MODEL BASED SYSTEMS ENGINEERING DEMONSTRATION FOR AIRWORTHINESS OF DEPARTMENT OF DEFENSE ROTORCRAFT

THESIS

Bretton M. Bethel, 2d Lt, USAF

AFIT-ENV-MS-21-M-205

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENV-MS-21-M-205

MODEL BASED SYSTEMS ENGINEERING DEMONSTRATION FOR AIRWORTHINESS OF DEPARTMENT OF DEFENSE ROTORCRAFT

THESIS

Presented to the Faculty

Department of Systems Engineering and Management

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Systems Engineering

Bretton M. Bethel, BS

2d Lt, USAF

March 2021

AFIT-ENV-MS-21-M-205

MODEL BASED SYSTEMS ENGINEERING DEMONSTRATION FOR AIRWORTHINESS OF DEPARTMENT OF DEFENSE ROTORCRAFT

Bretton M. Bethel, BS

2d Lt, USAF

Committee Membership:

Lt Col Amy Cox, Ph.D.
Chair

David Jacques, Ph.D.
Member

John Colombi, Ph.D.
Member

AFIT-ENV-MS-21-M-205

Abstract

Airworthiness is a process of certifying that an aircraft can be safety operated within specified bounds. This process begins with identifying the airworthiness requirements that apply to the specific aircraft. Next, an airworthiness plan is created for the aircraft and test data is collected. The data analyzed and the results of the analysis is used to verify and satisfy the airworthiness requirements. Finally, when the aircraft has verified and satisfied all requirements, the aircraft can receive an airworthiness certification. This process is essential to ensuring the safety of the aircraft, its personnel, and the surrounding assets.

A Model-Based Systems Engineering (MBSE) approach can be used as a method to improve the airworthiness process. MBSE is the methodology of creating and utilizing domain models as a means of exchanging and presenting information for a wide variety of disciplines to understand and replacing previous document-based exchange.

The objective of this research is to develop a reference architecture with a MBSE approach to perform the airworthiness process loop. The model features a logical system model, stores airworthiness requirements and flight test data, performs analysis, and uses analysis outputs to satisfy and verify airworthiness requirements. The reference architecture was applied to a Dolphin helicopter in hover and takeoff conditions to demonstrate the effectiveness. The results of the demonstration provide a proof of concept for the successful implementation of an MBSE approach to the airworthiness certification process.

Acknowledgments

I would like to express my sincere appreciation to my faculty advisor, Lt Col Amy Cox, for her guidance and support throughout the course of this thesis effort. The insight, experience, and donation of previously collected data was crucial for my success and was certainly appreciated. I would, also, like to thank the many individuals, both family and friends, that have supported me through this rigorous endeavor.

Bretton M. Bethel

Table of Contents

List of Figures

List of Tables

MODEL BASED SYSTEMS ENGINEERING DEMONSTRATION FOR AIRWORTHINESS OF DEPARTMENT OF DEFENSE ROTORCRAFT

I. Introduction

General Issue

Airworthiness is concerned with the safety of air vehicles relative to the passengers of the air vehicle, as well as people and assets on the ground. When air travel first began in 1903 with the Wright Flyer, the safety of the occupants and surrounding life was not a significant concern. As technology advanced and air travel increased, the potential harm of aviation activities to those directly involved and to participants increased as well. The field of airworthiness has captured best practices in design and operation towards the goal of improved safety.

Airworthiness is a process of certifying that an aircraft can be safety operated within specified bounds. This process begins with identifying the airworthiness requirements that apply to the specific aircraft, whether that be civilian (Federal Aviation Administration, Federal Aviation Regulations), Department of Defense (Military Standards), or both when the aircraft is operated in both domains. Next, an airworthiness plan is created for the aircraft and data is collected. The data can come from analysis and flight test data. The flight test data comes from two categories: pre-flight tests and bench tests for the components and the aircraft as a whole, and flight tests while the aircraft is operational. Once the data is collected and analyzed, the results of the analysis is used to verify and satisfy the airworthiness requirements. Finally, when the aircraft has verified and satisfied all requirements, the aircraft can receive an airworthiness certification. This process is essential to ensuring the safety of the aircraft, its personnel, and the surrounding assets.

The Department of Defense pushes the boundaries for developing new aircraft; however, the airworthiness approval for new aircraft and modifications to existing aircraft is an important factor to ensure safe operation. The DoD requires aircraft to be developed and modified to continue to be the superior force in warfighting, but also requires the aircraft to be safe for personnel. As the aircraft becomes increasingly more complicated and the safety requirements increase in depth and detail, the airworthiness process lengthens and becomes more complicated as a result. In order for the DoD to continue air superiority in warfighting, the airworthiness approval process must continue to evolve to be completed more efficiently.

The phases of an airworthiness approval in the DoD include pre-contract, pre-flight test, flight test, and operations. The pre-contract phase consists of an airworthiness plan and a certification basis. The pre-flight test phase includes analysis with sub-system test and ground test compliance data. The flight test phase is the iterative cycle of compliance review and risk assessment, risk acceptance, test memorandum for record (MFR) issuance, and flight test compliance data. The final phase is the operations phase, which includes another compliance review and risk assessment, another risk acceptance, and an operations MFR and Military Type Certificate (MTC) issuance (AFLCMC/EZZ, 2020). The phases described above can be seen in Figure 1.

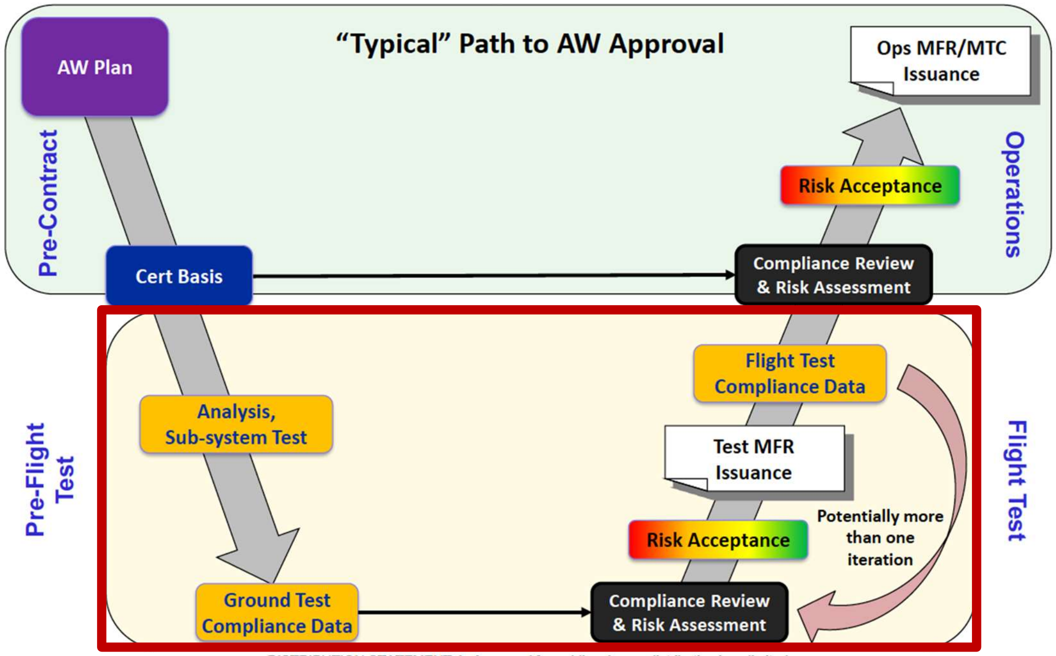


Figure 1: The DoD Airworthiness Process Phases (AFLCMC/EZZ, 2020).

The airworthiness process has been traditionally completed with a document-based process, where it is plagued by delays and confusion caused by version control, multi-organization influences, decentralized document locations, and widespread disorganization throughout the process. The Airworthiness Authority (AA) is in charge of managing the airworthiness process and issuing the airworthiness approval. For the Department of Defense (DoD), the AA is located at a single office at the Air Force Life Cycle Management Center (AFLCMC) but interacts with multiple other organizations across multiple geographically separated locations. The need for a more effective solution to conduct the airworthiness process in a collaborative environment has become an issue inside the DoD. The Air Force as a whole and Air Force Materiel Command have expressed a need for Digital Engineering solutions to existing issues with focus on improving Air Force processes, such as the airworthiness certification process (Roper, 2020).

The Government Accountability Office (GAO) released a report that described aircraft mission capable rates and they did not meet goals previously established. The capability rates are a metric used by the DoD to assess aircraft readiness levels. Capability rates are the percentage of total time when the aircraft can fly and perform at least one mission. The GAO examined the readiness of 46 selected fixed-wing and rotary aircraft across all branches of the DoD. The GAO cited that “for fiscal year 2019: 6 aircraft were 5 percentage points or fewer below the goal; 18 were from 15 to 6 percentage points below the goal; and 19 were more than 15 percentage points below the goal, including 11 that were 25 or more percentage points below the goal” (GAO, 2020). Figure 2 illustrates the readiness of the 46 aircraft across 9 fiscal years. The report notes multiple challenges affecting aircraft in the DoD that can cause the poor readiness levels. Some of the challenges are related to the unresponsive document-based approach to airworthiness. There are tools that have been developed that are better suited to handle the airworthiness process.

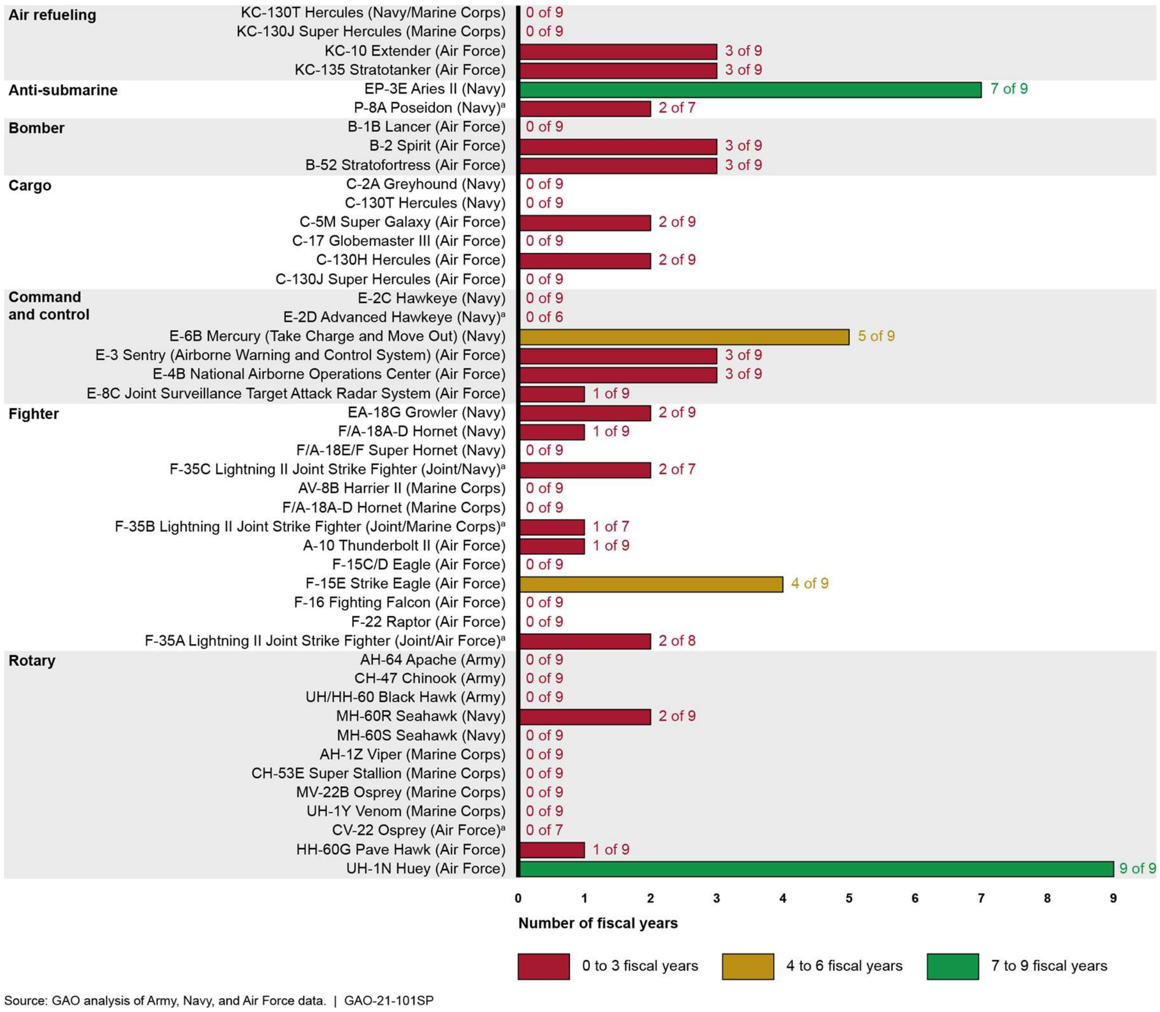


Figure 2: Number of Times Selected Aircraft Met Their Annual Mission Capable Goal, Fiscal Years 2011 through 2019 (GAO, 2020)

This research will focus on the Model-Based Systems Engineering (MBSE) approach utilizing the Systems Modeling Language (SysML) and Cameo Systems Modeler as the modeling tool to accomplish the airworthiness process. MBSE is the methodology of creating and utilizing domain models as a means of exchanging and presenting information for a wide variety of disciplines to understand and is intended to replace the previous document-based exchange. It is common to be achieved through the Unified Modeling Language (UML) or the Systems Modeling Language (SysML).

The process of architecting systems has been simplified by leveraging MBSE and UML or SysML, both in creating the architecture and implementing the architecture. As the benefits of systems architecture has increased, the emergence of reference architectures has unveiled a potential for providing a template for domain-specific solutions. MBSE is the modeling of the system requirements, system design, system analysis, and requirements verification and validation activities. The models show relationships among system requirements, functions, components, and actors.

Problem Statement

The airworthiness process is plagued by delays that contribute to low aircraft readiness levels that have been described by the GAO (GAO, 2020). Airworthiness has multiple phases as described above and a comprehensive model using MBSE provides many benefits. A model can better trace airworthiness requirements not only to system components and functions, but also mission and scenarios stored in use cases. The model can be used to store the flight test data, perform analysis, and complete a trade space analysis. MBSE is can be a collaborative method which can aid the airworthiness process. It provides a single source of truth that supports the needs of multiple stakeholders, eliminating confusion and reduces time. Implementing an MBSE approach to airworthiness has the potential to make the process more efficient. Utilizing MBSE for the airworthiness process can provide improvements in areas such as testing, analysis, and requirements verification. However, there is not a relevant example of airworthiness using MBSE in the scope of the Department of Defense for organizations to use a reference.

For this research, the focus will be on the Pre-Flight Test and Flight Test Phases of the airworthiness process, which includes compliance review and risk assessment, risk acceptance, test MFR issuance, and flight test compliance data. These phases can be seen highlighted in red in Figure 1. In these phases, test data from both pre-flight bench tests and flight tests will be analyzed and used to verify airworthiness requirements inside a MBSE reference architecture.

Research Objectives and Questions

As mentioned above, the overarching objective of this research is to produce a method of utilizing MBSE to create an improved airworthiness process. This research will focus on the application of a rotorcraft vehicle. A breakdown of this overarching objective follows:

1. Develop parametric models that automate analysis of test data with integrated MATLAB functions that will compute performance parameters and generate aircraft-specific safety charts.
2. Utilize automated analysis of flight test data to satisfy and verify airworthiness, performance, and mission requirements inside a reference architecture.
3. Leverage an MBSE approach to perform the airworthiness process loop for a helicopter.

The above resulted in the completion of the MBSE method for the Pre-Flight Test Phase and Flight Test phase of airworthiness for a rotorcraft application in order to answer the following questions:

1. How can flight test data be organized inside an MBSE reference architecture to optimize the usability of the model?
2. How can flight test data be used to satisfy and verify airworthiness requirements in a SysML model?
3. How can a reference architecture be constructed to automate the analysis of flight test data?
4. How can an MBSE approach be leveraged to improve the airworthiness process in the DoD?

Methodology

The most significant effort of this research was demonstrating existing methods of airworthiness by implementing an MBSE approach. The methods were adapted for the specific application of airworthiness of a helicopter in hover but provides a reference for full scale aircraft and can be adapted for more than this specific example. The airworthiness process demonstrated in this research will follow the process illustrated in Figure 3, which is featured below.

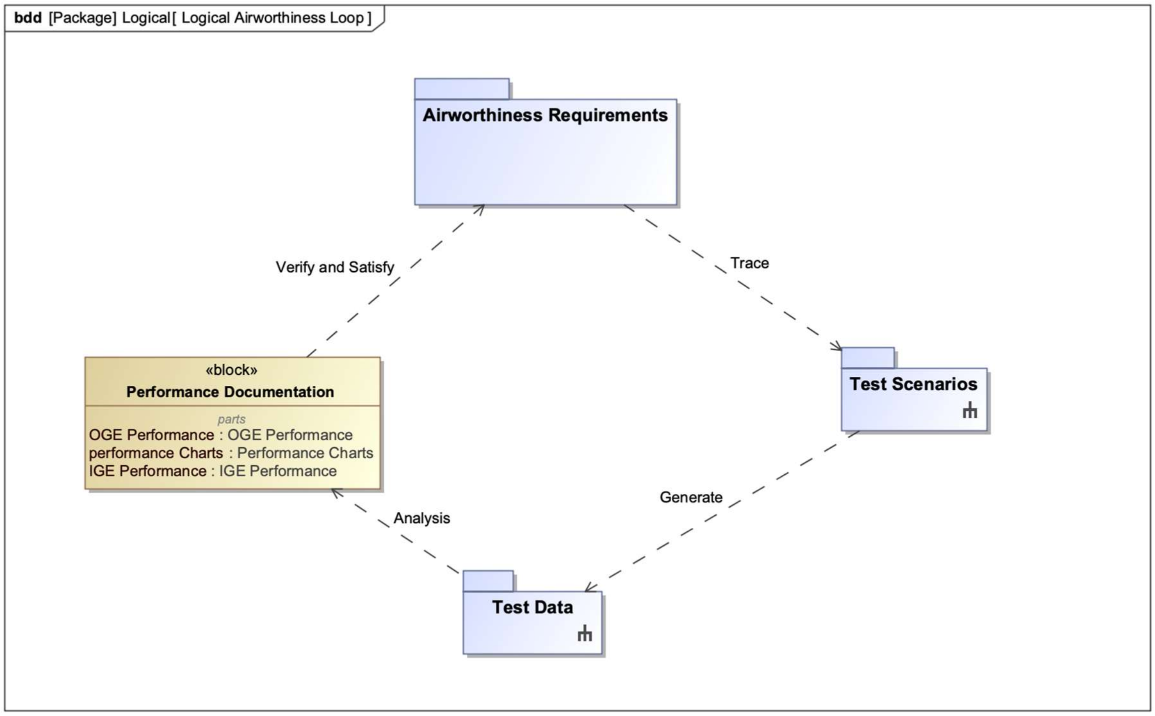


Figure 3: The Airworthiness Process Loop

Chapter III outlines the MBSE approach to architecting a model for a helicopter that accomplishes the analysis for airworthiness certifications for a helicopter in hover and takeoff conditions. The system architecture is designed with ease of operability where it utilizes the MBSE capabilities for requirements traceability, data analysis, chart generation, and requirements verification and satisfaction. The test data is stored in text files and it is represented as value properties of test data blocks in the model. This allows the data to be called in the analysis in the parametric models. The parametric models built in the architecture are where the automation capabilities are utilized to make the airworthiness process more effective. The analysis is accomplished by creating MATLAB functions that are represented as constraint blocks in the parametric models. Once the constraint blocks are created, the test data blocks and the constraint clocks are connected by ports and a simulation is run. The simulation is automated to generate performance parameters and charts. The performance parameters and charts are used to both verify and satisfy the airworthiness requirements and are displayed in requirements matrices.

Assumptions and Limitations

An assumption of this research is that the airworthiness process for an aircraft in practice, while utilizing an MBSE approach, can be repeated using this research method. This is assuming that in practice the airworthiness certification process would span multiple organizations and this research was completed by a single person. When this effort is performed by a single person, it is effective because there is no communication necessary to other people or organizations. When the process involves many organizations, often physically separated from one another geographically, the collaboration and efficiency is reduced as a result. In addition, the single person performing this research has knowledge and experience with MBSE and SysML that allows the process to be efficient. If the person, or people, performing this effort does not have knowledge and experience with MBSE and SysML, the effectiveness may not be the same.

There were limitations placed on this research which made it accomplishable in a constrained timeline. The first limitation is the application of a single type of aircraft, a helicopter, and a single helicopter, the Dolphin. This limits the ability of this research to be easily applied to other aircraft, such as fixed-wing aircraft and small unmanned aerial systems. In addition, the demonstration for only the Dolphin helicopter limits the ability to determine the effectiveness on other rotorcraft such as multiple rotor helicopters and larger helicopters, since the Dolphin is a small single rotor helicopter.

The second limitation is that only hover and takeoff airworthiness requirements were evaluated during the demonstration of the reference architecture. This limitation was placed on this research for multiple reasons such as only having access to test data for those conditions and a complete set of airworthiness requirements would require much more time than available.

The last limitation on this research was the exclusion of the motor and transmission limitations. The limitations on the motor and transmission affects the performance charts generated during the analysis. Without these considered during the analysis, the charts are incomplete because they do not accurately represent the actual capabilities of the aircraft; theoretically the chart shows aircraft performance that is not possible due to maximum allowable conditions for the motor and transmission. The motor and transmission limitations could have been included in the analysis if time permitted.

Preview

Chapter I provides an introduction to the topic area, discussed the problem, outlined methods for creating the reference architecture and explored the limitations of the research. Chapter II discusses airworthiness and Model Based Systems Engineering in the DoD, reference architectures for simulation and design, and how Model Based Systems Engineering is being utilized in the Test and Evaluation community. Chapter III describes in depth the methodology used to identify the relevant requirements, organize the test data that was acquired, and create the parametric models. Chapter IV provides an analysis of the effectiveness of the reference architecture when implementing the architecture for performing airworthiness process on the Dolphin helicopter in hover and takeoff conditions. Chapter V summarizes conclusions and provides answers to the research questions discovered during the analysis as well recommendations for future research and highlights the significance of this research.

II. Literature Review

Chapter Overview

Implementing a Model Based Systems Engineering (MBSE) approach to the airworthiness process of an aircraft is a complex issue to solve and requires a foundation of knowledge. First, an understanding of airworthiness and the certification process that is required for an aircraft, both in the civilian domain and the Department of Defense (DoD) domain. Both domains are relevant to this research as many aircraft are utilized as civilian and DoD aircraft. Second, a familiarity of the capabilities that MBSE can offer and the applied research that accompanies those capabilities. Next, a review of the ways industry and governmental agencies are leveraging MBSE reference architectures in domain fields for simulation and analysis of systems. Finally, the combination of MBSE in the airworthiness process and test and evaluation communities that have been researched in the commercial and government sectors. These topics will provide the foundation of knowledge required to architect a MBSE solution to airworthiness certification of a helicopter in takeoff and hover.

Airworthiness

Airworthiness has many definitions depending on the entity that is certifying the aircraft, whether it be a civilian or military entity. Filippo De Florio defines airworthiness for the civilian domain as “the possession of the necessary requirements for flying in safe conditions, within allowable limits” (De Florio, 2006). The Department of Defense (DoD) defines airworthiness in MIL-HDBK-516C as “the ability of an aircraft to obtain, sustain, and terminate flight in accordance with prescribed usage requirements”

(Department of Defense, 2014). The common theme between both definitions is the importance of the safety by verifying and validating the requirements.

