SYSTEM DEPENDENCY ANALYSIS FOR EVOLVING SPACE EXPLORATION SYSTEM OF SYSTEMS

A Thesis

Submitted to the Faculty

of

Purdue University

by

Christopher T. Brand

In Partial Fulfillment of the

Requirements for the Degree

of

Masters Degree in Aeronautics and Astronautics

August 2020

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF THESIS APPROVAL

Dr. Daniel A. DeLaurentis, Chair

School of Aeronautics and Astronautics

Dr. David Spencer

School of Aeronautics and Astronautics

Dr. Cesare Guariniello

School of Aeronautics and Astronautics

Approved by:

Dr. Gregory Blaisdell

Associate Head of the Gambaro Graduate Program of Aeronautics and Astronautics

ACKNOWLEDGMENTS

First, I would like to thank Dr. Daniel DeLaurentis for encouraging me to undertake this research and for providing many research opportunities that made my graduate school experience extremely rewarding. I would also like to recognize Dr. Cesare Guariniello for his constant eagerness to guide me through this project and for developing the tools on which this research is based. Thank you to my friends and colleagues in CISA, especially Zhong Thai, Prajwal Balasubramani, and Andrew Haines, for helping inject some fun between all the hard work.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

ABBREVIATIONS

SODA	System Operational Dependency Analysis
SDDA	System Developmental Dependency Analysis
SOD	Strength of Dependency
COD	Criticality of Dependency
IOD	Impact of Dependency
SE	Self-Effectiveness
Op	Operability
HLS	Human Landing System
AE	Ascent Element
DE	Descent Element
TE	Transfer Element
PPE	Power and Propulsion Element
HALO	Habitat and Logistics Outpost

ABSTRACT

Brand, Christopher T. MSAAE, Purdue University, August 2020. System Dependency Analysis for Evolving Space Exploration System of Systems. Major Professor: Daniel A. DeLaurentis.

Evolution is a key distinguishing trait of Systems-of-Systems (SoS) that introduces a layer of complexity in analysis that is not present when considering static systems. Some SoS analysis tools exist to determine and evaluate the evolution of an SoS, while other tools are better suited for studying individual instances of an SoS. System Operational Dependency Analysis (SODA) is one such method that has been used previously to study static SoS networks. SODA that has been proven effective in investigating the impacts of partial system disruptions and would benefit from a framework to apply SODA to evolving SoS. This thesis provides an approach to modeling evolving SoS in SODA and presents new data visualization methods to highlight the effects of changing network configurations across evolutionary phases. These visualization enhancements include Failure Impact Range sequence plots to show effects of deterministic system disruptions on capabilities of interest across evolutionary phases, as well as Stochastic Impact plots to quantify the impact of disruptions in particular systems in the context of the probabilistic operating statuses assigned to each system. Integration of SODA and the related method of System Developmental Dependency Analysis (SDDA) is explored to model how operational disruptions and developmental delays might interact and compound during the evolution of an SoS. The SODA enhancements provide decision makers with new information that can be used to explore design and implementation tradeoffs in an evolving SoS under budget and scheduling constraints. These ideas are demonstrated through a case study based on NASA's Artemis program to return humans to the Moon in commercially-built Human Landing Systems (HLS). The HLS concepts proposed to NASA consist of multiple elements that provide distinct capabilities in different phases of the lunar mission, and therefore can be considered an evolving SoS architecture. The operational dependencies of two HLS concepts are modeled across a four-phase lunar landing mission and results are generated using the new visualization methods to highlight the impacts of changing SoS configuration on the performance of key mission capabilities. The development timeline of the first three planned Artemis lunar landing missions is analyzed with SDDA and integrated with SODA results from one HLS concept to explore how developmental delays impact the likelihood of HLS mission completion and how operational failures requiring system redesign impact the program schedule. Connections between SDDA and Integrated Master Schedules (IMS) are discussed to show how SDDA results can be useful in a context more familiar to program managers.

