**A Framework for Safe System Design in Space Launch Vehicles**

**by**

**Barret William Schlegelmilch**

B.S. Astrophysics, University of California, Los Angeles, 2011

Submitted to the Department of Aeronautics and Astronautics and the MIT Sloan School of Management in Partial Fulfillment of the Requirements for the Degrees of

**Master of Science in Aeronautics and Astronautics**

**and**

**Master of Business Administration**

In conjunction with the Leaders for Global Operations Program at the

**Massachusetts Institute of Technology**

**June 2018**

Signature of Author Signature redacted
Department of Aeronautics and Astronautics
MIT Sloan School of Management
May 12, 2018

Certified by Signature redacted
Jeffrey A. Hoffman, Thesis Supervisor
Professor of the Practice, Aeronautics and Astronautics

Certified by Signature redacted
Roy E. Welsch, Thesis Supervisor
Professor of Statistics and Engineering Systems

Accepted by Signature redacted
Hamsa Balakrishnan, Chair of the Graduate Program Committee
Associate Professor of Aeronautics and Astronautics

Accepted by Signature redacted
Maura Herson
Director of MBA Program, MIT Sloan School of Management

*This page has been intentionally left blank.*

**A Framework for Safe System Design in Space Launch Vehicles**
by
**Barret William Schlegelmilch**
Submitted to the Department of Aeronautics and Astronautics and the
MIT Sloan School of Management on May 11th, 2018 in Partial Fulfillment of the
Requirements for the Degrees of
Master of Science in Aeronautics and Astronautics
and
Master of Business Administration

# Abstract

A hazard analysis for the test firing of NASA's Space Launch System core stage is performed using a systems-based alternative to the traditional reliability-based method. The method used, Systems-Theoretic Process Analysis (STPA), is shown to be a versatile and powerful tool in this application and by extension the development of future space launch vehicles.

The Boeing Company has been selected by NASA as the prime contractor for the Space Launch System (SLS) cryogenic stages. As such, they are working with NASA to develop a comprehensive hazard analysis for core stage test firing and eventual launch operations. Developing, testing, and launching rockets is an inherently complex and high risk endeavor. Preceding the launch itself, one of the highest risk times in the operation of a rocket is the static fire testing, also called a hot fire. Hundreds of parameters need to be monitored in real time in order to ensure the system is operating nominally and equipment damage (and possible injury or death) will not occur. Depending on the point of the testing and the resultant speed at which events are occurring, different levels of automatic safing conditions and operator actions are required to protect the vehicle. Traditionally, the way these safing conditions are derived is through the evaluation of hazard reports, which are themselves based on a "reliability" model: hazards are seen to arise from the failure of individual components and are thus primarily mitigated through increasing component reliability or adding in redundancy. With the level of complexity and required safety of today's launch systems, it is beneficial to evaluate a new approach to identifying the underlying hazards in a system, including ones that arise from unsafe component interactions and not simply failures.

Dr. Jeffrey A. Hoffman,
Thesis Supervisor, Professor of the Practice, Aeronautics and Astronautics

Dr. Roy E. Welsch,
Thesis Supervisor, Professor of Statistics and Engineering Systems

*The author wishes to acknowledge*
*the Leaders for Global Operations Program*
*and the National Science Foundation*
*for its support of this work.*

## Acknowledgements

There are a number of people that made critical contributions to this thesis, without whom this work would not be possible.

First, I would like to thank the entire Space Launch System Team at The Boeing Company. It was a privilege to be able to work on a system so important to the future of human spaceflight with some of the brightest minds in the industry. I would specifically like to thank members of the Systems Engineering, Integration, and Testing Green Run Team – Mike Wood, David Reiniger, Robert Bardwell, Danielle DeVaney, Jeff Newton, Jim Quinlan, and Mason Booth – for their invaluable mentorship and direction.

Second, I would like to thank Professor Jeff Hoffman, Professor Roy Welsch, and Professor Nancy Leveson for helping to guide me to a relevant and important thesis topic in the ever-evolving space industry. Their insights and experience gave me a strong foundation to base my research on.

Third, the support and feedback of my classmates and staff at the MIT Leaders for Global Operations program has been crucial to improving both the quality of this thesis and my own research abilities.

Finally, I would like to thank my friends and family for their patience, feedback, and support throughout this research.

*This page has been intentionally left blank.*

# Table of Contents

# List of Figures

*This page has been intentionally left blank.*

# List of Tables

*This page has been intentionally left blank.*

# Chapter 1

# Introduction

This chapter gives an overview of the motivation for and objective of the thesis. It also provides an overview of the structure of the thesis and the research approach.

## 1.1 Thesis Objective

This thesis aims to compare a systems-based hazard analysis framework called Sytems-Theoretic Process Analysis (STPA) to a traditional reliability-based hazard analysis for the derivation of automatic safety parameters on a new space launch vehicle architecture. The test firing and launch operations of the core stage of the Space Launch System, under development by NASA and The Boeing Company, will serve as a case study to compare the two methodologies in terms of speed and completeness. A discussion of the relative strengths and weaknesses of these two analysis methods will be presented and a recommendation made for which system to apply in future space launch vehicle development projects.

## 1.2 Project Motivation

The Boeing Company has been selected by NASA as the prime contractor for the SLS cryogenic stages. They are also responsible for the full suite of testing of the core stage of SLS - to include a modal test, tanking/de-tanking (Wet Dress Rehearsal), and a static test-fire - collectively known as the "Green Run Test." As such, they are working with NASA to implement automatic safety criteria and procedures for both testing and launch.

The parameters used and their associated limits and resultant automatic actions represent implicit tradeoffs between two major system traits: safety and resilience to spurious protective actions. Safety represents a component of direct equipment cost and a more intangible cost of human life in the case of crewed vehicles [2]. Resilience to spurious trips represents a cost both in terms of equipment and schedule. Designing a safe and robust launch vehicle test program is thus a process of identifying all of the applicable hazards that should result in safety actions and balancing this tradeoff. This is especially difficult when designing vehicles with extremely high cost and safety requirements, such as the National Aeronautics and Space Administration's (NASA) Space Launch System (SLS). Launch vehicles are typically produced in a high-mix/low-volume fashion, making it insufficient to apply the same automatic safety criteria to multiple platforms.

The first step to determining what criteria should result in an automatic safety action is to identify all of the applicable hazards for the system. Historically, hazards have been derived with a reliability-based model. Such models identify the root cause for an accident as a component "failing" and seek to mitigate such hazards through increasing reliability and redundancy of components. However, with today's increasingly complex systems, hazards can be present in a system without any individual component failures. Recognizing this fact is especially critical for extremely dynamic systems such as space launch vehicles, in which their internal conditions can drastically change on the order of milliseconds.

## 1.3 Research Methodology

To determine the effectiveness of a STPA analysis on the launch and test operations of a space launch vehicle, a case study of the Space Launch System Core Stage is performed. The results are then compared with the current industry standard reliability-based approach to inform future space launch vehicle development projects.

The case study focuses on a review of the currently used reliability-based framework, a literature review of applications of the STPA methodology to date, and a system-level study of the core stage components applicable to the STPA analysis. As part of the literature review, the conclusions of several case studies done to date comparing STPA and reliability-based frameworks in other areas of the aerospace industry are presented and used to drive our analysis. The system-level study of core stage components contains several abstractions in order to adhere to ITAR regulations, though its level of detail is adequate for the scope of our STPA analysis.

The STPA analysis is executed on a representative subsystem of the Core Stage, for the hot fire test. The test control interface with the avionics system is chosen due to its centrality to the rest of the system. The STPA analysis is broken down into five major steps: high level hazard identification (for test firing and launch), control model construction, identification of undesirable control actions, identification of causal scenarios, and the derivation of system requirements and mitigations (including automatic abort criteria).

The hazards derived from the STPA analysis are then evaluated in terms of severity and time sensitivity to determine automatic abort criteria. These hazards and recommended automatic abort criteria are then compared to those found from the reliability-based analysis. The differences in man-hours required to perform the analyses as well as the relative overlap and amount of hazards derived are presented to inform future space launch vehicle development programs.

## 1.4 Major Findings

The case study executed in this thesis shows that STPA is a powerful hazard analysis tool that accomplishes everything that a traditional reliability-based analysis does with greater speed, ease, and extensibility. It is particularly suited for the analysis of complex systems, including space launch vehicles. STPA is also able to provide a more complete list of potential system hazards that encompass those derived from a reliability-based analysis, and as such is the preferred basis for the derivation of automatic safety abort criteria.

Despite the apparent benefits of STPA and demonstrated success in many industries, implementation in the aerospace industry can be difficult due to the risk analysis requirements currently established by NASA and the Department of Defense. These requirements drive the development of major systems and the associated institutional knowledge generated dissuades developers from using any other method. Doing so would be considered superfluous when a reliability-based analysis is required by the customer regardless. Establishing a new industry standard, or at least STPA as an acceptable alternative, thus begins with adoption by the major customers in the industry (e.g. NASA, the DoD), which will require demonstrated evidence of STPAs efficacy through an adequate amount of case studies beforehand.