For the civilian definition by De Florio, the three important pieces of the definition are the requirements, the safe conditions, and the allowable limits. The necessary requirements refer to the aircraft, and its parts that must be designed and built within the tested criteria to operate in the safe conditions. These requirements are dictated by airworthiness authorities and are obtained through publications. Safe conditions refer to the normal operating conditions for the aircraft to reduce injury to persons in the aircraft or on the ground, and to reduce damage to property and the environment. The allowable limits refer to the 'flight envelope' for which a specific aircraft is designed to operate within, which depends on conditions such as speed, loading factors, and others. Airworthiness is focused around the safety of persons on or off the aircraft, the property on the ground, and the environment (De Florio, 2006).

For the DoD definition of airworthiness, all the phases of flight are addressed, and it focuses on the aircraft's ability to safely fly in these phases. The three phases of flight are to obtain (takeoff), sustain (fly), and terminate (land) flight (Department of Defense, 2014). These can be summarized as Safety of Flight (SoF), which is defined as "the property of an air system configuration to safely attain, sustain, and terminate flight within prescribed and accepted limits for injury/death to personnel and damage to equipment, property and/or environment" (Burke, Hall, & Cook, 2011). When combining the DoD definition with the definition of SoF, they cover all aspects of flight. The DoD airworthiness definition covers aspects corresponding to the safe operation of the aircraft. SoF covers the safety concerns that could occur as a result of aircraft operation.

As mentioned above, airworthiness is often tailored to the domain. When tailoring airworthiness, the criterion is chosen based on the application and configuration of the aircraft, a standard is assigned to each criterion, and a method of verification is assigned for each standard. The DoD airworthiness requirements described in MIL-HDBK-516C are usually less restrictive compared to civilian requirements for the purpose of incorporating new advances in aircraft technology. In addition, DoD aircraft operate in environments that are not common for civilian aircraft as a result of DoD missions (Burke, Hall, & Cook, 2011).

However, there are aircraft that are used in both the civilian domain as well as the DoD such as the Dolphin helicopter, the KC-46, and the B-737. These aircrafts are adapted and utilized differently based on their domain and their application within their domain. For each aircraft, in a specified domain with a specified application, there are prescribed standards which may or may not be identical or overlap with the same aircraft in a different domain or with a different application. The standard must be tailored to the specific aircraft application and domain. There is a prescribed method of compliance for the established standard and each aircraft has a tailored set of requirements that must be verified and satisfied to accomplish airworthiness certification.

Model Based Systems Engineering and Systems Modeling Language

Model-Based Systems Engineering (MBSE) is “the formalized application of modeling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases” (INCOSE, 2015). System modeling propels systems engineering efforts throughout the development, project, and acquisition life cycles. A model-based approach to Systems Engineering supports interdisciplinary analysis, design space optimization, requirements and architecture solutions. In addition, a model-based approach transitions Systems Engineering processes and management processes from a paper-based documentation to a paper-less method by allowing the system design and performance to be represented and reviewed in a digital form (Holt & Perry, 2018).

As an emerging approach to Systems Engineering, MBSE provides advantages to the field in communication between system development team members and system stakeholders, improved information capture, and easier traceability. Instead of managing documents for a system, the management is accomplished by controlling and updating the model of system. The system model is developed in a modeling language, such as Systems Modeling Language (SysML), and available within a modeling tool or program, such as Cameo (Ramos, Ferreira, & Barceló, 2012).

The model is represented on graphical diagrams and tables with modeling elements and contained in an integrated model repository. The model repository ensures that all parties are working and collaborating on the most up-to-date and accurate system. The system is created with modeling elements that are connected to represent structure, behavior, parametric, and requirements of the system. The model can be integrated with external engineering tools for simulation and analysis (Ramos, Ferreira, & Barceló, 2012).

MBSE and SysML feature the ability to input requirements into the system model and display the requirements with their relationships in a diagram or table. There are two properties that are common for every requirement, the ‘id’ and the ‘text’. The ‘id’ is an identifier for the requirement and the ‘text’ is used for describing the requirement. A requirement can be nested under a higher-level requirement. Thus, a requirement can have multiple sub-requirements. A requirement diagram allows a requirement to be related to system model elements that satisfy or verify a specific requirement (Holt & Perry, 2018). The relationships and model elements found in requirement diagrams can be found graphically in Figure 4.

- “Satisfy relationship. This is used to show that a model element satisfies a requirement. It is used to relate elements of a design or implementation model to the Requirements that those elements are intended to satisfy.
- Verify relationship. This is used to show that a particular test case verifies a given requirement and so can only be used to relate a test case and a requirement” (Holt & Perry, 2018).

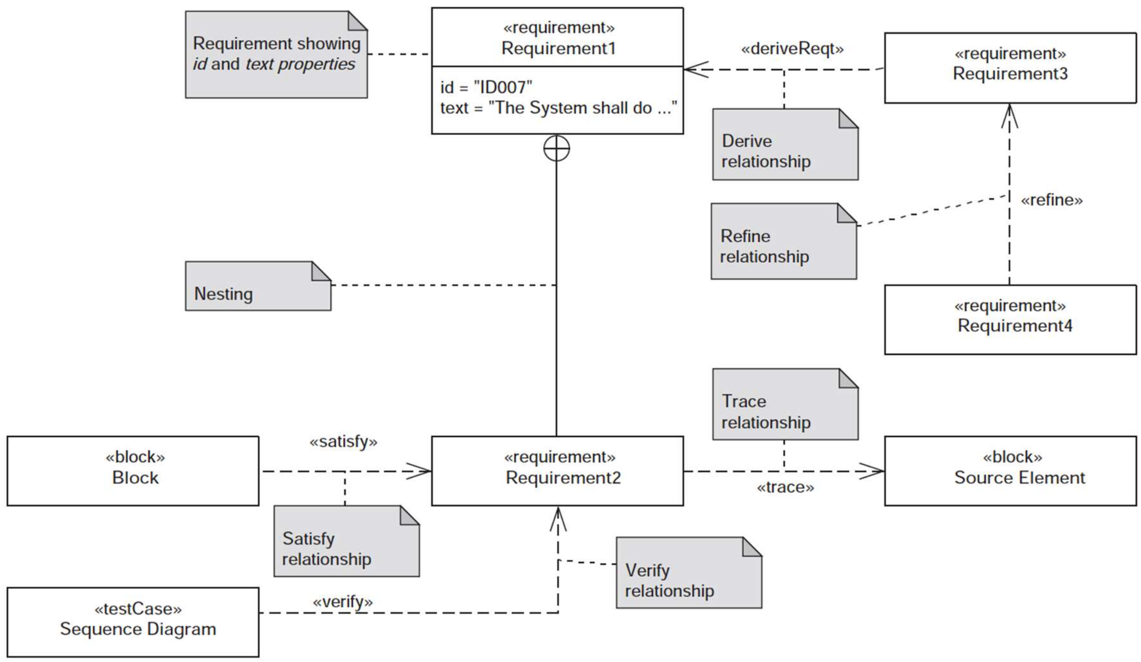


Figure 4: Illustration of the elements and relationships found in a requirement diagram (Holt & Perry, 2018)

The system model is the primary artifact of MBSE and any changes to the system requirements or design are reflected in the model. When a change is made, the change is propagated throughout the entire model. “The system model provides a consistent source of the system specification, design, analysis, and verification information, while maintaining traceability and rationale for key decisions. The information provides a context and critical input for more detailed hardware and software design and verification activities, which may also be model-based. In particular, the system model relates the text requirements to the system design, provides the system design information needed to support multi-disciplinary analysis, serves as a specification for the hardware and software design, and provides the test cases and related information needed to support verification” (Friedenthal, Moore, & Steiner, 2015). Figure 5 illustrates a SysML architecture for the purpose of combining a system model with design, analysis, and traceability utilizing SysML diagrams and elements.

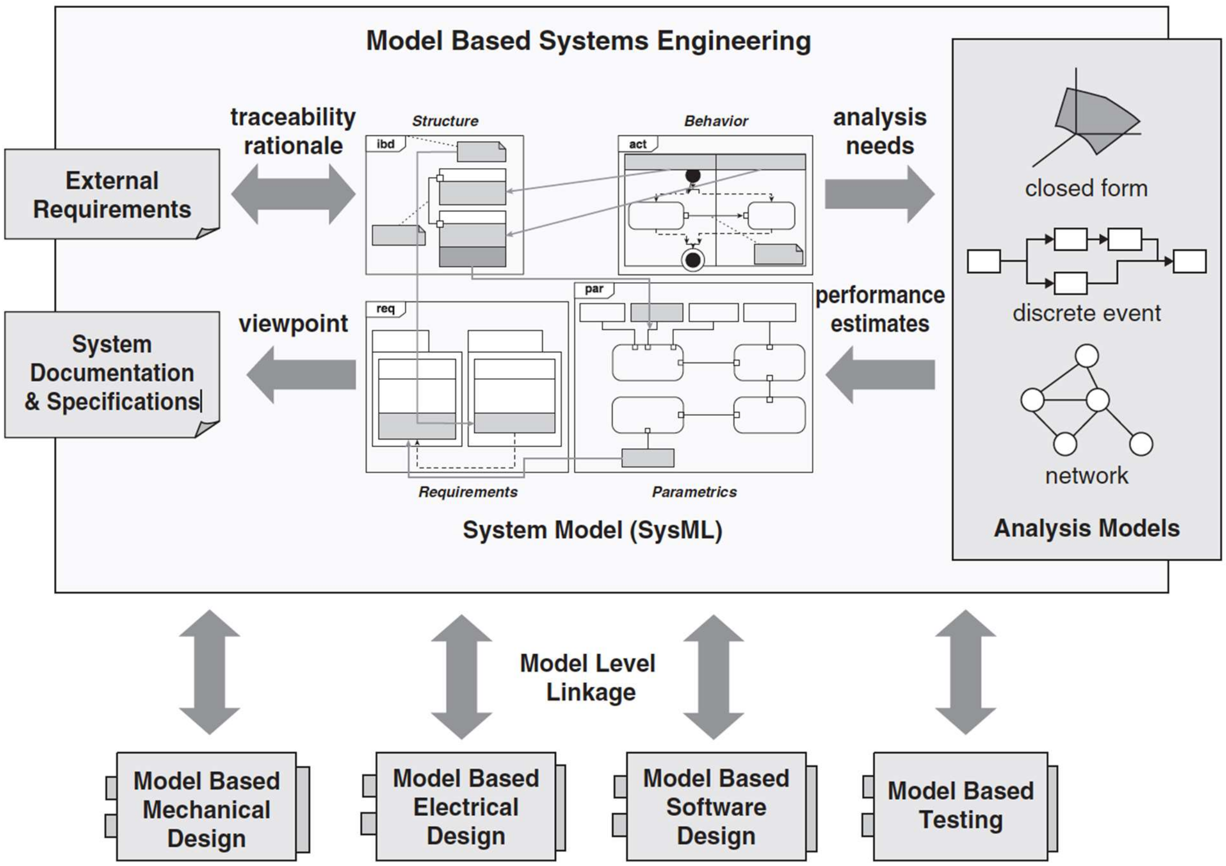


Figure 5: Architecture illustrating how the system model is related to exterior capabilities for design, simulation, and analysis (Friedenthal, Moore, & Steiner, 2015).

In order to construct an MSBE architecture specific to a helicopter system, a review of accepted modeling practices of the sub-systems of the helicopter is required. Raymond Prouty provides a breakdown of a military helicopter system into sub-systems and components of each system. The sub-systems are the powertrain, rotor, fuselage, control system, air speed system, and emergency system. Each sub-system contains components that perform a specific role to ensure the safe operation of the overall helicopter system. The helicopter features a breakdown of the system with sub-systems and components as described in Table 1 below:

Table 1: Hierarchical Breakdown of a Helicopter System (Prouty, 1998)

System	Sub-System	Component	
Helicopter System	Powertrain	Motor	Gas Generator
			Compression Stage
			Combustion Stage
			Free Turbine
		Fuel System	Fuel Regulation System
		Transmission	Drive Shaft
		Safety System	
	Rotor	Main Rotor	
		Tail Rotor	
	Fuselage	Integral Seat	
		Landing Gear	
		Static Mast	
		Door(s)	
	Control System	Collective System	
		Cyclic System	
		Hydraulic System	
		Side Arm Controller	
		Stability and Control Augmentation System	
		Trim System	
	Air Speed System	Pitot Tube	
		Static Port	
		Pressure Cell	
	Emergency System	Armor	
		Deicing System	
		Escape System	

MBSE for Simulation and Analysis

MBSE has been leveraged for simulation and analysis of systems to assist in development and sustainability life cycles. A simulation is a method of representing a system response as a function of time and space, but it can take many forms. In MBSE, simulations often include a dynamic model of the system, the environment it belongs to, the conditions, and the external inputs into the system. The simulation is performed by an executable model, which is a dynamic model that is expressed and executed by a specific environment. One classification of a simulation is the system performance simulation, which gives the ability to perform analysis of the behavior, resource usage, and other physical-based aspects of the system. Tools for data-analysis, visualization, and animation can be leveraged to display the results of the simulation (Friedenthal, Moore, & Steiner, 2015).

Combination of MBSE with Test and Evaluation

MBSE has the potential to revolutionize the process by which organizations such as the DoD perform test and evaluation. However, according to International Council on Systems Engineering (INCOSE) member William D. Miller, there has been a disconnect between Systems Engineering and MBSE. He argues that system architectures are not being created for ease of integration and testability. There must be communication between systems engineering and test and evaluation to enhance the development of the system architecture, which will result in a model that is useable across a system's life cycle (Miller, 2017).

The utilization of MBSE and SysML has enhanced the development of complex aerospace systems, especially the acquisition of such systems. The test and evaluation community has a need for test requirements, objectives, and assets with collaboration throughout the test domain. There have been increased efforts in developing MBSE architectures for test and evaluation purposes. Alvidrez created a simple integrated model using Department of Defense Architecture Framework (DoDAF) as an architecture framework combined with MBSE as the methodology to demonstrate early indemnification of test requirements, tasks, assets, and collaboration throughout the test process (Alvidrez, 2012).

An MBSE approach has found use in airworthiness certification plan management. “The model-based certification plan management is an approach to streamline the certification planning process by taking advantage of Model-Based Systems Engineering (MBSE) techniques” (Bleu-Laine, Bendarkar, Xie, Briceno, & Mavris, 2019). The airworthiness certification process is a prescribed systems engineering process, identifying requirements, choosing a means of verification, and the generation of proof for verification. An MBSE approach to airworthiness certification allows a system model to facilitate the most up-to-date requirements and analysis. A system model for airworthiness certification was architected to incorporate regulatory documents such as Federal Aviation Regulations and American Society for Testing and Materials requirements, aircraft-specific information. The model leverages SysML models for a certification basis and plan. Figure 6 shows an overview for a model-based process for airworthiness certification plan management (Bleu-Laine, Bendarkar, Xie, Briceno, & Mavris, 2019).

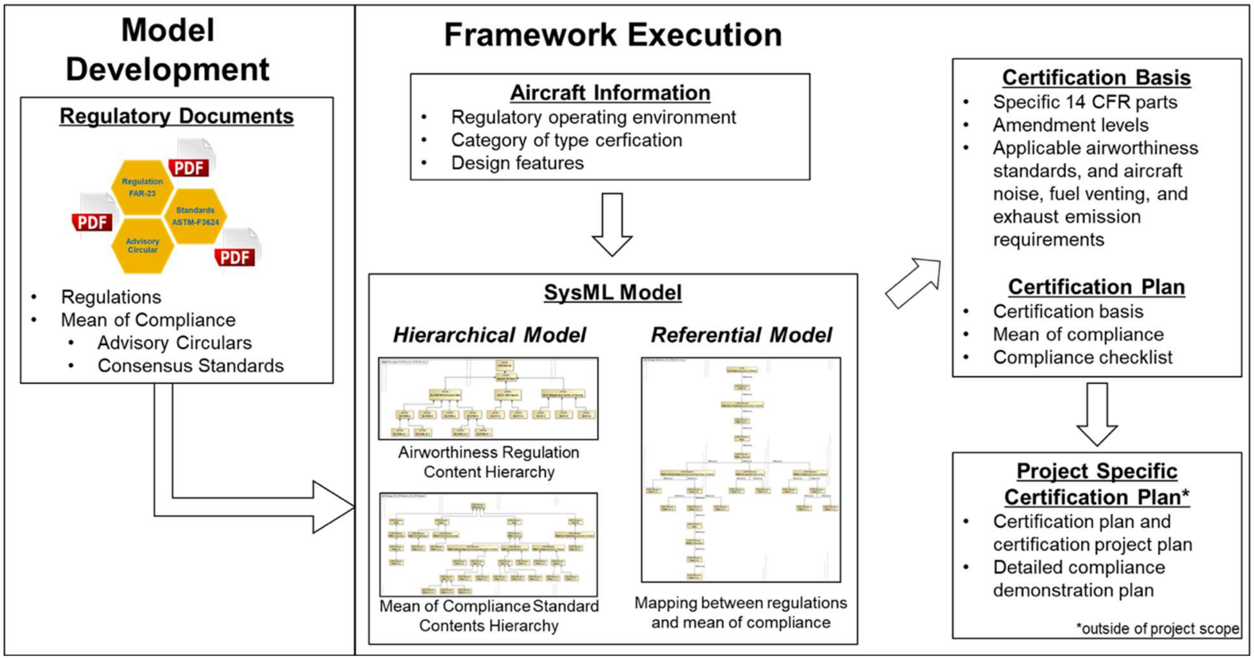


Figure 6: Model-Based Process for Certification Plan Management (Bleu-Laine, Bendarkar, Xie, Briceno, & Mavris, 2019).

There are recent advances in completing airworthiness certifications applied to Unmanned Aerial Systems (UAS). A System-Level Airworthiness Tool (SLAT) is a point-based tool that evaluates small UAS engineering practices in design, analysis, and testing. SLAT has been developed to assist airworthiness certifying authorities determine requirements, evaluate risk, leverage qualitative methods via safety tools, and provide verification and validation. This tool was created with the objective to provide help to those certifying authorities and aircraft manufactures that may not have experience in airworthiness certifications. SLAT is “a systems engineering framework for certifying small unmanned aircraft systems at the system level” (Burke, Hall, & Cook, 2011). Although the development of this tool is for the purpose of a small UAS, a similar tool can be developed for full scale, manned or unmanned, aircraft airworthiness at the system level. A tool for full scale aircraft can improve the airworthiness process.

Summary

Concluding this chapter, the reader should have an understanding of all the concepts used throughout this research. A fundamental review of airworthiness, both in the civilian domain and in the military domain. Next, an introduction to Model-Based Systems Engineering (MBSE) and Systems Modeling Language (SysML) with an emphasis on the capabilities that will be used in this research. Exploration of MBSE for simulation of design and analysis of performance of the system. Finally, a discussion of recent applications of MBSE within a test and evaluation domain. A literature review of these topics provides a foundation necessary to develop the methodology of Chapter III.

III. Methodology

Chapter Overview

This chapter describes the foundation of methods used in this research to develop a reference architecture and discusses the organization of the model to perform the airworthiness process for verifying and satisfying hover and take-off requirements of a helicopter. The main focus was on the implementation of a Model Based Systems Engineering (MBSE) approach to airworthiness, specifically how MBSE approach can be used to create a system model, store airworthiness requirements and flight test data, perform analysis, and verify and satisfy airworthiness requirements. This methodology was developed with SysML and with Cameo Systems Modeler as the tool. The airworthiness model consisted of individual packages that contained information, blocks, diagrams, tables, and matrices used to complete the airworthiness process. These packages were requirements, analysis, helicopter system, test, and documents.

Described first in this methodology is the formation of a logical model of a helicopter system. Second, a description of importing and storing requirements, followed by the test scenarios used to collect data. Next, a description of how flight test data is stored in a MBSE reference architecture. Next, an outline of the analysis performed in the model with integration of a math engine for the computations and how the analysis is utilized to verify and satisfy airworthiness requirements. Finally, a full traceability view is created to display an overview of the airworthiness completion. This chapter concludes with a description of traceability and how to create traceability in MBSE. The traceability follows the process as seen in Figure 7, the airworthiness process loop used throughout the duration of this methodology.

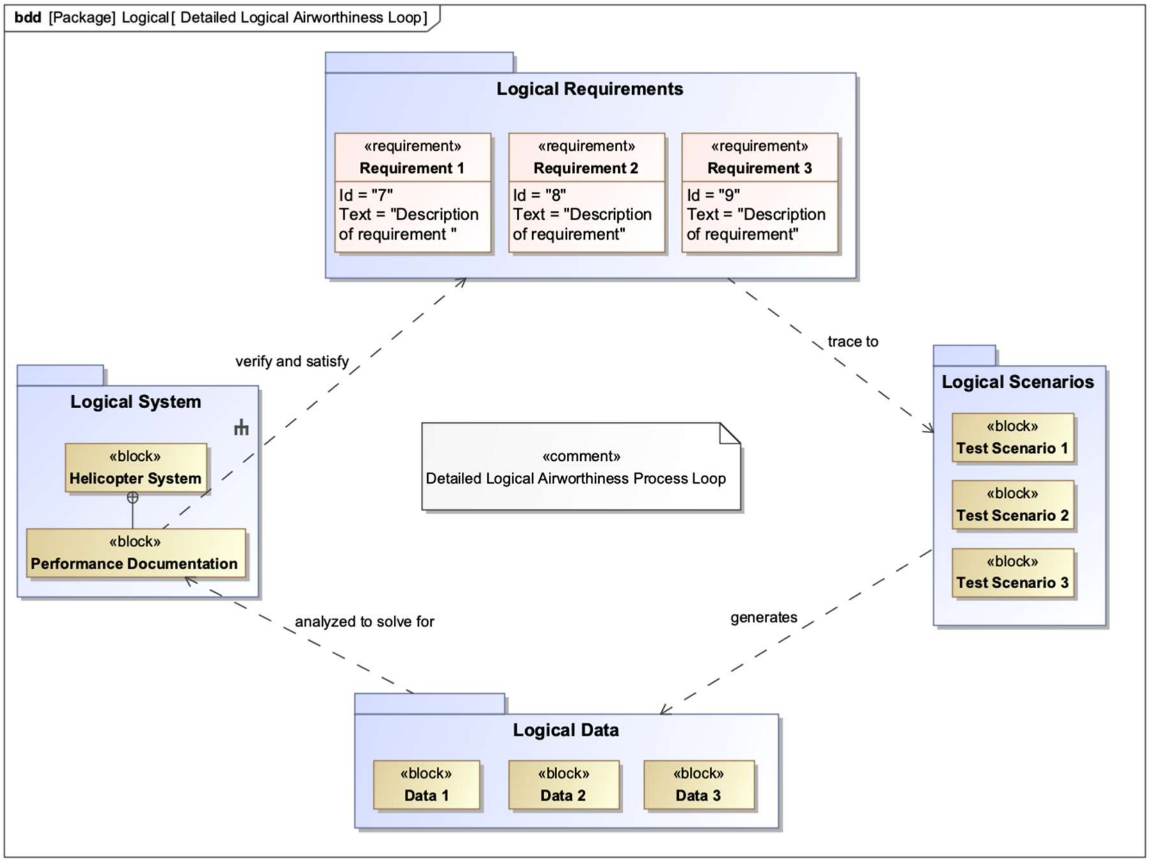


Figure 7: Detailed Logical Airworthiness Process Loop

Helicopter System Model

The usefulness of MBSE comes from the ability to create models of the particular area of interest. In this research, the area of interest is the ability to create models that accomplish the airworthiness process of a helicopter in hover and takeoff conditions. The initial step was to create a model of the helicopter system.

This is a hierarchy of the system components of the helicopter, where the helicopter is a complex system. A complex system is when an overall system, such as a helicopter, is comprised of components or other system but those components and other systems cannot perform their action by themselves. The components and other systems include the powertrain system, airspeed system, fuselage system, etc. Two types of models were created for the system: a logical system model and an instantiation of that logical system model. The logical system model is a general representation of the system being described.