1. INTRODUCTION

Recent developments in commercial and government human spaceflight endeavors are quickly opening up a new space age with visions ranging from space tourism, to returning to the Moon, to colonizing Mars. With these visions come prospects of long term human presence beyond low Earth orbit. This will undoubtedly require new technology developments and space exploration architecture concepts that focus on the long-term sustainability of human habitation away from Earth. But this leap will not all happen at once. NASA's Moon to Mars plan lays out several phases that will gradually see humans expand from largely Earth-dependent operations near and on the Moon to the Earth-independence required for stays on Mars [1]. The spacecraft and other systems needed to complete this plan will be gradually developed and introduced to build off the current infrastructure and expand humanity's reach.

NASA's Artemis and Moon to Mars plans require complex webs of interactions between the involved systems and subsystems to achieve the desired capabilities and complete mission objectives. As systems are introduced or removed, these interactions will compound and shift to provide new overall capabilities. In addition to developmental evolution, the interactions between systems may evolve over shorter operational timescales to more effectively complete mission goals. For example, several of the proposed concepts for Artemis' Human Landing System, meant to return humans to the Moon, involve multi-phase operations that exhibit system interdependency evolution on a smaller scale [2, 3].

Evolution is a key concept in SoS engineering, with Maier listing evolution as one of the five characteristics of SoS [4]. The evolution of an SoS can be described using the Wave Model [5] shown in Figure 1.1. This model splits the development of an SoS into a recurring sequence of steps including evolving the SoS, planning updates, implementing the updates, and continuously performing SoS analysis.

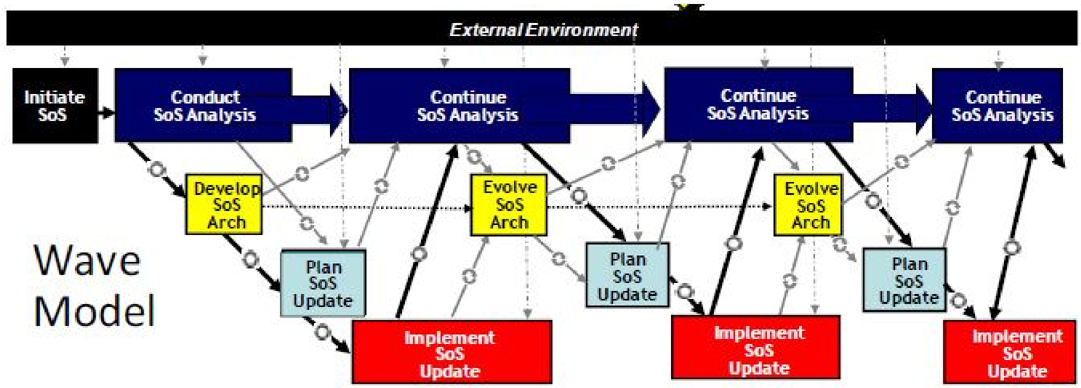

Figure 1.1. Wave Model of SoS evolution [5].

Some methods exist for designing SoS over time, such as Multi-Stakeholder Dynamic Planning (MUSTDO), which optimizes the intertwined decisions made by stakeholders to develop the best current and future SoS capabilities [6]. Once an SoS evolution plan is laid out, additional analysis, such as investigation of system interdependencies, can be helpful for gaining more insight into SoS performance. A method for studying the evolving interactions and dependencies between systems in an SoS such as the Artemis exploration architecture would be beneficial for stakeholders to help plan evolutionary steps to maintain the desired architecture capabilities under various disruption scenarios.

Systems Operational Dependency Analysis (SODA) is a methodology developed by Guariniello that has been demonstrated to be useful for analyzing the impacts of partial system disruptions on the operability of other systems or capabilities of interest in an SoS [7]. However, SODA has not been demonstrated for an SoS architecture that evolves over the course of the analysis. Restricting analysis to a static architecture may require some compromises in the accuracy of the operational dependency network to include all nodes and expected links in a single network. Expanding analysis to multiple phases would allow for more accurate representation of node interdependencies over the lifecycle of the SoS (Figure 1.2).

