation Risk Area	Synopsis of Evaluation Concern
1. Probability of Committing a Type I Statistical Error	Given the current paradigm of a single system reliability demonstration event of finite length, there is the potential to conclude that the system does meet the user's reliability requirement, when in fact, the system does not satisfy the user's requirement.
2. Probability of Committing a Type II Statistical Error	Given the current paradigm of a single system reliability demonstration event of finite length, there is the potential to conclude that the system does not satisfy the user's reliability requirement, when in fact, the system does satisfy the user's requirement.
3. Disparity between the System Test Configuration and the Planned System Field Configuration	Significant disparities between the test configuration and the planned field configuration may lead to inaccurate conclusions regarding the behavior of the actual system in its intended operational environment. Potential issues include, but are not limited to: • Insufficient capability to characterize peculiarities of distinct system components and/or variants • Reliability behavior not representative of complete system • May not capture the true human-machine interface issues • May not surface system-to-system interaction and/or integration related issues
4. OMS/MP Coverage	If the full OMS/MP range is not addressed, this could potentially lead to an inaccurate evaluation of fleet reliability.
5. Test Exposure Opportunity to Surface Key Failure Modes (Anticipated Unreliability Drivers)	Characterization of failure modes and failure recurrence rates may not be representative of the behavior of the actual system in the field. Need to consider: • Is there a sufficient quantity of test assets? • To what extent will each test asset be exercised in developmental and operational testing (type and duration of events, testing per asset, etc.)?
6. Power to Detect Statistically Significant Differences	Will the RTP design provide the opportunity to detect statistically significant differences, such as: • System Reliability between Test Assets • System Reliability of System Configurations • System Reliability during Developmental and Operational Testing • System Reliability by Test Location
7. Accuracy of System Reliability Evaluation Methodology	Evaluation accuracy pertains to how well the results from the application of a particular evaluation approach will correlate with the actual behavior of the system in the field. Considerations should include: • What assumptions must be satisfied for the approach to be valid? • What data are required? • What are the limitations of the approach?

Step 3: Perform up-front analysis to identify anticipated system unreliability drivers

The intent of this step is not to postulate all fathomable failure modes; rather, the intent is to identify the unreliability drivers that can be expected to occur under operationally realistic conditions, given the field configuration of the system. It is prudent to work with the system developer and consult the OMS/MP along with relevant historical data on similar systems to aid in this activity. Practitioners may wish to consider specific failure modes, subsystems, missions, or other elements during this step. A key aspect of this step is to accurately assess the expected loads and stresses in the operational environment. The results from this step of the RTP planning process will serve as input for steps 4 and 5.

Step 4: Establish essential RTP evaluation risk areas

As indicated in the definition from Section 6.2, an essential evaluation risk area pertains to an important aspect of the system that must be addressed in the evaluation. The degree to which such an aspect must be addressed is a subject to be considered during RTP planning process steps 5, 6, and 7. Some examples of essential evaluation risk areas that may be used appear above in Table 2.

Essential evaluation risk areas are intended to serve as the building blocks of an RTP design. The associated risk measures simultaneously indicate the degree to which each issue is addressed in an RTP along with the residual uncertainty regarding each issue. To be useful, risk measures must be defined in such a manner as to enable discrimination between distinct RTP designs. If a particular risk measure is not sensitive

to any types of changes in a given RTP design, the measure (and likely the essential evaluation risk area) is either ill-defined or of no practical value.

Step 5: Design a feasible RTP that addresses all essential evaluation risk areas

This step in the planning process is likely to be the most time-consuming as it requires coordination with multiple groups, including the system developer and test center personnel. The objective of this step is to build a complete RTP plan that is designed to exercise the system in such a manner as to expose all deficiencies that would have an adverse impact on mission capability in the field. Considerations for this step in the process should include the type of events, the test environment, the duration of testing (hours/miles/cycles/trials), and the quantity of test assets needed. Clearly, the feasibility of a given RTP plan will depend on the resources available to execute the RTP. In some cases, it may be possible to secure additional resources, depending on the level of risk associated with a particular RTP plan (refer to planning process step 7).

Step 6: Assess the level of risk associated with each essential evaluation risk area

Once the RTP plan exists, one must revisit the essential evaluation risk areas developed in step 4 to determine the extent to which each risk area is addressed. The methods to assess the level of risk may range from rudimentary to complex, and it is conceivable that the methods may result in either a qualitative or quantitative assessment. The results from this step are used to support decision-making in step 7.

Step 7: If any of the evaluation risk levels are higher than desired, reformulate the RTP

Typically, the question as to whether or not a given RTP plan is adequate arises. Indeed, this is a complex issue to address; however, by going through the 6 previous RTP planning steps, the answer to the question is appreciably easier to defend. The essential evaluation risk areas established in step 4 provide a convenient mechanism for assessing RTP adequacy. Referring back to Section 6.2, adequacy is achieved if the level of risk associated with each essential evaluation risk areas is below the decision authority's tolerance threshold. Here, the term decision authority may refer to a high-ranking official, or individual stakeholders (all of whom must achieve consensus).

6.5 Example Application of the RTP Planning Process

For the purpose of demonstration, we will now go through the 7-step RTP planning process for a notional air defense system-of-systems. We intentionally limit the complexity of this example; however, this approach can be applied to more complex situations and/or other types of military and commercial systems.

Step 1: Determine the system reliability requirement(s)

- *System Description*: The Air Defense SoS Battery (ADSB) is an air and missile defense system that consists of a control stations, launcher platforms, and radars. A launcher platform consists of missiles, a common military vehicle, launch rails, launcher electronics, and communications components. The ADSB will provide a significant means to defeat aerial reconnaissance, surveillance, and target acquisition platforms beyond their effective observation employment ranges.

- *Reliability Requirement*: The probability that the ADSB will complete a 24-hour mission without a system abort shall be greater than or equal to 0.90.

- *Failure Definition and Scoring Criteria*: The ADSB essential functions are

    1. Network Operations
    2. Monitor Airspace and Track targets
    3. Receive and Transmit Data
    4. Process and Display data
    5. Engage Targets
    6. Move

The reliability failures associated with the ADSB fall into 3 major categories of severity—system abort, essential function failure, non-essential function failure. Below are the definitions of each failure category that appear in the FDSC.

1. *System Abort (SA)*: A system abort is a failure or malfunction causing unacceptable degradation or the complete loss of one or more essential functions. An SA generally precludes the ADSB from continuing its mission or starting a mission. In other words, an SA results in a non-mission capable state for the system. At the SoS-level, an SA occurs if fewer than 2 out of 3 control stations, 3 out of 5 radars, or 9 out of 12 launchers are mission capable.

2. *Essential Function Failure (EFF)*: An essential function failure is a failure or malfunction causing degradation, or complete loss of one or more essential functions. However, if the failure causes only partial loss or an acceptable degradation of one or more essential functions, under certain circumstances, the

system may continue to operate under degraded conditions, usually until the completion of the current mission or task. This is analogous to going from a "fully mission capable" state to a "mission capable" state. In order to restore full mission capability, repair will be performed prior to the initiation of subsequent missions or tasks.

3. *Non-Essential Function Failure (NEFF)*: A non-essential function failure is a failure or malfunction that results in a loss of non-essential functions. An NEFF usually requires maintenance to correct, but that maintenance can be deferred without negatively impacting mission capabilities.

- *ADSB Mission Roles*

1. Provide air and missile defense protection of critical assets (static defense)
   - The ADSB performs the air and missile defense protection of a critical assets at strategic locations (including geopolitical assets like population centers, industrial resources, and sea/air ports of debarkation).
2. Provide air and missile defense protection of maneuver force
   - The ADSB provides air and missile defense protection of the maneuver forces during movement for combat/offensive operations to include periods of non-movement such as transitions and reconstitutions.

The ADSB is designed to accomplish the 2 mission roles listed above by providing:

- protection against a variety of air and cruise missile threats
- a critical non-line-of-sight and beyond line-of-sight overmatch capability against most rapidly evolving and projected air threats
- a means to defeat aerial reconnaissance, surveillance, and target acquisition platforms beyond their effective observation employment ranges, and
- integration of surveillance, command and control, fire direction, fire distribution, and engagement capabilities using an open, distributed, and networked architecture.

- *ADSB Modes*

1. Operation (assumed to be 24 hours/day when the ADSB is deployed)

2. Initialization (performed on an as-needed basis, e.g., when crew changes occur after every 8 hours of continuous operation)

3. Preventive Maintenance Checks Services (performed daily while the ADSB is in operation)

4. Scheduled Maintenance (staggered to ensure that the ADSB operates continuously)

5. Movement (only performed during continuous operation to establish/re-establish radio connectivity, otherwise, this mode occurs in order to emplace the ADSB); during movement, the breakdown of terrain types is assumed to be as follows: 70% primary roads, 25% secondary roads, and 5% cross-country terrain.

- *ADSB's Operational Environment*

ADSB will be capable of operations in climatic conditions, terrain, and geographic locations in which supported forces may be deployed. These conditions may include areas with salt-laden air and spray normally encountered in littoral area, use during times of limited visibility (including rain, smoke or fog, or darkness), use in areas with high humidity, sand, and dust, and under full solar radiation or at night.

  - Temperature: The system will be stored, transported, maintained, and operated in various temperature environmental conditions. Kits and procedures are allowed below -25º F and above 120° F.
  - Winds
    - Ground-level operation: ADSB shall meet its performance requirements during exposure to wind and gusts up to 35 mph without electrical or physical damage affecting the lifecycle expectancy or operation.
    - Ground-level non-operation: ADSB shall meet its performance requirements after exposure to wind to 50 mph and gusts up to 65 mph without electrical or physical damage affecting the lifecycle expectancy or operation.

- *Threats*

1. Primary threats to be countered or targeted by ADSB consist of unmanned aerial vehicles (UAVs) and cruise missiles (CMs).

2. The secondary target set consists of manned aircraft, both fixed-wing and rotary-wing.

Step 2: Define the field configuration of the system

The field configuration of the ADSB is depicted in Figure 2. The ADSB field configuration consists of 3 control stations, 5 radars, and 12 launchers. In order for the ADSB to be considered operational, at least 2 out of 3 control stations, 3 out of 5 radars, and 9 out of 12 launchers must be operational. All systems of a particular type are intended to be identical, and are considered to be configured in parallel, active redundancy.

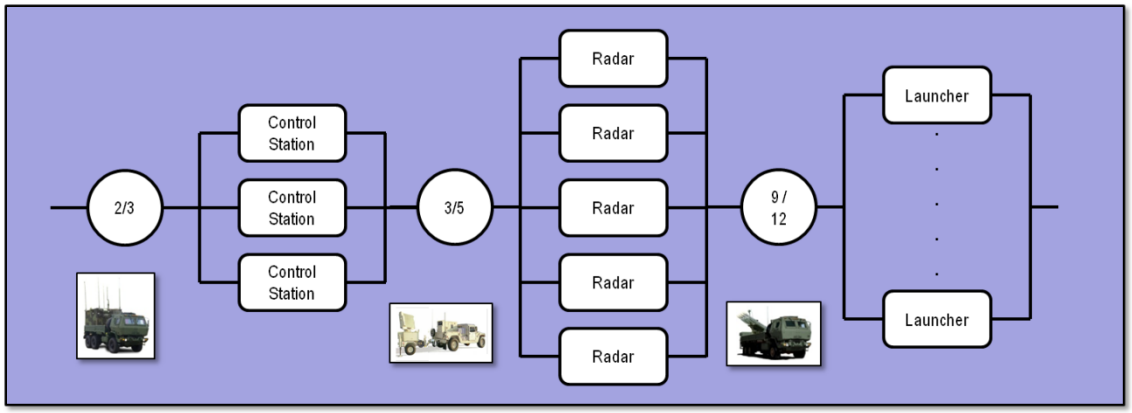


Figure 2. Reliability Block Diagram for the ADSB

Step 3: Perform up-front analysis to identify anticipated system unreliability drivers

Based on an examination of dominant failure modes on similar systems, the RTP planning team identified the following list of 11 failure modes as the anticipated unreliability drivers for ADSB. Although historical failure mode analysis was employed here, other approaches (as discussed in Chapter 3) could be applied as well.

1. Failures of the Target Tracking Software

2. Power System Failures
3. Loss of Data Communications Capability
4. Computer Boot-up Failures
5. Environmental Control Unit Failures
6. Computer System Crashes
7. Connector/Fastener Failures
8. Loss of Positional (GPS) Data
9. Radar Stops Radiating
10. Loss of Launcher Inclination/Rotation Capability
11. Suspension Failures on the Launcher Vehicle

Step 4: Establish essential RTP evaluation risk areas

For this example, we assume that the RTP planning team established the following 3 essential evaluation risk areas (EERAs). For a real RTP, planners would likely establish additional appropriate EERAs.

1. *Disparity between the System Test Configuration and the Planned System Field Configuration*

As discussed previously, there will almost always be a disparity between the operational test configuration and the field/production configuration of a given SoS. In this case, the risk is that evaluation of the ADSB's reliability based on the planned ADSB operational test configuration will not be sufficiently representative of the behavior of the ADSB field configuration. Note that this is EERA #3 from Table 2 in Section 6.4.

2. Operational Mode Summary/Mission Profile (OMS/MP) Coverage

The intent is to examine the extent to which the assumed OMS/MP for the ADSB is addressed during the RTP. For example, does the RTP include opportunities to exercise the ADSB under all anticipated environmental conditions? Will operational scenarios be executed with soldiers for all types of missions? Will all of the system modes be examined? Note that this is EERA #4 from Table 2 in Section 6.4.

3. Test Exposure Opportunity to Surface Key Failure Modes (Anticipated Unreliability Drivers)

The product of planning step 3 was a list of 11 anticipated unreliability drivers for the ADSB. Ideally, the RTP should include events/activities to verify or refute the list of anticipated unreliability drivers. Historical analysis indicates that certain failure modes may only surface under certain conditions, e.g., a particular mission in a particular environment. There is a natural connection between this EERA and the above EERA concerning the OMS/MP coverage associated with the RTP. However, the intent here is to focus on updating the list of actual unreliability drivers for the ADSB based on the ADSB RTP events, not historical data on similar systems. Practitioners need to hedge against the likely dominant failure modes, while also seeking out unanticipated, important failure modes. Note that this is EERA #5 from Table 2 in Section 6.4.

Step 5: Design a feasible RTP that addresses all essential evaluation risk areas

As the intent of this example is to demonstrate how the RTP planning process can be applied, we will not include all programmatic considerations here. Let us assume that we have 36 months to conduct all RTP activities up through the reliability demonstration event (referred to as the initial operational test [IOT]). The standard progression of events in an RTP is to begin with developmental testing, conduct a limited user test (LUT), build production-representative assets, perform reliability qualification testing of the production-representative assets, and execute the IOT. The IOT is the primary source of information for the formal evaluation of the system's reliability.

Given the above constraint of 36 months, we can design an RTP that offers the potential to address all of the essential evaluation risk areas established in step 4. In Table 3 below, we summarize the 5 test events that will constitute the RTP for the ADSB.

Table 3. ADSB RTP Events

RTP Event	Length (Hours)	Test Configuration
DT1	300	1-1-1
DT2	500	2-2-2
LUT	100	2-2-2
RQT	500	2-2-2
IOT	300	2-3-6

The "Length" column from Table 3 indicates the planned amount of time that the ADSB test configuration will be exercised. The "Test Configuration" column in Table 3 indicates the quantity of each type of system that will make up the ADSB test configuration. For example, 1-1-1 denotes that there will be 1 control station, 1 radar, and 1 launcher available during the test event. We will briefly discuss the rationale associated

with each of the 5 test events in Table 3. Below, Figure 3 depicts the schedule of all test events and other supporting RTP activities.

1. Developmental Test Phase 1 (DT1)

   The purpose of DT1 is to provide the opportunity for the first full-up SoS-level test of the ADSB. For this event, only 1 of each type of system will be available. Testing for this event will be conducted at the US Army Aberdeen Test Center (ATC) in Maryland. No live missile flights will occur. Opportunities to detect, track, and target rotary-wing aircraft will be emphasized, but no movement of the ADSB (recall one of the modes is “movement”) will be done during DT1. Immediately following completion of this test event, all equipment will be transported from ATC to White Sands Missile Range (WSMR) in New Mexico. Hence, there will not be time for the developer to implement corrective actions for observed failure modes, prior to the execution of DT2.

2. Developmental Test Phase 2 (DT2)

   During DT2, the test location will be WSMR, thus enabling live missile flight tests and tactical movement of the ADSB. No change to the ADSB or its constituent systems will occur prior to the conclusion of DT2. The test configuration will be augmented from 1 of each system type to 2 of each system type (see Table 3). Hence more information is likely to be obtained for this design of the ADSB. Subsequent to DT2, a corrective action period (CAP) will be conducted. The program manager and system developer have agreed to budget 4 months for the CAP to implement design changes based on observed failure modes to date (see Figure 3).

3. Limited User Test (LUT)

   The LUT will consist of live missile firings conducted at WSMR with representative soldier operators manning the system in a benign environment. The LUT test configuration of ADSB identified in Table 3 reflects an end-to-end evaluation (sensor-to-shooter) of the ADSB. The LUT will be conducted in the fourth quarter of 2014 (see Figure 3). Testing will exercise the ADSB within the safety constraints of a WSMR live missile flight test environment. The system will be employed against approved surrogate threat targets. Soldier training will be conducted at WSMR prior to the LUT and will be limited to air and missile defense employment of the ADSB, which will be focused on target engagement due to time and available funds. Field operations will consist of two live missile firings against threat aircraft platforms, as defined earlier. Following the LUT, a second CAP will be conducted, lasting approximately 5 months. The efforts of the CAP will be applied to the production-representative assets generated for the RQT and IOT (see below).

4. Reliability Qualification Test (RQT)

   The RQT will be conducted at the SoS-level on the low rate initial production (LRIP) assets to confirm that the LRIP design meets the ADSB reliability requirement. As well, an assessment of the effectiveness of vendor corrective actions to mitigate observed failure modes will be performed. The operators will be engineers and professional test personnel, not soldiers.

5. Initial Operational Test (IOT)

The IOT will be the reliability demonstration event for the ADSB. Testing will include soldier operators and maintainers previously trained and tested during the LUT. The objective of the IOT is to assess the capabilities of the ADSB to provide the mission functions discussed earlier. The IOT will be conducted in 3 phases: Field Sustained Operations, Hardware-in-the-Loop (HWIL), and Missile Flight Test. The ADSB test configuration for the IOT is listed in Table 3.

- *Phase I Field Sustained Operations*: Phase I will present the ADSB system employed in support of air and missile defense operations in a field environment under simulated combat conditions. Testing will include the system defending against live aircraft (approved threat surrogates) while operating under the ADSB operational Mission Summary/Mission Profile (OMS/MP). Testing will be completed in both active and benign electronic counter-measures (ECM) environments.

- *Phase II HWIL*: Phase II will be a HWIL test designed to exercise ADSB system performance against approved simulated threat scenarios. Computer-based, virtual stimulators will present aircraft scenarios representing a common air picture from external communications links. If simulation-based tools available at the time of IOT cannot be accredited for this application, then an additional number of live aircraft passes will be required during IOT Phase I testing.

- *Phase III Missile Flight Tests*: Phase III will consist of 5 live missile flight tests/target engagements.

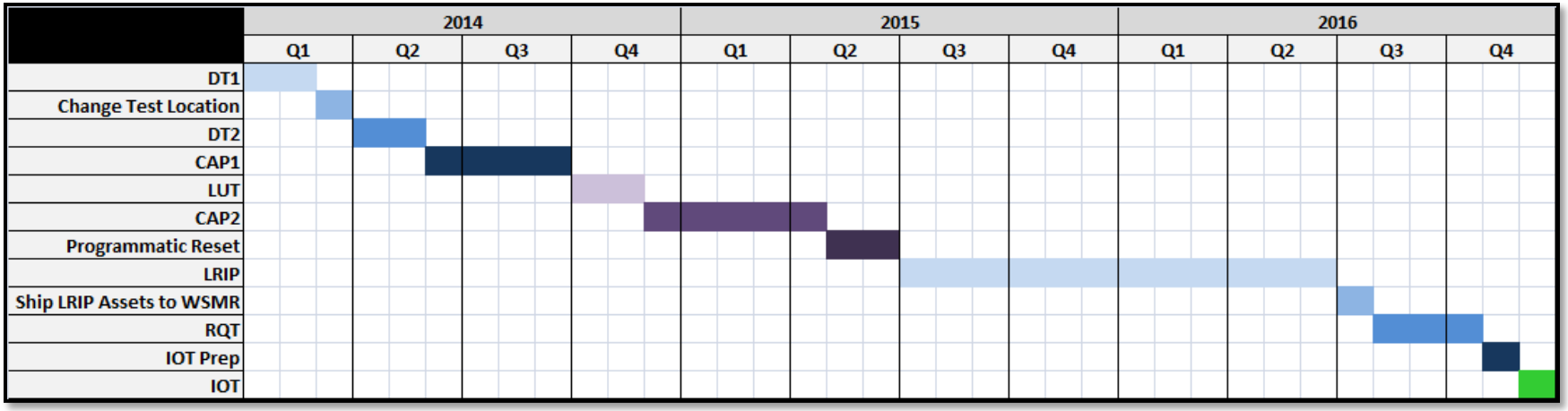


Figure 3. ADSB Reliability Test Program Schedule

Step 6: Assess the level of risk associated with each essential evaluation risk area

In step 4, we established 3 essential evaluation risk areas (EERAs) for this RTP. We will now comment on the risk assessment for each EERA.

1. *Disparity between the System Test Configuration and the Planned System Field Configuration*

Using a novel simulation-based approach discussed in Chapter 7, we consider the potential error in our estimate of ADSB reliability. Essentially, the accuracy in our evaluation of ADSB reliability is driven by (i) how close the test configuration is to the field configuration, and (ii) the length of the test event. Since DT1 is understood to be the first full-up SoS-level test event, it is likely that the SoS-level reliability will be fairly low. We assume that the true (but unknown) SoS-level reliability for ADSB at DT1is approximately 0.35.

Since no CAP is scheduled between DT1 and DT2, the true SoS-level reliability of ADSB does not change. Therefore, we continue to assume that the ADSB reliability is 0.35 for DT2. Between DT2 and the LUT, a 4-month CAP is scheduled, thus we expect the reliability of the ADSB to increase. In our example, we assume that the SoS-level reliability by the LUT is approximately 0.85. Again, a CAP is scheduled to occur following the LUT, and the findings will be leveraged in the design associated with the production-representative assets that will be available during the RQT and IOT. Just as we assumed improvement in SoS-level reliability as a result of the first CAP, we assume that the second CAP drives ADSB's reliability up to approximately 0.90. The "Relative Error" column in Table 4 is calculated as

$$Relative\ Absolute\ Error\ = \frac{|\ True\ SoS\ Reliability - Experimental\ SoS\ Reliability\ |}{True\ SoS\ Reliability}$$

where the true SoS reliability is the value listed in the "ADSB Reliability" column of Table 4, and the experimental SoS reliability is obtained via a stochastic simulation that emulates the outcome of testing based on the test configurations and test lengths listed in Table 4. The experimental SoS reliability values for each case are not shown in Table 4; however, an extensive treatment of this topic appears in Chapter 7. By inspection of Table 4, the important observation to make is that the anticipated relative absolute error associated with the ADSB reliability evaluation is decreasing as we progress through the RTP, with a reasonably low relative absolute error linked to the IOT—the most important event in the RTP when it comes to the evaluation of the ADSB's reliability.