## 1.5 Content Summary

As STPA is a relatively new tool, and its application to this type of system has not been widely studied, a thorough description of both the analysis framework and space launch industry is given first. Chapter 2 gives the history and current state of the space launch industry, as well as a description of how the major organizations in this case study operate with each other. In

Chapter 3, a literature review is presented that details the foundation for the current industry standard for risk assessment and hazard analysis, presents a summary of previous case studies comparing STPA and reliability-based approaches, and gives recommendations of how to implement a systems-based focus in hazard analysis and operations. An in-depth description of STPA, a summary of the major governing documents for current hazard analysis standards, and an overview of the implementation strategy for STPA in this thesis is given in Chapter 4. Chapter 5 presents a case study of STPA applied to the SLS core stage. This case study includes, hazard identification, control model construction, identification of undesirable control actions, causal scenarios, the derivation of system requirements and mitigations (including the development of automatic abort criteria), and a discussion of the results and overall analysis process. Overall conclusions about the effectiveness of STPA for space launch vehicles, as compared to traditional reliability-based hazard analysis methods, are presented in Chapter 6. Additionally, recommendations and lessons learned for application of STPA to future space launch vehicle development are presented.

*This page has been intentionally left blank.*

# Chapter 2

# Background

This chapter explores the history and present state of the space launch industry with a focus on Boeing and NASA in order to give context to the core case study in Chapter 5.

## 2.1 Space Launch Industry Overview

The space launch industry began in 1950 with the advent of national-level launch vehicle programs. Due to the high cost of development and initial lack of commercial customers, these programs were exclusively funded by large governments (initially the US and USSR). Military satellites formed the bulk of payloads until the 1970s when commercial communication satellites began to emerge. Initial commercial launch costs were extremely high based on the fact that the available launch vehicles were all originally developed for large and expensive military payloads. Around this time several other countries started their own space programs, including the Indian Space Research Organization and the European Space Agency (ESA). The ESA provided the initial development funding for the first commercial launch provider, Arianespace, founded in 1980. This increasing international interest in space brought with it a greater demand for cheaper access to orbit.

The standard of government-funded vehicles developed by third-party contractors being the sole method of access to space began to change first in the US with the signing of two key pieces of legislation: the Commercial Space Launch Act of 1984 and the Launch Services Purchase Act of 1990. The Commercial Space Launch Act expanded the available means of launching satellites to space past the Space Shuttle to also include privately operated launch systems. The Launch Services Purchase Act, which states *"…the National Aeronautics and Space Administration shall purchase launch services for its primary payloads from commercial providers whenever such services are required in the course of its activities,"* further incentivized the development of commercial launch vehicles. This set the stage for the global launch market to grow to its current size of approximately $5.5 billion, with the US constituting the largest portion at $2.2 billion [1]. This amount includes both privately developed launch vehicles as well as government-funded platforms.

### 2.2 NASA and SLS

In September 2010, shortly before the official retirement of the Shuttle program in 2011, Congress passed the ``National Aeronautics and Space Administration Authorization Act of 2010." This act stated that NASA must develop a heavy-lift launch vehicle called the Space Launch System by 2016. The act had two critical pieces that framed the SLS project in terms of cost and design: existing contract policy and retention of space shuttle legacy infrastructure.

Congress instructed NASA to maintain existing contracts in the portion of the Act that states *"In order to limit NASA's termination liability costs and support critical capabilities, the Administrator shall, to the extent practicable, extend or modify existing vehicle development and associated contracts necessary to meet the requirements in paragraph (1), including contracts for ground testing of solid rocket motors, if necessary, to ensure their availability for development of the Space Launch System."* In terms of current infrastructure, the Act stated *"The Administrator shall ensure critical skills and capabilities are retained, modified, and developed, as appropriate, in areas related to solid and liquid engines, large diameter fuel tanks, rocket propulsion, and other ground test capabilities for an effective transition to the follow-on*

*Space Launch System."* Congress' granularity of specification for SLS led NASA to the selection of many legacy systems for the SLS architecture which had to be integrated together in new ways. This makes SLS a prime case study for STPA to analyze the technique's flexibility and power to identify new hazards that arise when previously well-studied subsystems are combined into a new system.

SLS production and testing takes place at or near four major NASA centers: Michoud Assembly Facility (MAF), Stennis Space Center (SSC), Kennedy Space Center (KSC), and Marshall Space Flight Center (MSFC). Manufacturing takes place predominantly at MAF, with testing at SSC and MSFC, and final integration and eventual launch at KSC. This distributed operations network makes consistent and exhaustive hazard analysis methods between the involved organizations paramount.

## 2.3 The Boeing Company

NASA took the requirements outlined by Congress and established a plan to develop a vehicle with incremental improvements in capabilities through a series of upgrades ("Blocks"). To meet the requirements outlined by Congress, NASA maintained as many of the pre-existing contracts from the Constellation program as possible. For SLS, NASA has chosen four major prime contractors: Orbital ATK, Lockheed Martin, Aerojet Rocketdyne, and The Boeing Company. The Boeing Company was awarded a 6.5 year $2.8 billion contract for the delivery of two cores as well as the associated hydrogen tanks, oxygen tanks, and avionics [4].

Boeing Defense, Space & Security (BDS) is one of five of The Boeing Company's divisions. Its products include manned and unmanned aircraft programs, space and satellite systems, intelligence and security systems, and integration expertise. BDS is a $30 billion business with approximately 50,000 employees worldwide. It is the prime contractor for the core stage of SLS and is responsible for the manufacturing, testing, and certification of the rocket before delivery to NASA for flight.

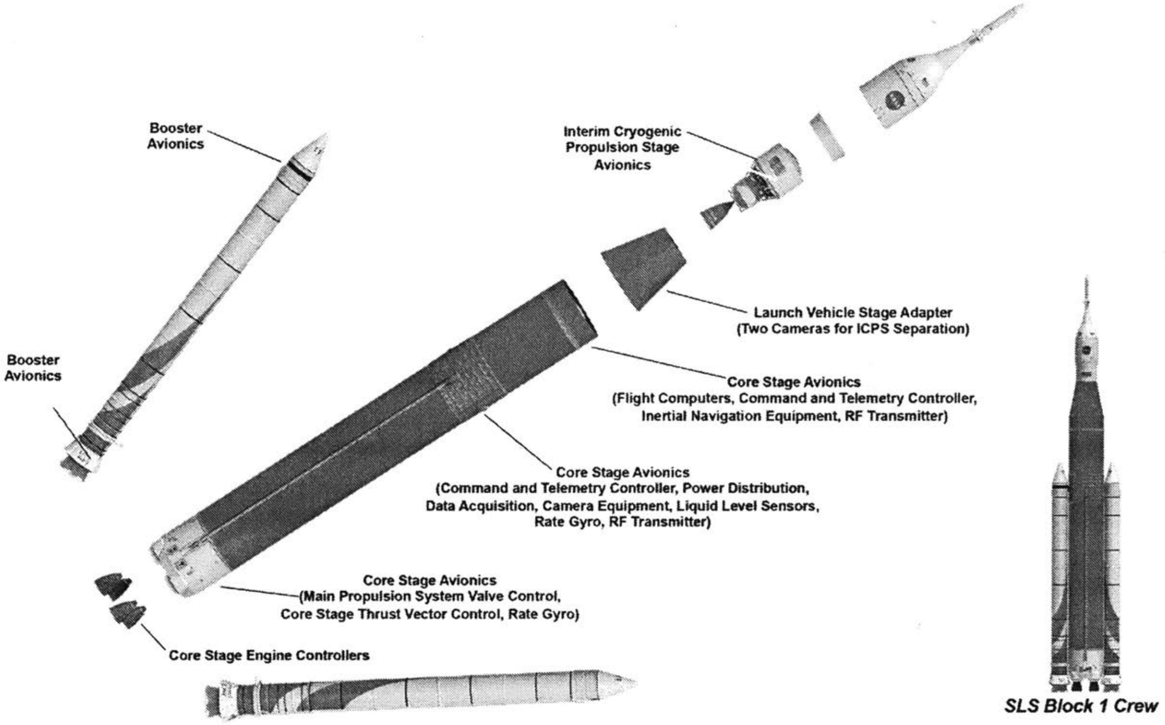

Booster Avionics
Interim Cryogenic Propulsion Stage Avionics
Booster Avionics
Launch Vehicle Stage Adapter (Two Cameras for ICPS Separation)
Core Stage Avionics (Flight Computers, Command and Telemetry Controller, Inertial Navigation Equipment, RF Transmitter)
Core Stage Avionics (Command and Telemetry Controller, Power Distribution, Data Acquisition, Camera Equipment, Liquid Level Sensors, Rate Gyro, RF Transmitter)
Core Stage Avionics (Main Propulsion System Valve Control, Core Stage Thrust Vector Control, Rate Gyro)
Core Stage Engine Controllers
SLS Block 1 Crew

**Figure 1: Space Launch System Block 1 Major Subsystems [7]**

### 2.4 Current Safety Development Standards for SLS

Hazard analysis has been incorporated into the design, development, manufacturing/construction, testing, transporting, maintenance, ground processing, and operations of all flight and ground hardware associated with SLS in accordance with SLS-RQMT-015 ("Space Launch System (SLS) Program Hazard Analysis Requirements"). This document directs hazard analysis to be initiated during the formation and design concept phase and matured throughout the SLS life cycle as designs evolve and operations commence [8]. Combined with SLS-RQMT-014 ("Space Launch System (SLS) Safety and Mission Assurance (S&MA) Requirements), these documents form the basis for the overall SLS program's approach to safety including risk management and probabilistic risk assessment (PRA) procedures [11].