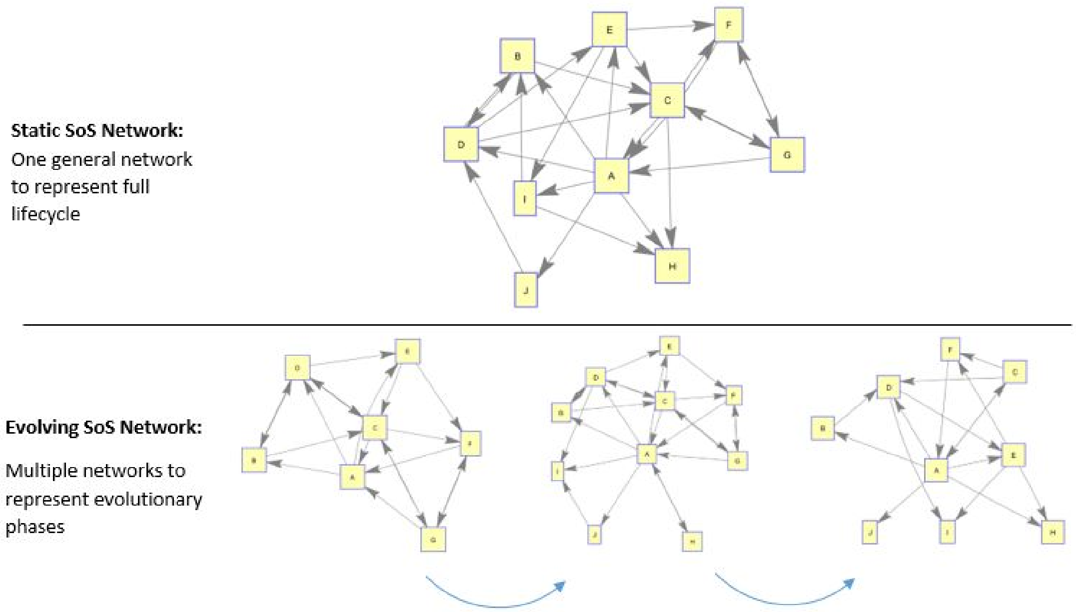

Figure 1.2. Considering the evolution of the SoS opens opportunities for more insightful analysis of node interdependencies across the lifecycle of the SoS.

1.1 Research Purpose

This research strives to develop a framework for analyzing evolving SoS architectures using SODA and investigate new types of conclusions that can be made when considering an architecture as it evolves over time. The scope is limited to analyzing the dependencies of architectures that have already been planned and that follow a defined evolutionary path. Specifically, this research answers the following research questions:

1. What enhancements to SODA's methodology and analysis tools are needed to consider the changing configurations of evolving SoS architectures and provide new insights and value to SoS designers and managers?

2. Do these enhancements provide new useful results for SoS decision makers when applied to a Artemis HLS case study?

The contributions of this work are as follows:

1. Definition of approach to SODA model setup for evolving and multi-phase architectures.

2. Development of new analysis and visualization methods that highlight the effects of configuration changes over the architecture's evolution.

3. Integration of SODA with Systems Developmental Dependency Analysis (SDDA) to investigate how developmental and operational dependencies interact as an architecture evolves.

This thesis begins with a review of the current state of the SODA methodology. Other work on dependency analysis of evolving SoS architectures is then reviewed for insights into how SODA may be enhanced. Next, several enhancements of SODA are discussed in comparison to current methodology and analysis tools. Finally, a case study is presented using the multi-phase lunar mission of proposed Artemis Human Landing System concepts. The scope of the case study is then expanded slightly to include the development of some elements of the Artemis program to demonstrate integration of SODA and SDDA for an evolving architecture.