Raymond Prouty, an authority in helicopter architectures, establishes a breakdown of the systems that make up helicopters. The breakdown of major systems is as follows: powertrain, fuselage, emergency system, control system, rotor, air speed system (Prouty, 1998). In Figure 8, the helicopter system is modeled in a block definition diagram based on the breakdown of the helicopter sub-systems identified by Prouty. A system performance block, named 'Performance Documentation', with in-ground effect, out-of-ground effect, and performance charts blocks were added to the overall system hierarchy to allow the performance to be stored and related to the overall system.

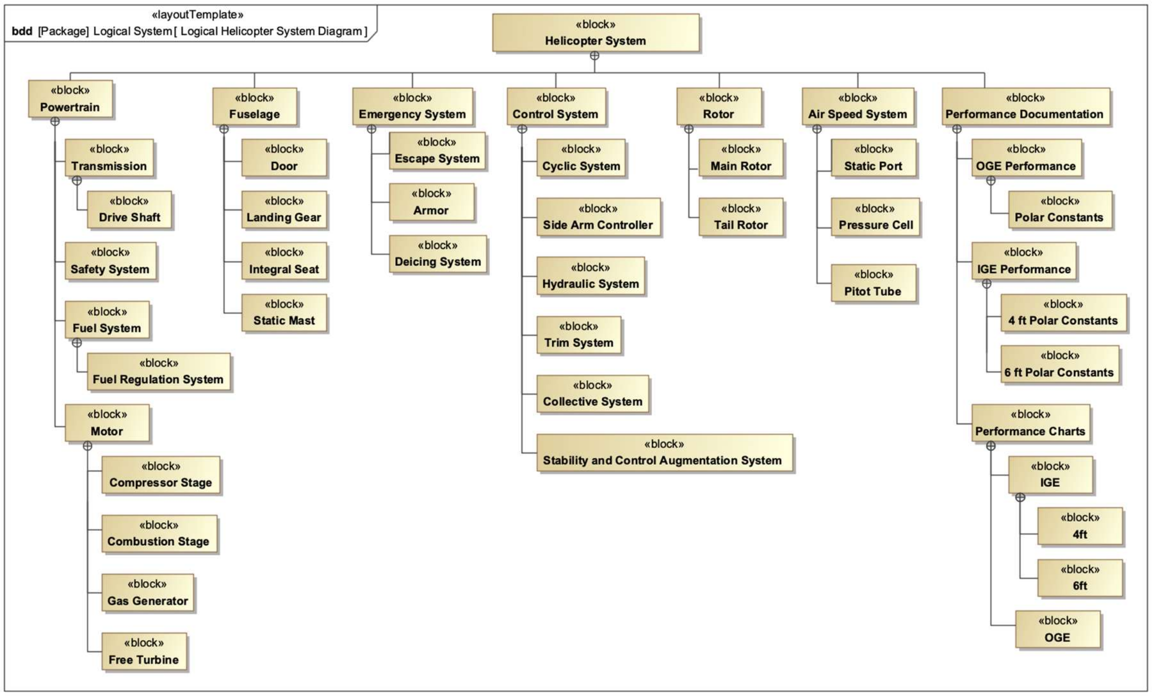


Figure 8: Logical Helicopter System Block Definition Diagram

The instantiation model is used to describe specific instances of the system, whether that be variations, modifications, upgrades, or configurations. This model provides the ability to describe a specific helicopter system with distinct value properties that are only associated with that system, where the logical system only shows general characteristics. There are aircraft that can be operated with a single motor or two motors depending on the variation of the model.

An instantiation model can be created with ease for each variation of the system, without having to change the original logical system. An instantiation model is created in MBSE by simply having a logical system model previously created, choosing the option to ‘create instance’, and selecting the system for which the instantiation is to be made.

Once the instantiation model is created, it can be modified to represent the specific system with unique value properties.

Airworthiness Requirements

The first step for introducing requirements into the model is to identify the airworthiness requirements necessary to achieve the airworthiness approval. The requirements are a negotiation between the AA and the program office. In some cases, there are requirements from multiple sources, such as FAA, FAR, and MIL-STD documents. Once the requirements are identified, they are entered into the MBSE model under the requirements package. Cameo, the MBSE software used for this research, offers a CSV file import option that can be utilized for importing requirements as long as the requirements file is saved in a CSV format. To utilize this feature, choose the import CSV option located under the ‘File’ tab at the top of the screen. This will open a window that will allow the user to open the CSV file in the computer directory. Next, the user can choose the location within the model where the requirements will be placed, called the ‘Target Package’. The user must also select the type of element and stereotype for the import, which for requirements, it will be ‘Class’ element type and ‘Requirement’ stereotype. Once all the required fields are selected for a successful import, the user will select the ‘Next’ option and continue to the second step of the import. An example of the first step can be found in Figure 9 for importing requirements from a CSV file and saving them in the model as requirements.

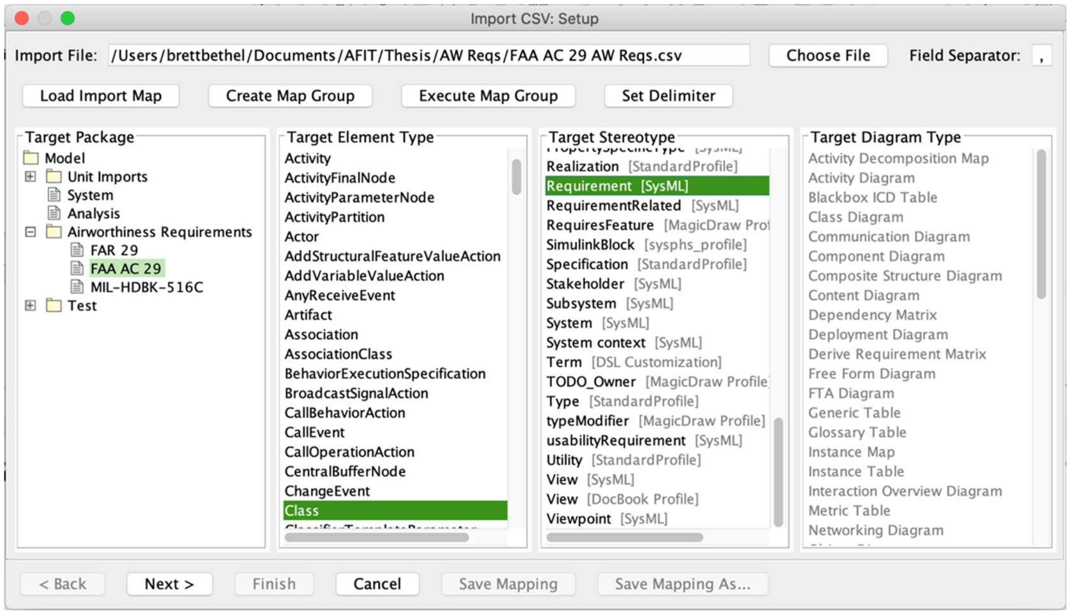


Figure 9: Import Window for Step 1

The second step for importing the requirements is to choose the property for each column in the import file. For this research, the requirements file had three columns. The first column was the requirement ID number, the second column was the name of the requirement, and the third column was the description of the requirement. The user must select the column of interest in the 'CSV Data' section of the window, then select the appropriate property in the 'Properties' list and click the 'Add' command to add the column to the property map. For this research, the ID column mapped to an ID property, the name column mapped to the Name property, and the description column mapped to the Text property. Once the mapping is complete, the user will finish the import by choosing the 'Finish' command. An example for the second step can be found in Figure 10.

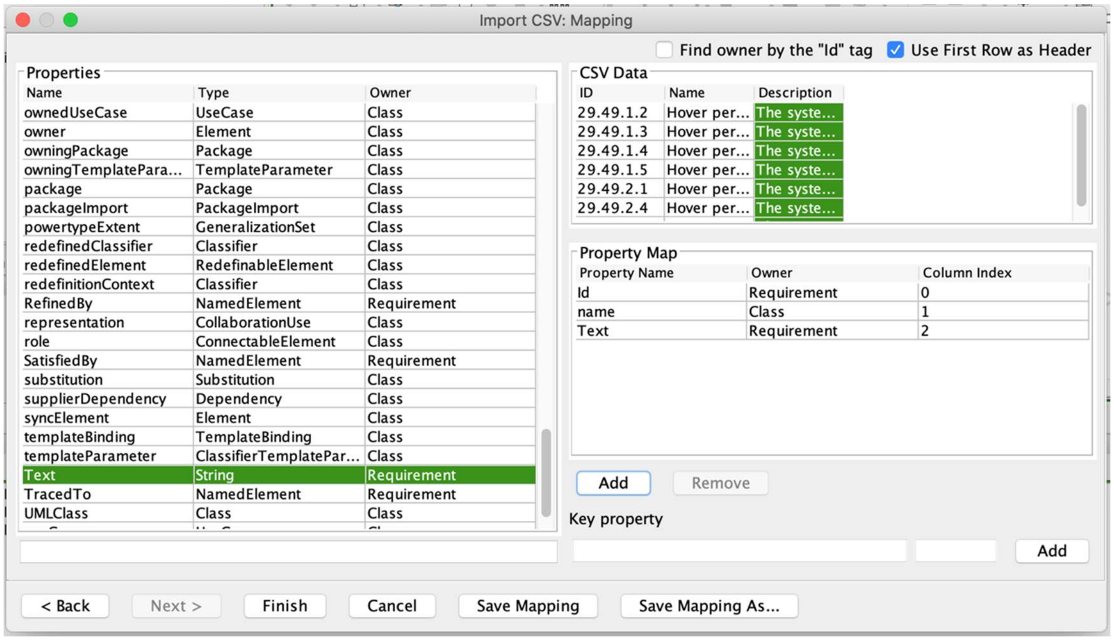


Figure 10: Import Window for Step 2

After the requirements are successfully imported into the model, a requirements diagram is created to illustrate the hierarchy and containment of all the requirements. This diagram can be scoped to the desire of the user. This allows for levels of abstraction for the user to only see the type of requirement of interest or to see all the requirements that pertain to the system. Once the requirements are in the model and the diagrams are created, the requirements are traced to the test scenarios.

Test Scenarios

Test scenarios are an essential aspect of airworthiness because they are used to plan the flight tests. The flight test team will choose test scenarios based on the airworthiness requirements that are being evaluated. In addition, there are test scenarios that are prescribed by certification standards. When reflecting this in MBSE, these test scenarios can be represented as SysML blocks. These blocks can have value properties to include additional information to further describe the test scenario.

A test scenario block should be created for each test scenario necessary to collect the data required. A block definition diagram (bdd) is created to show all the possible test scenarios and the hierarchy of the test scenarios related to one another. Another bdd is created to trace the airworthiness requirements to the test scenarios that will create the flight test data. This diagram is beneficial for the stakeholders to understand which the test scenario generated which test data. All the test scenarios and the diagrams are located under the 'Test' package and within the 'Test Scenarios' package. A package for all test related information, test scenarios and test data, allows for simplification of the containment tree.

Test Data

In the model, the test data is organized and stored under the 'Test' package and within the 'Test Data' package. The test data is represented by SysML blocks. The actual values of the test data are stored in text files. The test data blocks have value properties that are the file paths for each associated variable. The text files are then imported as an attachment to the corresponding value property. The complete data tables can also be stored in an attached file within the 'Test' package. The next step in the airworthiness loop is to utilize the parametric modeling ability in MBSE to analyze the test data.

Parametric Analysis

The analysis of the test data is accomplished by the parametric models in MBSE. This parametric modeling feature allows the user to create SysML constraint blocks that represent mathematical expressions. There are many options for the math engine to perform the computations. MATLAB was chosen as the math engine based on the previous experience with MATLAB and the nature of the analysis that was completed, which involved matrix solving. MATLAB was integrated with Cameo as the math engine to provide the ability to perform the analysis of the test data. The constraint blocks were used to represent MATLAB functions that performed the analysis. These MATLAB functions receive the test data, perform the analysis to produce the performance parameters, and send the parameters back into the parametric model.

The parametric analysis feature is in the form of a parametric diagram. The inputs and outputs of the MATLAB function are represented as value properties of blocks. The blocks are placed in the diagram and the value properties are displayed. The next step to set up the parametric diagram is to import the MATLAB function. This can be accomplished in many ways, but two methods were examined. The first way is to simply drag the MATLAB file from a file directory onto the diagram.

The second way is if the MATLAB function has already been imported as a constraint block for another diagram. This will only happen if the function is being reused. The MATLAB function will appear in the containment tree. The constraint block will be dragged and dropped from the containment tree onto the diagram. Once the constraint blocks, inputs, and outputs are placed on the diagram, the final step is to connect the elements. The ports on the constraint block are connected to the corresponding value properties. There is a parametric diagram for each analysis that is being performed.

To perform the analysis, a simulation is initiated on the block that owns the parametric diagram. Initiating the simulation will open a simulation window where the status and the results will be displayed. Once the simulation is complete, the results are found in the value properties of the blocks that were assigned when creating the parametric diagram and in the variables section of the simulation window. The results of the analysis can be found by expanding the block of interest to show the individual value properties that were assigned as outputs when creating the parametric diagram. These results can be saved as an instantiation of that specific block. This allows another simulation to be performed on the same parametric diagram without losing the previous results.

There are other parametric models that can be created for other characteristics of the aircraft such as a roll-up mass, roll-up cost, center of gravity calculation, structure analysis, etc. These other parametric models can be created with the same methodology as described above. All parametric models can be generalized to apply to all aircraft for any case that the analysis is valid. For example, the parametric models created for this research are associated with single rotor helicopter aircraft. If a parametric model is created for roll-up mass, it would apply to all aircraft regardless of classification or specification.

Requirements Satisfaction and Verification

The products generated as a result of the analysis of the test data were used to verify and satisfy requirements. First, an understanding of the difference between the verify relationship and the satisfy relationship must be reviewed. The verify relationship is used to show that a specific test case verifies a given requirement and so can only be used to relate a test case and a requirement. The satisfy relationship is used to show that a model element satisfies a requirement and is used to relate elements of a design or implementation model to the requirements that those elements are intended to satisfy (Holt & Perry, 2018).

For ease of completion, the optimal method of making the relationships, whether they be a verify relationship or a satisfy relationship, is to create a requirements matrix. There are two requirements matrices that are utilized, the requirements verification matrix and the requirements satisfaction matrix. One verification matrix and one satisfaction matrix were created for each set of requirements under the corresponding package in the model. Examples are described and displayed in the following paragraphs.

Once the matrices were created, the scope and element type for the row and column of the matrix were selected. For the verification matrices, the scope of the row was the requirements package of interest, the scope of the column was the test scenarios package, the element type of the row was 'AbstractRequirement', and the element type of the column was 'Block'. For the satisfaction matrices, the scope of the row was requirements package of interest, the scope of the column is the package where the blocks are located that will be involved with the relationship to the requirement (System, Analysis, Test, etc), the element type of the row was 'AbstractRequirement', and the element type of the column was 'Block'. After the matrix is properly configured, a double-click in the intersection of a requirement and block combination will create the relationship, either verify relationship or satisfy relationship depending on the type of matrix. An example of a logical satisfy matrix can be seen in Figure 11 and an example of a logical verify matrix can be seen in Figure 12.

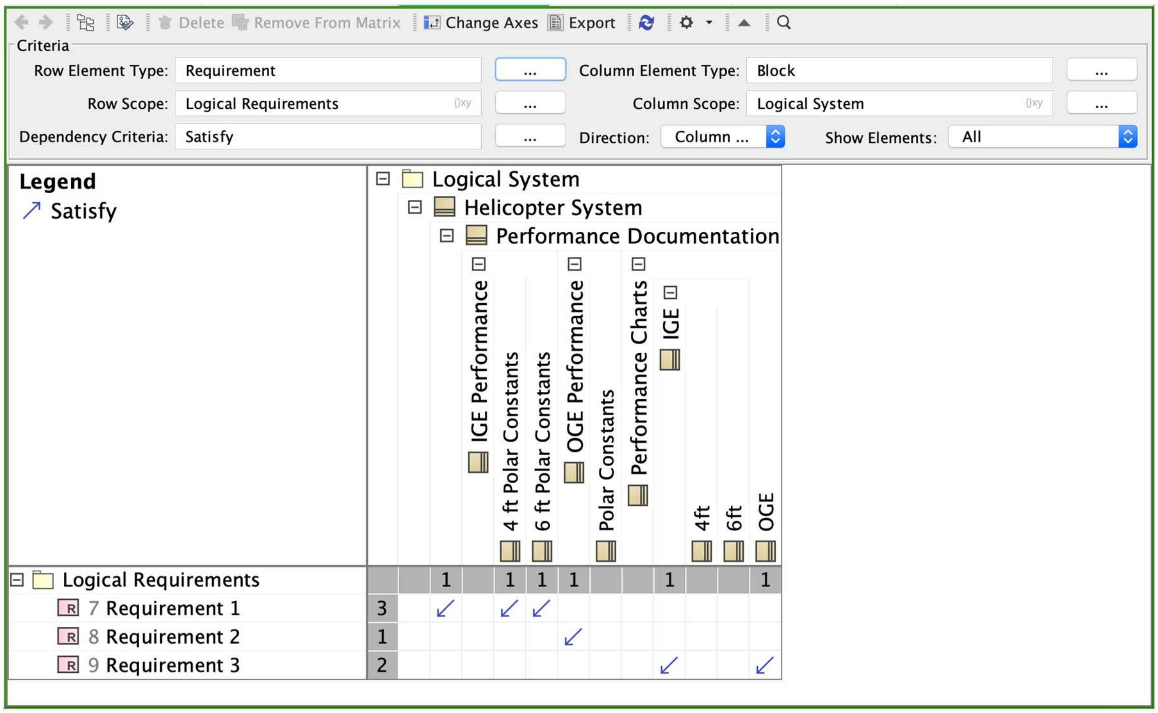


Figure 11: Logical Satisfy Matrix

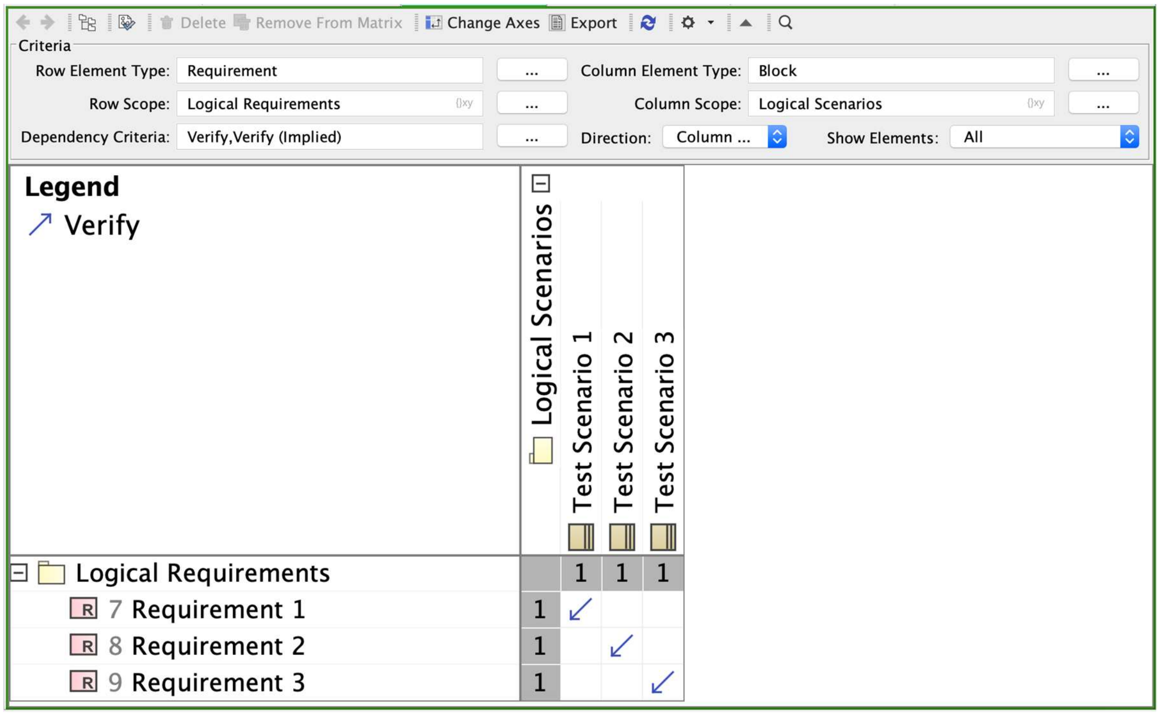


Figure 12: Logical Verify Matrix

There are two other methods of assigning and displaying the relationships between requirements and model elements, a requirements diagram and requirements table. First, the requirements diagram is the most visual representation of the relationship, which can be seen in Figure 13. This diagram is created by dragging the requirement of interest, in this case it was the logical requirements from the containment tree, onto the empty requirements diagram. Next, the incoming related elements must be displayed to show the contained requirements. To show the verify and satisfy relationships, a similar process is performed where the related elements are displayed for the requirements of interest. However, this is only applicable if the relationships already exist in the model. If the relationships do not exist yet, drag the elements that will satisfy or verify the requirements onto the diagram and assign the correct relationship by choosing the option from the 'Selection' window and under the 'Requirements' tab.

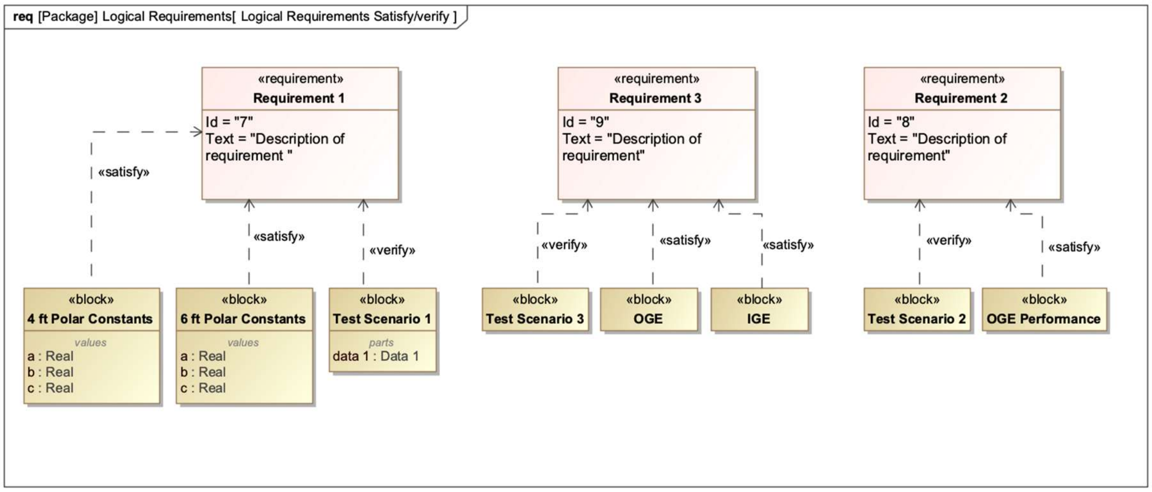
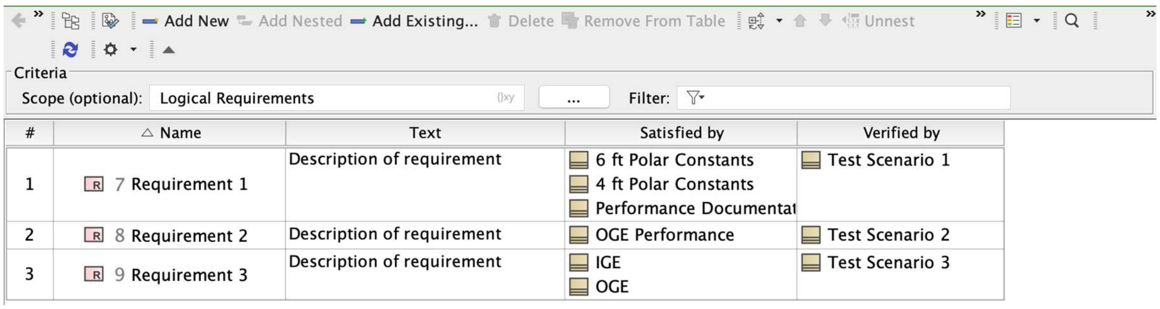


Figure 13: Logical Requirements Diagram with Satisfy/Verify Relationships

The final way to assign and display the relationships between requirements and the elements that verify or satisfy them is the requirements table. This table can be configured to show anything, but for this research it was configured to show the requirement ID, name, and text, as well as what elements verify and satisfy that specific requirement. This table is created by selecting the requirement table option when choosing to make a new diagram. The table will appear with the requirements as rows and standard columns: ID, name, and text. To add other columns of interest, such as the 'verified by' and 'satisfied by' columns, enter the specification menu for the table. Once the menu is open, click on custom columns and a new menu will appear. In that menu simply search for the 'verified by' and 'satisfied by' columns, select them, and add them to the table. If verify and satisfy relationships already exist in the model, they will appear without additional work. If they do not exist yet, double-click the 'verified by' or ‘satisfied by’ box in the requirement row of interest and click on the three dots option. This will open a menu to add the relationship by choosing the element in the model that completes the relationship. This table has the option to be exported into many other formats including an Excel spreadsheet. The requirements table for the logical requirements can be found in Figure 14.