Boeing, along with other major contractors on the SLS project, take these requirements and generate their own governing documents consistent with both their internal processes and the NASA requirements. These internal documents then drive all safety-related processes on the program – thus, Boeing's procedures within this area are directly driven by NASA requirements. Any official change in the framework and methods used to conduct hazard or risk analysis for the purpose of fulfilling S&MA requirements would thus have to happen at the direction of NASA.

*This page has been intentionally left blank.*

# Chapter 3

# Literature Review

This chapter gives an overview of the current state of the art of hazard analysis and risk modeling for space systems. It provides a summary of the major types of models and procedures in use in the space industry as well as the theoretical foundation of STPA.

## 3.1 Current Standards for Assessing Risk

Traditional risk assessment models used to analyze system safety are built largely on reliability-based frameworks. These frameworks form the basis for the current safety and risk assessment standards in use in the aerospace industry as well as the nuclear energy industry, transportation industry, and many other areas. We will start with an examination of the basic views of risk and its role in determining overall system safety first adopted by NASA during the Apollo era of the mid-1960s.

Probabilistic Risk Assessment (PRA) was one of the first formal safety analysis tools that NASA used. Developed with a reliability-based approach to safety, PRA assigns a quantitative probability of failure to a system through an analysis of the failure likelihood of individual components or subsystems. There are two implicit assumptions in this method; the first is the belief that a high reliability systems is inherently safe, and the second is that it is possible to assign an accurate probability of failure to individual components [9]. As was to be shown over the next several decades, neither of these assumptions are necessarily true.

Reliability does not necessarily imply safety; both unsafe systems can be reliable and unreliable systems can be safe. Reliability-based approaches originated with and are suited to simple electro-mechanical systems that do not have complex failure modes. However, in today's complex modern systems, unsafe states can arise with no failure mode present in any of the constituent components, resulting in an accident that is unaccounted for by the model. Conversely, an operator acting outside of procedures to prevent an unforeseen accident would be considered "unreliable" due to not acting according to the system design, but would still be keeping the system safe.

PRA breaks risk down into two quantities; severity and probability. Severity refers to the potential consequences for a given accident occurring, quantifiable in terms like cost of equipment damage or harm to life. Probability is defined as the chance of such an accident occurring per unit time. Of these two quantities, probability is much harder to analytically determine. Usually these probabilities are determined from top down or bottom up methods like Fault Tree Analysis (FTA) or Event Tree Analysis (ETA) respectively. Once determined, the only remaining question is the qualitative one of how much risk is acceptable [2]. These methods themselves must assume individual likelihoods of component failures or specific system states, which are difficult to accurately predict, sometimes giving either unrealistically low or high probabilities of an accident. It was partly for this reason that NASA initially chose not to utilize PRA, only formally adopting it after the Challenger disaster when a quantitative way to assess Space Shuttle mission risk was formally recommended by the Slay Committee [9].

### 3.2 Causality Chain Based Accident Models

The earliest formal accident models can be classified as causality chain models. The Domino Model, as well as the similar Swiss Cheese model, are the first and perhaps most widely known accident models. Many current models used in the aerospace industry today, including fault trees, are based on the concept of causality chains.

Herbert Heinrich's Domino Model, which has been slightly modified since its original inception, postulates that all accidents can be seen as proceeding in the following 5 domino "chain":

1. Unsafe Management Structure (Objectives, Organization, Operations)
2. Operational Errors (Manager Behavior, Supervisor Behavior)
3. Tactical Errors (Employee Behavior, Work Conditions)
4. Accident or Incident (Injury Producing, Property Damage, 'Near Miss')
5. Injury or Damage

In this model, accidents are seen as linear chains that occur as each 'domino' knocks down the next in the sequence. Properly mitigating any of the steps in the sequence (e.g. standard work procedures to prevent unreliable employee behavior) can break the chain.

The Swiss Cheese model, also called the cumulative act effect, was introduced by James Reason and Dante Orlandella in 1990. Instead of the analogy of dominos falling in succession, layers of Swiss Cheese are used and an accident occurs when holes, representing breaches of that layer, align. Common between the two models is the assumption that removing any of the individual 'dominos' or links in the chain will break the sequence and prevent an accident. This encourages a 'defense-in-depth' approach to accident prevention that focuses on using safety measures at each individual layer. As individual component failures are the most common causal events in models like these, two of the most common preventative techniques are improving component reliability and increasing redundancy. An inherent drawback to such causality based models is the oversimplification of complicated processes that can ignore external affecting factors (such as an undue sense of urgency from schedule or financial pressures) that allow such chains to proceed in the first place. They also cannot account for any nonlinear effects, which are extremely important in today's comparatively complicated systems.

### 3.3 Systems Based Hazard Analysis Models

In order to fully analyze all possible causes of an accident, they must not be seen just as a causal chain of events. The focus of such causal chain models, proximate events, are often symptoms of greater underlying issues or causes. A systems-level focus is thus needed to analyze hazards that can arise in today's complex and nonlinear systems. This includes taking into consideration not just the physical components of a complex system, but also the management structure of the organization that operates it, culture in which it is created and operated, the procedures used for its operation, and the design of its associated software.

One way to attempt to include these system level considerations was proposed by William Johnson in 1980. He proposed a model in which systemic factors drove contributory (environmental) factors, which in turn enabled the direct causal factors that resulted in an accident. Though this model incorporates systemic factors, the focus of this is still on an event chain of direct causal factors and thus insinuates that there is an identifiable "root cause," maintaining the focus on proximal events rather than the overall system.

Applying systemic factor analysis to event chain models is insufficient to accurately identify hazards in today's modern complex systems. Any model that focuses on identifying a direct chain of events for explanation of an accident, even ones that implicitly acknowledge the contributing systemic factors, shifts the focus from the analysis of the system as a whole and thus the identification of its unsafe traits. A more accurate systems-based model views safety as an emergent property of a system that arises when its components interact with each other and within the system's environment in a controlled manner. Events in a causality chain would then therefore be seen as symptoms of inadequate control of constraints on safe component interactions rather than the lowest level causes of the accident itself. STPA, examined in greater depth in Chapter 4, uses this idea as the basis for its approach [3].

### 3.4 Hazard Analysis Case Studies

Several previous case studies across a range of industries and applications motivated this thesis. The most applicable ones compare both traditional causality-chain based approaches (PRA, FTA) with new systems-based approaches such as STPA. There is a large body of literature spanning many industries showing the application of causality-chain analysis approaches, including the space industry, but comparatively little showing the application of STPA. Several of the studies most applicable to this thesis are highlighted here.

One of the most relevant studies of a systems-based approach to analyzing a technical and social system in the space industry is the "Demonstration of a New Dynamic Approach to Risk Analysis for NASA's Constellation Program." This study looked to specifically analyze the effects on cost, safety, and performance in NASA's Constellation program of management strategies, key parameters identified through employee interviews (e.g. hiring constraints, schedule pressure), and monitored metrics. A retrospective analysis of the growth of program risk during the Space Shuttle program is included along with prospective recommendations for future space exploration architecture development at an organizational level. This study also includes the development of a system dynamics model for safety-related decision making during the development of space exploration systems [10].

Another relevant case study involves a STPA hazard analysis for the Japanese Aerospace Exploration Agency (JAXA) H-II Transfer Vehicle approach and capture operations with the International Space Station [6]. The STPA analysis was compared with a fault tree analysis for the operations; STPA was able to identify all of the causal factors identified by the fault tree analysis as well as additional causal factors that would be otherwise impossible to find with that technique (particularly integration hazards between the ISS, the H-II Transfer Vehicle, and ground stations and hazards arising from multiple controllers present in the system). This case study also explored the benefits of including a systems-based hazard analysis methodology such as STPA early in the design process in order to proactively develop safety into a system, a further strength over traditional causality-chain methods that rely on a design already being complete to analyze.

### 3.5 Management Techniques to Implement a Systems-Based Safety Focus

A literature review of several case studies dealing with application of STPA in a new setting has provided the following findings. Successfully implementing a systems-based approach to safety, such as STPA, requires two main traits of an organization: first, an organization-wide prioritization of safety, and second, a culture of continuous improvement [5].

There are several techniques that can help promote an organization-wide prioritization of safety. Most critical is its visible adoption by senior leadership. This culture should go beyond just workplace safety and permeate into system design and operations. Creation of an independent safety division (e.g. Safety and Mission Assurance) and vesting of an appropriate amount of power into it is paramount to accomplishing this. Leadership of such a group should have a high status within the overall organization in order to send the message to employees that safety is being prioritized. By encouraging the early and close involvement of safety engineers in the design and development process, costly and time-consuming re-design work is prevented, overall systems are safer, and a stronger focus on safety is instilled across the organization and in the entire lifecycle of the system. Systems-based safety approaches such as STPA lend themselves especially well to this point, as they are able to be executed in parallel with the design and development process. This is critical in the development of space launch systems for which the severity of accidents and cost of re-work is typically very high.