2. LITERATURE REVIEW

2.1 SODA

System Operational Dependency Analysis is an SoS methodology for investigating how failures and partial disruptions of systems propagate through system interdependencies to impact SoS performance. It utilizes a network representation of an SoS in which systems or capabilities are represented as nodes and directed links between the nodes indicate operational dependence. Parameters assigned to each link dictate how disruptions propagate through the network. A small example network is shown in Figure 2.1. SODA can be used to determine which nodes in an SoS or complex system are most critical to the performance of nodes of interest and can also be used to generate SoS-level performance metrics such as robustness and resilience against disruption [7].

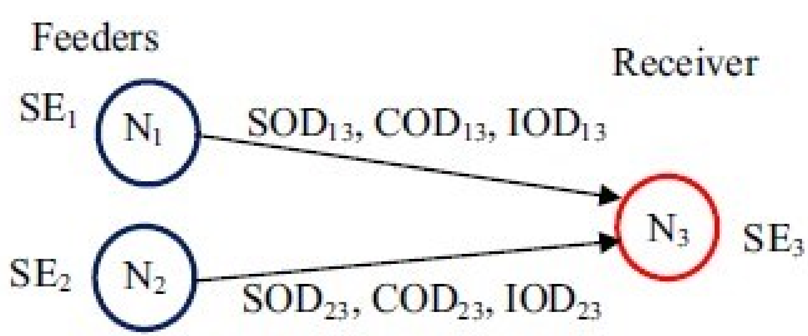

Figure 2.1. Example SODA network with two feeder nodes and one receiver node [7].

SODA considers dependencies between systems that can generally be defined a feeder node providing information, material/matter, or energy to a receiver node.

Each node in the network has an operability value that is a function of that node's internal operating status, or self-effectiveness (SE), and the operability of all feeder nodes from which it receives inputs. Disruptions in a node, represented as a decrease in SE, can impact the operability of the disrupted node and any nodes that are directly or indirectly dependent on the disrupted node. Guariniello notes that operability can be related to system performance but the two are not equivalent, and that the meaning of operability must be defined explicitly for each system [7]. A feeder node's impact on the operability of a receiver node is dictated by a piece-wise linear function defined by three parameters: strength of dependency (SOD), criticality of dependency (COD), and impact of dependency (IOD). SOD describes the decrease in operability of the receiver node (O_j) given a small reduction in the operability of the feeder node (O_i), with a higher SOD value indicating a steeper decrease. COD describes the maximum reduction in O_j that is achieved when O_i reaches zero. IOD represents the rate of increase in O_j when O_i is increased from a low value. These definitions are summarized in Table 2.1. A sample piece-wise operability relation is shown in Figure 2.2.

Table 2.1. Definition of SODA terms

Self-effectiveness (SE)	Internal operating status of a node
Operability	Operational status of a node considering its SE and inputs it receives from other nodes
SOD	Parameter describing the effect of small disruptions in a feeder node on the operability of the receiver node
COD	Parameter indicating the loss of operability in the receiver node that results from a complete loss of operability in the feeder node.
IOD	Parameter describing the effect of increasing the operability of the feeder node from low values