Table 4. Anticipated Relative Error based on ADSB Test Configuration

RTP Event	Length (Hours)	Test Configuration	ADSB Reliability	Relative Error
DT1	300	1-1-1	0.35	0.4078
DT2	500	2-2-2	0.35	0.3054
LUT	100	2-2-2	0.85	0.1858
RQT	500	2-2-2	0.90	0.0606
IOT	300	2-3-6	0.90	0.0483

2. Operational Mode Summary/Mission Profile (OMS/MP) Coverage

Based on the plans for the 5 test events in Tables 3 and 4, there are opportunities to observe the reliability behavior of the various ADSB test configurations across all mission capabilities and operational modes defined earlier. DT1 is the least representative of the ADSB OMS/MP, however, the full spectrum of mission capabilities and modes are

exercised from DT2 through the IOT. All operational testing (LUT and IOT) is planned to be conducted in accordance with the ADSB OMS/MP. Hence, the OMS/MP coverage should be adequate, and the risk of failing to capture issues associated with particular mission capabilities or modes is assessed to be low for this RTP.

3. Test Exposure Opportunity to Surface Key Failure Modes (Anticipated Unreliability Drivers)

As discussed earlier, it is important to determine if the historical observations regarding unreliability drivers for similar systems will hold true for the ADSB. Given the list of 11 anticipated unreliability drivers generated during planning step 3, we want to assess whether or not we should expect to surface these failure modes—if they are "present" in the ADSB. One approach that can be employed is to examine the time until the first failure of its kind was observed in prior RTPs for similar systems. Further, it is worth noting whether or not the failure ever manifested itself during developmental testing, or only during operational testing (such as in a LUT or IOT). In Table 5, we include the time to first failure for each of the 11 anticipated unreliability drivers as well as whether or not the failure occurred during developmental and/or operational testing.

Table 5. Historical Information for Anticipated Unreliability Drivers

	Failure Mode	Time to First Failure (Hours)	Observed in DT	Observed in OT
1	Failures of the Target Tracking Software	40	x	x
2	Power System Failures	70	x	x
3	Loss of Data Communications Capability	0	x	x
4	Computer Boot-up Failures	80	x	x
5	Environmental Control Unit Failures	160		x
6	Computer System Crashes	240		x
7	Connector/Fastener Failures	80		x
8	Loss of Positional (GPS) Data	50	x	x
9	Radar Stops Radiating	240		x
10	Loss of Launcher Inclination/Rotation Capability	60	x	x
11	Suspension Failures on the Launcher Vehicle	300		x

In Table 5, note that all 11 failure modes were previously observed in operational testing (OT); however, 5 out of 11 failure modes were not previously observed during developmental testing (DT). Of the 5 failure modes that were only observed in OT, the time to first failure for 4 out of the 5 failure modes was 160 hours or greater. Yet, the time to first failure for each failure mode in Table 5 is based on test time accumulated during DT and OT. It is valuable to know that the amount of time during OT associated with the observation of the first failure of each type ranged from 20 to 30 hours. Hence, although the failure modes were not surfaced in DT, it did not take very long to surface the failure modes during OT.

Based on the favorable assessment of the planned OMS/MP coverage across DT2, the LUT, and the IOT, and taking into account that all of these RTP events are planned to run longer than 100 hours, we conclude that there is low risk associated with not sufficiently exercising the ADSB to verify, refute, or augment the list of anticipated unreliability drivers. Consequently, we expect that the characterization of ADSB's

unreliability drivers based on the planned RTP will be representative of the dominant failure modes in the field.

Step 7: If any of the evaluation risk levels are higher than desired, reformulate the RTP

In this example, the risks associated with each of the 3 established EERAs are considered to be acceptable. However, should assumptions regarding the conduct of any of the test events change—such as a reduction in resources that precludes the execution of certain events, or activities within events—it will be necessary to re-evaluate the level of risk associated with each EERA as well as the overall assessment of RTP adequacy.

6.6 Reducing Evaluation Risk by Using the RTP Planning Process

In the previous section, we performed each of the 7 RTP planning process steps for a notional system to briefly demonstrate how the complete process can be implemented. In this section, we present 2 examples to illustrate how the proposed RTP planning process defined in Section 6.4 can be used to create RTPs with lower risk than existing RTPs without increasing RTP costs. For each of the examples in this section, we discuss how the decisions made regarding the existing RTPs resulted in certain evaluation risks. By conducting the RTP planning process steps 1 through 3 from Section 6.4, such concerns would have been established as essential evaluation risk areas (EERAs) during planning step 4. Based on these EERAs, a different RTP design would have been formulated (during planning step 5) to consciously address each EERA, thus reducing the level of risk associated with each EERA (assessed during planning step 6). Consequently, the overall assessment of the adequacy of the RTP during planning step 7 is a more

favorable outcome. The following two examples (based on actual RTPs) discuss the differences that the proposed RTP planning process can make.

Example 1: Tactical Wheeled Vehicle System

This system is 1 of the 10 that we examined using the RTP evaluation risk assessment process presented in Chapter 4. This Major Defense Acquisition Program consists of 2 vehicle variants with unique characteristics (equipment, weight, geometries, etc.). This tactical wheeled vehicle system (including both variants) is designed to be used to perform 5 types of missions. The system is expected to operate in 3 types of environments—nominal, tropical, and cold. Finally, this system is intended to negotiate 3 types of terrain, specifically, primary roads, secondary roads, and cross-country/unimproved surfaces. Based on the fielding plan for the fleet of vehicles, it is anticipated that 80% of the fleet will operate in the nominal environment, 10% will operate in the tropical environment, and 10% will operate in the cold environment. It is intended that the system can employed to execute of all of the 5 types of missions in any of the 3 environments.

The existing RTP plan is to conduct 4 out of 5 types of missions during the operational test. The reason that 1 mission type that will not be performed during the operational test event is because that mission type involves conditions that are unsafe for human subjects. Each of the 4 planned types of missions will be executed 7 times (trials). Since the test event is planned to be held at a single location, the missions will be conducted only in the nominal environment. All 3 terrain types are present at the operational test location. The RTP for this system assumes that both variants of the

system have identical reliability characteristics. Based on this assumption, the existing RTP includes testing of 6 assets of only one system variant during the developmental portion of the RTP. During the course of developmental testing, the existing plan is to accumulate a total of 120,000 test miles. For the operational test event, the plan is to accumulate a total of 40,000 vehicle test miles.

Using the proposed RTP planning process generates a significantly different and better RTP. First, based on the information gathered during RTP planning steps 1 through 3, 3 EERAs can be identified:

1. The OMS/MP coverage during the operational test event (reliability demonstration event),
2. The likelihood of detecting a 25% variation in system-to-system reliability during the developmental test portion of the RTP, and
3. The likelihood of detecting a 25% difference in reliability between developmental testing and the operational test event.

The proposed RTP planning process mitigates these risks. The resulting RTP includes testing at 3 distinct locations that represent the 3 environments. For each of the 4 missions that can be tested, 3 trials will be conducted in the first environment, 2 trials in second environment, and 2 trials in third environment. Instead of assuming commonality, 3 of each system variant will be tested during developmental testing. The allocation of test miles will be 100,000 miles across developmental testing and 60,000 miles during the operational test.

Relative to the existing RTP, the new RTP increases the variety of the tests but does not increase costs. The same number of trials are planned, the same number of

vehicles are tested, and the total miles are the same. The total time required is the same. As the following paragraphs discuss, however, the risks associated with the new RTP are lower than the risk associated with the existing RTP.

1. OMS/MP coverage during the operational test event

The existing RTP had 80% coverage of mission types in the environment in which 80% of the fleet is expected to operate once fielded. Thus, the planned OMS/MP coverage during the operational test event is 64%. Because the new RTP, with the distributed operational test event, has tests that represent all 3 environments in which the fleet is expected to operate, the OMS/MP coverage is now 80%.

2. The likelihood of detecting a 25% variation in system-to-system reliability during the developmental test portion of the RTP

Because the existing RTP does not include testing of any assets of the second system variant, there is absolutely no chance of detecting a difference in the reliability between the 2 distinct variants. The new RTP tests both system variants, however. Section 4.3 outlined a method for approximating the likelihood of detecting a 25% difference in reliability between 2 distinct system variants. Using this method, we calculated the likelihood of detecting a 25% difference in reliability between the 2 vehicle variants to be approximately 45%, which represents a significant reduction in risk without increasing test costs.

3. The likelihood of detecting a 25% difference in reliability between developmental testing and the operational test event

For the existing RTP, using the method from Section 4.3, we calculated the likelihood of detecting a 25% difference in system reliability between developmental testing and the operational test event to be approximately 29.4%. For the new RTP, which allocates the test miles differently, this likelihood improves to approximately 32.9%. Theoretically, this improvement is possible; however, planners must ensure that this is feasible and that the change will not increase overall test execution costs for the program manager. The new RTP is also superior because the information acquired during the operational test is considered to be of the greatest value in the overall evaluation of system reliability.

Example 2: Air Defense System of Systems (SoS)

Similar to the notional Air Defense SoS Battery (ADSB) discussed in Section 6.5, the SoS for this example is comprised of 3 types of systems—control stations, radars, and launchers. In this case, the full field configuration of the system includes 4 control stations, 3 radars, and 12 launchers. In order for this SoS to be considered operational, a minimum of 2 out of 4 control stations, 2 out of 3 radars, and 9 out of 12 launchers must be operational.

There are 5 essential functions defined for this SoS, namely: track targets, communicate (receive/transmit data), display target track information, engage targets (i.e., launch missile(s) with the intention of destroying the target), and move. The last of the 5 essential functions listed is expected to be executed only when this SoS is being emplaced (for the purpose of performing the other 4 essential functions). Based on the

OMS/MP document for this SoS, movement is anticipated to account for 30% of the operational use of the SoS.

The existing RTP has the following elements: The operational test will exercise 2 control stations and 4 launchers during the course of 13 24-hour missions (i.e., 312 operating hours). Radar subsystems will be used to pass simulated target data to the control stations and launchers; however, no actual radars will be available during the operational test. (Note that this situation violates minimum requirement #2 for RTPs: test all elements of the system.) The operational test event does not include any activities that will exercise the movement function. The RTP assumes that all system-level assets of a particular type behave identically.

Using the proposed RTP planning process generates a significantly different and better RTP. First, based on the information gathered during RTP planning steps 1 through 3, 3 EERAs can be identified:

1. The OMS/MP coverage during the operational test event (reliability demonstration event),
2. The likelihood of detecting a 25% variation in system-to-system reliability during the operational test event, and
3. The evaluation adequacy of the SoS operational test configuration.

The proposed RTP planning process mitigates these risks. This RTP will include 2 control stations, 2 launchers, and 2 radars. It will maintain the planned operational test duration of 312 hours but allocate a portion of that time to movement of the SoS. Per the official OMS/MP for this SoS, it is anticipated that each movement will be for 30 miles, with 15 miles on primary roads (at a speed of 40 miles per hour), 13 miles on secondary

roads (at a speed of 30 miles per hour), and 2 miles cross-country (at a speed of 10 miles per hour); the total time will therefore be 1 hour. The RTP operational test plan will allocate 24 hours to conduct movement of the SoS, however, and the 24 hours will be distributed so that movement is conducted prior the start of a mission and/or after the completion of a mission cycle. The total planned SoS operating time to perform the other 4 essential SoS functions (i.e., exercise the control stations, radars, and launchers) will be 288 hours.

Relative to the existing RTP, the new RTP adds additional tests but does not increase costs because the costs to produce the radars and the launchers are equivalent. The total time required is the same. As the following paragraphs discuss, however, the risks associated with the new RTP are lower than the risk associated with the existing RTP.

1. OMS/MP coverage during the operational test event

Although the 4 functions that are planned to be exercised during the operational test constitute the bulk of the anticipated operational use, it is still imperative for the movement function to be characterized prior to fielding. After all, if the SoS cannot be emplaced at a fixed site because it cannot make the journey, the other 4 functions are irrelevant. Including 24 hours to conduct movement and conducting the moves before the start of a mission and after the completion of a mission cycle improves operational realism, improving the value of the information captured during the operational test event. Consequently, the operational test event in the improved RTP will provide full coverage of all 5 SoS essential functions within the context of mission-based scenarios.

2. The likelihood of detecting a 25% variation in system-to-system reliability during the operational test event

The accuracy in the estimate of SoS-level reliability is driven by the accuracy in the estimate of system-level reliability. For this RTP, the approach to estimating system-level reliability hinges upon the assumption that all system-level assets of a particular type behave identically. However, if this assumption is not valid for 1 or more types of the systems that constitute the SoS, the estimate of system-level reliability will be less accurate. Since the accuracy of the SoS-level reliability is sensitive to the accuracy of the system-level estimates, it is important to have the capability to detect variation in system-to-system reliability. The knowledge that variation exists in the reliability among systems of a particular type can be used to inform sensitivity analyses in support of the overall evaluation of SoS-level reliability.

Since there are 3 distinct types of systems that constitute the SoS in this example, we will assess the likelihood of detecting a 25% (hereafter referred to as L25) difference in system reliability for each of the 3 types of systems using the method presented in Section 4.3. Since no radars will be available during the operational test, there is no chance of detecting system-to-system variation in reliability among the radars. The L25 difference in the reliability between the 2 control stations is approximately 26%. For the 4 launchers, the L25 difference is approximately 24%.

For the new RTP, which includes control stations, launchers, and radars, the resulting L25 differences for the launchers and the radars are 22% and 21%, respectively. The L25 difference for the control stations remains to be 26% since there is no change to the number of control stations from the original plan. Although the L25 difference for the

launchers decreased from 24% to 22%, improving the L25 difference for the radars from 0% to 21% is a considerable return on the investment.

3. The evaluation adequacy of the SoS operational test configuration

As discussed in Section 2.5—from the standpoint of evaluation adequacy—an operational test configuration that does not include all unique elements of the system to be evaluated is clearly deficient. Because the new RTP will exercise all elements of the SoS during the operational test, we can now assess the level of adequacy associated with the improved SoS operational test configuration including 2 of each type of system. Hereafter, we will use the shorthand notation 2-2-2 to represent the SoS operational test configuration.

In Section 6.5, we briefly referred to a general method for assessing the adequacy of a given SoS operational test configuration. The details underpinning this method are presented in Chapter 7. In essence, the method from Chapter 7 estimates the error between the actual (but unknown) SoS-level reliability and the SoS-level reliability estimate based on operational test data. Here, we apply this method for the revised operational test configuration. Given the preceding information, the estimated relative absolute error associated with exercising the 2-2-2 configuration during the operational test for 288 hours is 13%. When compared to the planned, inadequate operational test configuration of the SoS, this anticipated, approximate result would be considered superior by the decision authority with oversight on the RTP for this SoS.

6.7 Concluding Remarks

In this chapter, we presented a structured process for designing RTPs with an emphasis on evaluation adequacy. The process described herein can be modified and applied to other areas (such as system performance test program planning). In practice, it may be the case that resource constraints preclude the execution of an adequate RTP. However, by identifying, quantifying, and communicating the potential risks associated with the system reliability evaluation to all stakeholders, it may be possible to motivate the allocation of additional resources to conduct the RTP. On the other hand, there may be cases in which an established, adequate RTP plan becomes infeasible due to a reduction in available resources. It is important for all stakeholders to realize that an RTP can be deemed as "efficient" only if the RTP is—first and foremost—adequate because efficiency implies that all evaluation requirements are satisfied using the minimum amount of resources. This RTP planning process systematically establishes the essential elements for a given RTP along with the maximum acceptable levels of risk associated with each element. To ensure adequacy, there can be no reduction in the scale or scope of the RTP that would increase risks above acceptable levels.

7 A SIMULATION-BASED RISK ASSESSMENT METHODOLOGY TO DETERMINE THE EVALUATION ADEQUACY OF SYSTEM-OF-SYSTEMS OPERATIONAL TEST CONFIGURATIONS

7.1 Background

Per the Defense Acquisition Guidebook [46], a system-of-systems (SoS) is defined as "a set or arrangement of systems that results from independent systems integrated into a larger system that delivers unique capabilities." In step 4 of the reliability test program (RTP) planning process presented Chapter 6, we introduced the concept of an essential evaluation risk area (EERA). One of the EERAs from Table 2 in Section 6.4 concerns the disparity that may exist between the field/production configuration of an SoS and the operational test configuration of the SoS. Given practical RTP resource constraints, it is rare that the full field configuration of an SoS can be exercised during an operational test event. However, as we consider various potential operational test configurations for a given SoS during the RTP planning process, it is critical to acknowledge that there is a point at which a further reduction in the scale and scope of the test configuration will yield results that are not representative of how the field configuration will behave.

The intent of step 6 of the RTP planning process is to assess of the level of risk associated with each EERA. Hence, it is during step 6 that an assessment of the level of risk associated with the disparity between the field configuration and planned test configuration of the SoS must be performed. If the level of risk is deemed to be unacceptable, this indicates that the planned SoS test configuration is inadequate and an alternate (adequate) test configuration must be selected. In other words, the planned SoS

test configuration is not sufficiently representative of the SoS field configuration. Since the purpose of the RTP is to enable the evaluation of the field configuration of the SoS, an attempt must be made to characterize and control the error in the estimate of SoS reliability.

7.2 Problem Statement and Proposed Solution

Given that an EERA concerning the adequacy of the SoS operational test configuration has been established during step 4 of the RTP planning process, reliability evaluators require a quantitative method that can be employed during RTP planning process step 6 to assess the adequacy of the planned SoS operational test configuration. Conceptually, adequacy—in this context—is achieved if the information obtained by exercising a given SoS configuration during the operational test (also known as the SoS reliability demonstration event) can be expected to enable a representative evaluation of the reliability of the actual field configuration of the SoS.

As discussed previously, in a real-world RTP, te SoS operational test configuration will most likely be some subset of the full SoS field configuration. During the course of the operational test event, reliability failures are recorded in a database. Each failure is scrutinized—the severity of the failure is assessed and the failure is charged to a particular component or subsystem. In the case of an SoS, the next lower level of indenture beneath the SoS is the system-level. Thus, every failure is mapped to a particular system within the SoS operational test configuration.

Taking into account (1) the quantity of each type of system employed as a part of the SoS operational test configuration, (2) the number of failures charged to each type of

system, and (3) the cumulative operating time across systems of a particular type, the reliability of each system type is estimated. In most cases, an exponential model is determined to reasonably describe the underlying distribution of the inter-arrival times between system-level failures. Having established a validated SoS-level reliability model—usually in the form of a reliability block diagram (RBD)—prior to the operational test, the system data is used in the SoS-level model to evaluate the SoS reliability. This is accomplished through the execution of a stochastic time-to-failure (TTF) simulation for the full field configuration of the SoS.

The stochastic SoS simulation yields an empirical distribution of SoS TTF. Subsequently, goodness of fit with respect to various parametric statistical distributions is explored. Once an appropriate distribution is selected, the reliability of the full field configuration of the SoS is estimated. The overall evaluation of the operational suitability of the SoS for fielding is largely influenced by the results from this process.

To assess the evaluation adequacy of a given SoS operational test configuration, the solution that we propose in this chapter is a simulation-based method that can be applied to any military or commercial SoS. It is assumed that steps 1 through 5 of the RTP planning process described in Section 6.4 have already been completed, and this method for assessing the adequacy of a planned SoS operational test configuration is performed in partial fulfillment of step 6 (since this activity assesses the risk of only 1 of the EERAs established during RTP planning process step 4). Below, we list the steps in the method. In Section 7.3, we will present an example to illustrate how this general method can be applied. This method enables direct comparisons between specified SoS test configurations, resulting in a convenient mechanism for ranking alternatives.

GENERAL ASSUMPTIONS:

1. Individual systems are assumed to fail independently (all system states are independent).
2. Failures at the system-level occur according to a homogeneous Poisson process. Hence, the time between failures for each individual system in the SoS is assumed to follow an exponential distribution.
3. Individual systems within each k-out-of-n (KN) structure, i.e., of the same type, are identical and they share the same underlying constant Poisson failure rate.
4. Each individual system in the SoS is assumed to be "as good as new" at the start of each simulation run; concordantly, the SoS is assumed to be "as good as new" at the start of each simulation run.
5. For the SoS, if an individual system fails during operation, no repair is attempted (i.e., the SoS will operate continuously until an SoS-level abort occurs).

MODEL FORMULATION:

- Let L be the number of distinct types of systems in the SoS.
- For each system type i, $i = 1, \ldots, L$, let n_i be the number of systems of each type in the field configuration of the SoS, and let k_i be the number of such systems that must be operational for the SoS to be operational.
- Let $N = \sum_1^L n_i$ be the total number of systems (of all types) in the field configuration of the SoS.
- Let λ_i be the common Poisson failure rate for each system of type i.
- For a given time t, let $R_i(t)$ be the reliability of systems of type i at time t.

$$R_i(t) = e^{-\lambda_i t}.$$

- For a given time *t*, let $R_{KN_i}(t)$ be the reliability of the i^{th} KN structure in the SoS at time *t*.

$$R_{KN_i}(t) = P(X \geq k_i; n_i, R_i(t)) = \sum_{j=k_i}^{n_i} \binom{n_i}{j} [R_i(t)]^j [1 - R_i(t)]^{n_i - j}$$

- For a given time *t*, let $R_{SoS}(t)$ be the reliability of the SoS at time *t*.
- In the field configuration of the SoS,

$$R_{SoS}(t) = \prod_{i=1}^{L} R_{KN_i}(t).$$

- Let m_i represent the number of systems of type *i* that will be in the operational test configuration of the SoS.
- Let *T* represent the length of the operational event during which the SoS test configuration will be exercised.
- Let T_0 represent the time at which the SoS reliability will calculated.
- Let T_{ij} represent the actual operating time accumulated by the j^{th} system of type *i* during the operational test.
- Let *Q* represent the desired number of simulation trials.
- Let *P* represent the number of SoS times-to-failure drawn during each simulation trial.

ALGORITHM:

1. For i = 1, …, L, identify à priori values of the λ_i.
2. Calculate the à priori value for the SoS reliability $R_{SoS}(T_0)$.

For δ = 1, …, Q, perform steps 3 to 5 (this constitutes a trial):

3. Simulate an operational test of duration T, exercising the planned SoS operational test configuration to obtain system-level reliability data as follows:
    a. For i = 1, …, L, and j = 1, …, m_i, draw a random number of failures F_{ij} for each individual system from the Poisson distribution with mean $\lambda_i T_{ij}$.
    b. For i = 1, …, L, calculate point estimates of the mean time between failure (MTBF) for system type i as $\hat{\theta}_i = \frac{\sum_{j=1}^{m_i} T_{ij}}{\sum_{j=1}^{m_i} F_{ij}}$.
4. Build an empirical time-to-failure distribution for the full SoS field configuration that is characterized by the system-level behaviors obtained in step 3.b. For p = 1, …, P, perform steps a to c to generate P SoS times-to-failure.
    a. For i = 1, …, L, and j = 1, …, n_i, generate the time to failure TTF_{ij} for each individual system of the SoS field configuration by conducting a random draw from an exponential distribution with mean $\hat{\theta}_i$.
    b. For i = 1, …, L, sort the times-to-failure TTF_{ij} in descending order and let $STTF_i$ equal the k_i^{th} largest time. This represents the first time at which fewer than k_i systems of type i will be operational.
    c. Let $SoSTTF$ = min {$STTF_1$, $STTF_2$, …, $STTF_L$}. This represents the time at which the SoS fails. Store the SoS-level time-to-failure.

5. Estimate SoS-level reliability as follows:
    a. Based on the shape of the empirical distribution of SoS times-to-failure obtained in step 4, formulate a hypothesis regarding an appropriate parametric statistical distribution for the SoS times-to-failure.
    b. Calculate maximum likelihood estimates for the parameter(s) of the hypothesized distribution.
    c. Assess goodness-of-fit. If the hypothesized distribution does not provide a good fit, consider an alternative parametric distribution (as appropriate).
    d. Compute $R_{est}(T_0)$, the estimated SoS reliability (the probability that the SoS time-to-failure will exceed T_0) using the fitted parametric statistical distribution. If no parametric distribution provided an acceptable fit, use the empirical distribution obtained in step 4. Determine the relative absolute error for this trial $| R_{sos}(T_0) - R_{est}(T_0) | / R_{sos}(T_0)$.
6. Calculate the mean relative absolute error associated with this SoS test configuration across all Q trials.