A culture of continuous improvement is important for implementing any type of organizational change, especially one involving the way safety is prioritized or accomplished. Making the goal of implementing the change visible throughout the entire organization is imperative; it should again start with the senior leadership directly communicating the initiative and focus on safety. Clear and measurable incremental goals should be set for the adoption of the new safety standard. Measuring the results of the process (e.g. re-design time saved, man-hour decrease in executing STPA vs previous methods) and effectively communicating them to the organization will increase the buy-in as well. Many organizations are at first resistant to change, but instilling a culture of continuous improvement as described above will allow major shifts in thinking, such as a systems-based approach to safety, to take place.

*This page has been intentionally left blank.*

# Chapter 4

# Analysis Approach – Systems-Theoretic Process Analysis (STPA)

This chapter details the analysis approach for this thesis, STPA. A description of the analysis steps is included, along with the methodology and strategy for implementation within the SLS Core Stage case study in Chapter 5.

## 4.1 Description of STPA Methodology

STPA is built on STAMP (Systems-Theoretic Accident Model and Processes), a systems-based accident model. STAMP views safety as an emergent property of systems that arises when certain constraints on their operation or interaction with their environment are satisfied. When these constraints are not enforced, accidents occur. The model is thus based on system control rather than solely reliability. Though STAMP includes the ability to analyze individual component failures and their effects, it goes further to analyze accidents that arise from component interactions and where no individual failure is present. STAMP lays the foundation for analytical tools including CAST (Causal Analysis using System Theory), which is used for accident/event analysis, and STPA, which is used for hazard analysis.

Within the STAMP framework and STPA methodology, accidents and hazards are defined in the following way:

> **Accident:** *An undesired and unplanned event that results in a loss, including a loss of human life or human injury, property damage, environmental pollution, mission loss, financial loss, etc.*

**Hazard:** *A system state or set of conditions that together with a worst-case set of environmental conditions, will lead to an accident (loss).*

Though similar terms might be used in other industries to mean roughly the same thing (e.g. 'mishaps,' or 'incidents'), the above terminology is inclusive of most related terms used within the realm of safety engineering while still being precise. This terminology will be used for the case study in the following chapter.

STPA uses the basis of STAMP to provide for a systems and control based hazard analysis technique as opposed to the traditional limited in scope reliability based techniques. The ultimate goal of STPA is to identify hazards across the entire system lifecycle (design to operations) and provide the information necessary to properly eliminate or mitigate them. Due to its top-down nature, STPA can be applied at any point in the system lifecycle, even before a design is finalized. This gives it the ability to influence safety constraints and requirements from an early stage and to adapt the hazard analyses to the evolving design. STPA has been found to be able to identify all of the hazards identified by reliability based techniques as well as additional hazards related to the system design, software, or situations where no individual component failure is present.

One important difference between STPA and other hazard analysis techniques is that STPA eschews the practice of assigning a probability of occurrence to hazards. Because STPA takes into account factors for which non-stochastic probabilities cannot be reasonably derived, it does not attempt to predict the resultant likelihoods. Assigning a probability to a hazard can be dangerously misleading – such probabilities are many times based on stochastic processes or derived from a small set of component failure data or no data at all. Such probabilities can thus be off from observed occurrence rates by orders of magnitude and distract safety engineers from what are actually critical flaws in the system. Suspiciously high failure probabilities calculated for a roundtrip human mission to the Moon discouraged NASA from further developing their own quantitative risk assessment tools early in the Apollo program [9]. STPA avoids these issues and instead of generating severity/likelihood matrices, focuses on identifying the underlying control actions that need to be taken to prevent hazards from occurring. Thus instead of defining risk simply as an expectation value (product) of the quantitative probability of an accident and its severity as some High Reliability Organizations do, it is expanded to include the amount of exposure to a particular hazard and the likelihood of that hazard actually leading to an accident (Figure 1).

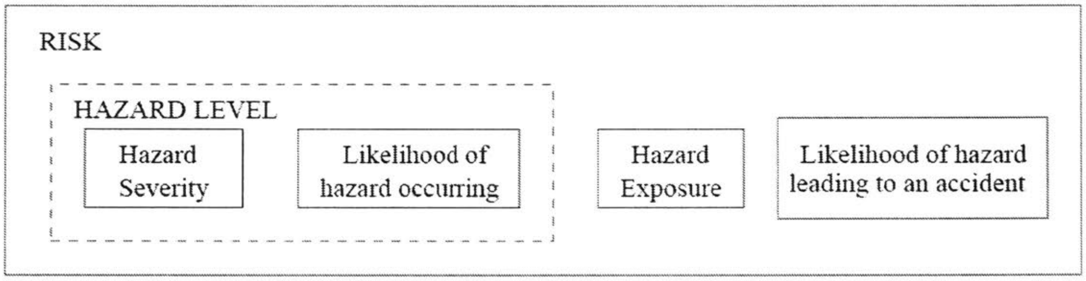

RISK
HAZARD LEVEL
Hazard Severity
Likelihood of hazard occurring
Hazard Exposure
Likelihood of hazard leading to an accident

**Figure 2: Definition of Risk Within STAMP/STPA [3]**

### 4.2 STPA Implementation and Analysis Strategy

The overall process used to complete the case study analysis in Chapter 5 followed multiple steps. As the design for the SLS Core Stage is solidified already, the first step of the process involved gaining a systems level understanding of the rocket in order to perform a well-informed analysis. This process took a single analyst with no prior launch system design or operational experience approximately 40 hours. Materials used included internal technical documentation and component level drawings, along with current testing operation procedures and requirements documents.

After obtaining a foundational understanding of the major components and operations of the system, an STPA analysis was carried out. The overall workflow can be summarized in the following 5 steps:

1. Identify losses to avoid (high level hazards)
2. Model the control structure (establish relationships between subcomponents)
3. Identify undesirable control actions (consequences for improper controls)
4. Identify causal scenarios (possible reasons for undesirable control actions existing)
5. Derive system requirements or mitigations depending on system maturity

Step 1 requires first establishing the boundaries of the system to be analyzed and then identifying the associated high-level hazards of concern. The boundary of analysis should be expanded to include only those components of the system under the control of the responsible engineers. After this is complete, high level accidents for the system are identified. These can be several system-specific examples that fit in to the general definition of an accident given at the beginning of the chapter, usually not more than four. Hazards, which should all be within the chosen system boundary and thus able to be controlled, are then derived from these high level accidents. A hazard, when combined with a worst-case set of conditions external to the system, can possibly lead to one or more of the defined accidents. Each accident should have at least one hazard mapped to it and vice versa.

After identifying the losses to avoid in Step 1, Step 2 involves the first STPA-unique action of creating a model of the control structure for the system to be analyzed. This functional diagram shows the various component relationships in terms of control actions, feedback, and other interactions between each component and thus translates the documented requirements of the system into a graphical format. The analysis can be focused at this point to include as much level of detail of the subsystems as is desired. When analyzing a system with one or more very modular subcomponents, e.g. a rocket's payload(s), these subcomponents can be abstracted to a level that allows interchangeability without the need to completely redo the analysis in the event such a subcomponent is changed. Laying out the model this way in terms of "zoomable" components greatly increases the clarity of the diagram and allows for more focused analysis where desired. An example control loop including a human and automated controller is seen in Figure 3.

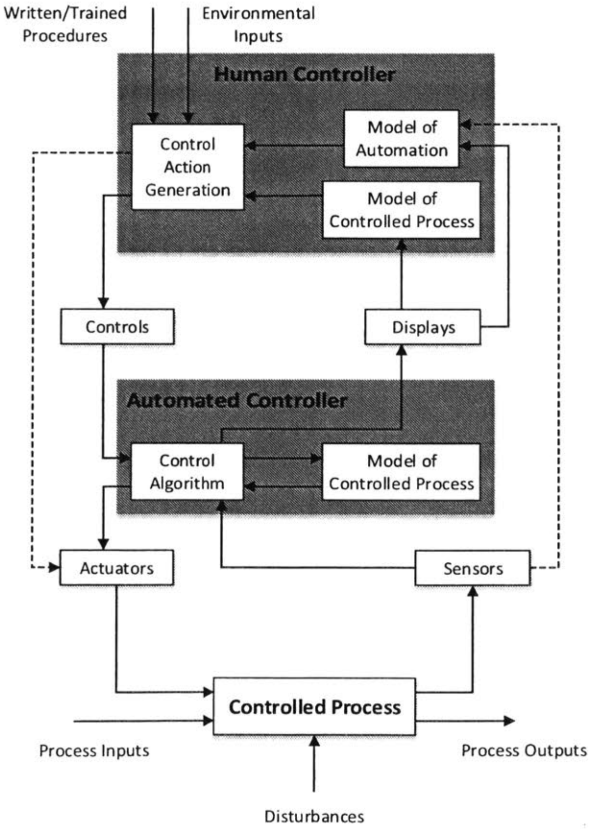


**Figure 3: Generic Control Loop with Human and Automated Controller [5]**

Step 3 uses the functional control diagram created in Step 2 to identify undesirable control actions that could create a hazardous system state. There are five types of unsafe conditions to be considered (note, the term ‘provided’ below can be taken to mean ‘generated’ or ‘issued.’ It can apply to the action of a human operator or electrical/mechanical controller):