#	Name	Text	Satisfied by	Verified by
1	7 Requirement 1	Description of requirement	6 ft Polar Constants 4 ft Polar Constants Performance Documentat	Test Scenario 1
2	8 Requirement 2	Description of requirement	OGE Performance	Test Scenario 2
3	9 Requirement 3	Description of requirement	IGE OGE	Test Scenario 3

Figure 14: Logical Requirements Table with Satisfy/Verify Relationships

These matrices, diagrams, and tables are useful for the airworthiness process. The airworthiness process has multiple aspects that are connected to one another and influence decisions about the safety and whether an aircraft can be operated within specified bounds. The matrices, diagrams, and tables show how the prescribed requirements are satisfied or verified by the model elements. The AA can see whether all the requirements are satisfied or verified and which specific model elements are satisfying or verifying those requirements to ensure the aircraft is safe to operate in the specified bounds.

Traceability

The final view that summarizes the efforts made in this research is a full traceability view that is shown with the use of a block definition diagram. This view displays how a requirement is traced to a test scenario, how the test scenario is traced to the test data, how the test data is traced to the analysis, how the analysis is traced to the performance of the system, and how the performance of the system is traced back to the requirement.

This diagram is developed by dragging on all the elements of the model that will be involved in the traceability. The final step is to connect the elements in order of the operation and feature the applicable stereotype. A diagram should be made for every requirement in the model. An example of a traceability diagram for 'Requirement 1' can be seen in Figure 15. This traceability view can be generalized for any airworthiness requirement as long as all aspects for tracing the requirement are modeled in the architecture.

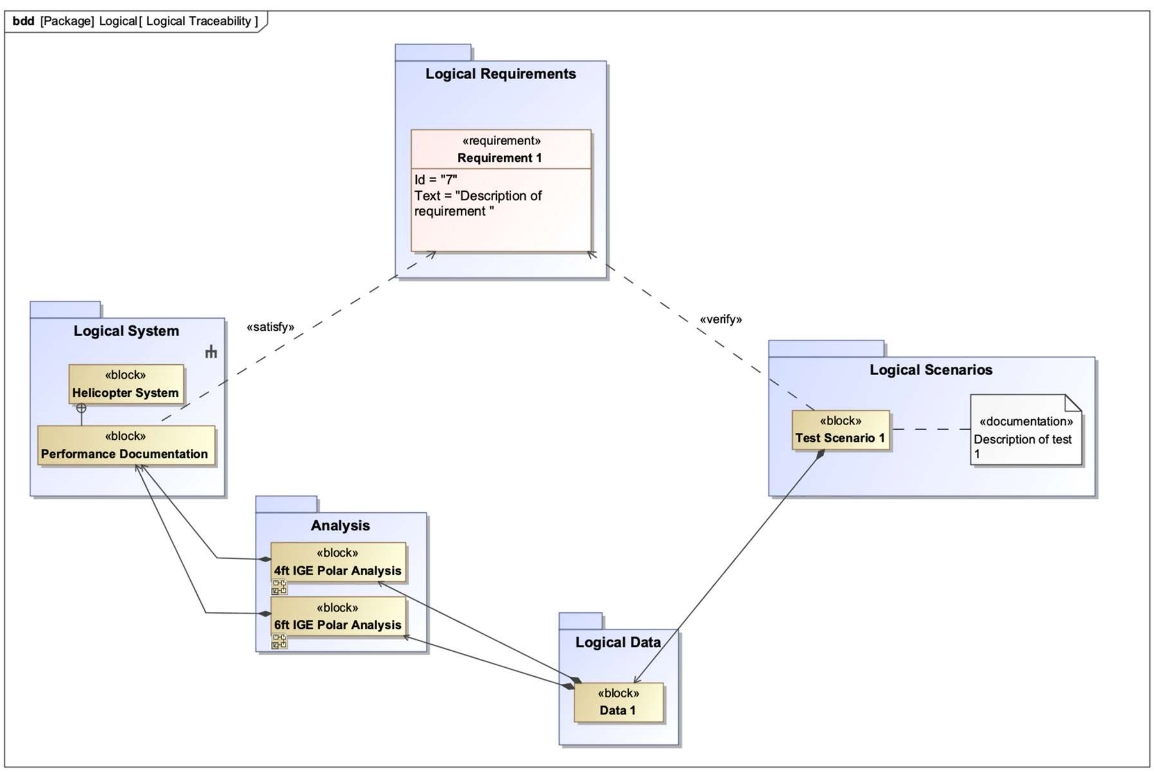


Figure 15: Logical Traceability Diagram

Summary

In summary, this chapter outlined the development of a logical helicopter model architected with a MBSE approach to demonstrate the airworthiness certification process. The development of the logical model followed the process loop described in Figure 7 and began with creating a model of the helicopter system, followed by a process description of the import and organization of the requirements. Next, the configuration of the test scenarios, followed by organization and storage of the test data. The formation of a parametric analysis model that is utilized to create satisfy and verify relationships between model elements and the airworthiness requirements. Finally, a description of how to create and display the traceability that is developed throughout the model. The methodology developed in this chapter provided the model that will be analyzed in Chapter IV.

IV. Analysis and Results

Chapter Overview

This chapter begins with a discussion of the results of implementing the system model, developed with the methodology outlined in Chapter III, for the Dolphin helicopter in hover and takeoff conditions. Following this, an interpretation of the results of the parametric analysis featured in the model will be presented. As outlined in Chapter III, the development of the model included airworthiness requirements identification, import, and storage, test scenarios storage and usage, test data storage, parametric diagrams, requirements satisfy and verify relationships, and traceability throughout the model. Following the discussion of the results of the modeling process for airworthiness of the Dolphin, a further description of the parametric analysis results and an interpretation of these parametric results is presented.

Results of Implementing the Model for Dolphin Airworthiness Certification

The MBSE approach to the development of a SysML model for airworthiness with integrated test data analysis was an issue that has not been solved. There were many questions to be answered through this model development including how to use test data for satisfy and verify relationships, how to organize test data in a model for efficiency, and how to architect a model for test data analysis. The overarching objective of this research was to architect a model that can be used to perform the airworthiness process loop. Thus, the development of this model was based on the airworthiness loop, which started with modeling requirements, then test scenarios and test data, then the analysis of the test data, ending with satisfying and verifying the requirements and displaying traceability throughout the loop. This airworthiness loop was implemented for a Dolphin helicopter in hover and takeoff conditions, as seen in Figure 16.

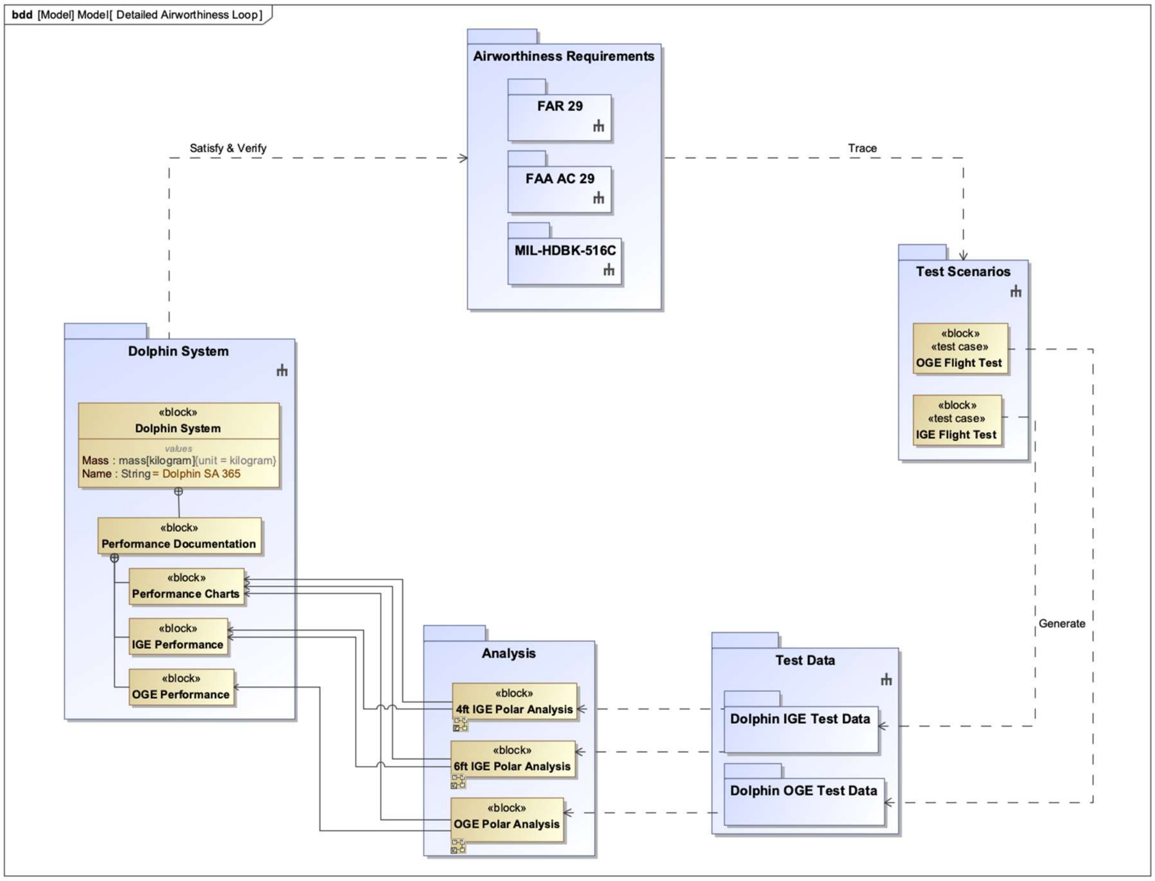


Figure 16: Detailed Airworthiness Loop for Dolphin - Hover and Takeoff

System Model Results

The modeling of the system is essential to any architecture regardless of application and the modeling of this helicopter system began as a logical system to allow for modification based on the specific helicopter being certified for airworthiness. In this research, the Dolphin was the specific helicopter to be certified. A clone of the logical system was created as an instance because when an instance is modified, it does not change the original logical system. This allows the logical system model to remain as a reference for future use. The clone system model was modified according to the unique characteristics of the Dolphin. This was accomplished by adding and editing the value properties of the system's component blocks. The Dolphin system model can be seen in Figure 17. Only the relevant modifications compared to the logical system model are displayed in the diagram.

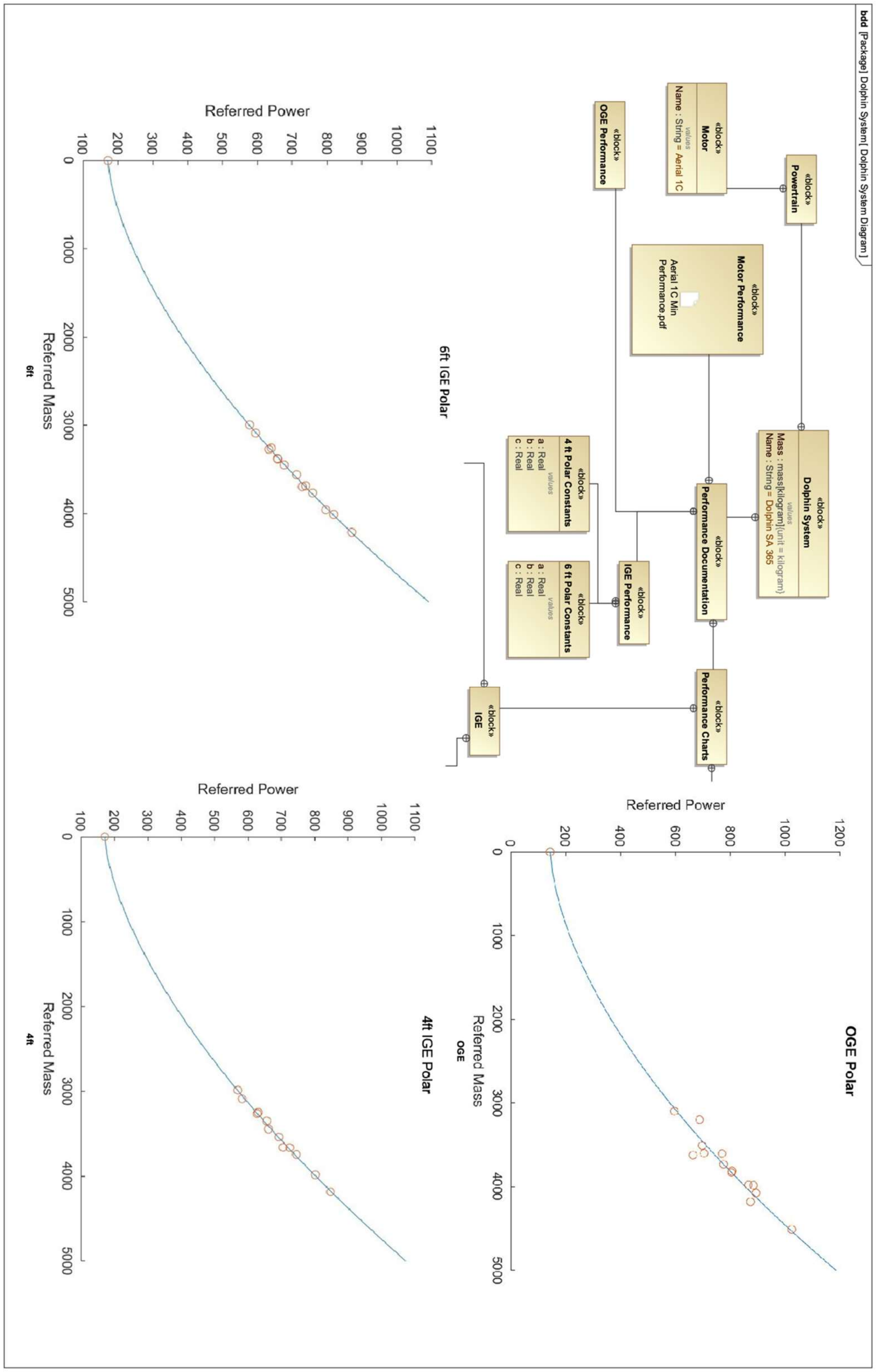


Figure 17: Dolphin Helicopter System Model Block Definition Diagram

Requirements Results

For this research, only the requirements for the hover and take-off performance were considered. The aircraft used for this research is the Dolphin, which is operated for both civilian and military use. This means that it must meet civilian and military airworthiness standards. The civilian airworthiness requirements come from the Federal Aviation Administration (FAA) and the Federal Aviation Regulations (FAR), while the military requirements come from Military Standard, Military Handbook, and Military Specification documents (MIL-STD, MIL-HDBK, MIL-SPEC). A complete list of requirements, both civilian and military, can be seen with full descriptions in Appendix A – MIL-HDBK-516C Requirements, Appendix B – FAR 29 Requirements, Appendix C – FAA AC 29 Requirements. Figure 18 illustrates the top-level requirements for each category: MIL-HDBK-516C, FAR 29, and FAA AC 29. Similar diagrams were created for each of the specific sets of requirements, FAR 29, FAA AC 29, and MIL-HDBK-516C, that show a lower level of abstraction for more detail on each of the sets. A hyperlink to these requirements diagrams can found inside the 'Airworthiness Requirements' diagram.

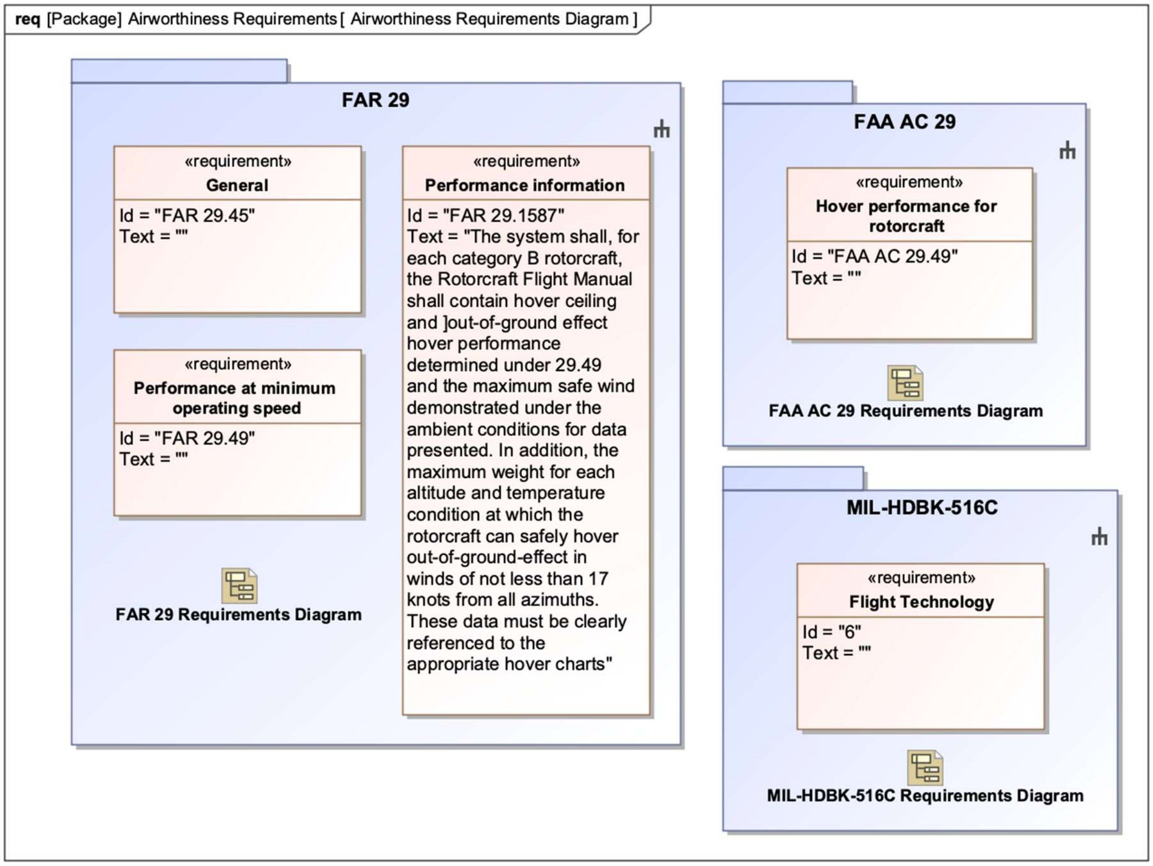


Figure 18: Requirements Diagram Featuring the Airworthiness Requirements with Each Package and Individual Requirements Diagrams

Specifically, for rotorcraft, or helicopters, in hover and takeoff conditions must satisfy and verify FAA AC 29 and FAR 29 requirements. These requirements cover the performance of the helicopter, specifically regarding power, speed, and environmental conditions. The FAA AC 29 requirements describe how the flight test should be conducted and how the data should be collected. The FAR 29 requirements describe airworthiness standards for certification of helicopters. The civilian requirements, both FAR 29 and FAA AC 29, related to the hover and take-off testing and system performance are listed below in Table 2 and Table 3 respectfully:

Table 2: FAA AC 29 Requirements

ID	Name	Description
29.49.1.2	IGE and OGE	The system shall determine hover performance at a height consistent with IGE and OGE for Category B rotorcraft
29.49.1.3	Power required	The system shall determine power required for hover at different gross weights, ambient temperatures, and pressure altitudes. Using non-dimensional power coefficients (Cp) and thrust coefficients for normalizing and presenting test results, a minimum amount of data are required to cover the rotorcraft's performance operating envelope
29.49.1.4	Conditions	The system shall be tested over a sufficient range of pressure altitudes and weights to cover the approved ranges of those variables for takeoff and landings. Additional data should be acquired during cold ambient temperatures, especially at high altitudes, to account for possible Mach effects
29.49.1.5	Height	The system shall prove that minimum hover height for which data should be obtained and subsequently presented in the flight manual should be the same height consistent with the minimum hover height demonstrated during the takeoff tests
29.49.2.1	Methods	The system shall be tested with two methods of acquiring hover performance data which are the tethered and free flight techniques
29.49.2.4	Techniques	The system shall if there are no provisions or equipment to conduct tethered hover tests, the free flight technique is also a valid method. The disadvantage of this technique as the primary source of data acquisition is that it is very time consuming. In addition, a certain element of safety is lost OGE in the event of emergency. The rotorcraft must be re-ballasted to different weights to allow the maximum Ct/Cp spread. When using the free flight technique, either as a primary data source or to substantiate the tethered technique, the considerations for wind, recorded parameters, etc., as used in the tethered technique apply. Free flight hover tests should be conducted at CG extremes to verify any CG effects. If the rotorcraft has any stability augmentation system which may influence hover performance, it must be accounted.