In practice, various methods can be used to identify values for the system-level failure rates in step 1 of the algorithm; however, the values selected should be vetted by subject matter experts (e.g., system engineers, testers, evaluators) to ensure validity of the simulation results. Furthermore, as it is impossible to know à priori what the true SoS reliability will be during the operational test, it is recommended to perform the above algorithm using a range of values for the system-level failure rates, resulting in a range of values for SoS reliability. This will enable practitioners to determine, for specific

applications, if the overarching results of this method are sensitive to the actual reliability of the SoS.

7.3 Illustration of the Simulation-Based Method to Assess the Adequacy of an SoS Operational Test Configuration

7.3.1 Overview

The simulation-based approach presented in this chapter mirrors the above process by which reliability of the SoS field configuration is evaluated. In contrast to the real-world case, we obtain failure data for the SoS operational test configuration via simulation. Based on the data from the simulated operational test, system-level point estimates for mean time between failure (MTBF) are calculated. Just as in the real-world case, these system-level MTBF point estimates are used in an SoS TTF simulation to evaluate SoS reliability. In Section 7.3.2, we outline the various scenarios for which this approach is exercised.

For the purpose of illustration, we will revisit the notional air defense SoS from Section 6.5, but we will consider 3 design variations for this SoS (see Section 7.3.5). To assess the accuracy of our simulation of the full SoS field configuration, we obtain the true SoS reliability values using the RBD logic method. For each SoS design that we consider in this study, we establish the corresponding SoS RBD (for example, see Figure 4 below) and we calculate the SoS reliability for the specified mission duration of 24 hours. In Section 7.3.6, we will discuss our approach for deriving and allocating the true system-level failure rates to achieve various SoS-level reliability targets defined in Section 7.3.2.

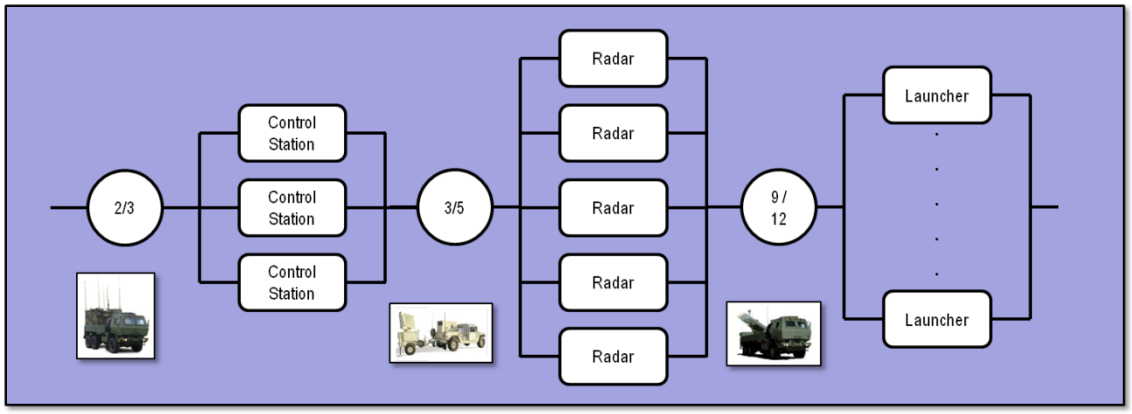


Figure 4. Reliability Block Diagram for SoS Design 3-5-12

7.3.2 Simulation Control Variables

- True SoS-level Reliability
  - 0.90, 0.85, and 0.35 (approximate values for each case—see Table 6 in Section 7.3.5 for actual values used)
- Number of *k*-out-of-*n* (KN) Structures within the SoS
  - 3 or 4
- k_i and n_i for each KN Structure
  - various choices for k_i and n_i
  - k_i ranges from 1 to 9, depending on n_i
  - n_i ranges from 2 to 12
- Operational Test Length
  - 100, 300, 500, or 1000 hours

7.3.3 Metrics

The primary metric that we consider for ranking each SoS test configuration is the relative absolute error between the true SoS reliability and the experimental SoS reliability, calculated as:

$$Relative\ Absolute\ Error\ = \frac{|\ True\ SoS\ Reliability - Experimental\ SoS\ Reliability\ |}{True\ SoS\ Reliability}$$

For all of the scenarios we consider in this chapter, we define SoS reliability as the probability that the SoS will complete a 24-hour mission without incurring a SoS-level abort (reliability failure). The SoS reliability requirement is assumed to be 90%, i.e., $P(T>24 \text{ hours}) \geq 0.90$. For Army air defense systems, the reliability requirement is almost always 90%, although, the specified mission duration may not be 24 hours (72 hours is another commonly specified mission duration).

For each simulation case, the secondary metric that we examine is the proportion of trials that result in a correct answer to the question "is $P(T>24 \text{ hours}) \geq 0.90$ for the SoS." This metric is simply intended as ancillary information since proximity of the estimate of SoS reliability—obtained via a test and evaluation program—to the true SoS reliability in the field is of the utmost importance. In practice, the operational test is not a pass or fail situation.

7.3.4 Simulation Assumptions

1. Individual systems are assumed to fail independently (all system states are independent).

2. Failures at the system-level occur according to a homogeneous Poisson process. Hence, the time between failure for each individual system in the SoS is assumed to follow an exponential distribution.
3. Individual systems within each *k*-out-of-*n* (KN) structure, i.e., of the same type, are identical and they share the same underlying constant failure rate.
4. Each individual system in the SoS is assumed to be "as good as new" at the start of each simulation run; concordantly, the SoS is assumed to be "as good as new" at the start of each simulation run.
5. The availability of a given test asset (system) during each simulated operational test/run is based on a random draw from U(0.6,0.9). This is based on the demonstrated availability of similar systems during actual government operational testing.
6. For the SoS-level simulation (see Section 7.3.6), if an individual system fails during the mission, no repair is attempted (i.e., the SoS will operate continuously until an SoS-level abort occurs). This aligns with the manner in which SoS-level reliability requirements are written by the US Army.

7.3.5 Simulation Run Matrices

In this section, we provide the various run matrices used for our simulation cases. Table 6 includes—for each SoS design—the values for the true SoS reliability along with the true system-level failure rates (λ_i).

Table 6. SoS Reliability by Design

SoS Design	SoS-Level Reliability	λ_1	λ_2	λ_3	λ_4
2-3-4	0.9044	0.0085	0.0048	0.0099	-
	0.8540	0.0104	0.0062	0.0121	-
	0.3590	0.0322	0.0183	0.0276	-
3-5-12	0.9039	0.0048	0.0073	0.0048	-
	0.8556	0.0062	0.0089	0.0053	-
	0.3551	0.0183	0.0203	0.0110	-
3-5-4-12	0.9045	0.0041	0.0066	0.0089	0.0044
	0.8533	0.0052	0.0078	0.0108	0.0050
	0.3596	0.0153	0.0176	0.0243	0.0096

For convenience, we have decomposed the primary simulation run matrix into Tables 7 through 9, which depict the 12 test configuration cases for each pairing of SoS design and SoS-level reliability from Table 6. Note that there are 3 SoS designs, 3 SoS-level reliability values, and 12 test configuration cases, for a total of 108 combinations. We conduct 20 trials for each of the 108 combinations, resulting in 2160 total simulation trials. Table 10 includes the k_i and n_i for each SoS design.

Table 7. Test Configuration Run Matrix for SoS 2-3-4

Test Configuration Case	Test Configuration	Operational Test Length
1	1-1-1	100
2	1-2-2	100
3	2-2-3	100
4	1-1-1	300
5	1-2-2	300
6	2-2-3	300
7	1-1-1	500
8	1-2-2	500
9	2-2-3	500
10	1-1-1	1000
11	1-2-2	1000
12	2-2-3	1000

Table 8. Test Configuration Run Matrix for SoS 3-5-12

Test Configuration Case	Test Configuration	Operational Test Length
1	1-1-1	100
2	2-2-2	100
3	2-3-6	100
4	1-1-1	300
5	2-2-2	300
6	2-3-6	300
7	1-1-1	500
8	2-2-2	500
9	2-3-6	500
10	1-1-1	1000
11	2-2-2	1000
12	2-3-6	1000

Table 9. Test Configuration Run Matrix for SoS 3-5-4-12

Test Configuration Case	Test Configuration	Operational Test Length
1	1-1-1-1	100
2	2-2-2-2	100
3	2-3-3-6	100
4	1-1-1-1	300
5	2-2-2-2	300
6	2-3-3-6	300
7	1-1-1-1	500
8	2-2-2-2	500
9	2-3-3-6	500
10	1-1-1-1	1000
11	2-2-2-2	1000
12	2-3-3-6	1000

Table 10. SoS KN Matrix

SoS Design	K1	N1	K2	N2	K3	N3	K4	N4
2-3-4	1	2	2	3	2	4	0	0
3-5-12	2	3	3	5	9	12	0	0
3-5-4-12	2	3	3	5	2	4	9	12

7.3.6 Simulation Activities

For this study, we conducted several activities via simulation using MATLAB (R2009b). As well, we performed additional activities using Mathematica (release 8) and MATLAB outside of the simulation in order to facilitate the overarching simulation-based activities. In this section, we will describe all of these activities. All of the source code is contained in the Appendix (Sections A.1 and A.2).

Establishing System-Level Failure Rates (Pre-Simulation Activity)

As we discussed in Section 7.3.2, we considered 3 levels of SoS reliability in our study—0.90, 0.85, and 0.35. In order to obtain the system-level failure rates (λ_i) in Table 6 that support the 3 levels of SoS reliability, we created a short routine in MATLAB. Given that each SoS design we consider is a series of either 3 or 4 KN structures, we calculate the n^{th} root of the desired reliability for the SoS (where n = {3,4}, depending on the number of KN structures in the SoS design). This yields the KN-level reliability for the given SoS design. For example, if the desired SoS-level reliability is 0.90 and there are 3 KN structures in the SoS, the KN-level reliability target would be $0.9^{1/3} = 0.9655$. Given values for k, n, λ, and t, we calculate the reliability of the KN structure, $R_{KN}(t)$ using RBD logic as follows:

$$R_{KN}(t) = P\left(X \geq k; n, e^{-\lambda t}\right) = \sum_{i=k}^{n} \binom{n}{i} [e^{-\lambda t}]^i [1 - e^{-\lambda t}]^{n-i}$$

The rationale governing the expression for $R_{KN}(t)$ above is that we have n identical systems configured in active parallel redundancy, and we need at least k of the systems to be operational for the associated KN structure to be operational. As stated in Section 7.3.4, the underlying distribution of the inter-arrival times between failures is assumed to

be exponential, hence the appearance of the $e^{-\lambda t}$ term representing the reliability of an individual system at time *t*.

In order to obtain values of the failure rates (λ_i) for the individual (and identical) systems within each KN structure, we use the KN-level reliability value as our target (say, 0.9655). We set the system-level failure rates equal to 1 as a starting point, and we calculate the reliability of the KN structure using RBD logic as discussed. If the calculated value of reliability for the KN structure is less than the target, we decrement the system-level failure rate (by a user-defined fraction) until the calculated KN reliability value is greater than or equal to the KN reliability target. In Table 11, we include an example progression of various choices for the system-level failure rate.

Table 11. Example Search Progression for the System-Level Failure Rate of a System in a KN Structure (*k*=2, *n*=3, *t*=24, target $R_{KN}(t)$=0.9655)

Iteration	λ_i	$R_{KN}(t)$
1	0.0100	0.8828
2	0.0095	0.8923
3	0.0090	0.9010
4	0.0086	0.9091
5	0.0081	0.9166
6	0.0077	0.9236
7	0.0074	0.9300
8	0.0070	0.9359
9	0.0066	0.9414
10	0.0063	0.9464
11	0.0060	0.9510
12	0.0057	0.9553
13	0.0054	0.9592
14	0.0051	0.9628
15	0.0049	0.9661

To reduce the number of iterations to display in Table 11, we used a start value for λ of 0.01 instead of 1. Note that in Table 11, for the i^{th} iteration ($i \geq 2$) $\lambda_i = 0.95\lambda_{i-1}$ because we specified a decrement value of 0.95. For the 15th iteration of the MATLAB routine, the final value of λ_i is 0.0049 and $R_{KN}(t)$ is 0.9661 which is close to our target of 0.9655. By

specifying decrement values for λ closer to 1, it is possible to obtain a value of λ that yields a value of $R_{KN}(t)$ that is as arbitrarily close to the target—0.9655, in this case. In turn, this will ensure that the resulting SoS-level reliability is as close to our desired value as we wish. Given each SoS is RBD is a series of m KN structures, $R_{SoS}(t)$ is computed as

$$R_{SoS}(t) = \prod_{j=1}^{m} R_{KN_j}(t)$$

Execution of Simulation Cases

In accordance with our discussion in section 7.3.5, we will use the terms "test configuration case" (or, simply case) and "trial" for our simulation. Figure 5 depicts the hierarchy of terms in our simulation construct.

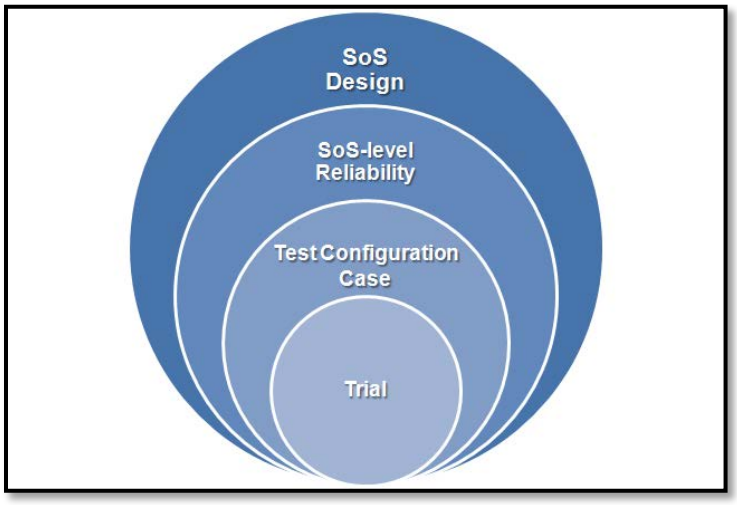


Figure 5. SoS Simulation Hierarchical Structure

As suggested by Figure 5, each trial requires the specification of the SoS design, SoS-level reliability, and the test configuration. In Table 12, we highlight the major simulation processes that are executed as part of each simulation trial. The "Input" column in Table

12 depicts the information required for the associated simulation process, and the product(s) of the simulation process are noted in the "Output" column.

Table 12. Summary of Simulation Processes for each Trial

Simulation Process	Input	Output
1. Simulation of an operational test event based on the given system-of-systems (SoS) test configuration	1. Number of system types 2. Number of systems of each type (i.e., number of test assets) 3. Vector consisting of the true system-level failure rates	Point estimates of mean time between failure (MTBF) for each type of system
2. SoS Time to Failure (TTF) Simulation	1. SoS design a. Number of system types (KN structures) b. k_i c. n_i 2. Vector consisting of system-level MTBF point estimates 3. Desired number of replications	Vector with 1000 SoS-level times to failure
3. Gamma distribution parameter estimation given empirical TTF distribution	TTF vector generated during the SoS-level simulation (process 2 above)	MLEs for gamma TTF distribution parameters α and β
4. Calculation of Key Metrics a. Experimental SoS-level Reliability b. Relative error between true SoS-level reliability and experimental SoS-level reliability	1. MLEs for gamma TTF distribution parameters α and β 2. True SoS-level reliability, i.e. P(T>24)	1. Experimental SoS-level Reliability, i.e., P(T>24) 2. Relative error between true SoS-level reliability and experimental SoS-level reliability

Simulation Process 1

The first major step in the simulation flow is to generate system-level "operational test data," i.e., MTBF point estimates for each type of system within the SoS. During actual operational testing, we exercise systems in accordance with how they are expected to be employed in the field, and we obtain system-level point estimates for MTBF. This process is designed to represent how we estimate system-level reliability in practice. Just as in our simulation, in real SoS reliability test and evaluation programs, we must use system-level reliability data to evaluate the reliability of the SoS field configuration.

Simulation Process 2

The second major step in the simulation flow is to take the system-level point estimates for MTBF, and use them in our SoS-level TTF simulation module. Given the SoS design and the vector of system-level MTBF point estimates, we simulate running the SoS until failure. We repeat this 1000 times in order to build an empirical distribution of 1000 SoS TTF.

Simulation Process 3

Having obtained the empirical SoS TTF distribution, we calculate the maximum likelihood estimates (MLEs) for the two-parameter gamma distribution, namely α and β. Based on a visual inspection of empirical TTF distributions for various SoS designs and levels of SoS reliability, we postulated that the two-parameter gamma distribution would serve as an appropriate distribution to represent the SoS TTF distribution. As well, we generated probability plots such as in Figure 7 and calculated Cramér-von Mises statistics to investigate goodness of fit. In all cases, there was extremely close agreement between the expected and observed values. Hence, we automated the process to calculate MLEs for α and β for all trials. We performed this automated process in Mathematica; although, it is quite easily done in MATLAB as well using the built-in `gamfit` function.

Simulation Process 4

The final major step to complete for each simulation trial is to calculate our 2 key metrics, namely (1) the experimental SoS-level reliability (i.e., the probability that the

SoS TTF will exceed 24 hours) based on the fitted gamma distribution from simulation process 3 and (2) the relative error in P(T>24) between the experimental and true values. Thus, a simulation trial consists of these 4 major processes. In this overarching study, we performed 20 such trials for each test configuration case. For each batch of 20 trials, we calculated the proportion of trials for which the answer to "Is P(T>24 hours) ≥ 0.90 for the SoS?" is correct. Lastly, we calculated the (arithmetic) mean relative absolute error across all 20 trials (results appear in Tables 14 through 22 in Section 7.3.7).

7.3.7 Simulation Results

Approximate Distribution for SoS-Level Time to Failure (TTF)

In section 7.3.6, we briefly indicated that the SoS-level simulation includes fitting a 2-parameter gamma distribution to each empirical SoS TTF distribution. Prior to conducting this study, our initial hypothesis was that the SoS TTF distribution would follow, approximately, a two-parameter gamma distribution. We expected this to be a reasonable starting point, conceptually, as the gamma distribution represents (in the physical world) a system that can handle n "shocks" prior to a catastrophic failure. Further discussion on this physical connection appears in [41] and [44] . By analyzing the data from the SoS-level simulation cases, we find that the SoS TTF distributions are well-characterized by the two-parameter gamma distribution. Figure 6 is a plot of the empirical TTF distribution for the SoS design 3-5-12 with true reliability of 0.35. The fitted gamma(3.851,5.567) probability density function is overlaid on top of the relative frequency histogram of SoS TTF.

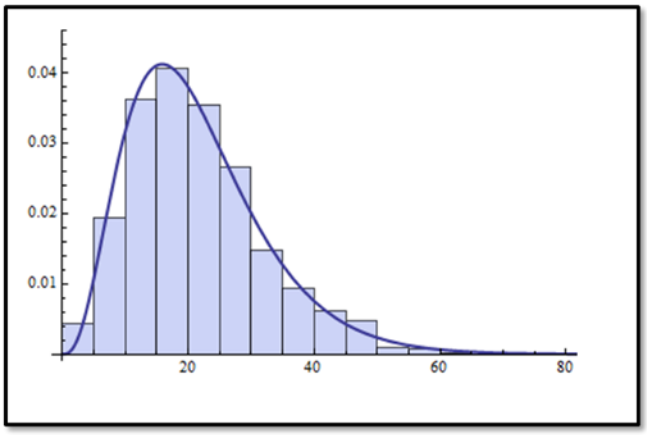


Figure 6. Empirical TTF Distribution for SoS Design 3-5-12 with Fitted Gamma Distribution Overlaid

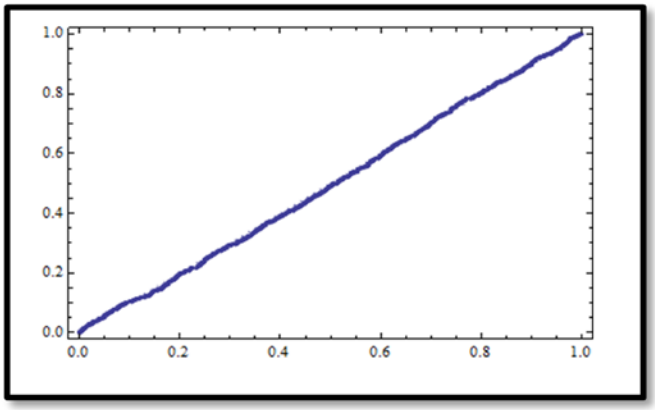


Figure 7. Probability Plot for Empirical and Fitted TTF Distributions for SoS Design 3-5-12

By inspection of the probability plot in Figure 7 above, there is extremely good agreement (in the goodness-of-fit sense) between the data and the theoretical distribution, i.e., gamma(3.851, 5.567). We also determined that the results for the SoS TTF are stable by 1000 replications. In Figure 8, this is visually confirmed, without loss of generality, for SoS design 3-5-12 by noting that the there is a nearly imperceptible difference between the fitted distributions as we increase the sample from 500 replications (light brown curve) to 1000 replications (green curve).

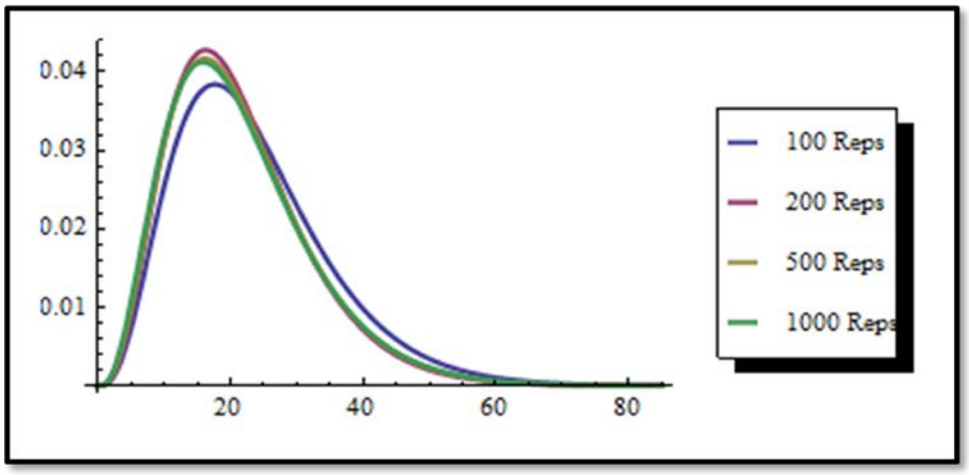


Figure 8. Plot of Fitted TTF Distribution for SoS Design 3-5-12 based on 100, 200, 500, and 1000 SoS TTF Simulation Replications

Mean Relative Absolute Error

Although the Army specifies its system reliability requirements a single threshold value that must be demonstrated during an operational test event, historically, the majority of Army systems fail to achieve this target. In practice, if a given system's demonstrated reliability falls short of its threshold requirement, the Army may still decide to field the system. The decision to field a system that does not demonstrate its operational reliability requirement is based on the assessed marginal impact of lower reliability to mission capability. Thus, it is critical for the Army's system reliability evaluators to obtain and present the most accurate system-level reliability estimate, subject to resource constraints. As we mentioned earlier, the notion of the anticipated relative error associated with a given test configuration is useful from an RTP planning standpoint as a means to assess RTP evaluation adequacy.

In this study, we calculate the relative absolute error in SoS-level reliability, i.e., P(T > 24 hours) for each simulation trial. As there are 20 trials for each simulation case, we present the mean relative error across all 20 trials in Tables 14 through 22. We note

that—irrespective of the SoS design or level of true SoS reliability—the mean relative absolute error decreases as either the operational test duration increases or the test configuration approaches the full field configuration of the SoS. This is not a surprising result, but it is important that this simulation-based methodology yields such a result.