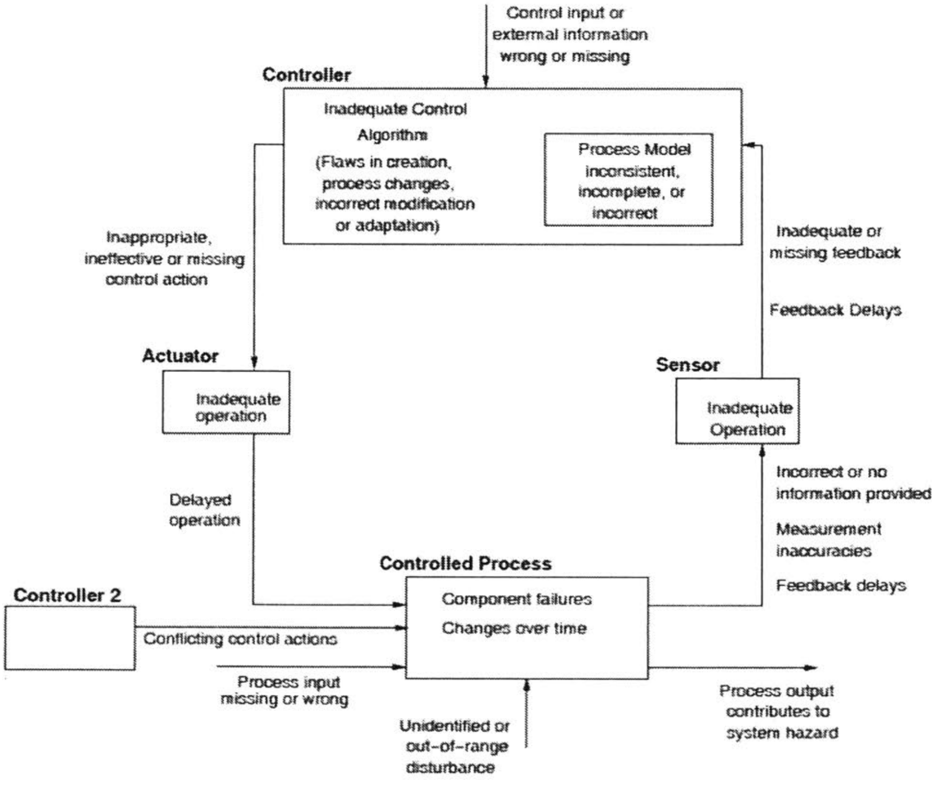
1. A control action required for safety is not provided
2. An unsafe control action is provided that leads to a hazard
3. A potentially safe control action is provided too late, too early, or out of sequence
4. A non-discrete safe control action is stopped too soon or applied too long
5. A safe control action is provided but not followed

The first four unsafe conditions are evaluated in this step and the fifth condition evaluated in the next step. Each control action determined in Step 2 is evaluated for the above unsafe states, with a table format being the most convenient (Table 1). Not every control action will necessarily have an unsafe result in each (or in some cases, any) condition. The derived unsafe results are then able to be used to create preliminary safety requirement statements.

| Control Action | Provided | Not Provided | Wrong Timing or Order | Wrong Duration |
|---|---|---|---|---|
| | | | | |

**Table 1: Example Unsafe Control Action Analysis Table**

Step 4 involves examining the unsafe control actions generated from step 3 in order to establish possible causal scenarios. This step also requires analyzing for the potential cause of a safe control action being provided but not followed as mentioned above. Additional requirements that can help mitigate or eliminate potential causes of hazards are potentially created here as well. Different from a standard FMEA, this process involves looking at not only single component problems or failures, but potential unsafe conditions that can arise from the interaction of several components through the control loops. An example of the different types of potential causes of unsafe control actions within a control loop is shown below in Figure 4.



Control input or external information wrong or missing
Controller
Inadequate Control Algorithm (Flaws in creation, process changes, incorrect modification or adaptation)
Process Model inconsistent, incomplete, or incorrect
Inappropriate, ineffective or missing control action
Inadequate or missing feedback
Feedback Delays
Actuator
Inadequate operation
Sensor
Inadequate Operation
Incorrect or no information provided
Measurement inaccuracies
Feedback delays
Delayed operation
Controlled Process
Component failures
Changes over time
Controller 2
Conflicting control actions
Process input missing or wrong
Unidentified or out-of-range disturbance
Process output contributes to system hazard

**Figure 4: Potential Unsafe Control Actions Within a Control Loop [5]**

After the causal scenarios have been identified in Step 4, design requirements or other possible mitigations (depending on system maturity) are identified to prevent the associated hazards. These requirements should be tied to specific components and referenced to the associated hazard they are designed to prevent. In the case of a system in its initial design phases, this portion of the STPA process can generate requirements that drive changes to the design, allowing this process to be performed iteratively. For systems that have already reached a mature point of design or where re-design is not possible, mitigations can be generated in the form of operations constraints or additions to the system.

# Chapter 5

# Case study of STPA Application to SLS Core Stage Test Firing

This chapter presents a case study of the application of STPA to the SLS Core Stage test firing with specific abstractions made to protect intellectual property and adhere to International Traffic in Arms Regulations (ITAR). This case study is meant to explore the applicability of the STPA method to a generalizable launch vehicle architecture.

## 5.1 Analysis Boundary Definition

The first step of the analysis is to define the boundary of the system. This case study will focus on the hot fire test for the SLS core stage. This test, which involves a simultaneous firing of all four RS-25D engines, is designed to replicate the launch profile of the vehicle and all of the core stage subsystems most work together safely to complete this dynamic procedure. During the hot fire, the core stage is mated to the test stand and does not have the two solid rocket boosters or upper stage attached, and they are thus excluded from this analysis. Additionally, the boundary of this analysis will not include the test management structure or operation procedures, though these can be analyzed using STPA as well.

## 5.2 Test Firing High-Level Accident Identification

With the system boundary defined, the next step is to identify the accidents of concern. The hot fire test is considered a success if all required test data is successfully obtained with no damage to the vehicle, ground equipment, or injury/death of humans. From these success criteria, accidents to be avoided can formally be defined as found in the table below.

| Accident | Description |
|---|---|
| A1 | Human injury or death. |
| A2 | Damage or destruction of the core stage. |
| A3 | Damage, destruction, or interruption of operations of the test stand or other infrastructure. |
| A4 | Test failure. |

**Table 2: Possible Accidents During Hot Fire**

Accident A1 states that the test should not result in injury or death of any humans in any location or at any point of time. Accident A2 is separated from A3 because the core stage is a self-contained system that could potentially be involved in the causation of A3. A3 states that the test should not damage or interfere with the operations of the supporting test infrastructure or any outside systems. Without the performance of the full test and the acquisition of specific required data, the hot fire may have to be reperformed. A4 captures this final case.

After identifying the high-level accidents, high-level hazards within the system boundary are derived. They may lead to an accident when combined with worst-case conditions external to the system boundary. All of the high-level hazards identified are within the boundary of the system and thus should be completely controllable. Each high-level hazard is linked to one or more of the above accidents, and all accidents have an associated precedent hazard. Table 3 contains the high-level hazards defined from the accidents in Table 2.

| Hazard | Description | Associated Accident |
|---|---|---|
| H1 | Humans are exposed to radiated energy or energetic components of the core stage or test hardware. | A1 |
| H2 | The core stage is exposed to radiated energy or loads in excess of its structural rating. | A2 |
| H3 | The test stand or other infrastructure are exposed to radiated energy or loads in excess of their structural rating from the core stage. | A3 |
| H4 | Loss of communication with or control of the core stage. | A2, A3, A4 |
| H5 | Required test data not recorded, transmitted, or properly saved. | A4 |
| H6 | Test aborted. | A4 |

**Table 3: Hazards Derived from Possible Hot Fire Accidents**

H1 sets humans apart from the core stage and related testing hardware. H2 and H3 refer to potential damage to the core stage and the test stand/supporting infrastructure respectively. H4 covers a hazard that could lead to A2, A3, or A4, depending on the timing and amplifying events. H5 and H6 more clearly delineate the conditions that could result in test failure.

High-level system requirements are now generated from the above hazards. These requirements are each tied to one or more of the above hazards and are designed to prevent their occurrence within the system. Table 4 contains the list of derived requirements which assume the design of both the core stage and testing procedure has already been finalized.

| Requirement | Description | Associated Hazard |
|---|---|---|
| R1 | *Radiated energy and energetic components must not be released in the proximity of humans.* | H1 |
| R2 | *The core stage must only be test fired when all initial conditions are met.* | H2 |
| R3 | *Radiated energy and energetic components must not impinge on materials not designed to withstand them as part of the planned testing procedure.* | H3 |
| R4 | *Sensors must record all required data for the duration of the test.* | H4 |
| R5 | *Test data must be transmitted from sensors to supporting recording equipment.* | H5 |
| R6 | *Communication must be maintained with the core stage at all times* | H4, H6 |
| R7 | *Core stage systems must operate within all testing limits.* | H1, H2, H3, H6 |

**Table 4: Requirements Derived from Hot Fire Hazards**

R1 and R3 deal with safety of human bystanders and the core stage and test equipment itself. R2 ensures that initial conditions are met before the test is commenced so that the core stage is operated within limits. R4 relates to the acquisition of required test data specifically. R5 ensures that all required test data is able to be recorded for post-test verification and analysis. R6 ensures that communication with and control of the core stage is maintained at all times so that an abort can be manually initiated if necessary. R7, similarly to R2, ensures that all parameters stay within their normal bounds to preclude the possibility of an automatic shutdown or core stage/testing infrastructure damage.