29.49.2.5	OGE tests	The system shall be tested in OGE. It is extremely difficult to determine when a rotorcraft is hovering OGE at high altitudes above ground level since there is no ground reference. In true hover, the rotorcraft will drift with the wind. Numerous techniques have been tried to allow OGE hover data acquisition at high altitudes, all of which have resulted in much data scatter. Until a method is proposed and found acceptable to the FAA/AUTHORITY, OGE hover data must be obtained at the various altitude sites where IGE hover data is obtained. Hover performance can usually be extrapolated up to a maximum of 4,000 feet

Table 3: FAR 29 Requirements

ID	Name	Description
29.45.1	Conditions	The system performance shall be determined with normal piloting skill and without exceptionally favorable conditions
29.45.2	Atmosphere	The system shall show compliance with the performance requirements for still air at sea level with a standard atmosphere and for the approved range of atmospheric variables
29.45.3	Power	The system shall prove the available power corresponds to engine power, not exceeding the approved power, less the installation losses and the power absorbed by the accessories and services at the values for which certification is requested and approved
29.45.4	Humidity	The system shall provide the performance, for turbine engine-powered rotorcraft, based on a relative humidity of 80% at and below standard temperature and 34% at and above standard temperature plus 50 degrees F
29.45.5	Takeoff	The system shall provide the pilot the ability to determine prior to takeoff that each turbine engine is capable of developing the power necessary to achieve the applicable rotorcraft performance
29.49.1	IGE Hover	The system shall determine, for each Category B helicopter, the hovering performance over the ranges of weight, altitude, and temperature for which certification is requested with takeoff power, the landing gear extended, and in ground effect at a height consistent with normal takeoff procedures
29.49.2	OGE Hover	The system shall determine the out-of-ground effect hovering performance over the ranges of weight, altitude, and temperature for which the certification is requested with takeoff power

29.1587	Performance information	The system shall, for each category B rotorcraft, the Rotorcraft Flight Manual shall contain hover ceiling and out-of-ground effect hover performance determined under 29.49 and the maximum safe wind demonstrated under the ambient conditions for data presented. In addition, the maximum weight for each altitude and temperature condition at which the rotorcraft can safely hover out-of-ground-effect in winds of not less than 17 knots from all azimuths. These data must be clearly referenced to the appropriate hover charts

For the military airworthiness requirements, the general aircraft requirements are outlined in the document, MIL-HDBK-516C. MIL-HDBK-516C is a document that describes airworthiness certification criteria for all military aircraft. The scope of this research will be narrowed down to "Flying qualities" and "Air vehicle aerodynamics and performance" of MIL-HDK-516C, which are Chapters 6.1 and 6.3 respectively (Department of Defense, 2014). In Chapter 6.1, requirements describe air vehicle configurations, modeling, simulation, and analysis tools, varies flying qualities such as launches and recoveries, vertical takeoff and landing, hover and the flight manual. Chapter 6.3 discusses requirements for performance information. Listed below, in Table 4, are the military requirements:

Table 4: MIL-HDBK-516C Requirements

ID	Name	Description
6.1.1.3.4	Determining air vehicle configurations	The system shall have all vehicle configurations define and assessed for safety of flight
6.1.1.5	Modeling, simulation, analysis tools and databases	The system shall be verified that all modeling, simulation, analysis tools and databases are of appropriate fidelity and accurately represent the air vehicle for evaluating airworthiness criteria and safety of flight
6.1.10.7	Launches and recoveries	The system shall be verified that launches and recoveries from any approved spot are safe

6.1.11.1	V/STOL operations	The system shall be verified that V/STOL operations are safe
6.1.11.1.2	Vertical takeoff	The system shall be verified that VTO is safe
6.1.11.1.5	Hover	The system shall be verified that V/STOL hover is safe
6.1.11.2.1	Flying qualities in hovering flight	The system shall be verified that V/STOL flying qualities in hover are safe
6.1.15	Manuals	The system shall be verified that the Flight, Performance, and Operations Manuals, and any supplements, contain the air vehicle's operating limits and instructions (e.g., Cautions, Warnings, Advisories, Notes, Corrective Actions, etc.) to assure flight safety for all conditions, configurations, loadouts, etc
6.3.2	Performance information	The system shall be verified that the air vehicle performance information and flight limits are provided to the pilot/operator is accurate to ensure safe flight

Test Scenarios Results

For this research, there were three main test scenarios included in the model. The first was an In-Ground Effect (IGE) test scenario, the second was an Out-of-Ground Effect (OGE) test scenarios, and the third was a motor bench test scenario. For the IGE test scenario, there were two test scenarios that fall under that specific scenario. These two tests are the four-foot and six-foot hover test scenarios because the helicopter will behave differently at these different heights above the ground. The test scenarios for this research are displayed in Figure 19.

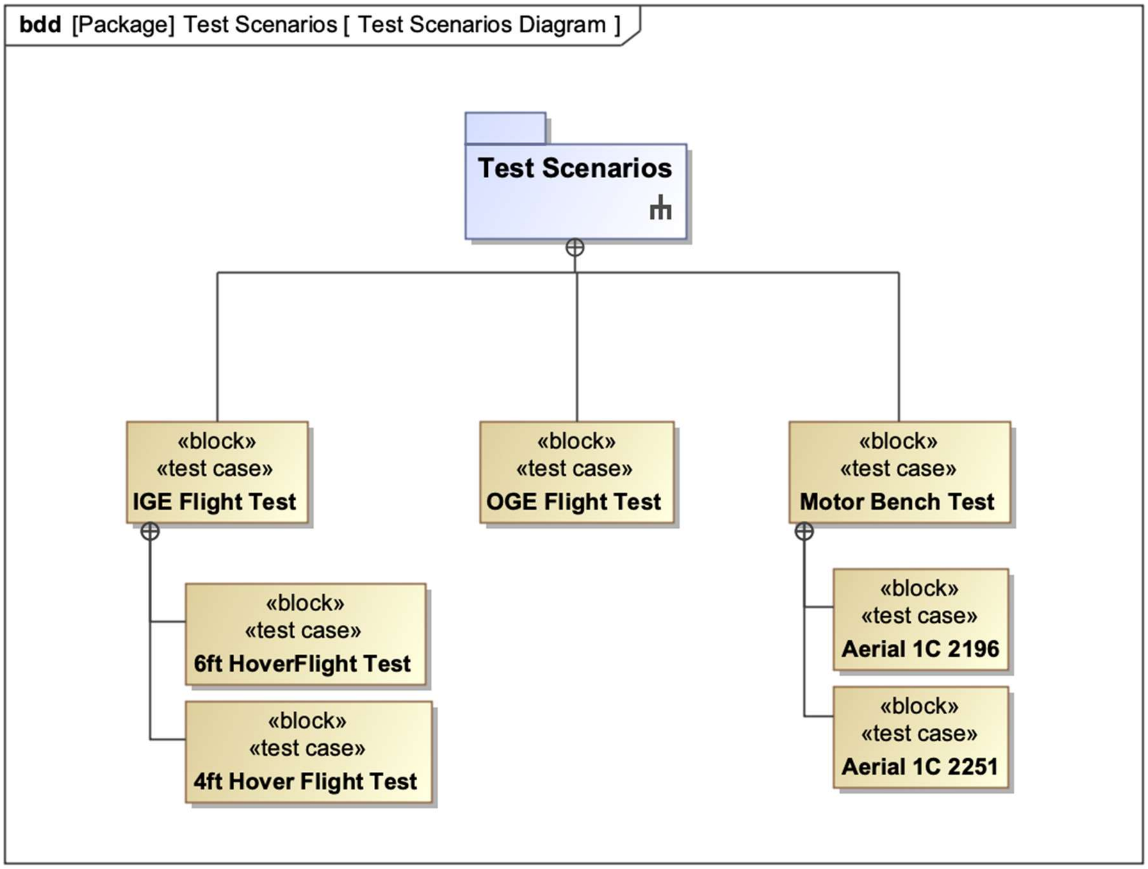


Figure 19: Test Scenarios Block Definition Diagram

Test Data Results

This research utilizes test data from the manufacturer of the motor and flight test data collected by students at the French test pilot school. Both sets of data are recorded in a document labeled "Performances en vol Stationnaire du Dauphin SA 365 N", which translates to Dolphin SA 365 N Hover Performance. The motor data provided by the manufacturer is bench data collected before the motor had been installed in any aircraft; thus, avoiding any installation losses. The bench data provides motor limits and a range of motor performance to be expected. The objective of the flight tests was to observe two separate phenomena that the helicopter experiences while taking-off and while hovering, the in-ground effect and the out-of-ground effect (Cox & Tortel).

The FAA defines ground effect as “the increased efficiency of the rotor disk caused by interference of the airflow when near the ground. The air pressure or density is increased, which acts to decrease the downward velocity of air” (Federal Aviation Administration, 2019). The flight test scenarios occurred at four-foot IGE, six-foot IGE, and OGE. The data from the manufacturer and the flight test data collected for both in-ground and out-of-ground effects can be found in Appendix D – Dolphin Flight Test Data. In Figure 20, the test data blocks with the value properties can be seen. Figure 21 shows the expanded view of the test data package in the containment tree of the model.

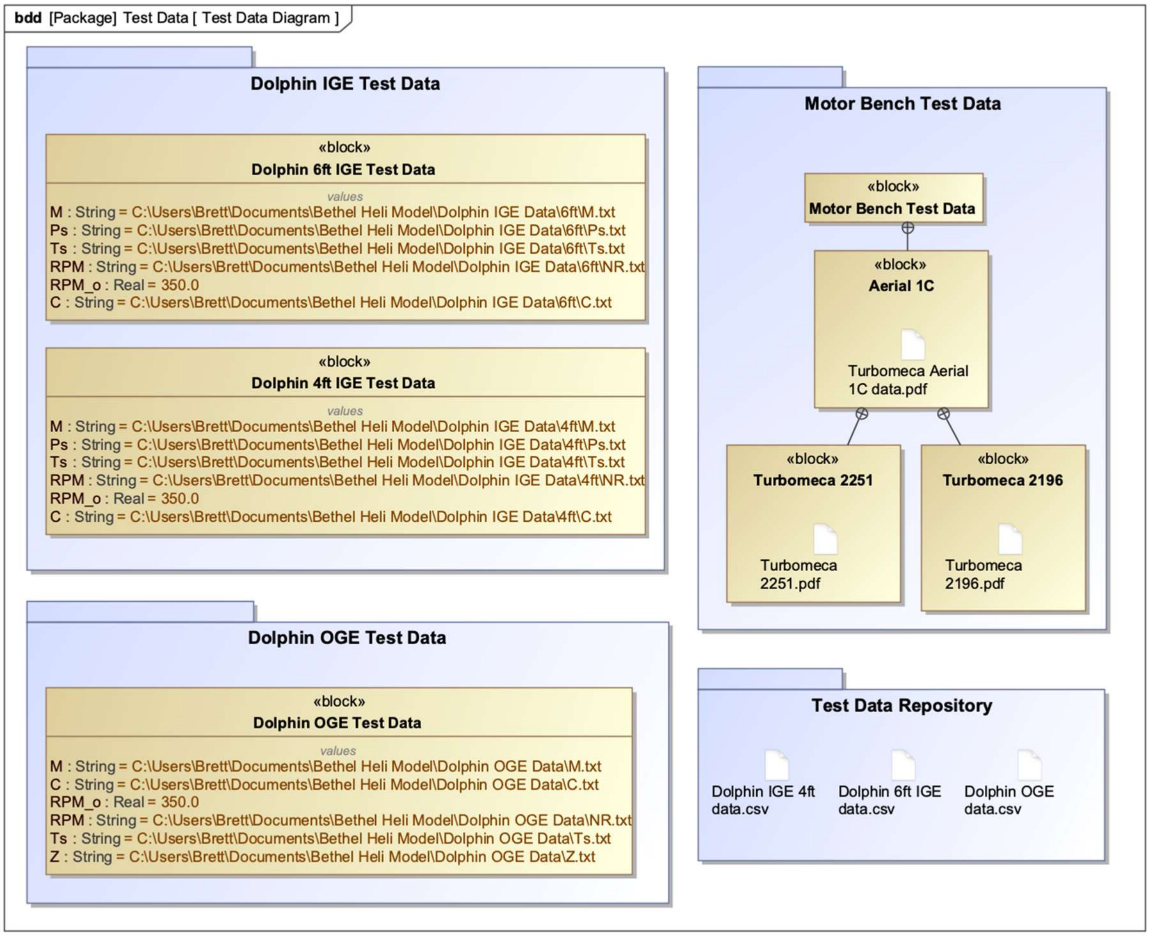


Figure 20: Block Definition Diagram of the Test Data Package

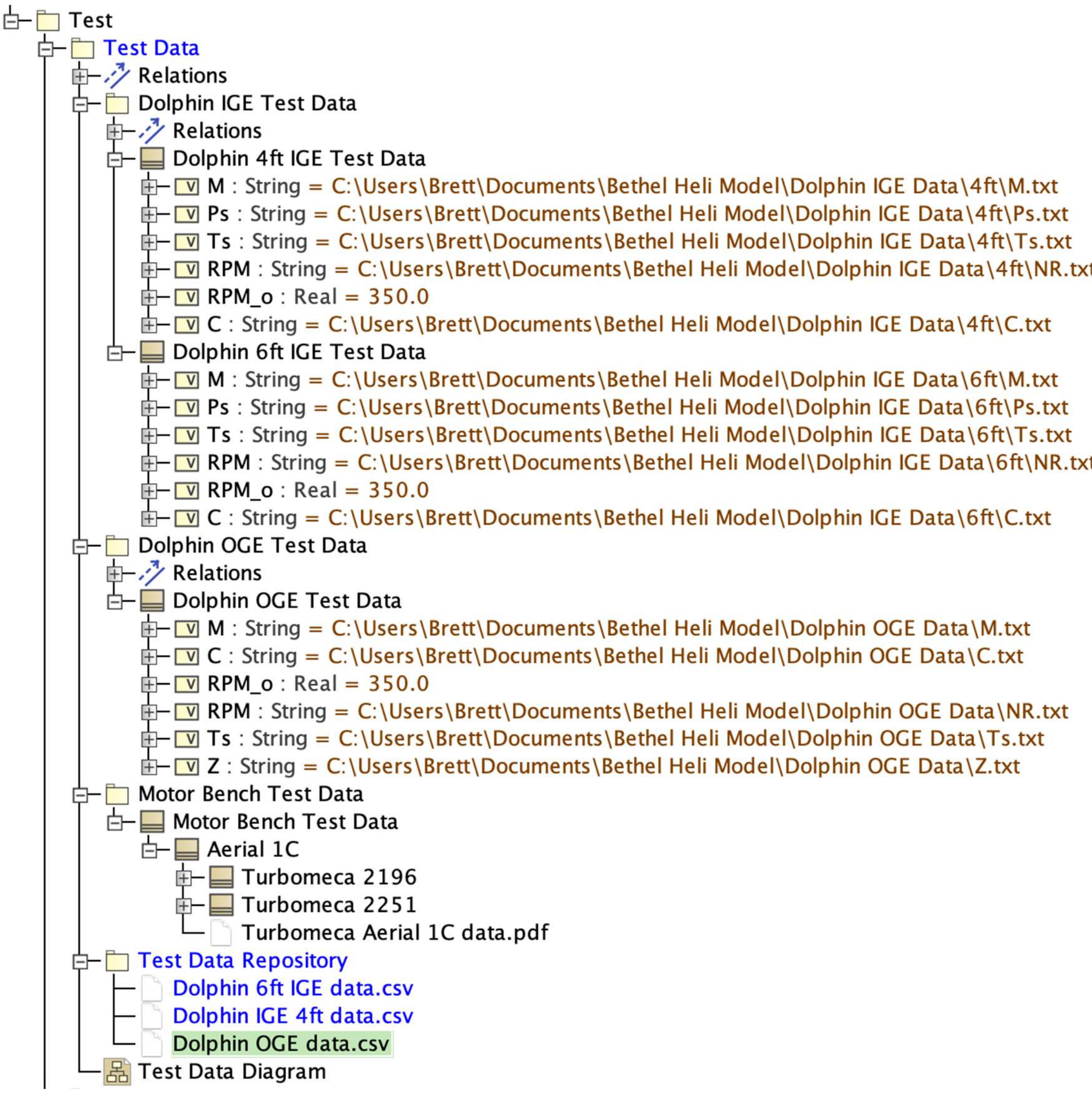


Figure 21: Containment Tree Expanded View of the Test Data Package

Parametric Analysis Results

The analysis of the test data for this research is to produce performance parameters and performance charts. The parametric analysis is a two-phase analysis. First, was the computation of the polar constants and the polar chart. The polar constants that are specific to each test scenario, four-foot IGE, six-foot IGE, and OGE. The polar constants are used to create performance charts (Cox & Tortel). Second, the polar constants were extrapolated and interpolated to create the predictive charts for hover performance. This analysis is specific to hover power required. This was one application for hover performance analysis. There are similar applications where test data is analyzed to create system performance parameters and those parameters are used to generate performance charts. The MATLAB code that was represented in the constraint blocks for the parametric analysis can be found in Appendix F – MATLAB Code: IGE Hover, Appendix G – MATLAB Code: OGE Hover, and Appendix H – MATLAB Code: Hover Power. For this research, a simulation was performed on each of the three parametric diagrams created; four-foot IGE, six-foot IGE, and OGE. Figure 22 shows the parametric diagram for the four-foot IGE polar analysis.

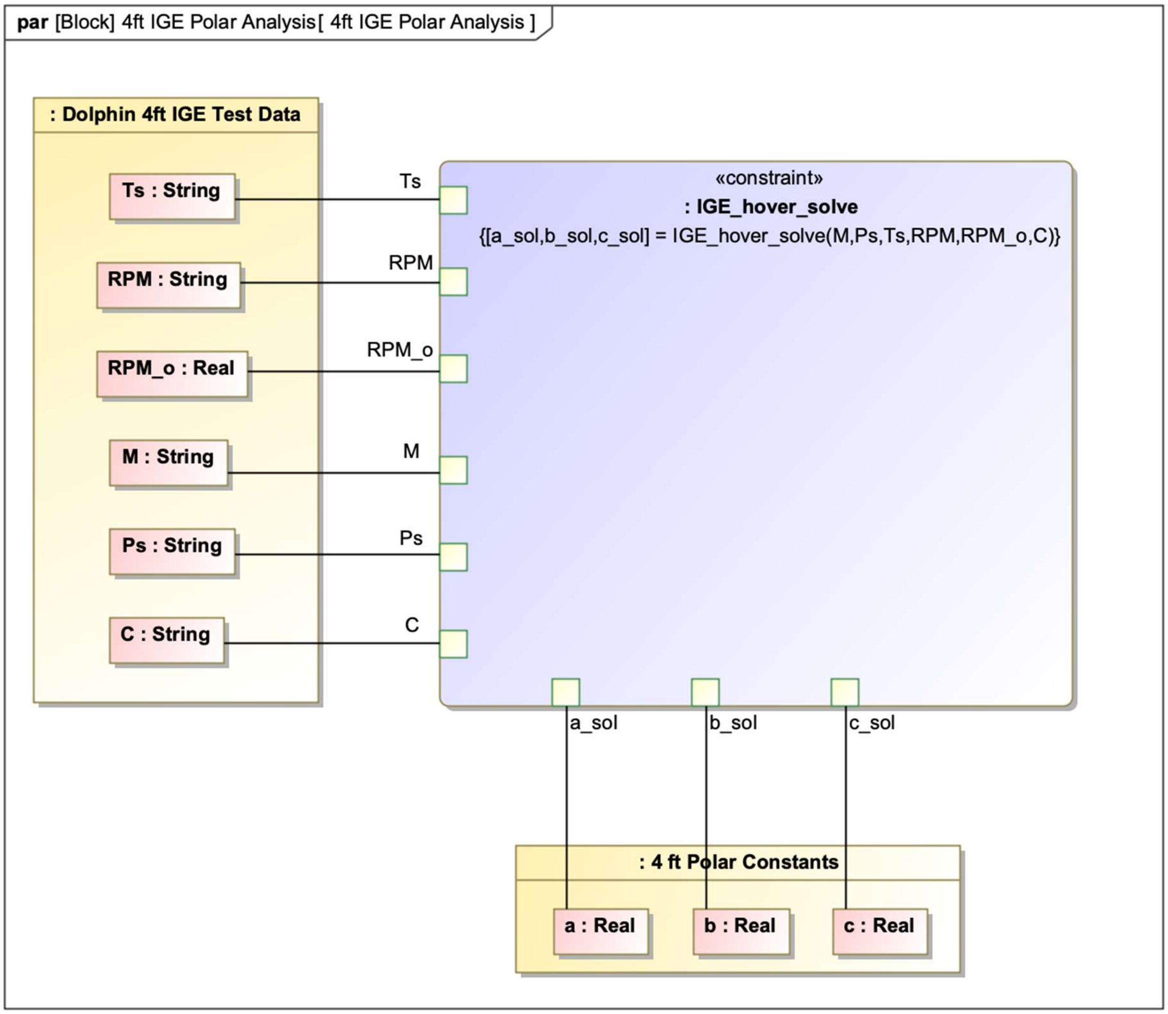


Figure 22: Parametric Diagram for 4ft IGE Polar Analysis

The first output of the parametric analysis is the performance parameters, which is the polar constants: 'a', 'b', 'c'. These polar constants are generated by the constraint block, which represents the MATLAB function. The polar constants are used to solve for referred power by $P' = f(M') = aM'^{3/2} + b + cM'^{2}$ Equation 1. The polar constants and this equation are necessary to create the performance charts. Further explanation and equations that were used to create the MATLAB code can be found in Appendix E – Determination of Polar Constants.

$$P' = f(M') = aM'^{3/2} + b + cM'^2$$

Equation 1

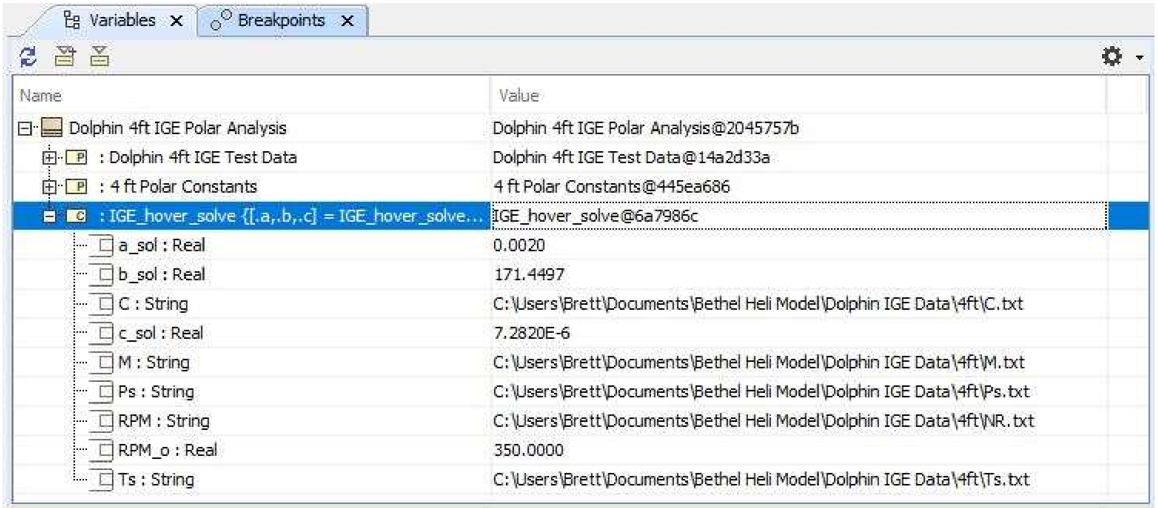
Where:

P' = referred power

M' = referred mass

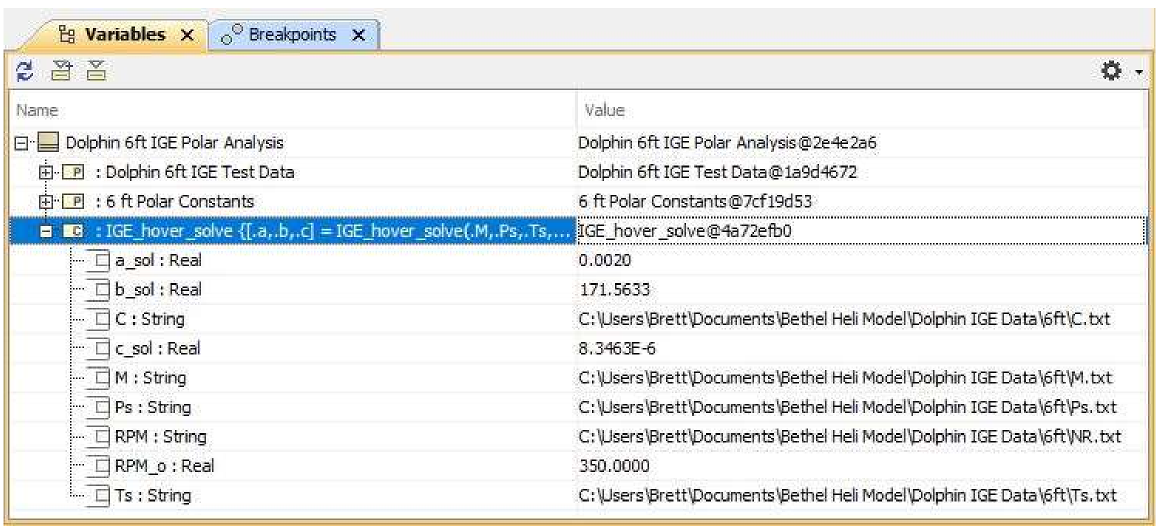
a, b, c = polar constants

Since the polar constants are unique for the four-foot IGE and six-foot IGE, the results of the analysis are saved to unique blocks. The OGE polar constants are valid for all situations where the aircraft is experiencing OGE. Figure 24, Figure 25, and Figure 25 show the results of the analysis completed in Cameo Systems Modeler for the four-foot IGE, six-foot IGE, and OGE respectfully. For this research, under the 'Dolphin System' package and within the 'Dolphin System' block hierarchy, there is a block for performance of the Dolphin helicopter. This is where all the performance analysis is stored. Within this hierarchy, there are polar constants blocks for each of the analysis: four-foot IGE, six-foot IGE, and OGE. Each polar constants block has three value properties, one for each polar constant: a, b, and c. The polar constants results for each simulation are saved within this value property, as illustrated in Figure 22.