Table 13. SoS Simulation Results using Actual System Failure Rates

SoS Design	True SoS Reliability	α	β	Experimental SoS Reliability	Relative Error
2-3-4	0.9044	2.980	23.893	0.9167	0.014
	0.8540	2.578	22.332	0.8427	0.013
	0.3590	2.596	8.320	0.3521	0.019
3-5-12	0.9039	3.672	16.017	0.9048	0.001
	0.8556	3.582	13.709	0.8473	0.010
	0.3551	3.684	5.785	0.3425	0.035
3-5-4-12	0.9045	3.852	14.268	0.8942	0.011
	0.8533	4.064	11.700	0.8563	0.004
	0.3596	3.922	5.407	0.3381	0.060

Table 14. Simulation Results for SoS Design 2-3-4 with SoS-Level Reliability 0.90

TC Case	Test Configuration	Operational Test Length	Mean Relative Error in P(T>24)
1	1-1-1	100	0.2264
2	1-2-2	100	0.1779
3	2-2-3	100	0.0981
4	1-1-1	300	0.0883
5	1-2-2	300	0.0496
6	2-2-3	300	0.0539
7	1-1-1	500	0.0544
8	1-2-2	500	0.0536
9	2-2-3	500	0.0450
10	1-1-1	1000	0.0409
11	1-2-2	1000	0.0412
12	2-2-3	1000	0.0340

Table 15. Simulation Results for SoS Design 2-3-4 with SoS-Level Reliability 0.85

TC Case	Test Configuration	Operational Test Length	Mean Relative Error in P(T>24)
1	1-1-1	100	0.2005
2	1-2-2	100	0.1248
3	2-2-3	100	0.1240
4	1-1-1	300	0.1196
5	1-2-2	300	0.0744
6	2-2-3	300	0.0515
7	1-1-1	500	0.0703
8	1-2-2	500	0.0706
9	2-2-3	500	0.0463
10	1-1-1	1000	0.0619
11	1-2-2	1000	0.0471
12	2-2-3	1000	0.0409

Table 16. Simulation Results for SoS Design 2-3-4 with SoS-Level Reliability 0.35

TC Case	Test Configuration	Operational Test Length	Mean Relative Error in P(T>24)
1	1-1-1	100	0.4116
2	1-2-2	100	0.4851
3	2-2-3	100	0.3664
4	1-1-1	300	0.3058
5	1-2-2	300	0.2716
6	2-2-3	300	0.1552
7	1-1-1	500	0.2601
8	1-2-2	500	0.2039
9	2-2-3	500	0.1654
10	1-1-1	1000	0.1968
11	1-2-2	1000	0.1493
12	2-2-3	1000	0.1540

Table 17. Simulation Results for SoS Design 3-5-12 with SoS-Level Reliability 0.90

TC Case	Test Configuration	Operational Test Length	Mean Relative Error in P(T>24)
1	1-1-1	100	0.3982
2	2-2-2	100	0.2555
3	2-3-6	100	0.1540
4	1-1-1	300	0.1028
5	2-2-2	300	0.1029
6	2-3-6	300	0.0483
7	1-1-1	500	0.1141
8	2-2-2	500	0.0606
9	2-3-6	500	0.0564
10	1-1-1	1000	0.0644
11	2-2-2	1000	0.0638
12	2-3-6	1000	0.0290

Table 18. Simulation Results for SoS Design 3-5-12 with SoS-Level Reliability 0.85

TC Case	Test Configuration	Operational Test Length	Mean Relative Error in P(T>24)
1	1-1-1	100	0.3485
2	2-2-2	100	0.1858
3	2-3-6	100	0.1530
4	1-1-1	300	0.1802
5	2-2-2	300	0.1336
6	2-3-6	300	0.0974
7	1-1-1	500	0.1256
8	2-2-2	500	0.0827
9	2-3-6	500	0.0570
10	1-1-1	1000	0.0606
11	2-2-2	1000	0.0459
12	2-3-6	1000	0.0599

Table 19. Simulation Results for SoS Design 3-5-12 with SoS-Level Reliability 0.35

TC Case	Test Configuration	Operational Test Length	Mean Relative Error in P(T>24)
1	1-1-1	100	0.5569
2	2-2-2	100	0.5644
3	2-3-6	100	0.4358
4	1-1-1	300	0.4078
5	2-2-2	300	0.3335
6	2-3-6	300	0.2564
7	1-1-1	500	0.3236
8	2-2-2	500	0.3054
9	2-3-6	500	0.2147
10	1-1-1	1000	0.2555
11	2-2-2	1000	0.1876
12	2-3-6	1000	0.1609

Table 20. Simulation Results for SoS Design 3-5-4-12 with SoS-Level Reliability 0.90

TC Case	Test Configuration	Operational Test Length	Mean Relative Error in P(T>24)
1	1-1-1-1	100	0.4210
2	2-2-2-2	100	0.2318
3	2-3-3-6	100	0.1704
4	1-1-1-1	300	0.0930
5	2-2-2-2	300	0.1104
6	2-3-3-6	300	0.0557
7	1-1-1-1	500	0.0974
8	2-2-2-2	500	0.0534
9	2-3-3-6	500	0.0350
10	1-1-1-1	1000	0.0612
11	2-2-2-2	1000	0.0494
12	2-3-3-6	1000	0.0464

Table 21. Simulation Results for SoS Design 3-5-4-12 with SoS-Level Reliability 0.85

TC Case	Test Configuration	Operational Test Length	Mean Relative Error in P(T>24)
1	1-1-1-1	100	0.5012
2	2-2-2-2	100	0.1839
3	2-3-3-6	100	0.1437
4	1-1-1-1	300	0.1122
5	2-2-2-2	300	0.1720
6	2-3-3-6	300	0.0727
7	1-1-1-1	500	0.1119
8	2-2-2-2	500	0.0939
9	2-3-3-6	500	0.0628
10	1-1-1-1	1000	0.0938
11	2-2-2-2	1000	0.0726
12	2-3-3-6	1000	0.0356

Table 22. Simulation Results for SoS Design 3-5-4-12 with SoS-Level Reliability 0.35

TC Case	Test Configuration	Operational Test Length	Mean Relative Error in P(T>24)
1	1-1-1-1	100	0.5381
2	2-2-2-2	100	0.4851
3	2-3-3-6	100	0.4161
4	1-1-1-1	300	0.3381
5	2-2-2-2	300	0.2636
6	2-3-3-6	300	0.2301
7	1-1-1-1	500	0.2775
8	2-2-2-2	500	0.2697
9	2-3-3-6	500	0.2045
10	1-1-1-1	1000	0.2382
11	2-2-2-2	1000	0.1609
12	2-3-3-6	1000	0.1622

As we would expect from real-world testing, by exercising more distinct test assets for longer periods of time, the accuracy of our system-level reliability estimates should improve. In our simulation, the primary reason that the we observe mean relative absolute error decreasing as (1) the SoS test configuration approaches the SoS field configuration and/or (2) as the operational test length increases is because the aggregate accuracy our system-level MTBF point estimates (PE) is improving.

It is difficult to get a good sense of this phenomenon simply from examination of values for the minimum, maximum, median, or mean, system-level MTBF PE across 20 trials. However, this phenomenon is easier to observe by visual inspection of box and whisker plots for the system-level MTBF PE such as appear in Figures 9 through 12 for our SoS design 3-5-4-12 (90% reliability). Each figure consists of box and whisker plots for each of the 12 test configuration cases by system type (recall that there are 4 types of systems in our SoS design 3-5-4-12). In each figure, the black dashed line represents the true system-level MTBF for that type of system. Without loss of generality, the trends that we note are (1) that the median MTBF values approach the true value for each system type and (2) that the interquartile range decreases as the quantity of test assets increases and the operational test duration (i.e., operating hours per test asset) increases. In other words, more of the system-level MTBF PE are closer to the true values, which results in the observed reduction in mean relative absolute error at the SoS-level.

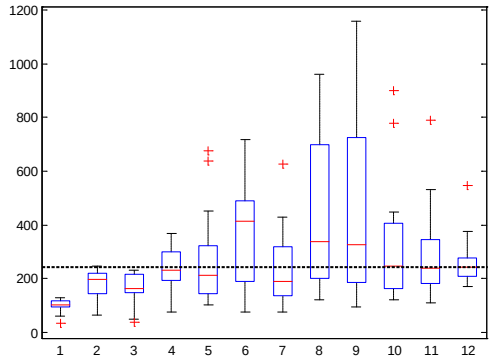


Figure 9. MTBF PE for SoS 3-5-4-12 (90% reliability) for System Type 1 Only

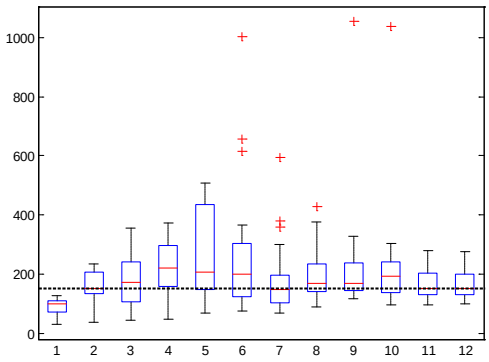


Figure 11. MTBF PE for SoS 3-5-4-12 (90% reliability) for System Type 2 Only

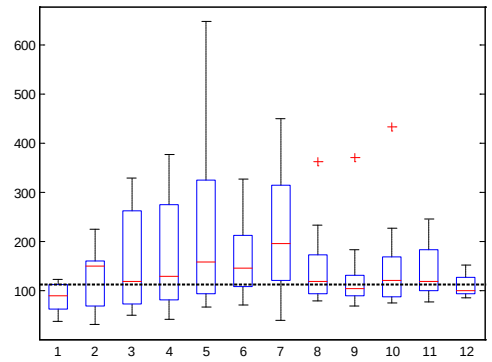


Figure 10. MTBF PE for SoS 3-5-4-12 (90% reliability) for System Type 3 Only

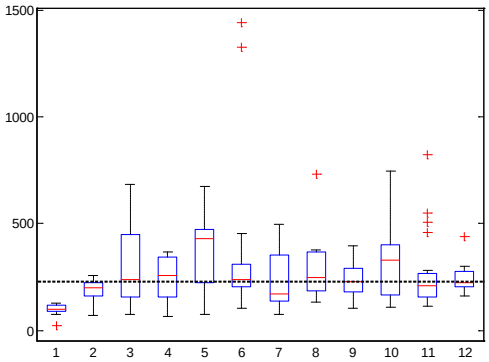


Figure 12. MTBF PE for SoS 3-5-4-12 (90% reliability) for System Type 4 Only

As we discussed in section 7.3.1, the relative absolute error calculations compare the experimental SoS-level reliability obtained via simulation to the true SoS-level reliability as calculated using each SoS RBD. As an excursion, we also performed 1000 simulation replications for each pairing of SoS design and SoS reliability using the true system-level MTBF values (instead of obtaining MTBF point estimates via simulated operational tests, as described in section 7.3.6). In essence, we removed the additional error associated with estimating the system-level MTBF values. Table 13 contains the SoS-level simulation reliability results (using the true system-level MTBF values), along with the parameter maximum likelihood estimates for α and β (recall that the TTF distribution for each SoS is well approximated by a two-parameter gamma distribution).

The relative absolute error for 7 of the 9 cases was between 0.001 and 0.019, with the other 2 cases resulting in relative errors of 0.035 and 0.060. All of the relative absolute errors for a given pairing of SoS design and SoS reliability from Table 13 are less than any case from Tables 14 through 22 associated with the same pairing. This is an encouraging result since we expected to observe less relative error when using the true system-level MTBF values in the SoS-level TTF simulation.

During RTP planning, the emphasis should not be placed on the particular relative absolute error values obtained for each trial via simulation; rather, practitioners should consider the marginal improvement that may be realized by increasing the operational test duration and/or augmenting the SoS test configuration. The sensitivity of relative absolute error to the test configuration and operational test duration is a tool to rank distinct options, ideally, while concurrently considering the cost to execute each option. In addition, using this approach during the RTP planning stage will enable documentation of the rationale for selecting a particular option and promote effective communication of that rationale to leadership.

Proportion of Trials where $P(T>24) \geq 0.90$

Given that we considered SoS-level reliability values of (approximately) 0.90, 0.85, and 0.35, we expected to see the highest proportion of trials where $P(T>24) \geq 0.90$ for cases in which the true SoS-level reliability was 0.90. We expected to see a lower proportion of trials where $P(T>24) \geq 0.90$ for cases in which the true SoS-level reliability was 0.85, and we expected it to be a rare event to observe such a result for cases in which

the true SoS-level was 0.35. By inspection of Tables 23 through 25 below, we see that this is, indeed, the trend.

For all cases in which the true SoS-level reliability was 0.35, there were no trials where $P(T>24) \geq 0.90$. For cases in which the true SoS-level reliability was 0.85, the proportion of trials where $P(T>24) \geq 0.90$ ranged from 0 to 0.35. Interestingly, for cases in which the true SoS-level reliability was 0.90, the highest proportion of trials where $P(T>24) \geq 0.90$ was 0.60. For cases in which the test configuration consisted of just a single test asset of each system type (this is always case 1 in Tables 23 through 25), there were no trials where $P(T>24) \geq 0.90$.

Since the 12 trials for a given pairing of SoS design and SoS reliability are essentially organized in groups of 3 (the operational test length is constant for each group of 3 trials, e.g., trials 1-3 or trials 9-12), we notice that the proportions demonstrate a group-oriented behavior. In general, the difference in the proportions within a group of 3 cases is not statistically significant (using the Fisher exact method for comparing 2 proportions, given "small" samples). As we transition from one group of 3 cases to the next group of 3 cases, the difference in the proportion may be statistically significantly different, but this is not always true of the results. Thus, it is fair to assert that—for the 0.90 and 0.85 SoS reliability levels—the trend with the proportion is that it increases as the operational test duration increases. There is not statistical evidence that the proportion depends on the test configuration, per se.

Table 23. Proportion of Trials where P(T>24)>=0.90 for SoS Design 2-3-4

SoS Reliability = 0.90		SoS Reliability = 0.85		SoS Reliability = 0.35	
Test Configuration Case	Proportion of Trials where P(T > 24) >= 0.90	Test Configuration Case	Proportion of Trials where P(T > 24) >= 0.90	Test Configuration Case	Proportion of Trials where P(T > 24) >= 0.90
1	0	1	0	1	0
2	0.15	2	0.05	2	0
3	0.30	3	0.30	3	0
4	0.45	4	0.20	4	0
5	0.55	5	0.30	5	0
6	0.30	6	0.20	6	0
7	0.50	7	0.15	7	0
8	0.45	8	0.20	8	0
9	0.55	9	0.05	9	0
10	0.60	10	0.10	10	0
11	0.45	11	0.40	11	0
12	0.60	12	0.10	12	0

Table 24. Proportion of Trials where P(T>24)>=0.90 for SoS Design 3-5-12

SoS Reliability = 0.90		SoS Reliability = 0.85		SoS Reliability = 0.35	
Test Configuration Case	Proportion of Trials where P(T > 24) >= 0.90	Test Configuration Case	Proportion of Trials where P(T > 24) >= 0.90	Test Configuration Case	Proportion of Trials where P(T > 24) >= 0.90
1	0	1	0	1	0
2	0.15	2	0.20	2	0
3	0.20	3	0.25	3	0
4	0.35	4	0.15	4	0
5	0.40	5	0.20	5	0
6	0.55	6	0.30	6	0
7	0.35	7	0.20	7	0
8	0.55	8	0.30	8	0
9	0.55	9	0.15	9	0
10	0.25	10	0.30	10	0
11	0.45	11	0.20	11	0
12	0.50	12	0.05	12	0

Table 25. Proportion of Trials where P(T>24)>=0.90 for SoS Design 3-5-4-12

SoS Reliability = 0.90		SoS Reliability = 0.85		SoS Reliability = 0.35	
Test Configuration Case	Proportion of Trials where P(T > 24) >= 0.90	Test Configuration Case	Proportion of Trials where P(T > 24) >= 0.90	Test Configuration Case	Proportion of Trials where P(T > 24) >= 0.90
1	0	1	0	1	0
2	0.10	2	0.10	2	0
3	0.20	3	0.35	3	0
4	0.45	4	0.20	4	0
5	0.35	5	0.15	5	0
6	0.40	6	0.25	6	0
7	0.30	7	0.30	7	0
8	0.35	8	0.05	8	0
9	0.35	9	0.15	9	0
10	0.55	10	0.35	10	0
11	0.45	11	0.15	11	0
12	0.35	12	0	12	0

Simulation Running Time

As we expect that practitioners who may wish to implement the simulation-based approach described herein will be interested in the temporal commitment involved, in Table 26, we include the running times for each pairing of SoS design and SoS reliability. Each time listed in Table 26 represents the time required to execute 240 simulation trials. Recall that each trial involves 1000 replications of the SoS TTF procedure described in section 7.3.6. Therefore, it takes less than 10 minutes to complete 240,000 simulation replications. The cumulative running time to complete all 2,160,000 simulation replications was approximately 86 minutes. Our simulation platform was a Dell Latitude Laptop Model E6510 running Microsoft Windows 7 Enterprise (64-bit) with dual Intel 2.4 GHz i5 Core processors and 4 GB of RAM.

Table 26. SoS Simulation Running Times for Test Configuration Cases

SoS Design	SoS Reliability	Running Time (min)
2-3-4	90	9.88
	85	9.68
	35	9.70
3-5-12	90	9.82
	85	9.52
	35	9.55
3-5-4-12	90	9.39
	85	9.29
	35	9.38
Total		**86.21**

7.3.8 Application Extension: The Iron Triangle of Cost, Schedule, and Accuracy

In order to demonstrate how practitioners can use the results from the simulation-based approach described herein to adopt a particular course of action, we will now provide an example application. Our system will be the SoS 3-5-12 with inherent reliability approximately equal to 90%. We want to make a recommendation regarding the size of the test configuration and the duration of the operational test. We will use results for mean relative absolute error from Table 17.

In our example, we will limit our planning considerations to (1) the cost to produce test assets for use during the operational test and (2) the cost to conduct the operational test. The values that we use are notional, although, they are representative of the costs that the program management office for the SoS can expect to incur. The costs (in millions of dollars) to produce each type of system for use during the operational test appear in Table 27, and the costs associated with test execution are summarized in Table 28. In Table 29, we indicate the number of test weeks required to complete the operational test.

Table 27. Production Costs Associated with SoS Operational Test Configurations

System Type	Cost per System ($M)	Systems in Test Configuration A	Systems in Test Configuration B	Systems inTest Configuration C
1	6	1	2	2
2	4	1	2	3
3	2	1	2	6
Total Cost ($M)		12	24	36

Table 28. Operational Test Work Schedule Options and Associated Costs

Work Schedule Option	Hours per Shift	Shifts per Day	Days per Week	Hours per Week	Cost per Week ($M)
1	8	2	5	80	1
2	8	2	6	96	1.3

Table 29. Number of Test Weeks Required for each Work Schedule Option

Operational Test Length	# Test Weeks (Work Schedule 1)	# Test Weeks (Work Schedule 2)
100	1.25	1.04
300	3.75	3.13
500	6.25	5.21
1000	12.50	10.42

In practice, it is generally possible to schedule two 8-hour shifts per day during the operational test. Typically, it is preferred to plan for a 5-day test week; however, adding a sixth day to the test week may be a feasible option. Now, we would expect to have to pay overtime rates for the 2 shifts on the sixth test day each week. For our example, we assume that the overtime rate is 50% higher than the regular labor rate (see Table 28), but this may be different in practice for different types of personnel participating in the execution of the test. Depending on the level of operational realism, the actual daily work schedules may vary if, say, the goal is to execute 72-hour continuous operations (missions) followed by a 24-hour pause. Such a sequence may be repeated 10 or more times for an operational test. It is a straightforward process to modify the cost estimation procedures and scheduling of labor accordingly.

Given the costs summarized in Tables 27 and 28, we are able to determine the costs associated with each of the 12 operational test cases considered in this study for

SoS 3-5-12. We summarize the costs associated with each of the 12 operational test cases in Table 30. Now, let us momentarily assume that practitioners do not wish to recommend a particular operational test plan where the mean relative error is expected to exceed 10%. In Table 30, we highlighted in yellow the 2 rows in which the mean relative absolute error (obtained via simulation) is 10%. As well, we have highlighted in yellow the total cost associated with each operational test configuration case (for each of the 2 work schedule options. i.e., 5-day or 6-day test weeks).

Table 30. Summary of Operational Test Costs for SoS Design 3-5-12 with 90% Reliability

Case	Test Configuration	Operational Test Length	Mean Relative Error	Total Test Asset Cost ($M)	Test Option 1 Cost ($M)	Total Cost for Option 1 ($M)	Test Option 2 Cost ($M)	Total Cost for Option 1 ($M)
1	1-1-1	100	0.40	12	1.25	13.25	1.35	14.60
2	2-2-2	100	0.26	24	1.25	25.25	1.35	26.60
3	2-3-6	100	0.15	36	1.25	37.25	1.35	38.60
4	1-1-1	300	0.10	12	3.75	**15.75**	4.06	19.81
5	2-2-2	300	0.10	24	3.75	27.75	4.06	31.81
6	2-3-6	300	0.05	36	3.75	39.75	4.06	43.81
7	1-1-1	500	0.11	12	6.25	18.25	6.77	25.02
8	2-2-2	500	0.06	24	6.25	**30.25**	6.77	37.02
9	2-3-6	500	0.06	36	6.25	42.25	6.77	49.02
10	1-1-1	1000	0.06	12	12.50	**24.5**	13.54	38.04
11	2-2-2	1000	0.06	24	12.50	36.5	13.54	50.04
12	2-3-6	1000	0.03	36	12.50	48.5	13.54	62.04

Naturally, it is preferable to recommend the option for which the estimated total cost is the lowest, relatively speaking. In Table 30, we note that (for a mean relative absolute error of 10%), the least costly option is the case in which we have 1 of each type of system in the SoS test configuration (case 4) and we plan to conduct a 300-hour operational test, following a 5-day test week work schedule. The cost for this option is estimated as $15.75 million. Based on Table 30, we anticipate that it will take just under 4 weeks to conduct the 300-hour operational test. The option to adopt a 6-day work

schedule would not be justifiable since we would still expect to take over 3 weeks to complete the operational test (see Table 29) and it would increase the total cost from $15.75 million to $19.81 million.

On the other hand, let us say that we wish to identify the best option for which the mean relative error does not exceed 6%. By inspection of Table 30, we note that there are 4 cases (highlighted in green) in which the mean relative absolute error is 6%. The lowest cost ($24.5 million) is associated with case 10, i.e., 1 of each type of system in the SoS operational test configuration and an operational test duration of 1000 hours. As in the previous scenario, this cost estimate is based on a 5-day work schedule. Adopting a 6-day work schedule would reduce the expected number of test weeks from 12.5 to 10.42 (see Table 29); however, the cost would increase from $24.5 million to $38.04 million (approximately a 55% increase in total cost). It is unlikely that the program manager would consider this to be a good investment.

In the two scenarios that we considered, we determined that the least costly options that satisfied our requirements (threshold tolerance for mean relative absolute error) were both associated with an operational test configuration including just 1 of each type of system. Generally speaking, we would prefer not to base our reliability evaluation on this minimalist SoS operational test configuration (as there is no guarantee that the observed behavior of the individual systems will necessarily be representative of the population). Although we included such SoS operational test configurations in Table 30 for the purpose of this example, practitioners will likely elect to exclude such an option from consideration at the beginning since it would generally be desirable to observe whether or not the redundancy in the SoS works as intended. If that is the case, let us

revisit the second scenario that we considered (mean relative absolute error not to exceed 6%). The best remaining option in Table 30 is the case in which we have 2 of each type of system and the operational test duration is 500 hours (instead of 1000 hours). The associated cost is approximately $30.25 million for this option.