### 5.3 Control Structure Model

After delineating the above high-level requirements, a control structure model is created to show how the major subsystems involved in the test interact to form safety control structures. Dashed arrows represent non-control physical and informational interactions, while solid lines represent control or feedback relationships. The high-level control structure model is shown in Figure 5. Individual loops can be focused on in more detail, as well as the subsystems themselves. Due to protection of proprietary information and ITAR controls, this case study will focus on this high-level structure, though the process is iterative through each individual subsystem.

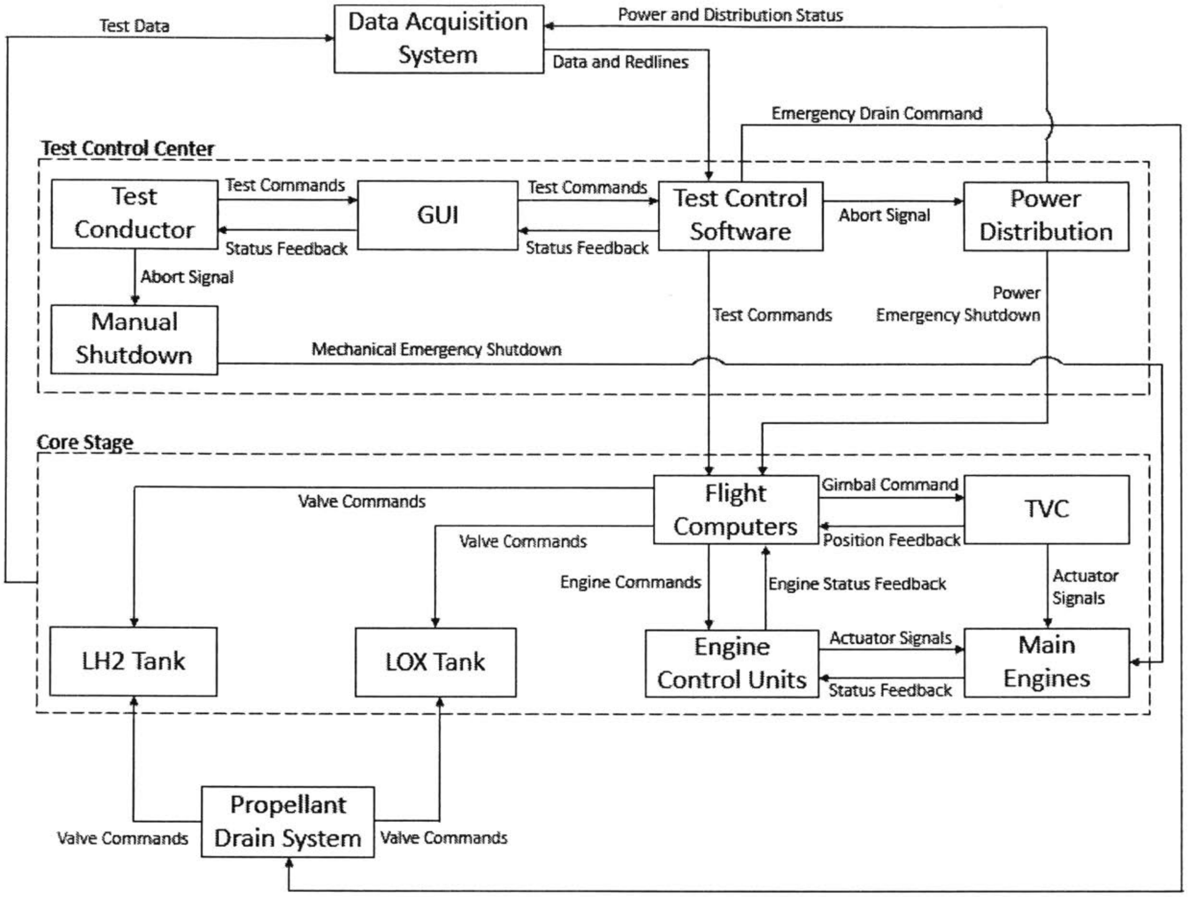


**Figure 5: High-Level Control Structure Model for Hot Fire Test**

### 5.4 Identifying Undesirable Control Actions

With the control structure model created, we next identify the applicable controllers that our analysis will focus on. As a demonstration, 29 undesirable control actions are derived for the Graphical User Interface (GUI) interface with the Test Control Software in Table 5.

| GUI to Test Control Software Actions | Provided | Not Provided | Wrong Timing or Order | Wrong Duration |
|---|---|---|---|---|
| 1. Commence/Abort Hot Fire test profile | UCA 1.1: Test profile is started while humans are present in the test area [H1]<br>UCA 1.2: Test profile is started when parameters are not at nominal levels [H2, H3, H5] | UCA 1.3: Test profile is not stopped when redline values are reached [H2, H3, H4, H5] | UCA 1.4: Test profile is started before data acquisition is ready [H5] | N/A |
| 2. Automatic redline response | N/A | UCA 2.1: Redline response not initiated when redline values are reached [H2, H3, H4, H5] | UCA 2.2: Redline response initiated when no redline values are reached [H5, H6]<br>UCA 2.3: Redline response delayed after a redline vale is reached [H2, H3] | N/A |
| 3. Initiate autonomous abort procedure for Loss of Communication | N/A | UCA 3.1: Abort procedure is not initiated while the vehicle is not under control, resulting in redline values being reached [H2, H3, H5] | UCA 3.2: Abort procedure initiated while communications are still maintained [H5, H6]<br>UCA 3.3: Abort procedure initiation after loss of communications is delayed [H2, H3] | N/A |

| GUI to Test Control Software Actions | Provided | Not Provided | Wrong Timing or Order | Wrong Duration |
|---|---|---|---|---|
| 4. Open MPS isolation valves | N/A | UCA 4.1: Open command signals not provided when required during test profile [H2, H3, H5] | UCA 4.2: Open command signals provided before other systems are ready [H2, H3, H5]<br>UCA 4.3: Open command signal provided too late after other systems have changed state [H2, H3, H5] | UCA 4.4: Open signal is provided too short of a duration for proper test sequence execution [H5]<br>UCA 4.5: Open signal is provided too long after test completion, emptying tanks [H2] |
| 5. Set monitored parameter redline thresholds and nominal ranges | UCA 5.1: Wrong redline thresholds or nominal ranges are selected [H2, H3, H5] | UCA 5.2: Redline thresholds and nominal ranges are not transmitted [H2, H3, H5] | UCA 5.3: Redline thresholds and nominal ranges are set too early, causing a premature abort [H5, H6]<br>UCA 5.4: Redline thresholds and nominal ranges are set too late, resulting in parameters exceeding thresholds with no abort [H2, H3] | N/A |

| GUI to Test Control Software Actions | Provided | Not Provided | Wrong Timing or Order | Wrong Duration |
|---|---|---|---|---|
| 6. Commence/End cryogenic ullage pressurization profile | UCA 6.1: Ullage pressurization is ended before test profile is complete [H5, H6] | UCA 6.2: Ullage pressurization is not commenced when required by test profile [H5, H6] | UCA 6.3: Ullage pressurization is commenced when other systems are not ready [H2, H3]<br>UCA 6.4: Ullage pressurization is commenced too late after other systems have changed state [H2, H3, H5] | N/A |
| 7. Execute engine throttle profile | N/A | UCA 7.1: Throttling profile is not initiated [H5] | UCA 7.2: Throttling profile is initiated too early in the test [H5]<br>UCA 7.3: Throttling profile is initiated too late in the test [H5] | N/A |
| 8. Execute gimbaling profile | N/A | UCA 8.1: Gimbaling profile is not initiated [H5] | UCA 8.2: Gimbaling profile is initiated too early in the test [H5]<br>UCA 8.3: Gimbaling profile is imitated too late in the test [H5] | |

**Table 5: Undesirable Control Actions Between GUI and Test Control Software**

### 5.5 Identifying Causal Scenarios

After identifying the undesirable control actions, the causal scenarios for those control actions are derived. Building upon the guiding logic of Figure 4, possible causes for improper control actions are identified by going beyond simple component failures. Table 6 lists the derived causal scenarios, along with their potential mitigations discussed in section 5.6.