Name	Value
Dolphin 4ft IGE Polar Analysis	Dolphin 4ft IGE Polar Analysis@2045757b
: Dolphin 4ft IGE Test Data	Dolphin 4ft IGE Test Data@14a2d33a
: 4 ft Polar Constants	4 ft Polar Constants@445ea686
: IGE_hover_solve {[.a,.b,.c] = IGE_hover_solve…	IGE_hover_solve@6a7986c
a_sol : Real	0.0020
b_sol : Real	171.4497
C : String	C:\Users\Brett\Documents\Bethel Heli Model\Dolphin IGE Data\4ft\C.txt
c_sol : Real	7.2820E-6
M : String	C:\Users\Brett\Documents\Bethel Heli Model\Dolphin IGE Data\4ft\M.txt
Ps : String	C:\Users\Brett\Documents\Bethel Heli Model\Dolphin IGE Data\4ft\Ps.txt
RPM : String	C:\Users\Brett\Documents\Bethel Heli Model\Dolphin IGE Data\4ft\NR.txt
RPM_o : Real	350.0000
Ts : String	C:\Users\Brett\Documents\Bethel Heli Model\Dolphin IGE Data\4ft\Ts.txt

Figure 23: Dolphin 4ft IGE Analysis Results



Name	Value
Dolphin 6ft IGE Polar Analysis	Dolphin 6ft IGE Polar Analysis@2e4e2a6
: Dolphin 6ft IGE Test Data	Dolphin 6ft IGE Test Data@1a9d4672
: 6 ft Polar Constants	6 ft Polar Constants@7cf19d53
: IGE_hover_solve {[.a,.b,.c] = IGE_hover_solve(.M,.Ps,.Ts,…	IGE_hover_solve@4a72efb0
a_sol : Real	0.0020
b_sol : Real	171.5633
C : String	C:\Users\Brett\Documents\Bethel Heli Model\Dolphin IGE Data\6ft\C.txt
c_sol : Real	8.3463E-6
M : String	C:\Users\Brett\Documents\Bethel Heli Model\Dolphin IGE Data\6ft\M.txt
Ps : String	C:\Users\Brett\Documents\Bethel Heli Model\Dolphin IGE Data\6ft\Ps.txt
RPM : String	C:\Users\Brett\Documents\Bethel Heli Model\Dolphin IGE Data\6ft\NR.txt
RPM_o : Real	350.0000
Ts : String	C:\Users\Brett\Documents\Bethel Heli Model\Dolphin IGE Data\6ft\Ts.txt

Figure 24: Dolphin 6ft IGE Analysis Results

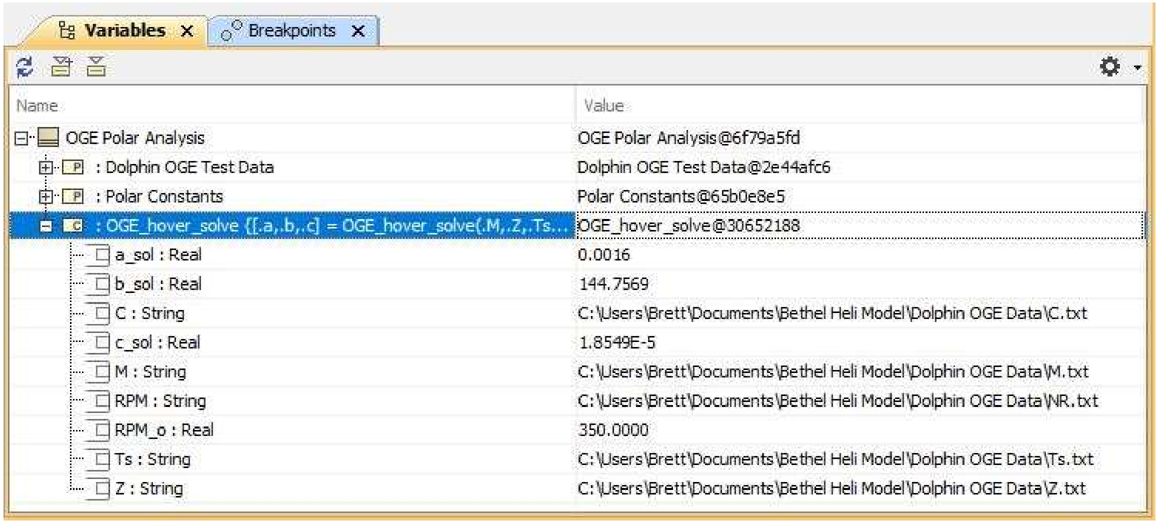


Figure 25: Dolphin OGE Analysis Results

The other output of the analysis is the performance charts. The performance charts can be displayed in many different forms, however, for this research, two types of performance charts will be generated. The first chart will be a polar chart that is referred power as a function of referred mass. The generation of the polar chart is an intermediate step that enables the generation of the performance chart for hover power.

The second chart is the performance chart that will be included in the flight manual, and the chart is a density altitude as a function of percent torque. This chart features multiple curves that represent different gross weights of the helicopter and fulfills both civilian and military airworthiness requirements. These charts are generated by the MATLAB function when it is executed. However, they are not an output in the parametric diagram. They are saved to a user defined file directory, where they can be imported into the model after the simulation has been completed. Importing these diagrams is achieved by attaching a file to a SysML block and choosing the correct chart in the file directory.

For this research, under the ‘Dolphin System’ package and within the ‘Dolphin System’ block hierarchy, there is a block for performance of the Dolphin helicopter. This is where all the performance analysis is stored. Within this hierarchy, there is a performance charts block that has blocks for four-foot IGE, six-foot IGE, and OGE. The performance charts are attached to each corresponding block. An example of the polar chart can be seen in Figure 26. An example of the hover power chart can be seen in Figure 27. Once the analysis is complete and the performance parameters and charts are saved accordingly, the next step is to verify and satisfy the airworthiness requirements.

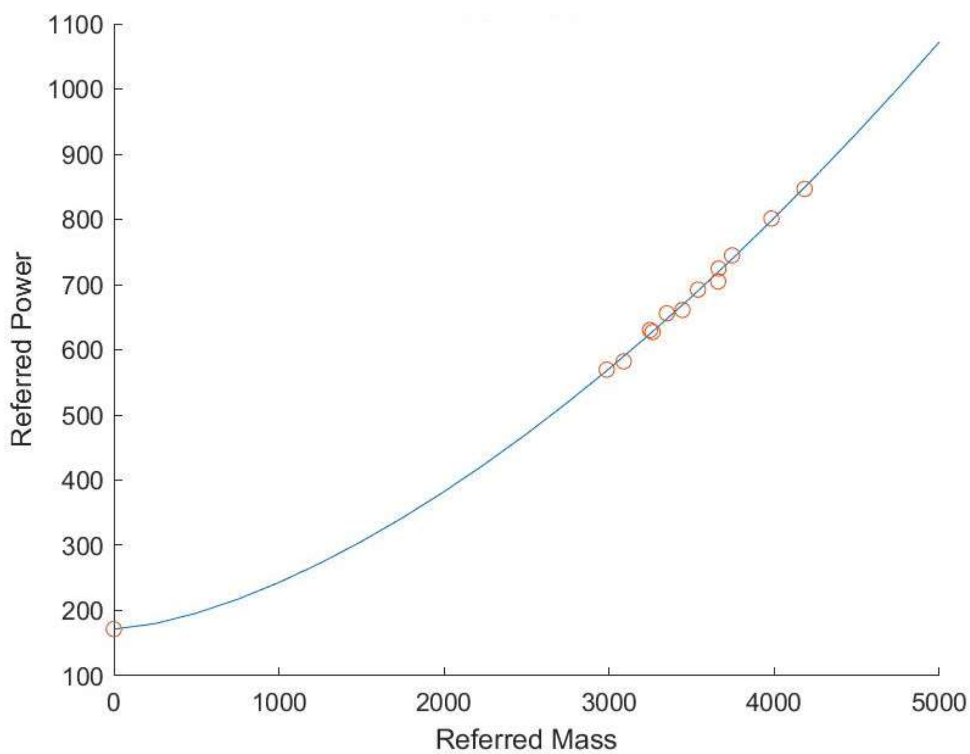


Figure 26: Dolphin 4ft IGE Polar Chart

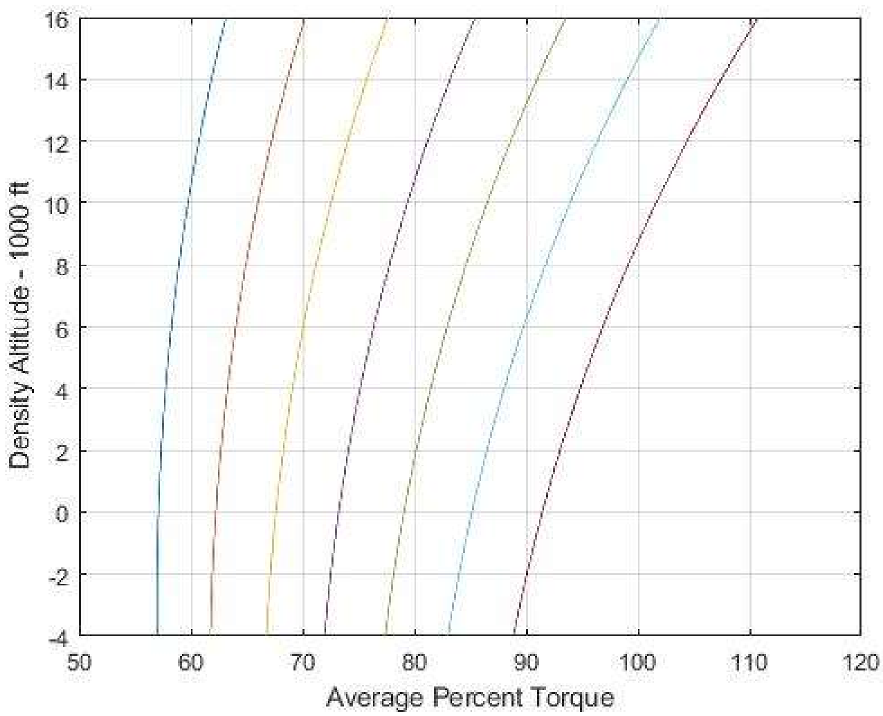


Figure 27: Dolphin 4ft IGE Power Chart

Requirements Satisfaction and Verification Results

After the analysis is complete, the final step in the airworthiness process loop is to satisfy and verify the requirements using the analysis outputs and other model elements. As discussed in Chapter III, there are three methods of creating the satisfy and verify relationships in the model: a satisfy and verify matrix, a block definition diagram, and a requirements table. There are advantages and disadvantages for each method, and each will be discussed in this section.

The method of assigning the satisfy and verify relationships by creating the satisfy and verify matrices is the optimal method because it is the most efficient and user-friendly. The user must simply double click at the intersection of a requirement and satisfying or verifying model element to create the relationship. This method shows the hierarchy of the requirements and allows the user to export the matrix to an Excel file. The matrix also makes it simple to identify which requirements are not satisfied or verified by the model elements. The disadvantage of this method is that the requirement description is not directly displayed in the matrix; however, the description can be read by hovering the cursor over the requirement of interest. An example of the satisfy and verify matrices for the MIL-HDBK-516C requirements can be seen in Figure 28.

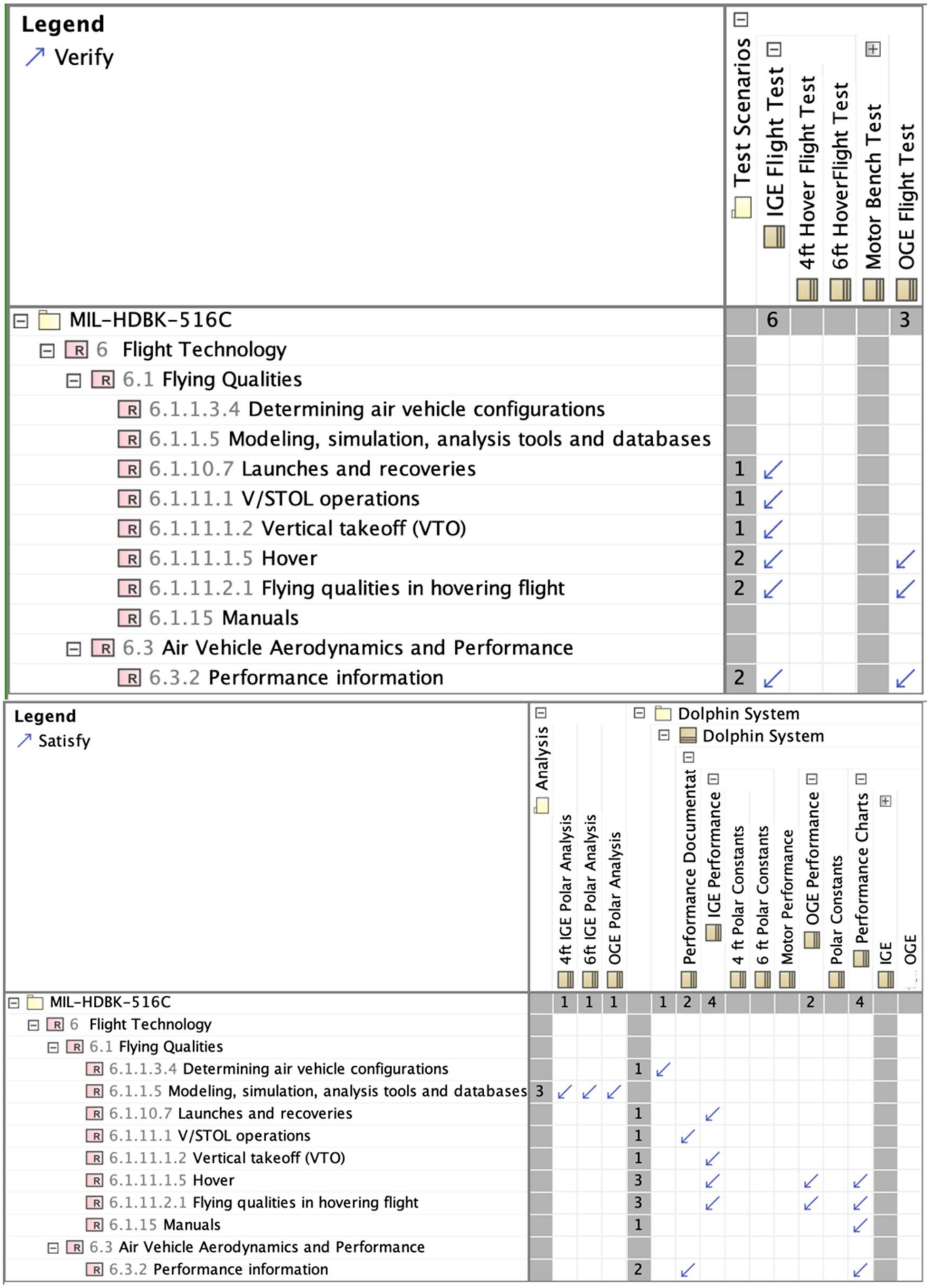
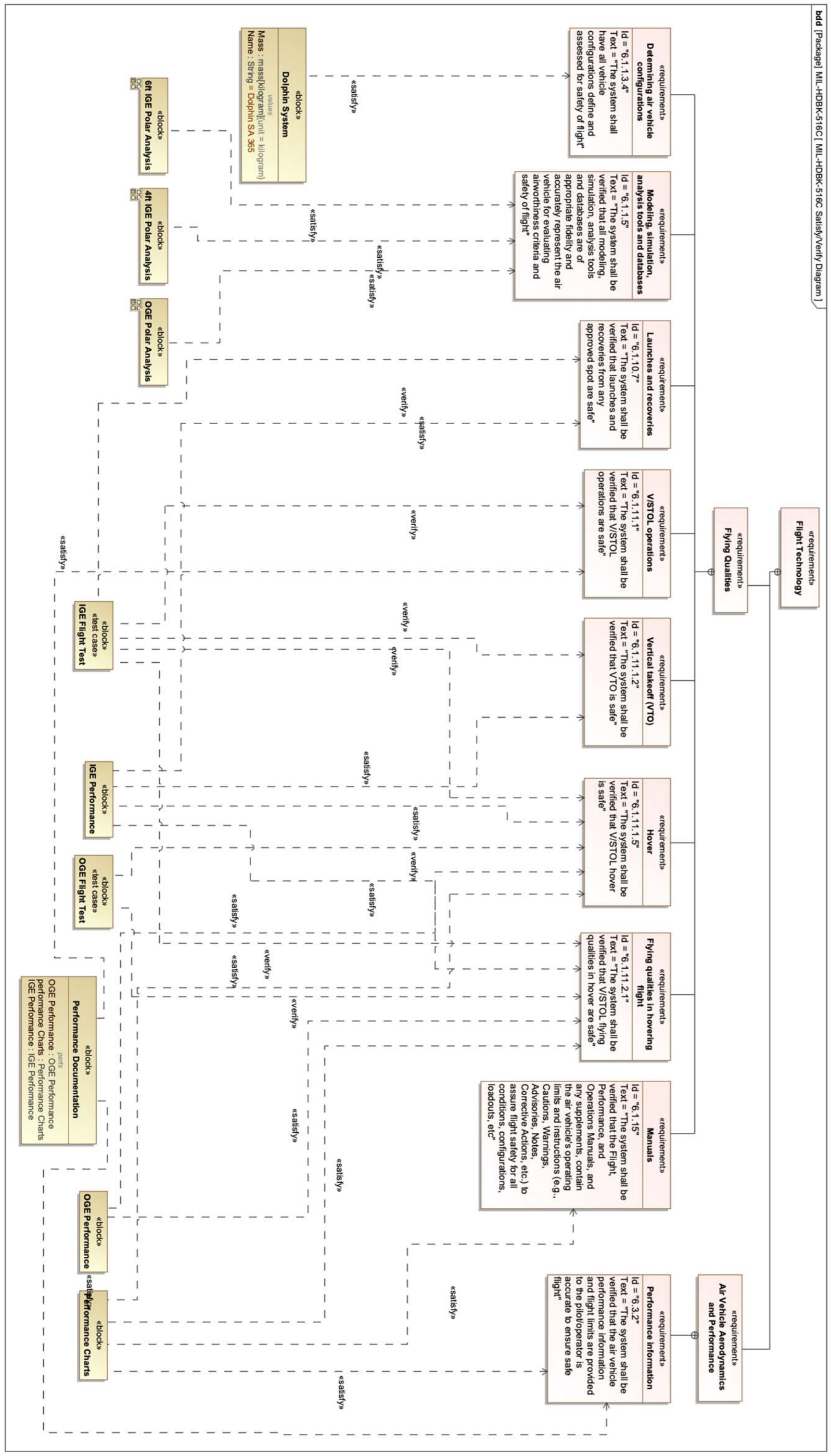


Figure 28: MIL-HDBK-516C Requirements Satisfy/Verify Matrices

The second method of creating the satisfy and verify relationships is the block definition diagram. This method is the most graphical method and allows the user the ability to display the requirement with the description. This also graphically displays the hierarchy of the requirements. The major disadvantage of this method is that it can become confusing and crowded when displaying multiple requirements. An example of the satisfy and verify block definition diagram for the MIL-HDBK-516C requirements can be seen in Figure 29.



The third method for creating the satisfy and verify relationships is the usage of the requirements table. The requirements table is created at an earlier stage when importing and organizing the requirements. The user can add the 'satisfied by' and the 'verified by' columns to the table. Once the columns are added to the table, the relationships can be assigned. An advantage to this method is that it displays the requirement with the description and the satisfy and verify relationships. Another advantage to this method is that it allows the user to identify the requirements that are not satisfied for verified by model elements. This method also displays the requirements hierarchy and can be exported to an Excel sheet. The disadvantage to this method is that it can be cumbersome and time-consuming to assign the satisfy and verify relationships. An example of the requirements table for the MIL-HDBK-516C requirements with the satisfy and verify relationships can be seen in Figure 30.

#	Name	Verified By	Satisfied By	Text
1	MIL-HDBK-516C			
2	6 Flight Technology			
3	6.3 Air Vehicle Aerodynamics and Performance			
4	6.3.2 Performance information	IGE Flight Test OGE Flight Test	Performance Docume Performance Charts	The system shall be verified that the air vehicle performance information and flight limits are provided to the pilot/operator is accurate to ensure safe flight
5	6.1 Flying Qualities			
6	6.1.15 Manuals		Performance Charts	The system shall be verified that the Flight, Performance, and Operations Manuals, and any supplements, contain the air vehicle's operating limits and instructions (e.g., Cautions, Warnings, Advisories, Notes, Corrective Actions, etc.) to assure flight safety for all conditions, configurations, loadouts, etc
7	6.1.11.2.1 Flying qualities in hovering flight	IGE Flight Test OGE Flight Test	IGE Performance OGE Performance Performance Charts	The system shall be verified that V/STOL flying qualities in hover are safe
8	6.1.11.1 V/STOL operations	IGE Flight Test	Performance Docume	The system shall be verified that V/STOL operations are safe
9	6.1.11.1.5 Hover	IGE Flight Test OGE Flight Test	IGE Performance OGE Performance Performance Charts	The system shall be verified that V/STOL hover is safe
10	6.1.11.1.2 Vertical takeoff (VTO)	IGE Flight Test	IGE Performance	The system shall be verified that VTO is safe
11	6.1.10.7 Launches and recoveries	IGE Flight Test	IGE Performance	The system shall be verified that launches and recoveries from any approved spot are safe
12	6.1.1.5 Modeling, simulation, analysis tools and databases		OGE Polar Analysis 6ft IGE Polar Analysis 4ft IGE Polar Analysis	The system shall be verified that all modeling, simulation, analysis tools and databases are of appropriate fidelity and accurately represent the air vehicle for evaluating airworthiness criteria and safety of flight
13	6.1.1.3.4 Determining air vehicle configurations		Dolphin System	The system shall have all vehicle configurations define and assessed for safety of flight
14	FAR 29			
25	FAA AC 29			

Figure 30: MIL-HDBK-516C Requirements Satisfy/Verify in Requirements Table

Traceability Results

Now that the airworthiness process loop has been completed, the final step is to display the traceability of the airworthiness process throughout the model. Requirements traceability can be accomplished in many ways but for this research, a block definition diagram was used. For ease of display, this is done for each requirement individually, which can be time consuming, because there are a large number of requirements and if there is more than one requirement per diagram, the information displayed becomes difficult to understand. Requirements traceability is important because it provides evidence for aircraft airworthiness certification. Evidence of traceability is a necessary component for the Airworthiness Authority (AA) to provide airworthiness certification to the aircraft. Once traceability for all requirements is complete, the model can be reviewed by the AA and the certification can be issued for the aircraft.