Another important consideration is the amount of calendar time (i.e., the number of test weeks) necessary to complete the operational test. Depending on the work schedule adopted, Table 29 indicates that it should be expected to take between 10.42 and 12.5 weeks to complete a 1000-hour operational test. An implicit assumption is that the test will remain on schedule, and there will be no delays. In reality, there are many factors that have an impact on the test schedule that cannot be controlled such as weather and excessive system down-time due to reliability failures. Therefore, it is reasonable to expect that we cannot, in practice, manage to accumulate the desired SoS-level operating time. Furthermore, as Army operational tests require the participation of Soldiers, it is almost certainly infeasible to plan a single operational test event that may last for a period of 10 to 13 consecutive weeks. Hence, test planning options associated with the accumulation of 500 hours or less during the operational test would be more appropriate courses of action.

By establishing the maximum acceptable threshold for the mean relative absolute error associated with P(T>24) for the SoS, practitioners are certifying that any SoS test configuration for which the mean relative absolute error exceeds that threshold—from a reliability evaluation standpoint—is inadequate. In other words, the empirical results obtained during an operational test employing an inadequate SoS test configuration should not be expected to be representative of the true behavior of the SoS field

configuration. Without going through this process, or at least a variant of this process, practitioners struggle to effectively articulate this to information to program managers. Concordantly, program managers are less likely to allocate the necessary resources to conduct an adequate operational test for a complex SoS.

Although not addressed in our example application, the decision concerning the maximum acceptable level of relative error associated with the SoS operational test configuration is a complex task. The reason that the accuracy in the evaluation of SoS reliability is of paramount importance to decision authorities is because even modest errors in the characterization of reliability can result in significant operational impacts. It is recommended that RTP planners consider the types of missions that the SoS is intended to support as a means to assess the impact of such errors.

For example, let us say that the maximum allowable mean relative absolute error is set to 5%, and there exists an SoS test configuration that satisfies this requirement. Without loss of generality, let us assume for the moment that our estimate for SoS reliability (based on exercising the pre-determined SoS test configuration during an operational test event) turns out to be 90%. If the relative error associated with this experimental estimate (versus the true but unknown SoS reliability) is 5%, the actual SoS reliability could be as low as 85.5%. However, program managers build logistical support campaigns based on the 90% estimate, not the 85.5% estimate. As well, unit commanders expect to maintain a certain level of operational readiness based on the 90% estimate of SoS reliability. Furthermore, long-term budget planning to sustain the system for 20 to 30 years in the field based on the 90% level of reliability would prove insufficient, and there is no guarantee that additional funding could be leveraged to cover the shortfall.

From a mission capability standpoint—depending on the type of system—the lower level of reliability means that it will be less likely for the unit employing the system to detect enemy forces, target enemy assets, counter/evade enemy attacks, reduce the risk of fratricide, or evacuate friendly casualties (where wounded Soldiers become fatalities). Unfortunately, it is impossible to know the magnitude of the error in the evaluation of SoS reliability that is fueled by the results obtained through exercising the selected SoS operational test configuration. Yet, during the planning stage, practitioners should consider the marginal impact of lower-than-estimated SoS reliability to mission capability and sustainment. By exploring the marginal impact to mission capability and sustainment via force-on-force combat simulations, it should be possible to identify cases when the SoS would cease to be suitable and/or affordable. Arguably, this is a more practical and meaningful method to assess RTP adequacy compared to the focusing on traditional consumer and producer risks.

7.4 Consideration of SoS Emergent Behavior

Coined by the English philosopher G.H. Lewes in 1875, the term emergence has been defined and re-defined over the years [49]. A definition of emergence offered by Goldstein in [50] is “the arising of novel and coherent structures, patterns and properties during the process of self-organization in complex systems.” Goldstein extends his discussion, asserting that—regardless of the origin—emergent behaviors exhibit 5 common characteristics:

1. radical novelty,
2. coherence or correlation,

3. a global or macro level,
4. dynamical, and
5. ostensive.

In the context of SoS reliability, emergence pertains to the potential for the manifestation of unpredictable behaviors due to the interaction among the constituent parts of the SoS. The method presented in this chapter for assessing the adequacy of a given SoS operational test configuration is not designed to account for potential emergent behaviors of the SoS. Instead, step 4 of the RTP planning process should consider the implications of behaviors that cannot be modeled or logically derived à priori to decide whether or not an EERA should be established to explicitly address this phenomenon. Arguably, if EERAs are established for (1) OMS/MP coverage and (2) SoS operational test configuration adequacy, these would likely handle the potential for emergent SoS behaviors reasonably well. Otherwise, it may be desirable to establish a separate EERA to reduce the risk of not characterizing such potential behavior.

7.5 Concluding Remarks

The novel simulation-based approach described herein is an appropriate technique to analytically assess the evaluation adequacy of a given SoS test configuration. This methodology provides insight into the relative absolute error sensitivity trends associated with various operational test lengths and SoS test configurations. Given existing data on the systems that constitute the SoS (or similar systems, based on subject matter expertise/professional engineering judgment), this methodology is easy to apply. This approach can be extended to other types of systems as well as more complex systems;

although, further investigation is necessary to determine how the methodology will perform in such cases.

8 SUMMARY AND CONCLUSIONS

8.1 Summary of Major Dissertation Contributions

1. Reliability Test Program Evaluation Risk Assessment Process (Chapter 4)

We devised a novel process to assess evaluation risks associated with reliability test programs (RTPs) to systematically examine poor design practices in observed in RTPs for military systems. This process, which was adapted from the traditional FMEA process, can be applied to any RTP, irrespective of the manner in which the plan was formulated.

2. Identification of Poor Design Practices for RTPs and Countermeasures (Chapter 5)

We applied the evaluation risk assessment process from Chapter 4 to RTPs for 10 Major Defense Acquisition Programs. This application led to the identification of several poor design practices for RTPs as well countermeasures to mitigate the level of risk associated with these practices.

3. Adaptive Reliability Test Program Planning Process (Chapter 6)

Using key considerations from the evaluation risk assessment process along with the insights regarding the poor design practices for RTPs, we formulated a novel, tailorable planning process designed to identify, assess, communicate, and mitigate evaluation risk in RTPs. This intuitive planning process can be applied to any type of commercial or military system.

4. Simulation-Based Risk Assessment Methodology to Assess the Evaluation Adequacy of a System-of-Systems (SoS) Operational Test Configuration (Chapter 7)

A general method was established to assess the adequacy of a given operational test configuration for any type of commercial or military SoS. This method is designed to support the risk assessment step in the RTP planning process.

8.2 Research Findings

Reliability Test Program Evaluation Risk Assessment Process

A reliability test program is designed to investigate and quantify the reliability characteristics of a given system, from a mission-based perspective. However, due to budget and schedule constraints, it is typically infeasible to conduct an exhaustive reliability test program. In Chapter 4, we proposed the following minimum requirements for reliability test programs (additional details can be found in Chapter 4):

- Requirement 1: Evaluate the Reliability of the Field Configuration of the System
- Requirement 2: Test all Elements of the System
- Requirement 3: Exercise all Anticipated Unreliability Drivers
- Requirement 4: Evaluate the Reliability of the System Correctly
- Requirement 5: Identify and Manage the Evaluation Risks of the RTP

Based on experience working in the area of reliability test and evaluation, we have noted that it is atypical for all of the 5 requirements listed above to be satisfied for a given reliability test program. This revelation served as the impetus to examine a cross-section of reliability test programs for 10 Major Defense Acquisitions Programs (documented in Chapter 5).

In order to examine each of the programs in a consistent, scientific manner, we developed an evaluation risk assessment process for reliability test programs. The evaluation risk assessment process that we defined in Chapter 4 is similar to a traditional failure mode and effects analysis (FMEA). This FMEA-like procedure yields a synopsis of identified reliability test program weaknesses along with the potential impact of each weakness with respect to the reliability evaluation. Below, we list the 7 steps in our evaluation risk assessment process.

a. Define the RTP.
b. Identify potential weaknesses in the RTP with respect to evaluation adequacy.
c. Identify monitoring/detection methods and contingency plans.
d. Develop RTP planning alternatives that mitigate, in whole or in part, the potential weaknesses identified in step b.
e. State the perceived impact associated with the implementation of planning alternatives developed in step d.
f. Document the analysis and summarize the problems which could not be corrected by changing the RTP design and identify the special controls which are necessary to reduce failure risk.
g. Provide a recommendation to decision makers based on the results from steps a through f of the RTP risk assessment process.

This evaluation risk assessment process can be employed to scrutinize any military or commercial system reliability test program plan and may be employed by practitioners to compare multiple reliability test program design alternatives for the same system. Ultimately, in a resource-constrained atmosphere, reducing the uncertainty associated

with the characterization of system reliability will need to be balanced with other competing evaluation priorities.

Poor RTP Design Practices Identified during the Application of the RTP Evaluation Risk Assessment Process to 10 Major Defense Acquisition Programs

Through our examination of the RTP details for 10 systems that are Major Defense Acquisition Programs, we identified certain weaknesses—issues that could potentially erode the adequacy of the system-level reliability evaluation. The following practices, which were found in these 10 RTPs, increase evaluation risk.

- *Commonality Assumption*: To reduce the overall test program, the program manager assumes that all system variants will exhibit the same reliability behavior (i.e., have the same failure modes and failure recurrence rates). The associated RTP may include little or no testing of certain system variants, which may preclude the identification of important differences among the variants.
- *Minimum Test Configuration*: For a large system-of-systems (SoS), it may be infeasible to exercise the full field configuration during the reliability demonstration event. The SoS "minimum test configuration" that is exercised may not yield representative information regarding the reliability behavior of the actual field configuration of the SoS. Flaws in assumed SoS redundant capabilities, interactions due to scale, integration issues, and similar concerns may be obscured.
- *"Bootstrap" Calculation of SoS Reliability*: Although the term "bootstrap" has been used by the community in other contexts, we are referring to generating a SoS-level reliability characterization using only system-level failure data. As with

the minimum test configuration approach, it may be impossible to construct an SoS-level reliability characterization that includes important failure modes due to integration and/or interaction of systems within the SoS.

- *Operational Mode Summary/Mission Profile (OMS/MP) Test Slice*: This refers to the case wherein only a subset of the operational conditions (e.g., missions and environmental conditions) will be included in the RTP. Such cases can lead to an inaccurate evaluation of fleet reliability.
- *Insufficient Test Time to Surface Key Failure Modes (FM)*: Due to resource constraints such as the calendar time leading up to a major program milestone decision, availability of facilities, and availability of personnel, the planned test time per asset in the RTP may be less than the time in which key failure modes have previously been observed for similar systems. This is considerable risk to the establishment of an effective logistical support plan after fielding the system, e.g., the number and type of spares, the number and qualifications of maintenance personnel, and the training materials for system operators.
- *Low Likelihood of Detecting System-to-System Variation in Reliability*: One test assumption often made is that all test articles are identical. If the RTP does not allow for a sufficient amount of test time per test article, the likelihood of detecting a variation among the individual test articles may be undesirably low.
- *Low Likelihood of Detecting Drop in System Reliability between DT and OT*: Typically, the total test time accumulated during the developmental test (DT) portion of the RTP is appreciably greater than the length of the operational test (OT). As systems transition from the DT portion of the RTP to the OT, the

demonstrated reliability tends to be lower. As even a 5% difference in the reliability of the system can lead to a significant impact to ownership costs, it is important to determine if this observed drop in reliability is real, or simply an artifact of chance. Hence, it is desirable to design the system RTP such that it will be possible to detect statistically-significant differences between DT and OT reliability.

- *Technology Moving Faster than RTP*: For certain types of systems (such as radios, IT equipment, and smart phones), the technology matures faster than the acquisition program can. New releases of software and hardware typically occur between the end of the RTP and the time of fielding. Thus, testing the exact version of the system that actually makes it to the field is impossible.
- *No Instrumented Data Collection Activities Planned*: Without an on-board instrumentation package, the reliability evaluator will have a limited amount of available information from which to develop insights. Furthermore, the accuracy of the system reliability evaluation will depend solely on the coverage and diligence of human data collectors. Instrumented data (e.g., system states and environmental conditions) can provide objective contextual information regarding reliability incidents and improve the accuracy and timeliness of failure mode root cause analysis. In the absence of on-board instrumentation, critical details may be unrecoverable.

Reliability Test Program Planning Process

As we discussed in Chapters 2, 4, and 6 of this dissertation, widespread variability currently exists with respect to the level of analytical rigor applied during the reliability test program design process. Concordantly, it has proven challenging for practitioners to assess reliability test program adequacy as well as defend reliability test program activities. The spirit and intent of the reliability test program planning process that was presented in Chapter 6 is to manage evaluation risk and support the following activities:

- document the rationale underpinning the reliability test program design,
- communicate the reliability test program plan to the decision authority,
- justify all elements of a given reliability test program design (number and type of events, quantity of test assets, duration of testing, test environment, etc.),
- assess overall reliability test program adequacy, and
- determine the impact of T&E resource reductions on the adequacy of an established reliability test program plan.

Chapter 6 presented the reliability test program (RTP) planning process, which consists of the following 7 steps:

1. Determine the system reliability requirement(s).
2. Define the field configuration of the system.
3. Perform up-front analysis to identify anticipated system unreliability drivers.
4. Establish essential RTP evaluation risk areas.
5. Design a feasible RTP that addresses all essential evaluation risk areas.
6. Assess the level of risk associated with each essential evaluation risk area.
7. If any of the evaluation risk levels are higher than desired, reformulate the RTP.

This planning process is not intended to be unforgivingly rigid; it can be easily modified or extended as desired. Hence, this planning process can be employed to design an adequate and feasible reliability test program for any type of system.

Although the 7 steps in our proposed planning process for reliability test programs may appear to be intuitive and fairly straightforward, in practice these steps—if attempted—have proven to be challenging to perform. For example, effective identification of anticipated system unreliability drivers (planning process step 3) prior to conducting system-level testing requires such activities as

- characterization of the operational loads and stresses on the system,
- component and subsystem testing,
- system-level modeling via simulation, and
- study of historical failure mode data (from developmental and operational testing as well as the field) on earlier configurations and/or similar systems.

Again, the objective of this step in the planning process is not to postulate all fathomable failure modes; rather, the intent is to identify the dominant operationally-relevant system failure modes. The reason that this step is explicitly included in our proposed planning process is because identifying the (anticipated) important system failure modes promotes wise selection of test activities to confirm or refute such assertions regarding the system.

Since the primary purpose of a reliability test program is to characterize how a system can be expected to behave in its intended operational environment, understanding the true unreliability drivers enables a more accurate characterization of system reliability. Furthermore, by understanding the dominant failure modes, program managers can devise an appropriate logistical support strategy to sustain the system in the field

(e.g., number and type of spares, number and type of maintainers, and requisite training for maintainers).

Another vital observation for practitioners involved in the planning of reliability test programs is that too great of an emphasis is placed on the so-called consumer and producer risks associated with the reliability demonstration event. For planning purposes, the null hypothesis is that the system under test does not meet the reliability requirement specified by the Army, and the alternative hypothesis is that the system does, in fact, meet the Army's requirement. Hence, the consumer risk is the probability of committing a type I statistical error, and the producer risk is the probability of committing a type II statistical error.

Consumer risk is driven by the planned length of the reliability demonstration event (operational test), and it includes an associated failure budget for the event. On the other hand, producer risk depends on the planned duration of the reliability demonstration event, the maximum number of allowable failures (associated with the pre-established level of acceptable consumer risk), and the assumed true system reliability achieved by the system developer by the time of the demonstration event. These are mathematical contrivances that are not based in evidence. In particular, the level of producer risk is driven largely by the strong assumption regarding what the achieved true (but unknown) system reliability will be. Based on practical resource constraints, the relatively short reliability demonstration event length necessitates the assumption that the developer will achieve a system-level reliability (say, mean time between system abort) that is 2-3 times higher than the Army requirement. Otherwise, the probability of "passing" the reliability demonstration test (equal to 1-producer risk) may be unacceptably low.

By allowing consumer and producer risks to serve as the foundation of the reliability test program plan, we ignore many other important risk areas that are acknowledged to exist and have more practical significance. Indeed, it is certainly possible to devise an overarching reliability test program that culminates with a reliability demonstration event that, theoretically, manages both consumer and producer risks. However, as we stated above, the common driver in the calculation of both consumer and producer risks is the length of the demonstration event. In no manner does either of these risks account for the operational realism of the demonstration event.

Clearly, the evaluation of a given system's operational reliability should be based on observed performance under operationally-realistic conditions. According to a paper by Hall et al. [32] from the Office of the Director of Operational Test and Evaluation, such conditions include, but are by no means limited to

- who the system operators are (e.g., Soldiers, contractors, or government testers),
- who the maintainers are (e.g., Soldiers, Field Support Representatives, government maintainers, or a combination thereof),
- what the environmental conditions of the tests may be (e.g., blowing dust, blowing sand, solar loading, rain, dense fog, snow, ambient temperature, relative humidity, day/night, etc.), and
- what operational aspects that the tests may or may not include (e.g., operational tempo, force-on-force missions with a credible opposing force, mission durations, mission types to be executed, electronic warfare, information assurance, threat computer network operations).

In 2004, the National Academy of Sciences (NAS) published findings [10] subsequent to a thorough examination of the planned initial operational test for Stryker, a wheeled combat vehicle program. In the report, the NAS panel asserts that the qualitative, non-statistical aspects of an operational test, such as in the case of Stryker, are substantially more important than the statistical aspects, such as consumer and producer risks. Unfortunately, the message has not yet had the desired impact with respect to reliability test program planning.

Thus, our proposed reliability test program planning process includes a step for establishing what we refer to as *essential evaluation risk areas*. The essential evaluation risk areas serve as the building blocks of a given system's reliability test program—they represent the key considerations that must be addressed. Now, the essential evaluation risk areas should be tailored to suit each system, but there are likely many categories of evaluation risk that are appropriate for most types of systems. In Chapter 6, we included the 7 examples of evaluation risk areas summarized below (refer to Chapter 6 for additional details and rationale):

1. Probability of Committing a Type I Statistical Error
2. Probability of Committing a Type II Statistical Error
3. Disparity between the System Test Configuration and the Planned System Field Configuration
4. Operational Mode Summary/Mission Profile Coverage (OMS/MP)
5. Test Exposure Opportunity to Surface Key Failure Modes (Anticipated Unreliability Drivers)
6. Power to Detect Statistically Significant Differences

7. Accuracy of System Reliability Evaluation Methodology

In practice, stakeholders from the Army operational test activity, the program management office, and the Army user community should work together to establish appropriate evaluation risk areas for each reliability test program.

The extent to which each essential evaluation risk area is addressed within the reliability test program is, of course, a matter of vital concern. Therefore, our proposed planning process includes an assessment of the level of risk associated with each essential evaluation risk area. Risk assessment is a non-trivial task that may demand the execution of several supporting analytical activities. Furthermore, although it is possible to establish thresholds for acceptable levels of risk using rational criteria, such criteria are subjective. Despite the inherently subjective nature of the establishment of acceptable evaluation risk thresholds, the cognitive process involved promotes an important dialogue between stakeholders. The outcome of such a process is not simply intended to be the acceptable thresholds for the essential evaluation risk areas—the higher purpose is to think critically about the objectives of the reliability test program, and the manner in which those objectives can be achieved.

Simulation-Based Risk Assessment Methodology for SoS Evaluation Adequacy

For the purpose of demonstration, in Chapter 7, we devised a general method to assess the level of risk associated with a particular essential evaluation risk area, namely, the disparity between the operational test configuration of the SoS and the field/production configuration of the SoS. This is a key reliability test program planning consideration because—given practical reliability test program resource constraints—it is rare that the full field configuration of an SoS can be exercised during an operational test

event. However, as we consider various operational test configurations for an SoS, it is critical to acknowledge that there is a point at which a further reduction in the scale and scope of the test configuration will yield results that are not representative of how the field configuration will behave. In other words, the level of accuracy associated with our evaluation of SoS reliability depends on the adequacy of the SoS test configuration.

The novel simulation-based approach described in Chapter 7 is a technique to analytically assess the evaluation adequacy of a range of distinct SoS operational test configurations and rank the courses of action accordingly. This methodology can quantify how the SoS operational test configuration and the duration of the operational test can have an impact on the accuracy associated with the SoS reliability evaluation. This approach can be applied to a wide range of commercial and military SoS.

In Section 7.3, we presented an illustrative application of the simulation-based risk assessment method. Specifically, we considered 3 SoS designs with k-out-of-n structures, 3 SoS-level reliability values (0.90, 085, and 0.35), and 12 test configuration cases (varying sizes of the SoS test configuration and operational test length), for a total of 108 combinations. We conducted 20 trials for each of the 108 combinations, resulting in 2,160 total simulation trials (refer to Chapter 7 for complete details). We note that—irrespective of the SoS design or level of true SoS reliability—the mean relative error decreases as either the operational test duration increases or the test configuration approaches the full field configuration of the SoS. This is not a surprising result, but it is important that this simulation-based methodology yields such a result.

In order to demonstrate how practitioners can use the results from the simulation-based approach described in Chapter 7 to adopt a particular course of action (i.e., make a

recommendation regarding the size of the test configuration and the duration of the operational test), an example application is included. In our example, we restricted our planning considerations to (1) the cost to produce test assets for use during the operational test and (2) the cost to conduct the operational test. Additional planning considerations can easily be incorporated as desired.

For each pairing of SoS operational test configuration and operational test length, we calculated the total cost associated with each pairing. Given a particular maximum acceptable threshold for the mean relative error associated with SoS reliability, we can identify the option(s) that do not exceed that threshold. If there are multiple options that do not exceed the threshold, the option with the lowest associated cost will likely be adopted. However, it is important to first decide if there are operational test configuration options that should not be considered (such as those that do not offer the potential to surface emergent behaviors of the SoS) before adopting a particular option.

By establishing the maximum acceptable threshold for the mean relative error associated with SoS reliability for the SoS, practitioners are certifying that any SoS test configuration for which the mean relative error exceeds that threshold—from a reliability evaluation standpoint—is inadequate. In other words, the empirical results obtained during an operational test employing an inadequate SoS test configuration should not be expected to be representative of the true behavior of the SoS field configuration. Without going through this process, or at least a variant of this process, practitioners struggle to effectively articulate this to information to program managers. Concordantly, program managers are less likely to allocate the necessary resources to conduct an adequate

operational test for a complex SoS. This is what is referred to as the Iron Triangle—in this context, the 3 vertices of the triangle are cost, schedule, and evaluation accuracy.

8.3 Future Work

Extension of the Simulation-Based Risk Assessment Methodology for SoS Evaluation Adequacy

The implementation of the technique discussed in Chapter 7 for assessing the evaluation adequacy of an SoS operational test configuration is designed to be used under the following conditions:

1. The SoS reliability requirement stipulates that no system-level repairs may be performed during the course of a 24-hour mission.
2. The time between failures for each system within the SoS follows an exponential distribution (i.e., the system-level failure recurrence rates are constant).
3. When a failed systems is repaired, it is assumed to be restored to "as good as new" condition—hence, the SoS is assumed to be "as good as new."
4. The SoS is comprised of multiple k-out-of-n structures arranged in series. Each k-out-of-n structure consists of n identical systems in configured in parallel active redundancy.

Condition 1 is explicitly specified in the definition of the SoS reliability requirement. Conditions 2 and 3 are standard assumptions made by practitioners, and they tend to be reasonable, given the inspection period for the systems that comprise the SoS. Condition 4 is driven by the actual SoS design—the majority of air defense systems are some variant of this design (unique configuration of k-out-of-n structures).

It would be worthwhile to adapt this approach to accommodate a wider range of potential SoS designs with different characteristics and/or assumptions. For example, although reliability requirements for air defense SoS are based on the assumption of no repair during the course of a mission (see condition 1 above), there may be certain system-level failures that could be repaired in the field. This has recently been a subject of much debate between the user community (authors of SoS reliability requirements) and the program management community (system developers). Assuming no repair during missions simplifies the evaluation of SoS reliability; however, if repair during the mission is allowed, the probability of completing a 24-hour mission without an SoS-level abort would be higher. Since we may see a paradigm shift with respect to how SoS reliability requirements are written in the near future, this would be a useful extension to the technique described in Chapter 7.