### 5.6 Deriving System Requirements and Mitigations

With the causal scenarios identified, we next develop potential mitigations. In the optimal case, the STPA analysis is performed during the time of system ideation and is thus able to drive iterative design changes early in the development process. However, in this case, the system that is already in a fully developed state and thus redesigns of hardware or major test flow changes are not feasible. We therefore focus on mitigations that are based in operations and procedures. This portion of the analysis benefits greatly from the involvement of multiple people involved with the development of the major subsystems to help brainstorm the most effective mitigations possible. Table 6 contains selected potential mitigations in the rightmost column.

| UCA | Causal Scenarios | Mitigations |
|---|---|---|
| UCA 1.1: Test profile is started while humans are present in the test area [H1] | 1.1.1 The test profile is started because…<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing.<br>Require independent all-clear verification. |
| UCA 1.2: Test profile is started when parameters are not at nominal levels [H2, H3, H5] | 1.2.1 The test profile is started because…<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Required pre-test value ranges defined with associated interlocks. |
| UCA 1.3: Test profile is not stopped when redline values are reached [H2, H3, H4, H5] | 1.3.1 The test profile is not stopped because…<br>• A hardware connection was improperly made.<br>1.3.2 The test profile is not stopped because…<br>• A connector was not installed. | Software health-checks and testing. Layered shutdown methods (manual shutdown backup). |
| UCA 1.4: Test profile is started before data acquisition is ready [H5] | 1.4.1 The test profile is started because…<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Backup data acquisition system available. |
| UCA 2.1: Redline response not initiated when redline values are reached [H2, H3, H4, H5] | 2.1.1 The test profile is not stopped because…<br>• A hardware connection was improperly made.<br>2.1.2 The test profile is not stopped because…<br>• A connector was not installed.<br>2.1.3 The test profile is not stopped because…<br>• The test sequence was incorrectly programmed. | Software health-checks and testing. Layered shutdown methods (manual shutdown backup). |
| UCA 2.2: Redline response initiated when no redline values are reached [H5, H6] | 2.2.1 The redline response is initiated too early because…<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Proper redline limits independently verified. |

| UCA | Causal Scenarios | Mitigations |
|---|---|---|
| UCA 3.1: Abort procedure is not initiated while the vehicle is not under control, resulting in redline values being reached [H2, H3, H5] | 3.1.1 The test profile is not stopped because…<br>• A hardware connection was improperly made.<br>3.1.2 The test profile is not stopped because…<br>• A connector was not installed.<br>3.1.3 The test profile is not stopped because…<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Manual shutdown backup. Redundant watchdog timer. |
| UCA 3.2: Abort procedure initiated while communications are still maintained [H5, H6] | 3.2.1 The test profile is not stopped because…<br>• A hardware connection was improperly made.<br>3.2.2 The test profile is not stopped because…<br>• A connector was not installed.<br>3.2.3 The test profile is not stopped because…<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Manual shutdown backup. Redundant watchdog timer. |
| UCA 3.3: Abort procedure initiation after loss of communications is delayed [H2, H3] | 3.3.1 The test profile is not stopped because…<br>• A hardware connection was improperly made.<br>3.3.2 The test profile is not stopped because…<br>• A connector was not installed.<br>3.3.3 The test profile is not stopped because…<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Manual shutdown backup. Redundant watchdog timer. |
| UCA 4.1: Open command signals not provided when required during test profile [H2, H3, H5] | 4.1.1 The open command is not provided when required because…<br>• A hardware connection was improperly made.<br>4.1.2 The open command is not provided when required because…<br>• A connector was not installed.<br>4.1.3 The open command is not provided when required because…<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Dry run pre-testing validation of command timing. |
| UCA 4.2: Open command signals provided before other systems are ready [H2, H3, H5] | 4.2.1 The open command is provided too early because…<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Dry run pre-testing validation of command timing. |

| UCA | Causal Scenarios | Mitigations |
|---|---|---|
| UCA 4.3: Open command signal provided too late after other systems have changed state [H2, H3, H5] | 4.3.1 The open command is provided too late because…<br>• A connector was not installed.<br>4.3.2 The open command is provided too late because…<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Dry run pre-testing validation of command timing. |
| UCA 4.4: Open signal is provided too short of a duration for proper test sequence execution [H5] | 4.4.1 The test profile is not stopped because…<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Pre-test dry run validation of command timing. |
| UCA 4.5: Open signal is provided too long after test completion, emptying tanks [H2] | 4.5.1 The test profile is not stopped because…<br>• A connector was not installed.<br>4.5.2 The test profile is not stopped because…<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Manual shutdown backup. Low tank level sensor interlock. |
| UCA 5.1: Wrong redline thresholds or nominal ranges are selected [H2, H3, H5] | 5.1.1 The wrong parameter thresholds and nominal ranges are selected because…<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Manual monitoring and shutdown backup. |
| UCA 5.2: Redline thresholds and nominal ranges are not transmitted [H2, H3, H5] | 5.2.1 The redline thresholds and nominal ranges are not transmitted because…<br>• A hardware connection was improperly made.<br>5.2.2 The redline thresholds and nominal ranges are not transmitted because…<br>• A connector was not installed.<br>5.2.3 The redline thresholds and nominal ranges are not transmitted because…<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Manual monitoring and shutdown backup. |
| UCA 5.3: Redline thresholds and nominal ranges are set too early, causing a premature abort [H5, H6] | 5.3.1 The redline thresholds and nominal ranges are enabled too early because…<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Proper redline limits independently verified before enabling. |

| UCA | Causal Scenarios | Mitigations |
|---|---|---|
| UCA 5.4: Redline thresholds and nominal ranges are set too late, resulting in parameters exceeding thresholds with no abort [H2, H3] | 5.4.1 The redline thresholds and nominal ranges are not transmitted because...<br>• A connector was not installed.<br>5.4.2 The redline thresholds and nominal ranges are not transmitted because...<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Manual monitoring and shutdown backup. |
| UCA 6.1: Ullage pressurization is ended before test profile is complete [H5, H6] | 6.1.1 The ullage pressurization is ended prematurely because...<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Pressurization interlock based on other systems status. |
| UCA 6.2: Ullage pressurization is not commenced when required by test profile [H5, H6] | 6.2.1 The ullage pressurization is not commenced because...<br>• A connector was not installed.<br>6.2.2 The ullage pressurization is not commenced because...<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Visible notification for pressurization not enabled. |
| UCA 6.3: Ullage pressurization is commenced when other systems are not ready [H2, H3] | 6.3.1 The ullage pressurization is when other systems are not ready because...<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Pressurization interlock based on other systems status. |
| UCA 6.4: Ullage pressurization is commenced too late after other systems have changed state [H2, H3, H5] | 6.4.1 The ullage pressurization is commenced too late because...<br>• A connector was not installed.<br>6.4.2 The ullage pressurization is commenced too late because...<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Pressurization interlock based on other systems status. |
| UCA 7.1: Throttling profile is not initiated [H5] | 7.1.1 The throttling profile is not initiated because...<br>• A connector was not installed.<br>7.1.2 The throttling profile is not initiated because...<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Pre-test dry run of testing profile (software only). Software interlocks with gimbaling profile. |

| UCA | Causal Scenarios | Mitigations |
|---|---|---|
| UCA 7.2: Throttling profile is initiated too early in the test [H5] | 7.2.1 The throttling profile is initiated too early because...<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Pre-test dry run of testing profile (software only) |
| UCA 7.3: Throttling profile is initiated too late in the test [H5] | 7.3.1 The gimbaling profile is not initiated because...<br>• A connector was not installed.<br>7.3.2 The throttling profile is initiated too late because...<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Pre-test dry run of testing profile (software only). Minimize throttling period to ensure it can be completed during the test fire. |
| UCA 8.1: Gimbaling profile is not initiated [H5] | 8.1.1 The gimbaling profile is not initiated because...<br>• A connector was not installed.<br>8.1.2 The gimbaling profile is not initiated because...<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Pre-test dry run of testing profile (software only). Software interlocks with throttling profile. |
| UCA 8.2: Gimbaling profile is initiated too early in the test [H5] | 8.2.1 The throttling profile is initiated too early because...<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Pre-test dry run of testing profile (software only). Ensure data acquisition system is able to record gimbaling data at any point. |
| UCA 8.3: Gimbaling profile is imitated too late in the test [H5] | 8.3.1 The gimbaling profile is not initiated because...<br>• A connector was not installed.<br>8.3.2 The gimbaling profile is not initiated because...<br>• The test sequence was incorrectly programmed. | Software quality assurance and testing. Pre-test dry run of testing profile (software only). Minimize gimbaling period to ensure it can be completed during the test fire. |

**Table 6: Causal Scenarios and Potential Mitigations**

## 5.7 Discussion of STPA Analysis and Results

The overall value of the STPA analysis as compared to traditional hazard analysis techniques in this case is seen in two main areas: the analysis workload and its 'completeness.'