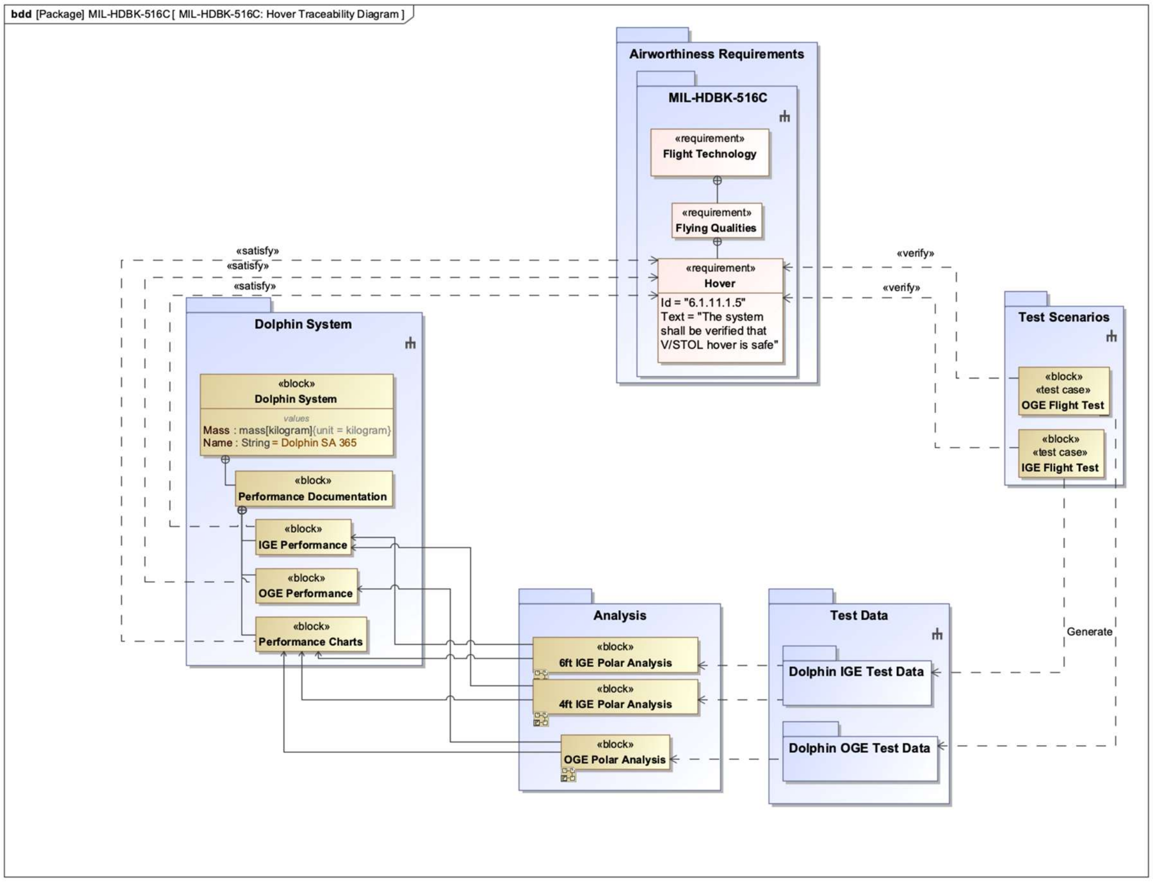


Figure 31: Traceability View for MIL-HDBK-516C - Hover Requirement

Chapter Summary

In summary, the results of implementing the logical model developed in Chapter III for a Dolphin helicopter in hover and takeoff conditions is discussed in this chapter. The implementation is applied throughout the airworthiness process loop: requirements management, test scenario and test data storage, assigning requirements relationships, and providing traceability. Chapter V will provide a conclusion of this research, answers to the research questions presented in Chapter I, discuss research limitations, recommendations for further research in this field, and review the significance of this research.

V. Conclusions and Recommendations

Chapter Overview

This chapter reviews the efforts of this research and presents conclusions about the reference architecture and the implementation of the architecture. Answers are provided for the research questions introduced in Chapter I. A discussion of the limitations encountered during this research and recommendations for future research. Finally, the significance of the completion of this research is provided.

Conclusion of Research

The overarching objective of this research was to develop a reference architecture for helicopter airworthiness with a Model-Based Systems Engineering (MBSE) approach. This architecture was developed to follow the airworthiness process loop. Airworthiness requirements were identified, imported, and organized in the model. The requirements are traced to test scenarios, which are created in the model. The test scenarios generate flight test data, which is also imported into the model and subsequently analyzed. The analysis is completed by parametric models that feature MATLAB functions, that are integrated with the model, represented by constraint blocks. The outputs of the parametric analysis are used to satisfy and verify the airworthiness requirements. Finally, a traceability view is created to graphically follow the airworthiness process throughout the model.

The demonstration of implementing this reference architecture provides a proof of concept for the effectiveness of introducing a MBSE approach to the existing airworthiness process. For this demonstration, the aircraft of interest was the Dolphin helicopter and the airworthiness requirements were limited to hover and takeoff. Since the Dolphin is utilized in both the civilian domain and in the DoD domain, the airworthiness requirements that govern the certification of the aircraft in both domains apply. The model was designed to able to separate the requirements and perform the airworthiness loop for one domain, or both domains. The overall demonstration of the usefulness of a MBSE approach to airworthiness was successful as the reference architecture was able to perform the airworthiness process loop.

Summary of Research Questions and Answers

1. *How can flight test data be organized inside an MBSE reference architecture to optimize the usability of the model?*

   When developing the reference architecture, a concern arose regarding how to organize the flight test data in the model to provide easy of usability for the user. One of the benefits to MBSE is the efficiency that it provides compared to a document-based systems engineering approach. However, the airworthiness process depends on the ability to collect, store, organize, and analyze test data, which presents an issue because there are multiple different methods of organizing data in a model. Since the airworthiness process relies on the ability to analyze the test data, the organization of the test data must facilitate the analysis.

   For this research, the optimal method of organizing the test data was to create a test specific package and test data package. Under the hierarchy of the test data package, SysML blocks with value properties are created to store the flight test data. This method is the most efficient method because when the analysis is performed in a parametric diagram, value properties of SysML blocks are used to assign inputs and outputs.

2. *How can flight test data be used to satisfy and verify airworthiness requirements in a SysML model?*

To obtain an airworthiness certification from the Airworthiness Authority for an aircraft, the evidence must be provided that the requirements for that aircraft are satisfied and verified. The evidence of satisfaction and verification of requirements depends on the analysis of the flight test data and production of performance parameters to be able to make the satisfy and verify relationships necessary for certification. When using a MBSE approach and a SysML model, all aspects, requirements, test data, and analysis, are integrated with relationships to one another to provide a single digital source for certification.

3. *How can a reference architecture be constructed to automate the analysis of flight test data?*

Another concern when developing a model for airworthiness is how to perform the analysis necessary to be able to satisfy and verify the airworthiness requirements. SysML features a diagram called a parametric diagram that allows the user to run a simulation. For this research, the simulation is the analysis of the test data. The SysML tool used for this research was Cameo Systems Modeler, which has the ability to do computations with a built-in math engine or be integrated with external math engines, such as Mathematica and MATLAB. MATLAB was chosen as the external math engine because the analysis necessary for the demonstration included matrix algebra, which is easily completed in MALAB. The simulation of the parametric diagram with MATLAB integrated functions automates the analysis of the flight test data in the SysML model. While MATLAB was used, other math engines could have been employed to perform the analysis.

4. *How can an MBSE approach be leveraged to improve the airworthiness process in the DoD?*

   A MBSE methodology when applied to the airworthiness process has the ability to improve the process. The demonstration of the effectiveness of a MBSE approach to airworthiness was provided in Chapter IV by implementing the reference architecture to the airworthiness of the Dolphin helicopter for hover and takeoff requirements. The model developed in this research provides a single digital source that not only contains the requirements, test scenarios, test data, analysis, aircraft-specific performance parameters, and satisfy and verify relationships between requirements and model elements, but also provides traceability throughout the airworthiness process. The power of a MBSE approach is the ability to conduct the entire airworthiness process in a single digital model. This directly translates to a more efficient process.

Research Limitations

There were limitations placed on this research which made it accomplishable in a constrained timeline. The first limitation was the application of a single type of aircraft, a helicopter, and a single helicopter, the Dolphin. This limits the ability of this research to be easily applied to other aircraft, such as fixed-wing aircraft and small unmanned aerial systems. In addition, the demonstration for only the Dolphin helicopter limits the ability to determine the effectiveness on other rotorcraft such as multiple rotor helicopters and larger helicopters, since the Dolphin is small single rotor helicopter.

The second limitation was that only hover and takeoff airworthiness requirements were evaluated during the demonstration of the reference architecture. This limitation was placed on this research for multiple reasons such as only having access to test data for those conditions and a complete set of airworthiness requirements would require much more time than available.

The last limitation on this research was the exclusion of the motor and transmission limitations. The limitations on the motor and transmission effects the performance charts generated during the analysis. Without these considered during the analysis, the charts are incomplete because they do not accurately represent the actual capabilities of the aircraft since theoretically the chart shows aircraft performance that is not possible due to maximum allowable conditions for the motor and transmission. The motor and transmission limitations can be included in the analysis if time permitted.

Recommendations for Future Research

Future research of this topic should continue to produce a more complete reference architecture on multiple aspects and a further demonstration of an entire set of airworthiness requirements. The migration of airworthiness requirements to SysML. The refinement of existing parametric analysis models and the addition of other parametric analysis models could enhance the model effectiveness. The ability to perform a trade space analysis and an analysis of a specific mission set in the model could be useful for the end user of the aircraft.

Migrating airworthiness requirements to a SysML would be beneficial for completing the airworthiness process with an MBSE approach. Although it would be an extensive undertaking, having the airworthiness requirements modeled in SysML would prevent the repetitive modeling of requirements for each application. Since aircraft airworthiness certification is subject to the same set of requirements, in an MBSE approach, the architecture for a specific aircraft could simply pull the relevant requirements from a database of already modeled requirements via a plug-in for the modeling program, such as Cameo Systems Model.

The existing parametric analysis models must be modified for each user that performs the simulation by downloading the data files and changing the file paths in the value properties of the test data blocks in the model. This is required because there was not a clear solution to have the data and analysis method self-contained in the system. If a method of a self-contained analysis in the model is accomplished, the effectiveness of the model will increase. The existing parametric analysis has no way to automate the import of the performance charts into the model as an output of the constraint block. Therefore, the user must edit the MATLAB function to specify a file path unique to their computer.

The addition of another parametric diagram for the purpose of automating the satisfy and verify relationships between model elements could be useful for the user of the model because it can be time consuming to manually enter those relationships. For this to be possible there must be a method of quantifying the airworthiness requirements, which would require a line of research alone, but could improve the airworthiness process in the future when a MBSE approach is adopted to perform airworthiness certification.

The model being able to perform trade space analysis and mission set analysis is a powerful tool for the aircraft program office and the end user. The program office could utilize a trade space analysis when considering new components of the aircraft such as a motor, transmission, rotor, etc. They could input the performance parameters into the model and run a simulation to determine how the new component could affect overall system performance. The end user could utilize a mission set analysis to easily determine if the aircraft is capable to perform the mission. This would be accomplished by a parametric model of the system where the end user would input mission parameters such as weight, environmental conditions, flight parameters, etc. and the analysis would determine if the aircraft could safely perform the mission.

Significance of Research

This research developed a functional reference architecture to perform airworthiness certification of a helicopter through the entire airworthiness process loop. It was developed to be able to be applied to all helicopters. This reference architecture was demonstrated to be successful when implemented for the airworthiness of a Dolphin helicopter in hover and takeoff conditions. Given that there are different governing organizations based on the application of aircraft, whether it be used in the civilian domain or the DoD domain, the ability of the model to perform airworthiness for one domain or both is a testament to the usefulness of the reference architecture. The effectiveness of this architecture has the potential to revolutionize the airworthiness process in the DoD, where the ability to quickly and efficiently certify airworthiness for aircraft is crucial for mission success and to continue a legacy of air supremacy.

Appendix A – MIL-HDBK-516C Requirements

Table 5: Full MIL-HDBK-516C Requirements (Department of Defense, 2014)

ID and Name	Criterion	Standard	Method of Compliance
6.1.1.3.4 Determining air vehicle configurations	Verify that all air vehicle configurations have been defined and assessed for safety of flight	The air vehicle meets the standards within MIL-STD-1797, 4.1.3.4 Determining air vehicle configurations. For rotorcraft, the air vehicle meets standards within ADS-33-PRF, 3.1.7 Configurations	Verification methods include inspection of requirements, design, and configuration documentation
6.1.1.5 Modeling, simulation, analysis tools and databases	Verify that all modeling, simulation, analysis tools and databases are of appropriate fidelity and accurately represent the air vehicle for evaluating airworthiness criteria and safety of flight	Verify and validate that modeling, simulation, analysis tools and databases which are utilized for evaluating airworthiness criteria across the flight envelope, for all expected center-of-gravity ranges and mass properties, for all flight phases, tasks and flight control modes, for all configurations and store loadings as tailored from tables I, II and III of MIL-STD-1797, and in the expected atmospheric disturbances for which the air vehicle is to perform its mission(s) are of sufficient fidelity and accuracy. A suitable verification, validation and accreditation (VV&A) process, as outlined in MIL-STD-3022, is demonstrated. Configuration control across all such tools is demonstrated to assure currency and traceability	Verification methods include inspection of maturity, fidelity and accuracy of analysis, modeling and simulation tools and databases, as well as the processes in place to assure their currency, traceability and configuration control. Analysis, modeling and simulation tools and databases, including the verification and validation of their results, reflect industry best practices for the purpose of their intended use
6.1.10.7 Launches and recoveries	Verify that launches and recoveries from any approved spot are safe	Provide piloted simulation, land-based flight test data, or analysis against historical standards to show this is safe. A logical and	Verification methods include analysis, test, demonstration, simulation, and inspection of process, design, test, or configuration documentation

		measured flight test build-up from benign to more stressing conditions may be allowable in lieu of this data	
6.1.11.1 V/STOL operations	Verify that V/STOL operations are safe	The air vehicle meets the standards within MIL-STD-1797, 5.2.7.1 V/STOL operations	Verification methods include analysis, test, demonstration, simulation, and inspection of process, design, test, or configuration documentation
6.1.11.1.2 Vertical takeoff (VTO)	Verify that vertical takeoff (VTO) is safe	The air vehicle meets the standards within MIL-STD-1797, 5.2.7.1.2 Vertical takeoff (VTO)	Verification methods include analysis, test, demonstration, simulation, and inspection of process, design, test, or configuration documentation
6.1.11.1.4 Powered-lift landing	Verify that V/STOL powered-lift landing is safe	The air vehicle meets the standards within MIL-STD-1797, 5.2.7.1.4 Powered-lift landing	Verification methods include analysis, test, demonstration, simulation, and inspection of process, design, test, or configuration documentation
6.1.11.1.5 Hover	Verify that V/STOL hover is safe	The air vehicle meets the standards within MIL-STD-1797, 5.2.7.1.5 Hover	Verification methods include analysis, test, demonstration, simulation, and inspection of process, design, test, or configuration documentation
6.1.11.2.1 Flying qualities in hovering flight	Verify that V/STOL flying qualities in hover are safe	The air vehicle meets the standards within MIL-STD-1797, 5.2.7.2.1 Flying qualities in hovering flight	Verification methods include analysis, test, demonstration, simulation, and inspection of process, design, test, or configuration documentation.
6.1.15 Manuals	Verify that the Flight, Performance, and Operations Manuals, and any supplements, contain the air vehicle's operating limits and instructions (e.g., Cautions, Warnings, Advisories, Notes, Corrective Actions, etc.) to assure flight safety for all conditions, configurations, loadouts, etc	The manuals accurately document/identify aircraft operating limits and emergency characteristics and procedures	Review of the manuals verifies that the limits and emergency procedures documented are appropriate and adequate

6.3.2 Performance information	Verify that air vehicle performance information provided to the pilot/operator is accurate to ensure safe flight	Flight manual air vehicle performance for all flight phases including, but not limited to, launch, takeoff, climb, cruise, endurance, maneuver, hover, in-flight refueling, descent, approach, landing, and recovery is sufficiently accurate to allow safe operations	An air vehicle force and moment accounting system is defined for all air vehicle variants, configurations, and flight conditions. All coordinate systems, sign conventions, control effectors, aerodynamic and propulsion forces and moments, and aerodynamic/propulsion reference conditions have been defined to support performance simulation
		Flight manual air vehicle performance includes the full range of mass properties and atmospheric conditions for all air vehicle variants, configurations, and loadings within the flight envelope	Aerodynamic, installed propulsion, and mass properties databases are based on the latest information available, have been placed under configuration control, and are sufficient in scope for all air vehicle configurations, loadings, and flight conditions. All aerodynamic data corrections of the original source analysis/test data to the final, full-scale, flight representative configuration are defined. All propulsion data is corrected for losses and efficiency changes going from uninstalled to installed configurations. Mass properties are representative of all air vehicle configurations and loadings
		Air vehicle performance information provided to the pilot/operator by other means (e.g., checklist, calculator, laptop, mission planning tool, onboard embedded system) is sufficiently accurate to allow for safe operation	Predictions of trimmed lift and drag in and out of ground effect, installed thrust, power available, power required, fuel flow, fuel quantity, inertias, center of gravity, and weights allow for accurate simulation of air vehicle performance for all atmospheric conditions within the flight envelope
			Flight manual air vehicle performance is based on simulation models that have been verified against actual air vehicle flight performance and accounts for flight test data measurement uncertainty
			All flight manual air vehicle performance charts, procedures, and instructions are defined, clearly written, and traceable back to the supporting analysis and data basis

			Flight manual performance is verified by inspection of documentation. Performance information provided to the pilot/operator by other means is verified against the flight manual or simulation model by test and inspection of documentation
	Verify that all air vehicle performance flight limits are provided to the pilot/operator to ensure safe operation	Any flight performance limitation that affects safe operation of the air vehicle for both normal and degraded/emergency operating conditions is identified and documented including, but not limited to, weight, center of gravity, acceleration, speed, altitude, stall, buffet, engine operability, propulsion system limits, rate-of-climb, rate-of-descent, maneuverability, structural load limit, landing gear, brake energy, store carriage, temperature, wind, runway condition, and icing	An air vehicle force and moment accounting system is defined for all air vehicle variants, configurations, and flight conditions. All coordinate systems, sign conventions, control effectors, aerodynamic and propulsion forces and moments, and aerodynamic/propulsion reference conditions have been defined to support performance simulation
			Aerodynamic, installed propulsion, and mass properties databases are based on the latest information available, have been placed under configuration control, and are sufficient in scope for all air vehicle configurations, loadings, and flight conditions. All aerodynamic data corrections of the original source analysis/test data to the final, full-scale, flight representative configuration are defined. All propulsion data is corrected for losses and efficiency changes going from uninstalled to installed configurations. Mass properties are representative of all air vehicle configurations and loadings
			Predictions of trimmed lift and drag in and out of ground effect, installed thrust, power available, power required, fuel flow, fuel quantity, inertias, center of gravity, and weights allow for accurate simulation of air vehicle performance for all atmospheric conditions within the flight envelope
			Flight manual air vehicle performance is based on simulation models that have been verified against actual air vehicle flight performance and accounts for flight test data measurement uncertainty

			Air vehicle buffet and stall characteristics accounting for Mach number effects as well as deployed flaps, spoilers, landing gear, and store carriage are identified and assessed using wind tunnel and flight test data
			Stall angle-of-attack and/or stall speed account for air vehicle weight, center of gravity, configuration, and store loading
			Charts depicting bank angle versus minimum speed to maintain altitude account for air vehicle weight, configuration, and store loading
			For air vehicles without adequate anti-ice protection, the effect of icing on air vehicle aerodynamics and performance is characterized using analysis, wind tunnel, and/or flight test data to establish operational limits
			Rotorcraft performance limits account for vortex ring state, settling with power, retreating blade stall, and advancing blade compressibility effects
			All flight limit charts, procedures, and instructions are defined, clearly written, and traceable back to the supporting analysis and data basis
			Flight manual performance is verified by inspection of documentation. Performance information provided to the pilot/operator by other means (e.g., checklist, calculator, laptop, mission planning tool, onboard embedded system) is verified against the flight manual or simulation model by test and inspection of documentation

Appendix B – FAR 29 Requirements

Table 6: Full FAR 29 Requirements (Federal Aviation Administration, 2008)

ID and Name	Description
29.45 General	the performance prescribed must be determined: with normal piloting skill and without exceptionally favorable conditions
	Compliance with the performance requirements of this subpart must be shown— For still air at sea level with a standard atmosphere and for the approved range of atmospheric variables
	The available power must correspond to engine power, not exceeding the approved power, less installation losses and the power absorbed by the accessories and services at the values for which certification is requested and approved
	For turbine engine-powered rotorcraft, the performance, as affected by engine power, must be based on a relative humidity of— 80 percent, at and below standard temperature and 34 percent, at and above standard temperature.
	For turbine-engine-power rotorcraft, a means must be provided to permit the pilot to determine prior to takeoff that each engine is capable of developing the power necessary to achieve the applicable rotorcraft performance prescribed in this subpart.
29.49 Performance at minimum operating speed	For each Category B helicopter, the hovering performance must be determined over the ranges of weight, altitude, and temperature for which certification is requested, with— Takeoff power; the landing gear extended; and the helicopter in ground effect at a height consistent with normal takeoff procedures.
	For each helicopter, the out-of-ground effect hovering performance must be determined over the ranges of weight, altitude, and temperature for which certification is requested with takeoff power.
29.1587 Performance information	*Category B.* For each category B rotorcraft, the Rotorcraft Flight Manual must contain—hover ceiling and out-of-ground effect hover performance determined under §29.49 and the maximum safe wind demonstrated under the ambient conditions for data presented. In addition, the maximum weight for each altitude and temperature condition at which the rotorcraft can safely hover out-of-ground-effect in winds of not less than 17 knots from all azimuths. These data must be clearly referenced to the appropriate hover charts

Appendix C – FAA AC 29 Requirements

Table 7: Full FAA AC 29 Requirements (Federal Aviation Administration, 2014)

ID and Name	Description
29.49.1.2: IGE and OGE	Under § 29.49, hover performance should be determined at a height consistent with the takeoff procedure for category A rotorcraft and in ground effect (IGE) for category B rotorcraft. Additionally, out of ground effect (OGE) hover performance should be determined for both category A and B rotorcraft
29.49.1.3: Power required	The objective of hover performance tests is to determine the power required to hover at different gross weights, ambient temperatures, and pressure altitudes. Using non-dimensional power coefficients (Cp) and thrust coefficients (Ct) for normalizing and presenting test results, a minimum amount of data are required to cover the rotorcraft's performance operating envelope.
29.49.1.4: Conditions	Hover performance tests must be conducted over a sufficient range of pressure altitudes and weights to cover the approved ranges of those variables for takeoff and landings. Additional data should be acquired during cold ambient temperatures, especially at high altitudes, to account for possible Mach effects.
29.49.1.5: Height	The minimum hover height for which data should be obtained and subsequently presented in the flight manual should be the same height consistent with the minimum hover height demonstrated during the takeoff tests. Refer to section 29.51 of this AC for the procedure to determine the minimum allowable hover height.
29.49.2.1: Methods	Two methods of acquiring hover performance data are the tethered and free flight techniques.
29.49.2.4: Techniques	If there are no provisions or equipment to conduct tethered hover tests, the free flight technique is also a valid method. The disadvantage of this technique as the primary source of data acquisition is that it is very time consuming. In addition a certain element of safety is lost OGE in the event of emergency. The rotorcraft must be re-ballasted to different weights to allow the maximum Ct/Cp spread. When using the free flight technique, either as a primary data source or to substantiate the tethered technique, the same considerations for wind, recorded parameters, etc., as used in the tethered technique apply. Free flight hover tests should be conducted at CG extremes to verify any CG effects. If the rotorcraft has any stability augmentation system, which may influence hover performance, it must be accounted for.
29.49.2.5: OGE Tests	It is extremely difficult to determine when a rotorcraft is hovering OGE at high altitudes above ground level since there is no ground reference. In a true hover, the rotorcraft will drift with the wind. Numerous techniques have been tried to allow OGE hover data acquisition at high altitudes, all of which have resulted in much data scatter. Until a method is proposed and found acceptable to the FAA/AUTHORITY, OGE hover data must be obtained at the various altitude sites where IGE hover data is obtained. Hover performance can usually be extrapolated up to a maximum of

Appendix D – Dolphin Flight Test Data

Table 8: Dolphin 4ft IGE Test Data (Cox & Tortel)

C (%)	M (kg)	NR (t/mn)	Ps (hPa)	Ts (°K)
18.7	0	350.6	1000.7	291
87.3	3882.2	340	1002.8	290
78.5	3668.9	354	1002.8	291
81.7	3654.6	338.7	1002.7	291
73.4	3442	353.5	1002.6	291
75.8	3429.4	338.4	1002.6	291
73.3	3334.7	337.4	1002.6	291
69.1	3234.9	352.1	1002.5	291
70.1	3224.3	337	1002.5	290
63.9	3050.7	351.5	1002.3	291
66.1	3037.9	336.6	1002.2	290.6
62.5	2948.9	351.5	1002.2	291
63.4	2939.7	335.9	1002.2	290