Risk Assessment Methods for RTP Essential Evaluation Risk Areas

The technique in Chapter 7 is a technique to assess the level of risk associated with a particular evaluation risk area, namely, the disparity between the test configuration and the field/production configuration of the system. In Chapter 6, we proposed other examples of evaluation risk areas having to do with

- reliability test program coverage of the system operational mode summary/mission profile (OMS/MP),
- the power to detect statistically-significant differences (such as system-to-system variation, effectiveness of vendor corrective actions, or degradation/improvement between test events),

- test exposure opportunity to surface key failure modes, and
- accuracy of the system reliability evaluation methodology.

Additional investigation could be performed in any of the above areas to devise appropriate methods and/or activities to assess the level of risk in each of these areas. Furthermore, there are potentially innumerable evaluation risk areas, and one could certainly strive to postulate other meaningful evaluation risk areas and develop associated risk assessment procedures.

APPENDIX

A.1 MATLAB Code

In this section, we include all MATLAB code developed and applied in the conduct of this dissertation research. Before inserting the MATLAB code, i.e., function and script m-files, we will provide a brief description of the purpose of each m-file.

- ultraDriver.m: This script m-file is the main engine that is used to execute all simulation cases, and it relies upon the following 3 m-files
  - ultracase.m: For each simulation case, this function m-file simulates a single operational test, obtains system-level MTBF point estimates, uses the system-level MTBF point estimates in the SoS-level reliability simulation (see ultraSoS.m immediately below) to obtain a vector of 1,000 SoS-level times to failure (TTF), and writes the TTF vector to an output file specified by the user.
  - ultraSoS.m: Given a particular SoS design and system-level MTBF point estimates, this function m-file performs a user-specified number of replications of a TTF simulation for the specified SoS, and it returns a vector of SoS TTF (to ultracase.m).
  - getUltraMTBF.m: This function returns a vector of MTBF point estimates. The MTBF point estimates are based on a single simulated operational test. The length of the operational test as well as the actual quantity of each type of system "present" during the OT are inputs specified by the user.
- ultraTrue.m: Script m-file to run SoS-level TTF (ultraSoS.m) simulation using actual system-level failure rates.
- ultraResults.m: Script m-File to calculate mean relative error in P(T>24) across 20 simulation trials as well as calculate the proportion of trials for which P(T>24) ≥ 0.90. The results are written to a user-specified file.
- knrel.m: This function m-file calculates the reliability for a k-out-of-n structure, given any value of k an n. This function is used by sosDesign234.m, sosDesign3512.m, and sosDesign35412.m (see below).
- sosDesign234.m: This function m-file finds system-level failure rates that result in an SoS-level reliability that is approximately equal to a user-specified target. In this case, the SoS has 3 KN structures in series. The k_i are {1,2,2} and the n_i are {2,3,4}. The user specifies a fraction that is used to obtain system-level failure rates that produce an SoS-level reliability that is arbitrarily close to the user-

specified target. The function returns a vector of 3 system-level failure rates and the calculated SoS-level reliability.

- sosDesign3512.m: This function m-file finds system-level failure rates that result in an SoS-level reliability that is approximately equal to a user-specified target. In this case, the SoS has 3 KN structures in series. The k_i are {2,3,9} and the n_i are {3,5,12}. The user specifies a fraction that is used to obtain system-level failure rates that produce an SoS-level reliability that is arbitrarily close to the user-specified target. The function returns a vector of 3 system-level failure rates and the calculated SoS-level reliability.

- sosDesign35412.m: This function m-file finds system-level failure rates that result in an SoS-level reliability that is approximately equal to a user-specified target. In this case, the SoS has 4 KN structures in series. The k_i are {2,3,2,9} and the n_i are {3,5,4,12}. The user specifies a fraction that is used to obtain system-level failure rates that produce an SoS-level reliability that is arbitrarily close to the user-specified target. The function returns a vector of 4 system-level failure rates and the calculated SoS-level reliability.

- mtbfBoxPlotter.m: Script m-File to generate side-by-side box and whisker plots for system-level MTBF point estimates over 20 trials for a given SoS design and SoS reliability.

- runMTBFplotter.m: Script m-file to plot several comparisons of system-level MTBF point estimates; uses plotMTBFscatter.m.

    - plotMTBFscatter.m: Given 2 sets of system-level MTBF point estimates (each set is based on 20 simulation trials), generate a scatter plot with both data sets (set 1 in blue, set 2 in red) along with a black dashed horizontal line representing the true system-level MTBF.

- fisherPropTest.m: Function m-File to calculate 2-sided p-value for equality of 2 proportions (i.e., the null hypothesis assumes equality). This is the Fisher exact method, and it is appropriate for small samples.

getUltraMTBF.m

```
function mtbf = getUltraMTBF(sosDesign, lambda)
% This function returns a vector of MTBF point estimates.
% The MTBF point estimates are based on a single simulated operational
% test.  The length of the operational test as well as the actual
% quantity of each type of system "present" during the OT are inputs
% specified by the user.

% The expected format of the sosDesign vector is as follows:
% The length of the vector will be 6.
% sosDesign(1) is the number of distinct system types in the SoS
% (either 3 or 4, depending on the scenario considered--can be modified
% to handle other SoS designs)
% sosDesign(2) is the operational test length for the current case
% sosDesign(3) through sosDesign(6) represent the quantity of each type
% of system that will be exercised during the operational test


% Determine the length of the MTBF vector and pre-allocate storage.
mtbf = zeros(1,sosDesign(1));

% Generate individual system availability values.
% Note: In actual OT, the availability (up-time) of each system will
% vary based on the nature of reliability failures that occur.  Hence,
% we do not wish to assume the amount of operating time accumulated by
% all of a single type of system during the OT is equal to
% (num systems)*OT_length.  We use a uniform distribution U(0.6,0.9)
% based on operational availability demonstrated during previous OT
% events for a range of system types.

avail = zeros(sosDesign(1),max(sosDesign(3:6)));

for i = 1:sosDesign(1)
    for j = 1:sosDesign(i+2)
        avail(i,j) = 0.6 + 0.3.*rand();
    end
end

% Calculate system operating times for the OT based on availability
% Note that these are vectors with operating times for multiple systems
% of each type.  The operating times will vary based on the random
% sample from above.

optime = zeros(size(avail));

for i = 1:sosDesign(1)
    for j = 1:sosDesign(i+2)
        optime(i,j) = sosDesign(2)*avail(i,j);
    end
end

% Generate number of system failures.
% We are assuming that the failures occur according to a Poisson
```

```
% Process, and we use the true system-level MTBF values as the
% parameter to randomly generate the number of failures for each
% individual system during each replication.

fails = zeros(size(avail));

for i = 1:sosDesign(1)
    for j = 1:sosDesign(i+2)
        fails(i,j) = poissrnd(optime(i,j)*lambda(i));
    end
end

% Calculate point estimates for the MTBF for each system type.

for i = 1:sosDesign(1)
    % If no failures occur, use the 50% lower confidence bound for
    % MTBF.
    if (sum(fails(i,:)) == 0)
        mtbf(i) = 2*sum(optime(i,:))/chi2inv(0.5,2);
    % If at least 1 failure occurs, calculate the MTBF point estimate
    % assuming that the times between failure follow an exponential
    % distribution (i.e., optime/failures).
    else
        mtbf(i) = sum(optime(i,:))/sum(fails(i,:));
    end
end

end
```

ultraSoS.m

```
function TTF = ultraSoS(numreps,sosDesign,mtbf)

TTF = zeros(numreps,1);

for ctr = 1:numreps

% Create status matrix for all systems.
% Initially,elements are set to 1 to denote that the system is not
% in a failed state.  An initial value of zero denotes that the
% corresponding matrix element is only a placeholder, i.e., the
% element does not correspond to a system.  This is because the
% number of columns in the matrix is driven by the largest number
% of systems for a given KN structure (structure 3, in this case).

% Get number of system types
numTypes = sosDesign(1);
% Load K vector
K = [sosDesign(7) sosDesign(9) sosDesign(11) sosDesign(13)];
% Load N vector
N = [sosDesign(8) sosDesign(10) sosDesign(12) sosDesign(14)];
% Get max quantity of systems within a given KN structure of the SoS
maxKN = max(N);

KN = zeros(numTypes,maxKN);
for i = 1:numTypes
    KN(i,1:N(i)) = ones(1,N(i));
end

% For each simulation replication, perform random draws from the
% exponential distribution for the time-to-failure (TTF) for each
% system. We are using TTF because it is assumed that there will be no
% repair during a mission.

TTF1 = exprnd(mtbf(1),1,N(1));
TTF2 = exprnd(mtbf(2),1,N(2));
TTF3 = exprnd(mtbf(3),1,N(3));
if (numTypes == 4)
    TTF4 = exprnd(mtbf(4),1,N(4));
end

% Create a vector of all TTF from previous section.
if (numTypes == 3)
    TTFmaster = [TTF1 TTF2 TTF3];
else
    TTFmaster = [TTF1 TTF2 TTF3 TTF4];
end
```

```
% Create an index vector to associate TTF with each particular system.
if (numTypes == 3)
    TTFmaster_index = [ones(1,N(1)) 2*ones(1,N(2)) 3*ones(1,N(3)); ...
        1:N(1) 1:N(2) 1:N(3)];
else
    TTFmaster_index = [ones(1,N(1)) 2*ones(1,N(2)) 3*ones(1,N(3)) ...
        4*ones(1,N(4)); 1:N(1) 1:N(2) 1:N(3) 1:N(4)];
end

% Create ordered list of TTF (eventList) and the transition vector
% (eIX) to keep track of the original position in TTF123.
[eventList eIX] = sort(TTFmaster);

% Pre-allocate matrix to identify the failed systems (in order).
failedSystems = zeros(2,length(TTFmaster));

% Populate the matrix with the systems in the order of failure.
for i = 1:length(TTFmaster)
    failedSystems(:,i) = TTFmaster_index(:,eIX(i));
end

% Set SA (as in system abort for the complete SoS) to be false
% since the SoS is operational at time = 0.
SA = false;

% Initialize loop counter for while loop below.
SA_index = 0;

while (SA == false)
    SA_index = SA_index + 1;
    % Update system state matrix KN as systems fail.
    KN(failedSystems(1,SA_index),failedSystems(2,SA_index)) = 0;

    for j = 1:numTypes
        % Check to see if there are >= k out of n systems that are
        % still operational for each KN structure.
        if ( sum(KN(j,:)) < K(j) )
            SA = true;
        end
    end
end

TTF(ctr) = eventList(SA_index);

% End of TTF simulation loop
end

end
```

ultracase.m

```
function ultracase(inFile, caseRange, lambdaRange, outFile)

% Retrieve simulation case inputs
sosDesign = xlsread(inFile,'TC Run Matrix',caseRange);
lambda = xlsread(inFile,'Reliability Run Matrix',lambdaRange);

sosDesignSize = size(sosDesign);
numCases = sosDesignSize(1);

% For each simulation case:  simulate a single operational test, obtain
% system-level MTBF point estimates, use the system-level MTBF point
% estimates in the SoS-level reliability simulation to obtain a vector
% of 1000 SoS-level times to failure (TTF), and write the TTF vector to
% outFile specified by the user.  Repeat 20 times for each simulation
% case.
for i = 1:numCases
    % Extract single case from the matrix of cases.
    currentCase = sosDesign(i,:);
    for j = 1:20
        % Obtain system-level MTBF point estimates.
        mtbf = getUltraMTBF(currentCase,lambda);
        % Use system-level point estimates in SoS-level reliability
        % model to generate TTF distribution.
        TTF = ultraSoS(1000,sosDesign(i,:),mtbf);
        % Write the TTF vector to the outFile specified by the user.
        xlswrite(outFile,TTF,strcat('Sheet',num2str(i)), ...
            strcat(char(64+j),'1'));
    end
    disp(['Case ', num2str(i), ' complete.']);
end

end
```

ultraDriver.m

```
% Script M-File to execute all SoS study cases; written on 09/08/13.

% Create timekeeping file name
timeFile = 'SoS Time Log.xls';
% Create input file name
inFile = 'SoS Ultra Simulation Run Matrix.xls';

%*********************************************************************
% 1 SoS Design {2,3,4}
%*********************************************************************
disp(sprintf('\nBeginning SoS Part 1 of 3.'))

caseRange = 'B2:O13';

% 1.1 SoS Reliability 0.90
outFile = 'SoS 2-3-4 90.xls';
lambdaRange = 'C2:E2';

tStart = tic;

ultracase(inFile, caseRange, lambdaRange, outFile);

tElapsed = toc(tStart);
xlswrite(timeFile,1290,'Sheet1','A1');
xlswrite(timeFile,tElapsed,'Sheet1','B1');
disp(sprintf('\nPart 1.1 complete.'))

% 1.2 SoS Reliability 0.85
outFile = 'SoS 2-3-4 85.xls';
lambdaRange = 'C3:E3';

tStart = tic;

ultracase(inFile, caseRange, lambdaRange, outFile);

tElapsed = toc(tStart);
xlswrite(timeFile,1285,'Sheet1','A2');
xlswrite(timeFile,tElapsed,'Sheet1','B2');
disp(sprintf('\nPart 1.2 complete.'))

% 1.3 SoS Reliability 0.35
outFile = 'SoS 2-3-4 35.xls';
lambdaRange = 'C4:E4';

tStart = tic;

ultracase(inFile, caseRange, lambdaRange, outFile);

tElapsed = toc(tStart);
xlswrite(timeFile,1235,'Sheet1','A3');
xlswrite(timeFile,tElapsed,'Sheet1','B3');
```

```
disp(sprintf('\nPart 1.3 complete.'))


%**********************************************************************
% 2 SoS Design {3,5,12}
%**********************************************************************
disp(sprintf('\nSoS Part 1 complete. Beginning SoS Part 2 of 3.'))

caseRange = 'B14:O25';

% 2.1 SoS Reliability 0.90
outFile = 'SoS 3-5-12 90.xls';
lambdaRange = 'C5:E5';

tStart = tic;

ultracase(inFile, caseRange, lambdaRange, outFile);

tElapsed = toc(tStart);
xlswrite(timeFile,2490,'Sheet1','A4');
xlswrite(timeFile,tElapsed,'Sheet1','B4');
disp(sprintf('\nPart 2.1 complete.'))

% 2.2 SoS Reliability 0.85
outFile = 'SoS 3-5-12 85.xls';
lambdaRange = 'C6:E6';

tStart = tic;

ultracase(inFile, caseRange, lambdaRange, outFile);

tElapsed = toc(tStart);
xlswrite(timeFile,2485,'Sheet1','A5');
xlswrite(timeFile,tElapsed,'Sheet1','B5');
disp(sprintf('\nPart 2.2 complete.'))

% 2.3 SoS Reliability 0.35
outFile = 'SoS 3-5-12 35.xls';
lambdaRange = 'C7:E7';

tStart = tic;

ultracase(inFile, caseRange, lambdaRange, outFile);

tElapsed = toc(tStart);
xlswrite(timeFile,2435,'Sheet1','A6');
xlswrite(timeFile,tElapsed,'Sheet1','B6');
disp(sprintf('\nPart 2.3 complete.'))

%**********************************************************************
% 3 SoS Design {3,5,4,12}
%**********************************************************************
disp(sprintf('\nSoS Part 2 complete. Beginning SoS Part 3 of 3.'))
```

```
caseRange = 'B26:O37';



% 3.1 SoS Reliability 0.90
outFile = 'SoS 3-5-4-12 90.xls';
lambdaRange = 'C8:F8';

tStart = tic;

ultracase(inFile, caseRange, lambdaRange, outFile);

tElapsed = toc(tStart);
xlswrite(timeFile,3690,'Sheet1','A7');
xlswrite(timeFile,tElapsed,'Sheet1','B7');
disp(sprintf('\nPart 3.1 complete.'))

% 3.2 SoS Reliability 0.85
outFile = 'SoS 3-5-4-12 85.xls';
lambdaRange = 'C9:F9';

tStart = tic;

ultracase(inFile, caseRange, lambdaRange, outFile);

tElapsed = toc(tStart);
xlswrite(timeFile,3685,'Sheet1','A8');
xlswrite(timeFile,tElapsed,'Sheet1','B8');
disp(sprintf('\nPart 3.2 complete.'))

% 3.3 SoS Reliability 0.35
outFile = 'SoS 3-5-4-12 35.xls';
lambdaRange = 'C10:F10';

tStart = tic;

ultracase(inFile, caseRange, lambdaRange, outFile);

tElapsed = toc(tStart);
xlswrite(timeFile,3635,'Sheet1','A9');
xlswrite(timeFile,tElapsed,'Sheet1','B9');
disp(sprintf('\nPart 3.3 complete.'))

%**************************************************************************
disp(sprintf('\nSoS simulation complete.'))
```

ultraResults.m

```
% Script M-File to analyze the SoS simulation results; created 09/09/13
tStart = tic;

inFile = 'SoS Ultra Simulation Run Matrix.xls';
relTab = 'Reliability Run Matrix';
relRange ='B2:B10';
sosRel = xlsread(inFile,relTab,relRange);
outFile = 'SoS Ultra Simulation Results Summary.xls';

%*********************************************************************
% 1 SoS Design {2,3,4}
%*********************************************************************

% 1.1 90% SoS Reliability
trials01 = xlsread('ultra SoS calcs 2-3-4 90.xls','Sheet1','F2:F241');
trialCtr = 1;
relError01 = zeros(1,240);
meanRelError01 = zeros(1,12);
prop01 = zeros(1,12);
for i = 1:12
    for j = 1:20
        if( trials01(trialCtr) >= 0.9 )
            prop01(i) = prop01(i) + 1;
        end
        relError01(trialCtr) =
            abs(trials01(trialCtr)-sosRel(1))/sosRel(1);
        trialCtr = trialCtr + 1;
    end
    meanRelError01(i) = mean(relError01(20*(i-1)+1:trialCtr-1));
end

prop01 = prop01./20;

outMat01 = [ones(12,1) 0.90*ones(12,1) (1:12)' prop01'
meanRelError01'];
xlswrite(outFile,outMat01,'Sheet1','A2');
```

```
% 1.2 85% SoS Reliability
trials02 = xlsread('ultra SoS calcs 2-3-4 85.xls','Sheet1','F2:F241');
trialCtr = 1;
relError02 = zeros(1,240);
meanRelError02 = zeros(1,12);
prop02 = zeros(1,12);
for i = 1:12
    for j = 1:20
        if( trials02(trialCtr) >= 0.9 )
            prop02(i) = prop02(i) + 1;
        end
        relError02(trialCtr) =
            abs(trials02(trialCtr)-sosRel(2))/sosRel(2);
        trialCtr = trialCtr + 1;
    end
    meanRelError02(i) = mean(relError02(20*(i-1)+1:trialCtr-1));
end

prop02 = prop02./20;

outMat02 = [ones(12,1) 0.85*ones(12,1) (1:12)' prop02'
meanRelError02'];
xlswrite(outFile,outMat02,'Sheet1','A14');


% 1.3 35% SoS Reliability
trials03 = xlsread('ultra SoS calcs 2-3-4 35.xls','Sheet1','F2:F241');
trialCtr = 1;
relError03 = zeros(1,240);
meanRelError03 = zeros(1,12);
prop03 = zeros(1,12);
for i = 1:12
    for j = 1:20
        if( trials03(trialCtr) >= 0.9 )
            prop03(i) = prop03(i) + 1;
        end
        relError03(trialCtr) =
            abs(trials03(trialCtr)-sosRel(3))/sosRel(3);
        trialCtr = trialCtr + 1;
    end
    meanRelError03(i) = mean(relError03(20*(i-1)+1:trialCtr-1));
end

prop03 = prop03./20;

outMat03 = [ones(12,1) 0.35*ones(12,1) (1:12)' prop03'
meanRelError03'];
xlswrite(outFile,outMat03,'Sheet1','A26');
```

```
%**************************************************************************
% 2 SoS Design {3,5,12}
%**************************************************************************

% 2.1 90% SoS Reliability
trials04 = xlsread('ultra SoS calcs 3-5-12 90.xls','Sheet1','F2:F241');
trialCtr = 1;
relError04 = zeros(1,240);
meanRelError04 = zeros(1,12);
prop04 = zeros(1,12);
for i = 1:12
    for j = 1:20
        if( trials04(trialCtr) >= 0.9 )
            prop04(i) = prop04(i) + 1;
        end
        relError04(trialCtr) =
            abs(trials04(trialCtr)-sosRel(1))/sosRel(1);
        trialCtr = trialCtr + 1;
    end
    meanRelError04(i) = mean(relError04(20*(i-1)+1:trialCtr-1));
end

prop04 = prop04./20;

outMat04 = [2*ones(12,1) 0.90*ones(12,1) (1:12)' prop04'
meanRelError04'];
xlswrite(outFile,outMat04,'Sheet1','A38');


% 2.2 85% SoS Reliability
trials05 = xlsread('ultra SoS calcs 3-5-12 85.xls','Sheet1','F2:F241');
trialCtr = 1;
relError05 = zeros(1,240);
meanRelError05 = zeros(1,12);
prop05 = zeros(1,12);
for i = 1:12
    for j = 1:20
        if( trials05(trialCtr) >= 0.9 )
            prop05(i) = prop05(i) + 1;
        end
        relError05(trialCtr) =
            abs(trials05(trialCtr)-sosRel(2))/sosRel(2);
        trialCtr = trialCtr + 1;
    end
    meanRelError05(i) = mean(relError05(20*(i-1)+1:trialCtr-1));
end

prop05 = prop05./20;

outMat05 = [2*ones(12,1) 0.85*ones(12,1) (1:12)' prop05'
meanRelError05'];
xlswrite(outFile,outMat05,'Sheet1','A50');
```

```
% 2.3 35% SoS Reliability
trials06 = xlsread('ultra SoS calcs 3-5-12 35.xls','Sheet1','F2:F241');
trialCtr = 1;
relError06 = zeros(1,240);
meanRelError06 = zeros(1,12);
prop06 = zeros(1,12);
for i = 1:12
    for j = 1:20
        if( trials06(trialCtr) >= 0.9 )
            prop06(i) = prop06(i) + 1;
        end
        relError06(trialCtr) =
            abs(trials06(trialCtr)-sosRel(3))/sosRel(3);
        trialCtr = trialCtr + 1;
    end
    meanRelError06(i) = mean(relError06(20*(i-1)+1:trialCtr-1));
end

prop06 = prop06./20;

outMat06 = [2*ones(12,1) 0.35*ones(12,1) (1:12)' prop06'
meanRelError06'];
xlswrite(outFile,outMat06,'Sheet1','A62');


%**************************************************************************
% 3 SoS Design {3,5,4,12}
%**************************************************************************

% 3.1 90% SoS Reliability
trials07 = xlsread('ultra SoS calcs 3-5-4-12
90.xls','Sheet1','F2:F241');
trialCtr = 1;
relError07 = zeros(1,240);
meanRelError07 = zeros(1,12);
prop07 = zeros(1,12);
for i = 1:12
    for j = 1:20
        if( trials07(trialCtr) >= 0.9 )
            prop07(i) = prop07(i) + 1;
        end
        relError07(trialCtr) =
            abs(trials07(trialCtr)-sosRel(1))/sosRel(1);
        trialCtr = trialCtr + 1;
    end
    meanRelError07(i) = mean(relError07(20*(i-1)+1:trialCtr-1));
end

prop07 = prop07./20;

outMat07 = [3*ones(12,1) 0.90*ones(12,1) (1:12)' prop07'
meanRelError07'];
xlswrite(outFile,outMat07,'Sheet1','A74');
```

```
% 3.2 85% SoS Reliability
trials08 = xlsread('ultra SoS calcs 3-5-4-12
85.xls','Sheet1','F2:F241');
trialCtr = 1;
relError08 = zeros(1,240);
meanRelError08 = zeros(1,12);
prop08 = zeros(1,12);
for i = 1:12
    for j = 1:20
        if( trials08(trialCtr) >= 0.9 )
            prop08(i) = prop08(i) + 1;
        end
        relError08(trialCtr) =
            abs(trials08(trialCtr)-sosRel(2))/sosRel(2);
        trialCtr = trialCtr + 1;
    end
    meanRelError08(i) = mean(relError08(20*(i-1)+1:trialCtr-1));
end

prop08 = prop08./20;

outMat08 = [3*ones(12,1) 0.85*ones(12,1) (1:12)' prop08'
meanRelError08'];
xlswrite(outFile,outMat08,'Sheet1','A86');


% 3.3 35% SoS Reliability
trials09 = xlsread('ultra SoS calcs 3-5-4-12
35.xls','Sheet1','F2:F241');
trialCtr = 1;
relError09 = zeros(1,240);
meanRelError09 = zeros(1,12);
prop09 = zeros(1,12);
for i = 1:12
    for j = 1:20
        if( trials09(trialCtr) >= 0.9 )
            prop09(i) = prop09(i) + 1;
        end
        relError09(trialCtr) =
            abs(trials09(trialCtr)-sosRel(3))/sosRel(3);
        trialCtr = trialCtr + 1;
    end
    meanRelError09(i) = mean(relError09(20*(i-1)+1:trialCtr-1));
end

prop09 = prop09./20;

outMat09 = [3*ones(12,1) 0.35*ones(12,1) (1:12)' prop09'
meanRelError09'];
xlswrite(outFile,outMat09,'Sheet1','A98');
%**************************************************************
tElapsed = toc(tStart);
disp(sprintf('\nTotal time elapsed for executing all cases: %0.2f
seconds.',tElapsed))
```