The analysis workload is unique in two respects as compared to other methods. First, a large portion of the analysis, up to deriving system requirements and mitigations, can be performed with a very small team (or a single person as in this case). The systems based approach encourages a high-level understanding of how overall subsystems interact before diving in to more detailed analysis. This helps lower the initial resources required for the analysis which would otherwise require the coordination of multiple departments to execute. It also requires the analyst to understand not just how one specific piece of the system works, but how things function together as a whole. Second, the total amount of man-hours required for the analysis is significantly less than alternative hazard analysis methods based on the reduced number of personnel required. In this case, the analysis took approximately 23 man-hours (excluding approximately 40 man-hours for the analyst to gain sufficient understanding of the system from available documentation). STPA was also able to identify all of the hazards identified through the causality-based approach while highlighting two redundant cases. The large difference in analysis time is tied to the 'bottom-up' flow of information in the alternative hazard analysis technique. Individual subject matter experts (SMEs) are implicitly tasked with performing the entire analysis on their own because they need to understand how their subsystem impacts the rest of the system. In practice, this manifests itself through siloed teams performing their hazard analysis separately, then meeting to discuss and compare their results which causes redundancy and delays in the process. Approaching the analysis from an integrated systems perspective first and then involving the key SMEs to analyze individual subsystem failure modes and interactions not only ensures a more complete analysis, but drastically reduces the required time.

The strength of the STPA analysis comes from its holistic approach to the system. By focusing on the overall system first and analyzing the component interactions, we were able to draw attention to several potential problem areas in the test that other hazard analysis methods did not emphasize as much. The basis of STPA, from the control structure model on, lends itself as a building block for future hazard analysis of similar tests or operations of the core stage (e.g. integration and launch). This also highlights the value of the analysis for iterative design changes – once the foundational control structure model is created, slight design changes can easily be included. Outside of its use for the STPA analysis, the high-level control structure model serves as an excellent training tool for teaching new team members about the various functions of the system or project in an intuitive way. As many organizations shift to Model-Based Systems Engineering (MBSE) methodologies, control structure models and STPA will find a natural fit due to their domain model-like nature.

*This page has been intentionally left blank.*

# Chapter 6

# Conclusion and Recommendations for Forward Work

Systems-Theoretic Process Analysis is a robust framework with a proven track record in a range of industries and applications. This case study has demonstrated evidence of the feasibility of its application within the space industry, specifically in the realm of launch vehicle development and testing. The key potential benefits seen here and in other case studies include its impact on schedule, low learning curve, scalability, and inclusivity. The main issues that challenge its adoption include slow-changing industry standards and organizational resistance to change.

## 6.1 Benefits

A clear benefit of STPA over traditional hazard analysis techniques is the iterability of the analysis throughout the development lifecycle, combined with saved time and thus schedule flexibility. This saved time stems from the methodical and structured approach of the analysis that does not require subject matter expertise to carry out. By taking a systems-based approach to hazard analysis, subject matter experts can be brought in to brainstorm hazard mitigations and clarify subsystem interactions after the core analysis has been done, rather than each person or group essentially performing a separate hazard analysis on their own.

The STPA analysis is ideally performed by someone with a system-level understanding of the project or product. The pre-requisite knowledge required is minimal as the analyst only needs to understand the major functions and relationships of the subsystems of concern. STPA is broken down into clear and methodical steps that aid in both creating a low learning curve and providing consistency of results between analysts. These strengths can be further amplified through the creation of company or even project specific directives or checklists for how to perform the analysis.

STPA's scalability is another major strength of the analysis. Once a core control structure model has been created, the analysis can be zoomed in or out to any level of controller desired. This both cuts down on total amount of time required for separate hazard analyses and allows the focusing of mitigation efforts to specific portions or functions of the system. The scalability also aids those not directly involved in the analysis with understanding how the system operates at multiple levels in a clear and intuitive way.

One of the most important traits of STPA is its inclusivity as compared to traditional reliability-based hazard analysis techniques. As discussed in section 5.6, all of the hazards identified through the causality-based approach were identified by STPA. STPA is thus able to give the same output of other hazard analysis techniques in a fraction of the time. The method has also been shown to identify additional hazards in many cases that cannot be derived through other techniques, especially in systems at earlier stages of development. [6]

## 6.2 Challenges

A major barrier to adoption for STPA in the space industry, along with any other new hazard analysis procedure or quality assurance tool, is current regulations. Hazard analysis and risk assessment methodologies are performed to the standard of the customer, in this case NASA, and thus would require a change in federal policy to modify. There is a large amount of bureaucracy involved in effecting change at this level that will require continued efforts from those actually responsible for the analyses to overcome. Additional case studies demonstrating the cost savings and power of the STPA process can potentially help with this.

Organizational resistance to change is the other more immediate barrier to successfully implementing the STPA process. Many of the current hazard analysis techniques have been used for decades, and the primary analysts (e.g. S&MA members) are likely to be resistant to significantly changing their processes. There are several tools to help overcome this. One is assigning a small portion of the team to use STPA in parallel with the rest of the team using their usual tools and processes. As more projects are completed, the team can be rotated through this pilot group to see the efficacy of the technique for themselves. The STPA process can then gradually supplant other techniques completely at a rate based on the team's enthusiasm about the pilot program and resultant adoption at the company level.

### 6.3 Forward Work

This case study focused on the implementation of the STPA methodology late in the product development lifecycle. This meant that hazard mitigations had to rely on operational or quality assurance-based approaches rather than fundamental system design improvements. As such, the technique could only be used in a comparative fashion to the current hazard analysis methods and its core strength of enabling iterative safety-centric system design was not utilized. A useful next step would be to apply the STPA methodology at the system ideation or initial design phase to help maximize the potential benefits. A larger body of evidence is needed to prove STPA's efficacy in these cases, in the form of more comparison with traditional hazard analysis techniques or case studies of STPA applied to relevant projects in the space industry.

*This page has been intentionally left blank.*

# APPENDIX A: LIST OF ACRONYMS

| | |
|---|---|
| CDR | Critical Design Review |
| CS | Core Stage |
| ECU | Engine Control Unit |
| FMEA | Failure Mode and Effects Analysis |
| FTA | Fault Tree Analysis |
| GUI | Graphical User Interface |
| HR | Hazard Report |
| IPT | Integrated Product Team |
| ITAR | International Traffic in Arms Regulations |
| KSC | Kennedy Space Center |
| LEO | Low Earth Orbit |
| LH2 | Liquid Hydrogen |
| LO2 | Liquid Oxygen |
| MAF | Michoud Assembly Facility |
| MBSE | Model Based Systems Engineering |
| MPS | Main Propulsion System |
| MSFC | Marshall Space Flight Center |
| NASA | National Aeronautics and Space Administration |
| PDR | Preliminary Design Review |
| RS-25D | Rocket System 25 (Space Shuttle Main Engine) |
| SEIT | System Engineering Integration and Testing |
| STPA | Systems-Theoretic Process Analysis |
| SLS | Space Launch System |
| SME | Subject Matter Expert |
| SSC | Stennis Space Center |
| S&MA | Safety & Mission Assurance |
| TCC | Test Commit Criteria |
| TVC | Thrust Vector Control |

*This page has been intentionally left blank.*

# BIBLIOGRAPHY

[1] "2017 State of the Satellite Industry Report" [Online]. Available: https://www.sia.org/wp-content/uploads/2017/07/SIA-SSIR-2017.pdf [Accessed: 27-Feb-2018].

[2] S. Dick, "Why We Explore" [Online]. Available: https://www.nasa.gov/missions/solarsystem/Why_We_02.html [Accessed: 12-Dec-2017].

[3] N. Leveson, *Engineering a Safer World.* Cambridge, MA: MIT Press, 2012.

[4] D. Leone, "NASA, Boeing Finalize $2.8B SLS Core Stage Contract" *SpaceNews.com*, 3-Jul-2014 [Online]. Available: http://spacenews.com/41139nasa-boeing-finalize-28b-sls-core-stage-contract/ [Accessed: 8-Oct-2017].

[5] N. Leveson, "An STPA Primer: Version 1" [Online]. Available: http://sunnyday.mit.edu/STPA-Primer-v0.pdf [Accessed: 8-Dec-2017].

[6] Ishimatsu et al. "Modeling and Hazard Analysis Using STPA", Proceedings of the 4th IAASS Conference, Making Safety Matter, 19–21 May 2010, Huntsville, Alabama, USA SP-680.

[7] NASA, "SLS Avionics Locations" [Online]. Available: https://blogs.nasa.gov/Rocketology/wp-content/uploads/sites/251/2016/07/Rocketology-Avionics-Locations.jpg [Accessed: 20-Nov-2017].

[8] NASA, "Space Launch System (SLS) Program Hazard Analysis Requirements SLS-RQMT-015" [Online]. Available: https://prod.nais.nasa.gov/eps/eps_data/152816-OTHER-001-005.pdf [Accessed: 12-Dec-2017].

[9] M. Stamataletos et al. "Probabilistic Risk Assessment Procedures Guide for NASA Managers and Practitioners" [Online]. Available: https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20120001369.pdf [Accessed: 10-Dec-2017].

[10] N. Dulac, B. Owens, N. Leveson et al "Demonstration of a New Dynamic Approach to Risk Analysis for NASA's Constellation Program" [Online] Available: http://sunnyday.mit.edu/ESMD-Final-Report.pdf [Accessed: 14-Nov-2017].

[11] NASA, "Space Launch System (SLS) Program Safety and Mission Assurance (S&MA) Requirements SLS-RQMT-014" [Online]. Available: https://prod.nais.nasa.gov/eps/eps_data/152816-OTHER-001-004.pdf [Accessed: 09-Dec-2017].