Table 9: Dolphin 6ft IGE Test Data (Cox & Tortel)

C (%)	M (kg)	NR (t/mn)	Ps (hPa)	Ts (°K)
18.7	0	350.6	1000.7	291
87	3891.5	350.1	1002.6	290
89.3	3886.5	339.2	1002.6	290
80.3	3663.6	352.2	1002.5	291
83.1	3659.3	337.9	1002.5	291
75	3437.1	353.1	1002.4	291
76.9	3433.1	337.8	1002.3	291
72.4	3341.9	352	1002.3	291
74.4	3338	336.7	1002.3	291
69.5	3232	351.6	1002.3	291
71.8	3228	336.6	1002.3	290.6
65.2	3045.4	351.3	1002	291
65.9	3042.1	335.6	1001.9	291
63.2	2945.7	350.9	1001.9	291
64.3	2942.6	335.7	1001.9	290

Table 10: Dolphin OGE Test Data (Cox & Tortel)

C (%)	M (kg)	NR (t/mn)	Ts (°K)	Z (ft)
16.2	3731	350.3	285	-127
88.9	3652	351.7	283	3572
91.7	3637	333.9	284	3519
89.3	3620	351	284	3007
84.4	3593	353.8	284	2484
83.1	3580	335.7	281	2522
72.5	3562	357.5	282	2108
85.8	3550	339.8	282	1998
81.4	3529	350.3	282.2	2055
83.4	3517	351.6	281	1543
76.2	3505	351.7	281	1609
81.5	3495	339.7	281	1487
76.6	3469	355.9	280	1915
75.3	3152	352.3	280	1478
65	3040	351.2	279	1510

Appendix E – Determination of Polar Constants

<u>Determination of polar constants:</u>

$$P' = f(M') = aM'^{3/2} + b + cM'^2$$

Equation 2

Where:

P' = referred power

M' = referred mass

a, b, c = polar constants

To reduce the errors between P' and f(M'), the creation of a function q, the sum of the square deviations:

$$q = \sum_{j=1}^{n} \left[P' - (aM'^{3/2} + b + cM'^2\right]^2$$

Equation 3

The optimal values of a, b, and c are found at the point where this function is minimal (where the partial derivatives are zero):

$$\frac{\delta q}{\delta a} = -2\sum_{j=1}^{n}(P' - M'^{3/2} - b - cM'^2)(M'^{3/2}) = 0$$

Equation 4

$$\frac{\delta q}{\delta b} = -2\sum_{j=1}^{n}(P' - M'^{3/2} - b - cM'^{2}) = 0$$

Equation 5

$$\frac{\delta q}{\delta c} = -2\sum_{j=1}^{n}(P' - M'^{3/2} - b - cM'^{2})(M'^{2}) = 0$$

Equation 6

The equations are changed into the form below:

$$a\sum_{j=1}^{n} M'^{3} + b\sum_{j=1}^{n} M'^{3/2} + c\sum_{j=1}^{n} M'^{7/2} = \sum_{j=1}^{n} P'M'^{3/2}$$

Equation 7

$$a\sum_{j=1}^{n} M'^{3/2} + bn + c\sum_{j=1}^{n} M'^{2} = \sum_{j=1}^{n} P'$$

Equation 8

$$a\sum_{j=1}^{n} M'^{7/2} + b\sum_{j=1}^{n} M'^{2} + c\sum_{j=1}^{n} M'^{4} = \sum_{j=1}^{n} P'M'^{2}$$

Equation 9

a, b, and c can be found by matrix multiplication:

$$\begin{bmatrix} \sum_{j=1}^{n} M'^{3} & \sum_{j=1}^{n} M'^{3/2} & \sum_{j=1}^{n} M'^{7/2} \\ \sum_{j=1}^{n} M'^{3/2} & n & \sum_{j=1}^{n} M'^{2} \\ \sum_{j=1}^{n} M'^{7/2} & \sum_{j=1}^{n} M'^{2} & \sum_{j=1}^{n} M'^{4} \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} \sum_{j=1}^{n} P'M'^{3/2} \\ \sum_{j=1}^{n} P' \\ \sum_{j=1}^{n} P'M'^{2} \end{bmatrix}$$

Equation 10

Appendix F – MATLAB Code: IGE Hover

```
function [a_sol,b_sol,c_sol] = IGE_hover_solve(M,Ps,Ts,RPM,RPM_o,C)
% This function generates the polar constants and polar chart for a
helicopter in IGE hover

% Explanation of inputs:
% M - mass (kg)
% Ps - pressure (hPa)
% Ts - temperature (degree Kelvin)
% RPM - revolutions per min for the rotor (t/mn)
% RPM_o - initial RPM
% C - precent couple (%)

% % IGE - in ground effect
% % OGE - out of ground effect

% imports C and RPM
% solves for power
fileC = fopen(C,'r');
C_data = fscanf(fileC, '%f');
fileRPM = fopen(RPM,'r');
RPM_data = fscanf(fileRPM, '%f');
P = (C_data.*24555.33408/100).*(RPM_data*2*pi/60)./1000;

% imports pressure and temperature
% solves for air density
filePs = fopen(Ps,'r');
Ps_data = fscanf(filePs, '%f');
fileTs = fopen(Ts,'r');
Ts_data = fscanf(fileTs, '%f');
ad = (Ps_data.*288.15)./(Ts_data.*1013.25);

% solves for referred power
Pp = (P./ad).*((RPM_o./RPM_data).^3);

% imports mass
% solves for referred mass
fileM = fopen(M,'r');
M_data = fscanf(fileM, '%f');
Mm = (M_data./ad).*((RPM_o./RPM_data).^2);

% solves for polar constants
m3 = sum(Mm.^3);
m3_2 = sum(Mm.^(3/2));
m7_2 = sum(Mm.^(7/2));
n = size(Mm);
m2 = sum(Mm.^2);
m4 = sum(Mm.^4);
pm3_2 = sum(Pp.*(Mm.^(3/2)));
p = sum(Pp);
pm2 = sum(Pp.*(Mm.^2));
x = [m3, m3_2, m7_2;...
```

```
    m3_2, n(1), m2;...
    m7_2, m2, m4];
xinv = inv(x);
y = [pm3_2;...
    p;...
    pm2];
sol = xinv*y;
a_sol = sol(1);
b_sol = sol(2);
c_sol = sol(3);

% Creates array of referred mass values
% Uses polar constants to solve for referred power array when given
values for referred mass
Mprime = 0:250:5000;
Pprime = (a_sol.*(Mprime.^(3/2))) + b_sol + (c_sol.*(Mprime.^2));

% the function hoverpower is used to generate curves for power required
hoverpower(a_sol,b_sol,c_sol)

% This saves the figure that is generated by the hoverpower function
figuresdir = 'C:\Users\Brett\Documents\Bethel Heli Model';
saveas(gca, fullfile(figuresdir, 'IGE Hover Power'), 'jpeg')

% Generates the polar chart
figure
hold on
plot(Mprime,Pprime)
scatter(Mm,Pp)
xlabel('Referred Mass')
ylabel('Referred Power')
title('OGE Polar')
hold off

% This saves the polar chart figure
saveas(gca, fullfile(figuresdir, 'IGE Polar'), 'jpeg')
end
```

Appendix G – MATLAB Code: OGE Hover

```
function [a_sol,b_sol,c_sol] = OGE_hover_solve(M,Z,Ts,RPM,RPM_o,C)
% This function generates the polar constants and polar chart for a
helicopter in OGE hover

% Explanation of inputs:
% Z - height from altimeter (ft)
% M - mass (kg)
% Ts - temperature (degree Kelvin)
% RPM - revolutions per min for the rotor (t/mn)
% RPM_o - initial RPM
% C - precent couple (%)

% % IGE - in ground effect
% % OGE - out of ground effect

%imports percent and RPM
%solves for power
fileC = fopen(C,'r');
C_data = fscanf(fileC, '%f');
fileRPM = fopen(RPM,'r');
RPM_data = fscanf(fileRPM, '%f');
P = (C_data.*24555.33408./100).*(RPM_data.*2.*pi./60)./1000;

%imports height and temperature
fileZ = fopen(Z,'r');
Z_data = fscanf(fileZ, '%f');
fileTs = fopen(Ts,'r');
Ts_data = fscanf(fileTs, '%f');

%solves for pressure using height
Zm = Z_data*0.3048;
Ps = 1013.25*((1 - (22.558*(10^(-6))*Zm)).^5.525611);

%solves for air density
ad = (Ps.*288.15)./(Ts_data.*1013.25);

%solves for referred power
Pp = (P./ad).*((RPM_o./RPM_data).^3);

%imports mass
%solves for referred mass
fileM = fopen(M,'r');
M_data = fscanf(fileM, '%f');
Mm = (M_data./ad).*((RPM_o./RPM_data).^2);
Mm(1) = 0;

%solves for polar constants
m3 = sum(Mm.^3);
m3_2 = sum(Mm.^(3/2));
m7_2 = sum(Mm.^(7/2));
```

```
n = size(Mm);
m2 = sum(Mm.^2);
m4 = sum(Mm.^4);
pm3_2 = sum(Pp.*(Mm.^(3/2)));
p = sum(Pp);
pm2 = sum(Pp.*(Mm.^2));
x = [m3, m3_2, m7_2;...
    m3_2, n(1), m2;...
    m7_2, m2, m4];
xinv = inv(x);
y = [pm3_2;...
    p;...
    pm2];
sol = xinv*y;
a_sol = sol(1);
b_sol = sol(2);
c_sol = sol(3);

% the function hoverpower is used to generate curves for power required
hoverpower(a_sol,b_sol,c_sol)

% This saves the figure that is generated by the hoverpower function
figuresdir = 'C:\Users\Brett\Documents\Bethel Heli Model';
saveas(gca, fullfile(figuresdir, 'OGE Hover Power'), 'jpeg')

% Creates array of referred mass values
% Uses polar constants to solve for referred power array when given
values
% for referred mass
Mprime = 0:250:5000;
Pprime = (a_sol.*(Mprime.^(3/2))) + b_sol + (c_sol.*(Mprime.^2));

% Generates the polar chart
figure
hold on
plot(Mprime,Pprime)
scatter(Mm,Pp)
xlabel('Referred Mass')
ylabel('Referred Power')
title('OGE Polar')
hold off

% This saves the polar chart figure
saveas(gca, fullfile(figuresdir, 'OGE Polar'), 'jpeg')
end
```

Appendix H – MATLAB Code: Hover Power

```
function [hp] = hoverpower(a,b,c)
% This function generates power required curves for a helicopter in
hover

% Explanation of inputs:
% a, b and c are parameters from the hover polar, either IGE or OGE
% IGE - in ground effect
% OGE - out of ground effect
%
% This formulation assumes that a system is operating at constant and
% nominal rotor rpm.

% Possible upgrades to this function include:
% - adding NR, both nominal and operational... in the event someone
wants
% to assess performance changes due to beep trim adjustments to rotor
rpm
% - adding entry parameters for helicopter torque reading to kilowatts.
% for now this is hard coded in section 4 of the code below

% 1.0 Altitude and atmoshpere vectors

% 1.1 Generate a vector for geopotential altitude (in feet)
zp = -4000:10:16000;

% 1.2 Unit conversion; our equations need altitude in meters.
zpm = 0.3048*zp;

% 1.3 Generate the pressure associated with that geopotential altitude.
% Pressure generated based on the 1976 International Standard
Atmosphere
% model, this provide pressure in hPa.

ps = 1013.25*((1-(22.558*(10^(-6))*zpm)).^5.525611);

% 1.4 Generate the temperature associated with that geopotential
altitude.
% Temperature generated based on the ICAO and 1976 International
Standard
% Atmosphere model, this provide pressure in degrees Kelvin.  Due to
the
% nature of helicopters we are only considering

tempk = 288.15-(1.98*zp/1000);

% 1.5 Determine sigma for this combination of atmospheric conditions

sigma = (ps*288.15)./(1013.25*tempk);

% 2.0 Developing Mass Vectors
```

```
% We are generating power required curves.  The curves will provide the
% percent torque required for a given take-off weight and density
altitude.
% Given a constant take-off weight, density altitude will be varied and
% power will be determined for that take-off weight.
%
% The form of the curves provided is based on performance curves found
in
% the flight manual for the UH-1N.

% 2.1 Mass in imperial units (pounds)... yah... I know this is weight
m_imperial = [6000 6500 7000 7500 8000 8500 9000];

% 2.2 Mass in kilograms (the required units for our equation)
mk = m_imperial*0.45359237;

% 3.0 Generating Power Required Vectors
for ii = 1:max(size(mk))

    % 3.1 Generating a referred mass vector for each constant mass for
all
    % values of sigma
    Mp = mk(ii)./sigma;

    % 3.2 Generating the referred power vector to compliment the
referred
    % mass vector
    Pp = (a*(Mp.^1.5))+b+(c*(Mp.^2));

    % 3.3 Generating the power in kilowatts that is required to hover
    P(ii,:) = Pp.*sigma;
end

% 4.0 Converting from kilowatts to percent torque
% Power = torque * angular momentum of rotor
% At nominal Nr, 100% torque is equivalent to 900 KW

P = P * 100/900;

plot(P,zp/1000)
grid
xlabel('Average Percent Torque')
ylabel('Density Altitude - 1000 ft')
title('Hover Power')
```

Appendix I – Helicopter Model User Manual

Helicopter Airworthiness Reference Architecture - User Manual

In the reference architecture, Cameo Systems Modeler (CSM) project is used to create a framework for accomplishing airworthiness with a Model-Based Systems Engineering (MBSE) approach.

Tools needed:

- Cameo Systems Modeler 19+
- MATLAB R2017b+

**Minor changes in the CSM model and in the MATLAB code are required for the model to operate as designed. This will be discussed later.*

About the Example

This reference architecture with an example implementation is provided as a tool for understanding how to manage the airworthiness process with performance data for an aircraft. It displays methods for airworthiness completion through the airworthiness process. It features requirements import and organization, test scenario development, flight test data organization, integration between CSM and MATLAB for data analysis, and assigning satisfy/verify relationships between model elements and requirements. All of this is completed in the digital model to provide a proof of concept for future work in this topic.

Acronyms

CSM – Cameo Systems Modeler

MBSE – Model-Based Systems Engineering

FAA – Federal Aviation Administration

FAR – Federal Aviation Regulations

MIL-HDBK – Military Handbook

Getting Started

To get started, before using the model, the user must download and install CSM and MATLAB to their computer. The user must integrate MATLAB, set up CSM to use MATLAB as the default math engine, and download the MATLAB functions from the model. The following steps are required:

1. Download the model file named “AW_Helicopter.mdzip” and open it in CSM
2. Integrate MATLAB with CSM. Within CSM:
    a. Click the ‘Tools’ menu item and select ‘Integrations’
    b. In the new window, select ‘MATLAB’ and click ‘Integrate’
3. Make MATLAB the math engine for CSM
    a. Click the ‘Options’ menu item and select ‘Environment’.
    b. In the new window, select ‘Simulation’ from the left-hand column.
    c. In the right-hand grid, browse to the section ‘Parametric Evaluator’
        i. Click ‘External Solver Timeout’ and change its value to 90. This should provide enough time to allow the simulation to be initialized and run before a timeout error is activated. If a timeout error is activated, simply increased the timeout value until the simulation initializes and runs without the timeout error.
        ii. Click ‘Default Parametric Evaluator’ and change it to ‘MATLAB’
    d. Press OK to save the changes and close the window

4. Download the MATLAB function files from the CSM model
    a. The MATLAB functions files are located in the ‘Documents’ package and under the ‘MATLAB files’ package.
    b. The files to download are ‘IGE_hover_solve.m’, ‘OGE_hover_solve.m’, and ‘hoverpower.m’
    c. The MATLAB files must be located in the same file directory as the CSM ‘.mdzip’ file

Using the Model

1. In the CSM model, import the requirements by clicking ‘File’ in the menu and select ‘Import From’ and then ‘Import CSV’
2. Save the test data in a location that is convenient. The test data should be in ‘.txt’ file formats
    a. In the CSM model, in the ‘Test’ package and within the ‘Test Data’ package, there are test data blocks.
    b. Change the value properties for each test block to the file path of the test data located on the user’s computer (e.g. C:\...\filename.txt)
3. In the MATLAB functions, ‘IGE_hover_solve.m’ and ‘OGE_hover_solve.m’:
    a. Edit the variable ‘figuresdir’ to a file path of the user’s choosing. This is where the performance charts will be saved to
    b. Save the functions
4. Import the functions into CSM by dragging and dropping the MATLAB function files onto the corresponding parametric diagrams in the model.

'IGE_hover_solve.m' goes on the IGE diagrams and 'OGE_hover_solve.m' goes on the OGE diagram

5. Delete the old constraint blocks and make the necessary port connections. The ports on the constraint block will be labeled the same as the value properties of the blocks they need to be associated to
6. Expand the constraint block and ensure the value property's specification matches the input specification (i.e. 'String' matches 'String', 'Real' matches 'Real')
7. Run the simulation in CSM
    a. While on parametric diagram, left-click and select the 'Simulation' and then click 'Run'
    b. The analysis will initialize and start the math engine, MATLAB
    c. Once the analysis is complete, the results will appear in the 'Variables' tab of the 'Simulation' window
    d. The results can be saved as an instance of the blocks within the 'Variables' tab by selecting 'Export to New Instance'
8. After the math engine has completed its analysis, the user will have a value based on the parameters included. The user can now close the programs. Alternatively, the user can configure CSM to run repeated iterations of the simulation and gather the computed results

Understanding the Process

The high-level description of this process begins with the CSM model. The example of the Dolphin helicopter provided in the model shows a demonstration of the complete airworthiness loop. The loop begins with requirements, which are traced to test scenarios. The test scenarios are executed with flight test and generate test data. The test data is analyzed by the parametric models in CSM. The analysis results and model elements are used to satisfy and verify the requirements; thus, closing the loop. The requirements are a negotiation between the AA and the program office. In some cases, there are requirements from multiple sources, such as FAA, FAR, and MIL-STD documents. For the Dolphin example in the model, FAA, FAR, and MIL-HDBK-516C requirements are included in the airworthiness loop.

Understanding the Code

The status updates form MATLAB will be visible in the simulation window of CSM as it is run, but the actual calculation happens without visual feedback. Once done calculating, MATLAB will shut down immediately. It may take a few moments before the returned values appear in CSM. Any changes to the input variables in CSM will cause the simulation process to re-run.

When debugging, it is best to use a tester file to pass values into the function that CSM will be calling. Feedback from MATLAB only shows in the CSM output window if there is an error. Generic output from MATLAB code, such as statements from a ‘disp’ call, are not shown. Also, this allows use of the MATLAB debugger, something that is not available from within CSM. One error that may be seen is a miscellaneous mix of MATLAB errors that can occur if the ‘External Solver Timeout’ value is not set to a large enough number to allow the external solver to finish. Since it is time-based, it is likely that each time the timeout exception shuts down MATLAB, a different error message for whatever line of code was being executed at that time.

Bibliography

AFLCMC/EZZ. (2020). *USAF Airworthiness Policy anf Implementation Course.* WPAFB: Life Cycle Management Center.

Alvidrez, C. F. (2012). Using Model Based Systems Engineering in Flight Test. *ITEA Journal*, 145-152.

Bleu-Laine, M.-H., Bendarkar, M. V., Xie, J., Briceno, S., & Mavris, D. N. (2019). *A Model-Based System Engineering Approach to Normal Category Airplane Airworthiness Certification.* AIAA Aviation.

Burke, D. A., Hall, C. E., & Cook, S. P. (2011). System-Level Airworthiness Tool. *Journal of Aircraft*, 777-785.

Colombi, J., Miller, M. E., Schneider, M., McGrogan, J., Long, D. S., & Plaga, J. (2012). Predictive mental workload modeling: implications for system design. *Journal of Systems Engineering, 15*(4), 448-460.

Cox, A., & Tortel, F. (n.d.). *Rapport D'Esssais: Performances en vol Stationnaire du Dauphin SA 365 N.* École du Personnel Navigant D'Essais et de Réception.

De Florio, F. (2006). *Airworthiness: An Introduction to Aircraft Certification : A Guide to Understanding JAA, EASA and FAA Standards.* Butterworth-Heinemann.

Department of Defense. (2014). *Airworthiness Certification Criteria.*

Department of Defense. (2014). *MIL-HDBK-516C: Airworthiness Certification Criteria.* Wright-Patterson AFB.

Federal Aviation Administration. (2008). *Fedaral Aviation Regulations Part 29.* Department of Transportation.

Federal Aviation Administration. (2014). *FAA Advisory Circular 29: CERTIFICATION OF TRANSPORT CATEGORY ROTORCRAFT.* Department of Transportation.

Federal Aviation Administration. (2019). *Helicopter Flying Handbook.* Oklahoma City: US Department of Transportation.

Friedenthal, S., Moore, A., & Steiner, R. (2015). *A Practical Guide to SysML: The Systems Modeling Language: Vol. Third edition.* Morgan Kaufmann.

GAO. (2020). *Weapon System Sustainment: Aircraft Mission Capable Rates Generally Did Not Meet Goals and Cost of Sustaining Selected Weapon Systems Varied Widely.* U.S. Goverment Accountability Office.

Holt, J., & Perry, S. (2018). *SysML for Systems Engineering: A Model-Based Approach.* London: The Institution of Engineering and Tecnology.

INCOSE. (2015). *INCOSE Systems Engineering Handbook – A Guide for System Life Cycle Processes and Activities, Version 4.0.* International Council on Systems Engineering (INCOSE).

Miller, W. D. (2017). The Engagement of Systems Engineering with Test and Evaluation. *The ITEA Journal of Test and Evaluation*, 8-11.

Prouty, R. W. (1998). *Military Helicopter Design Technology.* Malabar: Krieger Publishing Company.

Ramos, A., Ferreira, J., & Barceló, J. (2012). Model-Based Systems Engineering: An Emerging Approach for Modern Systems. *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS*, 101-111.

Roper, W. (2020). *There is No Spoon: _The New Digital Acquisition Reality.* Department of the Air Force.

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 074-0188*

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	
02-03-2021	Master's Thesis	August 2020 – March 2021

TITLE AND SUBTITLE

Model-Based Systems Engineering Demonstration of Airworthiness for Department of Defense Rotorcraft

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Bethel, Bretton M., 2d Lt, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way, Building 640
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENV-MS-21-M-205

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Intentionally left blank

10. SPONSOR/MONITOR'S ACRONYM(S)

AFRL/RHIQ (example)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

DISTRUBTION STATEMENT A. APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Airworthiness is a process of certifying that an aircraft can be safety operated within specified bounds. This process is essential to ensuring the safety of the aircraft, its personnel, and the surrounding assets. A Model-Based Systems Engineering (MBSE) approach can be utilized as a method to improve the airworthiness process. MBSE is the methodology of creating and utilizing domain models as a means of exchanging and presenting information for a wide variety of disciplines to understand and replacing previous document-based exchange.
The objective of this research is to develop a reference architecture with a MBSE approach to perform the airworthiness process loop. The model developed features a system model, stores airworthiness requirements and flight test data, performs analysis, and uses analysis outputs to satisfy and verify airworthiness requirements. The reference architecture was applied to a Dolphin helicopter in hover and takeoff conditions to demonstrate the effectiveness. The results of the demonstration provide a proof of concept for the successful implementation of an MBSE approach to the airworthiness certification process.

15. SUBJECT TERMS

Model-Based Systems Engineering, Airworthiness, Parametric Analysis

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			Lt Col Amy Cox, AFIT/ENV
					19b. TELEPHONE NUMBER *(Include area code)*
U	U	U	UU	112	(937) 257-363, ext 4352 Amy.cox@afit.edu

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39-18