ultraTrue.m

```
% Script M-file to run SoS-level TTF simulation using actual system-
% level failure rates.  The results will be used for comparison
% purposes. Created: 09/14/13

% Create input file name
inFile = 'SoS Ultra Simulation Run Matrix.xls';
outFile = 'SoS Ultra Results True MTBF.xls';

%*********************************************************************
% 1 SoS Design {2,3,4}
%*********************************************************************
disp(sprintf('\nBeginning SoS Part 1 of 3.'))
caseRange = 'B2:O2';

% 1.1 SoS Reliability 0.90
lambdaRange = 'C2:E2';
sosDesign = xlsread(inFile,'TC Run Matrix',caseRange);
lambda = xlsread(inFile,'Reliability Run Matrix',lambdaRange);
TTF = ultraSoS(1000,sosDesign,1./lambda);
[params,ci]=gamfit(TTF);
P90 = 1-gamcdf(24,params(1),params(2));
xlswrite(outFile,[params P90],'Sheet1','A2');
xlswrite(outFile,ci,'Sheet1','A3');
disp(sprintf('\nPart 1.1 complete.'))

% 1.2 SoS Reliability 0.85
lambdaRange = 'C3:E3';
sosDesign = xlsread(inFile,'TC Run Matrix',caseRange);
lambda = xlsread(inFile,'Reliability Run Matrix',lambdaRange);
TTF = ultraSoS(1000,sosDesign,1./lambda);
[params,ci]=gamfit(TTF);
P90 = 1-gamcdf(24,params(1),params(2));
xlswrite(outFile,[params P90],'Sheet1','A5');
xlswrite(outFile,ci,'Sheet1','A6');
disp(sprintf('\nPart 1.2 complete.'))

% 1.3 SoS Reliability 0.35
lambdaRange = 'C4:E4';
sosDesign = xlsread(inFile,'TC Run Matrix',caseRange);
lambda = xlsread(inFile,'Reliability Run Matrix',lambdaRange);
TTF = ultraSoS(1000,sosDesign,1./lambda);
[params,ci]=gamfit(TTF);
P90 = 1-gamcdf(24,params(1),params(2));
xlswrite(outFile,[params P90],'Sheet1','A8');
xlswrite(outFile,ci,'Sheet1','A9');
disp(sprintf('\nPart 1.3 complete.'))
```

```
%**************************************************************************
% 2 SoS Design {3,5,12}
%**************************************************************************
disp(sprintf('\nSoS Part 1 complete. Beginning SoS Part 2 of 3.'))
caseRange = 'B14:O14';

% 2.1 SoS Reliability 0.90
lambdaRange = 'C5:E5';
sosDesign = xlsread(inFile,'TC Run Matrix',caseRange);
lambda = xlsread(inFile,'Reliability Run Matrix',lambdaRange);
TTF = ultraSoS(1000,sosDesign,1./lambda);
[params,ci]=gamfit(TTF);
P90 = 1-gamcdf(24,params(1),params(2));
xlswrite(outFile,[params P90],'Sheet1','A11');
xlswrite(outFile,ci,'Sheet1','A12');
disp(sprintf('\nPart 2.1 complete.'))

% 2.2 SoS Reliability 0.85
lambdaRange = 'C6:E6';
sosDesign = xlsread(inFile,'TC Run Matrix',caseRange);
lambda = xlsread(inFile,'Reliability Run Matrix',lambdaRange);
TTF = ultraSoS(1000,sosDesign,1./lambda);
[params,ci]=gamfit(TTF);
P90 = 1-gamcdf(24,params(1),params(2));
xlswrite(outFile,[params P90],'Sheet1','A14');
xlswrite(outFile,ci,'Sheet1','A15');
disp(sprintf('\nPart 2.2 complete.'))

% 2.3 SoS Reliability 0.35
lambdaRange = 'C7:E7';
sosDesign = xlsread(inFile,'TC Run Matrix',caseRange);
lambda = xlsread(inFile,'Reliability Run Matrix',lambdaRange);
TTF = ultraSoS(1000,sosDesign,1./lambda);
[params,ci]=gamfit(TTF);
P90 = 1-gamcdf(24,params(1),params(2));
xlswrite(outFile,[params P90],'Sheet1','A17');
xlswrite(outFile,ci,'Sheet1','A18');
disp(sprintf('\nPart 2.3 complete.'))

%**************************************************************************
% 3 SoS Design {3,5,4,12}
%**************************************************************************
disp(sprintf('\nSoS Part 2 complete. Beginning SoS Part 3 of 3.'))
caseRange = 'B26:O26';

% 3.1 SoS Reliability 0.90
lambdaRange = 'C8:F8';
sosDesign = xlsread(inFile,'TC Run Matrix',caseRange);
lambda = xlsread(inFile,'Reliability Run Matrix',lambdaRange);
TTF = ultraSoS(1000,sosDesign,1./lambda);
[params,ci]=gamfit(TTF);
P90 = 1-gamcdf(24,params(1),params(2));
xlswrite(outFile,[params P90],'Sheet1','A20');
xlswrite(outFile,ci,'Sheet1','A21');
disp(sprintf('\nPart 3.1 complete.'))
```

```
% 3.2 SoS Reliability 0.85
lambdaRange = 'C9:F9';
sosDesign = xlsread(inFile,'TC Run Matrix',caseRange);
lambda = xlsread(inFile,'Reliability Run Matrix',lambdaRange);
TTF = ultraSoS(1000,sosDesign,1./lambda);
[params,ci]=gamfit(TTF);
P90 = 1-gamcdf(24,params(1),params(2));
xlswrite(outFile,[params P90],'Sheet1','A23');
xlswrite(outFile,ci,'Sheet1','A24');
disp(sprintf('\nPart 3.2 complete.'))

% 3.3 SoS Reliability 0.35
lambdaRange = 'C10:F10';
sosDesign = xlsread(inFile,'TC Run Matrix',caseRange);
lambda = xlsread(inFile,'Reliability Run Matrix',lambdaRange);
TTF = ultraSoS(1000,sosDesign,1./lambda);
[params,ci]=gamfit(TTF);
P90 = 1-gamcdf(24,params(1),params(2));
xlswrite(outFile,[params P90],'Sheet1','A26');
xlswrite(outFile,ci,'Sheet1','A27');
disp(sprintf('\nPart 3.3 complete.'))

%**************************************************************************
disp(sprintf('\nSoS simulation complete.'))
```

knrel.m

```
function rel = knrel(k,n,lambda,t)

rel = 0;

for i = k:n
    rel =
        rel + nchoosek(n,i)*exp(-lambda*t)^i*(1-exp(-lambda*t))^(n-i);
end

end
```

sosDesign234.m

```
function [sysrel, lambda] = sosDesign234(sysrel_tgt,dec)

lambda = ones(1,3);
KNR = zeros(1,3);
KNrel_tgt = sysrel_tgt^(1/3);

% Step 1: Obtain value for lambda(1)
lambdaFound = false;
while (lambdaFound == false)
    KNR(1) = knrel(1,2,lambda(1),24);

    if (KNR(1) >= KNrel_tgt)
        lambdaFound = true;
    else
        lambda(1) = dec*lambda(1);
    end
end

% Step 2: Obtain value for lambda(2)
lambdaFound = false;
while (lambdaFound == false)
    KNR(2) = knrel(2,3,lambda(2),24);

    if (KNR(2) >= KNrel_tgt)
        lambdaFound = true;
    else
        lambda(2) = dec*lambda(2);
    end
end

% Step 3: Obtain value for lambda(3)
lambdaFound = false;
while (lambdaFound == false)
    KNR(3) = knrel(2,4,lambda(3),24);

    if (KNR(3) >= KNrel_tgt)
        lambdaFound = true;
    else
        lambda(3) = dec*lambda(3);
    end
end

sysrel = KNR(1)*KNR(2)*KNR(3);

end
```

sosDesign3512.m

```
function [sysrel, lambda] = sosDesign3512(sysrel_tgt,dec)

lambda = ones(1,3);
KNR = zeros(1,3);
KNrel_tgt = sysrel_tgt^(1/3);

% Step 1: Obtain value for lambda(1)
lambdaFound = false;
while (lambdaFound == false)
    KNR(1) = knrel(2,3,lambda(1),24);

    if (KNR(1) >= KNrel_tgt)
        lambdaFound = true;
    else
        lambda(1) = dec*lambda(1);
    end
end

% Step 2: Obtain value for lambda(2)
lambdaFound = false;
while (lambdaFound == false)
    KNR(2) = knrel(3,5,lambda(2),24);

    if (KNR(2) >= KNrel_tgt)
        lambdaFound = true;
    else
        lambda(2) = dec*lambda(2);
    end
end

% Step 3: Obtain value for lambda(3)
lambdaFound = false;
while (lambdaFound == false)
    KNR(3) = knrel(9,12,lambda(3),24);

    if (KNR(3) >= KNrel_tgt)
        lambdaFound = true;
    else
        lambda(3) = dec*lambda(3);
    end
end

sysrel = KNR(1)*KNR(2)*KNR(3);

end
```

sosDesign35412.m

```
function [sysrel, lambda] = sosDesign35412(sysrel_tgt,dec)

lambda = ones(1,4);
KNR = zeros(1,4);
KNrel_tgt = sysrel_tgt^(1/4);

% Step 1: Obtain value for lambda(1)
lambdaFound = false;
while (lambdaFound == false)
    KNR(1) = knrel(2,3,lambda(1),24);

    if (KNR(1) >= KNrel_tgt)
        lambdaFound = true;
    else
        lambda(1) = dec*lambda(1);
    end
end

% Step 2: Obtain value for lambda(2)
lambdaFound = false;
while (lambdaFound == false)
    KNR(2) = knrel(3,5,lambda(2),24);

    if (KNR(2) >= KNrel_tgt)
        lambdaFound = true;
    else
        lambda(2) = dec*lambda(2);
    end
end

% Step 3: Obtain value for lambda(3)
lambdaFound = false;
while (lambdaFound == false)
    KNR(3) = knrel(2,4,lambda(3),24);

    if (KNR(3) >= KNrel_tgt)
        lambdaFound = true;
    else
        lambda(3) = dec*lambda(3);
    end
end

% Step 4: Obtain value for lambda(4)
lambdaFound = false;
while (lambdaFound == false)
    KNR(4) = knrel(9,12,lambda(4),24);

    if (KNR(4) >= KNrel_tgt)
        lambdaFound = true;
    else
        lambda(4) = dec*lambda(4);
    end
```

```
end

sysrel = KNR(1)*KNR(2)*KNR(3)*KNR(4);

end
```

mtbfBoxPlotter.m

```
% Script M-File to generate side-by-side box and whisker plots for
% system-level MTBF point estimates over 20 trials for a given SoS
% Design and SoS reliability.  Created on 09/28/13.

pause on;

inFile = 'SoS 3-5-4-12 90 MTBF.xls';

m1true = 243.90;
m2true = 151.51;
m3true = 112.36;
m4true = 227.27;

m1pe = zeros(20,12);
m2pe = zeros(20,12);
m3pe = zeros(20,12);
m4pe = zeros(20,12);
tv = 0:13;

for i = 1:12
    m1pe(:,i) = xlsread(inFile,strcat('Sheet',num2str(i)),'A1:A20');
end

boxplot([m1pe(:,1) m1pe(:,2) m1pe(:,3) m1pe(:,4) m1pe(:,5) ...
    m1pe(:,6) m1pe(:,7) m1pe(:,8) m1pe(:,9) m1pe(:,10) ...
    m1pe(:,11) m1pe(:,12)]);
hold on;
mv1 = m1true*ones(1,14);
plot(tv,mv1,'--k','LineWidth',2);
pause;

hold off;
for i = 1:12
    m2pe(:,i) = xlsread(inFile,strcat('Sheet',num2str(i)),'B1:B20');
end

boxplot([m2pe(:,1) m2pe(:,2) m2pe(:,3) m2pe(:,4) m2pe(:,5) ...
    m2pe(:,6) m2pe(:,7) m2pe(:,8) m2pe(:,9) m2pe(:,10) ...
    m2pe(:,11) m2pe(:,12)]);
hold on;
mv2 = m2true*ones(1,14);
plot(tv,mv2,'--k','LineWidth',2);
pause;

hold off;
for i = 1:12
    m3pe(:,i) = xlsread(inFile,strcat('Sheet',num2str(i)),'C1:C20');
end

boxplot([m3pe(:,1) m3pe(:,2) m3pe(:,3) m3pe(:,4) m3pe(:,5) ...
    m3pe(:,6) m3pe(:,7) m3pe(:,8) m3pe(:,9) m3pe(:,10) ...
    m3pe(:,11) m3pe(:,12)]);
```

```
hold on;
mv3 = m3true*ones(1,14);
plot(tv,mv3,'--k','LineWidth',2)
pause;

hold off;
for i = 1:12
    m4pe(:,i) = xlsread(inFile,strcat('Sheet',num2str(i)),'D1:D20');
end

boxplot([m4pe(:,1) m4pe(:,2) m4pe(:,3) m4pe(:,4) m4pe(:,5) ...
    m4pe(:,6) m4pe(:,7) m4pe(:,8) m4pe(:,9) m4pe(:,10) ...
    m4pe(:,11) m4pe(:,12)]);
hold on;
mv4 = m4true*ones(1,14);
plot(tv,mv4,'--k','LineWidth',2);
```

plotMTBFscatter.m

```
function plotMTBFscatter(d1,d2,trueMTBF)

numtrials = 1:length(d1);
numtrials = numtrials';

mt = trueMTBF*ones(length(d1)+2,1);

plot(0:21,mt,'--k');
hold on;
scatter(numtrials,d1,'filled','MarkerFaceColor','b');
scatter(numtrials,d2,'filled','MarkerFaceColor','r');
xlabel('Trial Number');
xlim([0 21]);
ylim([0 400]);
ylabel('MTBF Point Estimate');
title('Comparison of 2 Sets of System-level MTBF Point Estimates across
20 Trials');
hold off;
end
```

runMTBFplotter.m

```
% Script M-file to plot several comparisons of system-level MTBF point
% estimates. Created on 09/28/13.

pause on;

inFile = 'SoS 2-3-4 90 MTBF.xls';

% Compare {1,1,1}*100 to {2,2,3}*100
d1 = xlsread(inFile,'Sheet1','A1:A20');
d2 = xlsread(inFile,'Sheet3','A1:A20');
plotMTBFscatter(d1,d2,117.6);
pause;

% Compare {1,1,1}*100 to {1,1,1}*300
d3 = xlsread(inFile,'Sheet4','A1:A20');
plotMTBFscatter(d1,d3,117.6);
pause;

% Compare {1,1,1}*100 to {1,1,1}*500
d4 = xlsread(inFile,'Sheet7','A1:A20');
plotMTBFscatter(d1,d4,117.6);
pause;

% Compare {1,1,1}*100 to {1,1,1}*1000
d5 = xlsread(inFile,'Sheet10','A1:A20');
plotMTBFscatter(d1,d5,117.6);
pause;

% Compare {1,1,1}*100 to {2,2,3}*1000
d6 = xlsread(inFile,'Sheet12','A1:A20');
plotMTBFscatter(d1,d6,117.6);
pause;
```

fisherPropTest.m

```
function [PV2, PL, PU] = fisherPropTest(prop1,prop2,n1,n2)
% Function M-File to calculate 2-sided p-value for equality of 2
% proportions.  This is the Fisher exact method, and it is appropriate
% for small samples.  Written: 09/15/13.

% Get parameters for the MATLAB implementation of the hypergeometric
% distribution, i.e., {X,M,K,N}.

K = n1;
M = n1 + n2;
X = prop1*n1;
Y = prop2*n2;
N = X + Y;

% Calculate lower one-sided p-value, PL.

PL = 0;
for i = max(0,N-M+K):X
    PL = PL + hygepdf(i,M,K,N);
end

% Calculate upper one-sided p-value, PU.

PU = 0;
for i = X:min(K,N)
    PU = PU + hygepdf(i,M,K,N);
end

% Calculate 2-sided p-value, PV2.

PV2 = 2*min(PL,PU);

end
```

A.2 Mathematica Code

Build a list of input file names that store results from the MATLAB SoS time-to-failure (TTF) simulation.

```
fn = {"SoS 2-3-4 90.xls","SoS 2-3-4 85.xls","SoS 2-3-4 35.xls", "SoS 3-
5-12 90.xls", "SoS 3-5-12 85.xls", "SoS 3-5-12 35.xls", "SoS 3-5-4-12
90.xls", "SoS 3-5-4-12 85.xls", "SoS 3-5-4-12 35.xls"};

TableForm[fn]

SoS 2-3-4 90.xls
SoS 2-3-4 85.xls
SoS 2-3-4 35.xls
SoS 3-5-12 90.xls
SoS 3-5-12 85.xls
SoS 3-5-12 35.xls
SoS 3-5-4-12 90.xls
SoS 3-5-4-12 85.xls
SoS 3-5-4-12 35.xls
```

Create a table element to store calculated values.

```
extable = Table[0,{241},{6}];
```

Make the first row in the table a header row.

```
extable[[1]]={"Case","Trial","□","□","1st Moment","P(T>24)"};
```

For all 12 cases in each input file:

1. calculate MLEs for α and β for the two-parameter gamma distribution
2. write the case number (1-12), trial number (1-20), α, β, the first central moment, and P(T>24) to each row in the table “extable”

```
For[j = 3; rowctr = 2, j<13,j++, For[k = 1, k<21, k++; rowctr++,
TTF = Import["MATLAB/ultra/"<>fn[[1]],{"Data",j,Range[1,1000],k}];
TTFedist=EstimatedDistribution[TTF,GammaDistribution[α,β]];
extable[[rowctr]]={j,k,TTFedist[[1]],TTFedist[[2]],Moment[TTFedist,1],
1-CDF[TTFedist,24]}]];

Export["UMD/SoS Adequacy/ultra SoS calcs 2-3-4 90.xls",extable];
```

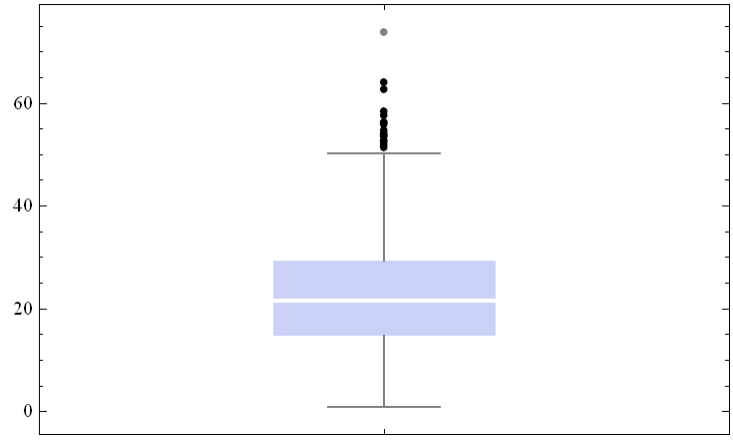
The above statements can be repeated for all fn[[i]] where i = {1,2,...,9}

The following loop can be used to take the empirical TTF data for n cases (16 cases in this example) to:

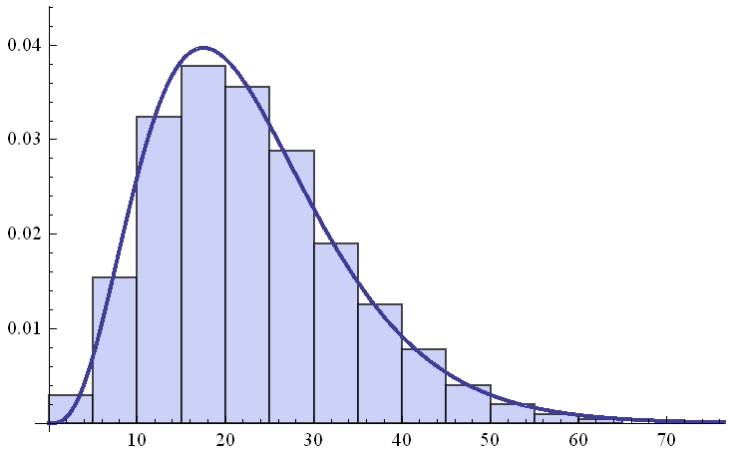
(1) generate a relative frequency histogram of the empirical TTF data
(2) calculate MLEs α and β for the two-parameter gamma distribution
(3) plot the PDF of the fitted gamma(α, β) distribution
(4) calculate the first central moment of the gamma(α, β)
(5) calculate P(T>24 hours), using the gamma(α, β) CDF
(6) create a probability plot to visually assess goodness of fit
(7) create box & whisker plot of the empirical TTF data
(8) calculate the arithmetic mean of the TTF distribution

```
For[i=1,i<17,i++,sosData =
Import[fnxls[[i]],{"Data",1,Range[1,1000],1}];
BoxWhiskerChart[sosData,{"Outliers",{"MedianMarker",1,
Directive[Thick,White]}}]//Print;
Print["Arithmetic Mean: ",Mean[sosData]];
SoSedist2=EstimatedDistribution[sosData,GammaDistribution[α, β]];
Show[Histogram[sosData,Automatic,"ProbabilityDensity"],
Plot[PDF[SoSedist2,x],{x,0,85},PlotStyle→Thick]]//Print;
Print[SoSedist2];Print["First Moment: ",Moment[SoSedist2,1]];
Print["P(T>t_mission) = " ,1-CDF[SoSedist2,24]];
ProbabilityPlot[sosData,SoSedist2]//Print]
```

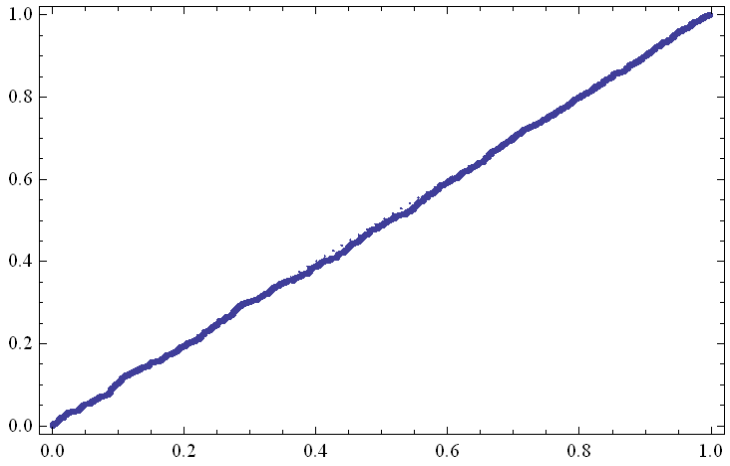
Without loss of generality, we only show results from the first iteration of the above for loop.




```
Arithmetic Mean: 22.9863
```




```
GammaDistribution[4.20304,5.46896]
First Moment: 22.9863
P(T>t_mission) = 0.400536
```



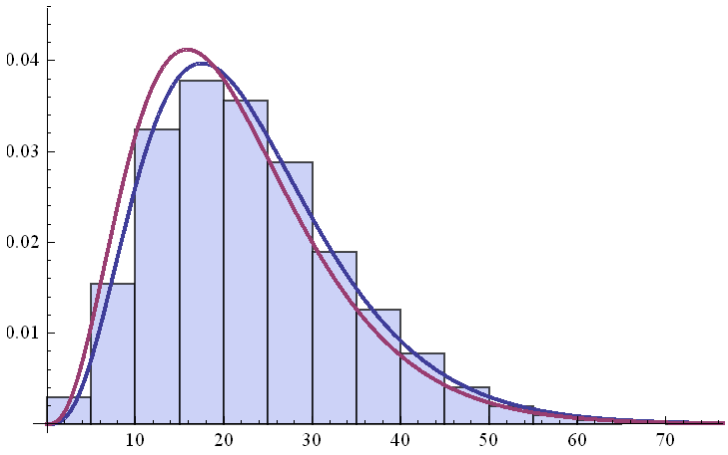
Given empirical TTF data, the following loop will:

1. generate a relative frequency histogram of the empirical TTF data
2. calculate MLEs α and β for the two-parameter gamma distribution
3. plot the PDF of the fitted gamma(α, β) distribution
4. plot the PDF associated with another underlying TTF distribution (e.g., the true TTF distribution)
5. calculate the first central moment of the gamma(α, β)
6. plot two TTD CDFs on the same axes along with a dashed black vertical line at T = 24 hours
7. calculate P(T>24 hours), using the gamma(α, β) CDF

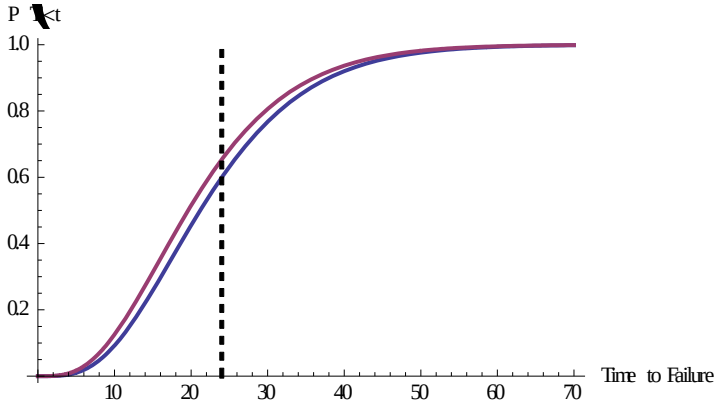
```
For[i=1,i<17,i++,
sosData = Import[fnxls[[i]],{"Data",1,Range[1,1000],1}];
Print[Style["Case " <> ToString[i],"Section"]];
SoSedist2=EstimatedDistribution[sosData,GammaDistribution[α, β]];
Show[Histogram[sosData,Automatic,"ProbabilityDensity"],
Plot[{PDF[SoSedist2,x],
PDF[GammaDistribution[3.8509,5.5670],x]},{x,0,80},
PlotStyle→{Thick,Thick}]]//Print; Print[SoSedist2];
Print["First Moment: ",Moment[SoSedist2,1]];
Plot[{CDF[SoSedist2,x],CDF[GammaDistribution[3.851,5.567],x]},{x,0,70},
PlotStyle→{Thick,Thick}, AxesLabel→{"Time to Failure","P(T<t)"},
Epilog➞{Dashed,Thick,Line[{{24,0},{24,1}}]}]//Print;
Print["P(T>t_mission) = " ,1-CDF[SoSedist2,24]]]
```

Without loss of generality, we only show results from the first iteration of the above for loop.

Case 1




```
GammaDistribution[4.20304,5.46896]
First Moment: 22.9863
```




```
P(T>t_mission) = 0.400536
```

A.3 Overview of Reliability Growth Planning Curves (RGPC)

Per DoD [2] and Army [3] policy, all acquisition programs with reliability requirements that are under DoD oversight must develop a RGPC. The RGPC is intended as a management tool for devising a structured approach to exposing design weaknesses (i.e., failure modes), applying corrective actions, and growing system-level reliability. The latest Army policy [3] signed by the Army Acquisition Executive on 26 June 2011, mandates the use of the U.S. Army Materiel Analysis Activity (AMSAA) Planning Model based on Projection Methodology (PM2). The theoretical underpinnings of PM2 may be found in AMSAA Technical Report 652: AMSAA Reliability Growth Guide [5], the Army Test and Evaluation Command (ATEC) Implementation Guide for U.S. Army Reliability Policy [42], and Military Handbook 189 C: Reliability Growth Management [37].

The idealized growth curve is the theoretical MTBF that would be achieved if all expected failures were found and corrected instantly at any given point in test time. The steepness of the curve is an important factor. Steep curves require rapid growth in a very short period of time—which might be infeasible for some programs. The curve also allows the IPT to work backwards from the reliability needed at the end of the test program to find the reliability needed at the beginning of the test program. In Figure 13, we have included a notional RGPC.

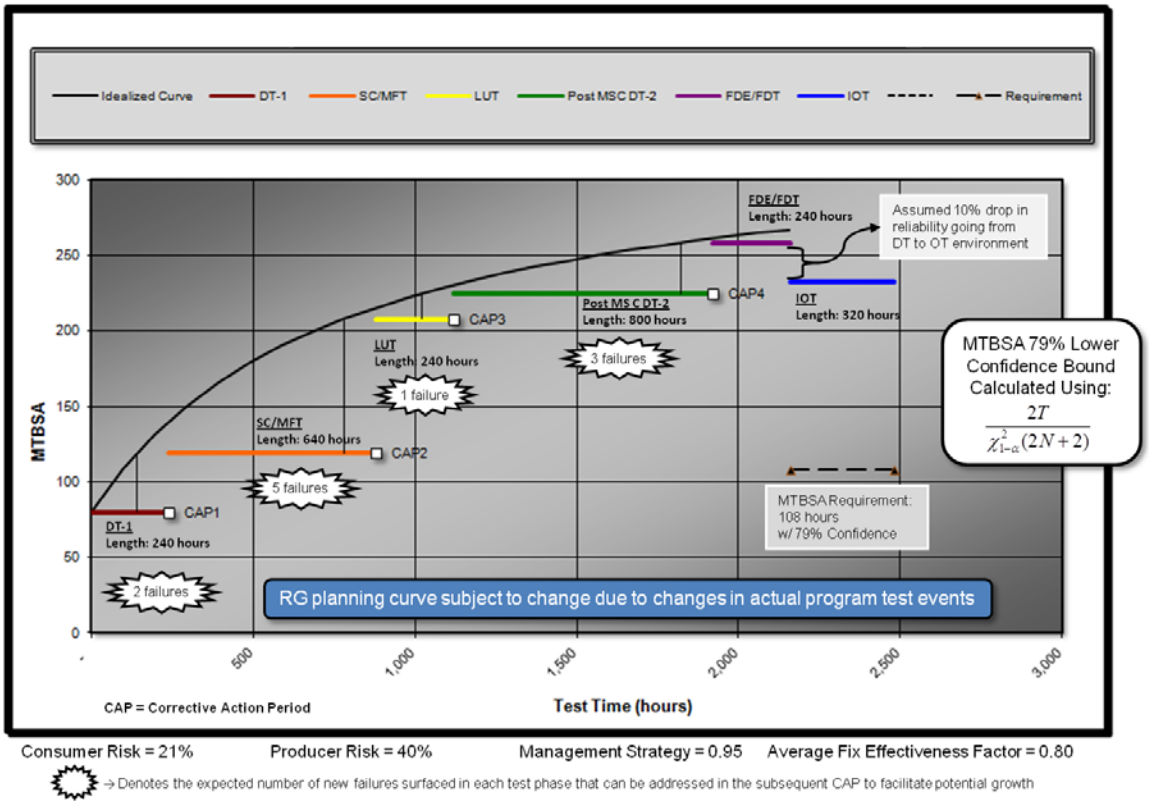


Figure 13. Notional System Reliability Growth Planning Curve

Key inputs into the RGPC development process include:

- The reliability requirement (M_R), established by the user
  - Per Department of the Army Pamphlets 70-3 [7]and 73-1 [8], system reliability requirements must be demonstrated with high statistical confidence (typically 80% or higher, depending on the type of system).
- The length of the reliability demonstration test (length of the Initial Operational Test (IOT))
- The reliability design goal (M_G)
  - Using standard Operating Characteristic (OC) curve analysis procedures, M_G is set in order to balance the consumer and producer risks. Since the reliability requirement, M_R, must be demonstrated during the IOT with

high statistical confidence, the target for M_G must be sufficiently higher than M_R. As well, we typically anticipate a drop in system-level reliability as we transition from a developmental test environment to an operational test environment. This drop may occur for various reasons, such as: more stressful test conditions, employment by representative operators, etc.

- The Management Strategy (MS)
  - The fraction of the initial failure intensity that will be addressed via corrective action.
- The planned average Fix Effectiveness Factor (μ)
  - As corrective actions are implemented, each has a degree of effectiveness. The symbol μ represents the average across all corrective actions to be implemented.
- The initial reliability (M_i) for the reliability test program
  - M_i is frequently estimated using contractor test data. If the initial reliability is too low, it may be infeasible for a program to ever "get on the curve." By implementing design-for-reliability (DfR) best practices, it is possible to identify the majority of system failure modes, and implement corrective actions prior to the start of the government reliability growth program.
- The reliability growth potential (M_{GP})
  - By definition, $M_{GP} \equiv \frac{M_i}{1-\mu \cdot MS}$. This represents the theoretical limit to system-level reliability that could be achieved, given infinite time.

RGPC Risk Assessment

For every RGPC, the system reliability evaluator should conduct a risk assessment. Table 31 summarizes the risks associated with 10 categories related to RGPCs. Notice that all categories are medium risk for our notional RGPC. The guidance for the risk levels was developed by AMSAA and appears in the Army Center for Reliability Growth (CRG) Short Course on Reliability [43].

Table 31. Notional Reliability Growth Planning Curve Risk Assessment Matrix

Category	Low Risk	Medium Risk	High Risk
MTBF Goal (DT) M_G = 258 hrs; M_{GP} = 333 hrs M_G/M_{GP} = 0.77	Less than 70% of Growth Potential	70 – 80% of Growth Potential	Greater than 80% of Growth Potential
IOT&E Producer's Risk (40%)	20% or less	20-40%	Greater than 40%
IOT&E Consumer's Risk (21%)	20% or less	20-30%	Greater than 30%
Management Strategy (95%)	90% or lower	90-96%	Greater than 96%
Fix Effectiveness Factor (80%)	70% or lower	70-80%	Greater than 80%
MTBF Goal (DT)/MTBF Initial (258 hrs / 80 hrs = 3.23)	Less than 2	2-4	4 or Larger
Time to Incorporate and Validate Fixes in IOT&E Units Prior to Test	Adequate time and resources to have fixes implemented & verified with testing or strong engineering analysis	Time and resources for almost all fixes to be implemented & most verified w/ testing or strong engineering analysis	Many fixes not in place by IOT&E and limited fix verification
Corrective Action Periods (CAP) 4 CAPs in JLENS RGPC	5 or more CAPS which contain adequate calendar time to implement fixes prior to major milestones	3 - 4 CAPS but some may not provide adequate calendar time to cut in all fixes	1- 2 CAPS of limited duration
Reliability Increases after CAPs see jump between SC/MFT and LUT	Moderate reliability increases after each CAP result in lower-risk curve that meets goals	Some CAPs show large jumps in reliability that may not be realized because of program constraints	Majority of reliability growth tied to one or a couple of very large jumps in the reliability growth curve
Percent of Initial Problem Mode Failure Intensity Surfaced	Growth appears reasonable (i.e. a small number of problem modes surfaced over the growth test do not constitute a large fraction of the initial problem mode failure intensity)	Growth appears somewhat inflated in that a small number of the problem modes surfaced constitute a moderately large fraction of the initial problem mode failure intensity	Growth appears artificially high with a small number of problem modes comprising a large fraction of the initial problem mode failure intensity

A.4 RTP Questionnaire for Army Reliability Evaluators

The following questions were posed to the reliability evaluators for major acquisition programs of interest, and the responses have been archived:

1. Does the IOT test configuration of the system match the intended fielding configuration of the system?
    - If differences exist between the IOT test configuration of the system and the intended fielding configuration of the system, describe the extent of the differences.
    - In addition to a description of the differences between the IOT test configuration and the fielding configuration, include the reason(s) behind the differences.
        - For example, assets were not available because there was not enough money in the PM's T&E budget.
2. Did the vendor attempt to identify unreliability drivers (i.e., portions of the system expected to contribute the failure modes constituting the majority of the initial failure intensity) prior to the start of the reliability test program?
    - If the vendor did conduct activities intended to identify unreliability drivers:
        - Which activities did the vendor elect to conduct (e.g., PoF analyses, HALT/ALT, shakedown testing, bench testing)?
        - Why did the vendor select each activity (historical success with the technique, etc.)?
        - Which activities were successful in enabling the identification of failure modes?
        - Which failure modes were surfaced by each of the DfR activities?
3. If the IOT test configuration of the system does not match the fielding configuration of the system:
    - Was the test configuration chosen based solely on availability of assets?
    - Was the availability of assets in support of the chosen test configuration driven by which portions of the system were anticipated to be linked to key failure modes (i.e., failure modes that contribute to the majority of the initial failure intensity)?

- If the test configuration was not expected to include portions of the system that were anticipated to heavily influence the overall reliability of the system, what actions (if any) were taken to communicate the deficiency to the RAM sub-IPT or the AST?

4. What approach was taken to communicate risk to stakeholders and leadership?
    - What was the consumer risk?
    - What was the producer risk?
    - What was the risk to the evaluation, i.e., the risk that the test configuration would not enable us to adequately characterize the reliability behavior of the system? If this risk was not explicitly considered in the planning stages, then consider the question given what is currently known.
5. Was there a tentative plan to leverage late DT data along with IOT data in support of the reliability evaluation?
    - Did the RAM sub-IPT coordinate with DOT&E during the development of the reliability test program?
    - What were the conditions/criteria (if any) that DOT&E established for it to be acceptable to combine the reliability results from late DT along with the IOT reliability results?
    - If the IOT has occurred and there was a tentative plan in place to potentially combine the results from late DT along with the results from the IOT, were the conditions satisfied so that the data could be pooled? In other words, were the failure modes and their respective recurrence rates consistent? Were the test conditions (e.g., loads, stresses, operators) comparable?

REFERENCES

[1] C. M. Bolton, “Reliability of US Army Materiel Systems,” Memorandum issued by the Office of the Assistant Secretary of the Army for Acquisition, Logistics and Technology, 6 December 2007.

[2] F. Kendall, “Reliability Analysis, Planning, Tracking, and Reporting,” Directive Type Memorandum (11-003) issued by the Office of the Under Secretary of Defense for Acquisition, Technology and Logistics, 21 March 2011.

[3] H. Shyu, “Improving the Reliability of US Army Materiel Systems.” Memorandum issued by the Office of the Assistant Secretary of the Army for Acquisition, Logistics and Technology, 26 June 2011.

[4] R. N. Tamburello and M. J. Cushing, “Lessons Learned during the Test and Evaluation of US Army Reliability Growth Programs,” Paper presented at the 17th Annual Canada/US Operations Research Symposium, Ottawa, ON, 28 August 2012.

[5] W. J. Broemm, P. M. Ellner, and W. J. Woodworth, “AMSAA Reliability Growth Guide,” AMSAA Technical Report 652, 2000.

[6] Director of Operational Test and Evaluation, “FY 2012 Annual Report to Congress,” 2012.

[7] Headquarters, Department of the Army, “Department of the Army Pamphlet 70-3: Army Acquisition Procedures,” 2008.

[8] Headquarters, Department of the Army, “Department of the Army Pamphlet 73-1: Test and Evaluation in Support of Systems Acquisition,” 2003.

[9] National Academy of Sciences, “Improved Operational Testing and Evaluation: Better Measurement and Test Design for the Interim Brigade Combat Team with Stryker Vehicles, Phase I Report,” 2003.

[10] National Academy of Sciences, “Improved Operational Testing and Evaluation and Methods of Combining Test Information for the Stryker Family of Vehicles and Related Army Systems, Phase II Report,” 2004.

[11] US Department of Defense, “Military Standard 1629A: Procedures for Performing a Failure Mode, Effects and Criticality Analysis,” 1980.

[12] Information Technology Association of America (ITAA), “GEIA-STD-0009: Reliability Program Standard for Systems Design, Development, and Manufacturing,” Arlington, VA: ITAA, 2008.

[13] TechAmerica, "TechAmerica Engineering Bulletin: Reliability Program Handbook TA-HB-0009," Washington, DC: TechAmerica, 2013.

[14] K. E. Murphy, C. M. Carter, and R. H. Gass, "Who's Eating Your Lunch? A Practical Guide to Determining the Weak Points of Any System," *Proceedings of the Annual Reliability and Maintainability Symposium*, 2003, pp. 263-268.

[15] O. P. Yadav, N. Singh, and P. S. Goel, "A Practical Approach to System Reliability Growth Modeling and Improvement. *Proceedings of the Annual Reliability and Maintainability Symposium*, 2003, pp. 351-359.

[16] O. P. Yadav, N. Singh, P. S. Goel, "Reliability demonstration test planning: A three dimensional consideration," *Reliability Engineering & System Safety*, vol. 91, no. 8, 2006, pp. 882-893.

[17] W. B. Nelson and J. B. Hall, "Statistical Reliability Demonstration for a Complex System," unpublished manuscript, 2011.

[18] D. K. Lloyd, M. Lipow, *Reliability: Management, Methods, and Mathematics*, Englewood Cliffs, New Jersey: Prentice Hall, 1962.

[19] P. M. Ellner, R. Easterling, *et al*., "Handbook for the Calculation of Lower Statistical Confidence Bounds on System Reliability (Report #1, Ad-Hoc Methodology Working Group on Nuclear Weapons Reliability Assessment)," Maximus. February 1980.

[20] C. S. Reese, M. Hamada, and D. Robinson, "Assessing System Reliability by Combining Multilevel Data from Different Test Modalities," *Quality Technology and Quality Management*, vol. 2, no. 2, 2005, pp. 177-188.

[21] E. T. Jaynes, "Information Theory and Statistical Mechanics," *Physical Review*, vol. 106, no. 4, 1957, pp. 620-630.

[22] E. T. Jaynes, *Probability Theory The Logic of Science*, New York: Cambridge University Press, 2003.

[23] M. Modarres, *Risk Analysis in Engineering: Techniques, Tools, and Trends*, New York: CRC Press, 2006.

[24] R. M. Herman and K. M. Janasak, "Using FMECA to Design Sustainable Products," *Proceedings of the Annual Reliability and Maintainability Symposium*, 2011, pp. [1-6].

[25] J. B. Bowles, "The New SAE FMECA Standard," *Proceedings of the Annual Reliability and Maintainability Symposium*, 1998, pp. 48-53.

[26] Society of Automotive Engineers, “SAE J1739: Potential Failure Mode and Effects Analysis in Design (Design FMEA) and Potential Failure Mode and Effects Analysis in Manufacturing and Assembly Processes (Process FMEA) and Effects Analysis for Machinery (Machinery FMEA),” 2009.

[27] Society of Automotive Engineers, “SAE ARP 5580:Recommended Failure Modes and Effects Analysis (FMEA) Practices for Non-Automobile Applications,” 2001.

[28] US Department of Defense, “Military Standard 785B: Reliability Program for Systems and Equipment Development and Production,” 1980.

[29] M. R. Wayne and M. Modarres, “Reliability Growth Planning using Combined Developmental and Operational Test Data for Reliability Demonstration,” unpublished manuscript, 2013.

[30] C. E. Shannon, “A Mathematical Theory of Communication,” *Bell System Technical Journal*, vol. 27, 1948, pp. 379–423, 623–656.

[31] H. F. Martz, R. A. Waller, *Bayesian Reliability Analysis*, New York: Wiley, 1982.

[32] J. B. Hall, A. Fries, B. Simpson, L. Freeman, “Ground Combat Vehicle Reliability,” Enclosure to memorandum issued by the Director of Operational Test and Evaluation on “The Bayesian Continuous Planning Model (BCPM),” 16 September 2013.

[33] J. M. Gilmore, “The Bayesian Continuous Planning Model (BCPM),” Memorandum issued by the Director of Operational Test and Evaluation, 16 September 2013.

[34] D. W. Coit, “System-Reliability Confidence-Intervals for Complex-Systems With Estimated Component Reliability,” *IEEE Trans. on Reliability*, vol. 46, no. 4, 1997, pp. 487-493.

[35] W. B. Nelson, *Recurrent Events Data Analysis for Product Repairs, Disease Recurrences, and Other Applications*, Philadelphia: SIAM, 2003.

[36] C. S. Reese, “A Hierarchical Model for the Reliability of an Anti-aircraft Missile System (LA-UR-05-9281),” Submitted to *Journal of the American Statistical Association*, dated 7 December 2005.

[37] US Department of Defense, “Military Handbook 189C: Reliability Growth Management,” 2011.

[38] Headquarters, Department of the Army, “Army Regulation 73-1: Test and Evaluation Policy,” 2006.

[39] US Army Test and Evaluation Command Headquarters, “ATEC Command Overview,” Information briefing provided to the Program Executive Office for Missiles and Space, Huntsville, AL, 4 June 2012.

[40] Office of the Under Secretary of Defense for Acquisition, Technology and Logistics (USD[AT&L]), “Department of Defense Instruction 5000.02, Operation of the Defense Acquisition System,” 2008.

[41] L. J. Bain and M. Engelhardt, *Statistical Analysis of Reliability and Life-Testing Models: Theory and Methods*, New York: Marcel Dekker, ed. 2, 1991.

[42] J. B. Hall, M. J. Cushing, P. M. Ellner, *et al.*, “Army Test and Evaluation Command (ATEC) Implementation Guide for U.S. Army Reliability Policy,” ATEC Technical Note, 15 June 2009.

[43] M.R. Wayne, “Reliability Growth Planning,” presentation during the Army Center for Reliability Growth Short Course on Reliability, 5 September 2013.

[44] M. Modarres, M. Kaminskiy, and V. Krivtsov, *Reliability Engineering and Risk Analysis: A Practical Guide*, New York: CRC Press, ed. 2, 2009.

[45] S. L. Jeng and W. Q. Meeker, “Comparison of Approximate Confidence Interval Procedures for Type I Censored Data,” *Technometrics*, vol. 42, no. 2, 2000, pp. 135-148.

[46] US Department of Defense, “Defense Acquisition Guidebook,” 2012.

[47] US Department of Defense, “Systems Engineering Guide for System of Systems,” 2008.

[48] Y. Hatamura, editor, Learning from Design Failures, Tokyo: Springer, 2009.

[49] P. A. Corning, “The Re-emergence of ‘Emergence’: A Venerable Concept in Search of a Theory,” Complexity, vol. 7, no. 6, 2002, pp. 18-30.

[50] J. Goldstein, “Emergence as a Construct: History and Issues,” *Emergence*, vol. 1, no. 1, 1999, pp. 49-72.