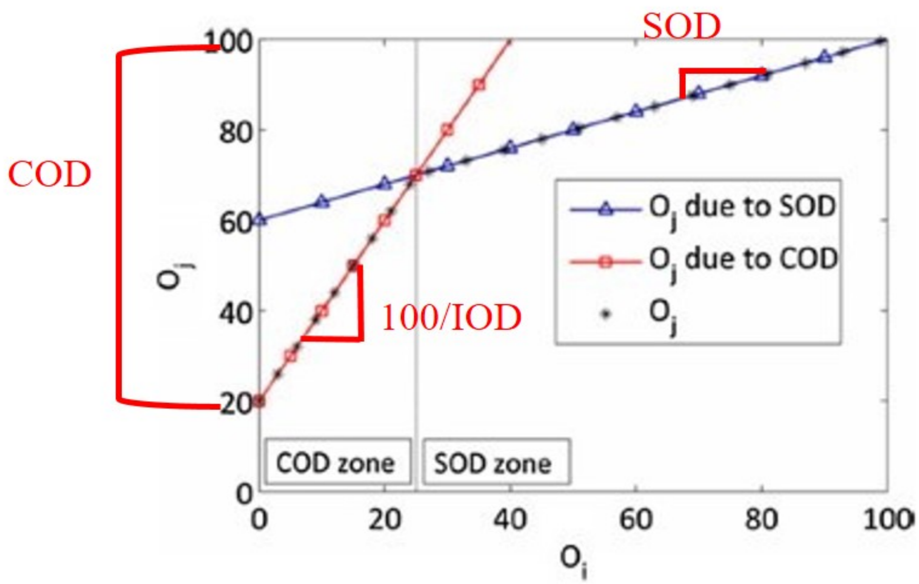

Figure 2.2. Example Piece-wise Operability relationship based on SOD, COD, and IOD [7].

2.1.1 Comparison of SODA to other Failure and Dependency Analysis Methods

SODA is partly based on Functional Dependency Network Analysis, which was created by Garvey to incorporate concepts of Failure Modes and Effects Analysis (FMEA), dependency analysis, input-output models, and other methods [8]. A key differentiator between SODA and traditional FMEA analysis is the focus on partial disruptions rather than binary failures. The ability to consider partial disruptions allows for more nuanced analysis into how system dependencies interact under different disruption scenarios. This information can be more representative of real-world situations and therefore more useful to designers and engineers [7].

2.1.2 Deterministic and Stochastic Analysis

SODA can be used to analyze a network with either deterministic or stochastic information for node functionality. In deterministic analyses, the SE values of all nodes are selected and the dependency model is used to calculate the resulting operability of all nodes. The impact of disrupting various sets of nodes on the operability of nodes of interest can be visualized with a Failure Impact Range (FIR) plot. This bar plot gives a succinct summary of the impact that disrupting a given system has on the operability of a system or capability of interest. The Disruption Impact Matrix (DIM) was later introduced by Guariniello et al. to show deterministic results for the impacts of disrupting each node on all other nodes in a single view [9].

A limitation of the deterministic analysis is that it only provides information about the specific disruptions that the user enters into the model. The stochastic capability of SODA addresses this issue by allowing each node's SE value to follow a probability distribution. Then a Monte Carlo analysis is conducted to provide probabilistic results for the operability of each node. As opposed to the deterministic analysis, these results do not give information for specific disruptions, but provide generalized information such as the expected value of operability for each node. The expected operability can be used to identify overall weaknesses of the architecture, and expected operability of key nodes can be used to compare the suitability of alternative architectures.

2.1.3 Strengths and Limitations

The main strength of SODA is that it can provide information regarding the impacts of partial disruptions when complex simulations aren't practical and when details of system or subsystem performance are sparse [7]. SODA is best applied at an early stage of development to give architecture designers a preliminary understanding of the dependencies within a given architecture. As more detailed information becomes available, SODA models can be revised with more precise dependencies between nodes. SODA is meant to complement other systems engineering tools like Probabilistic Risk Assessment (PRA) rather than replace them.

A current limitation of SODA, and the focus of this thesis, is the lack of demonstration on an architecture that has evolving configurations. An approach must be defined for handling changing configurations and new analysis tools are presented to highlight the impacts of the evolution.

2.1.4 Applications of SODA

SODA has previously been used to analyze various static architectures that maintain the same node-link configuration throughout the analysis [7, 9, 10]. Guariniello analyzed how operability of a satellite constellation can be maintained or improved over the lifespan of the constellation by replacing modular disrupted components in individual satellites, however the network topology remained constant [10]. Guariniello also explored SODA's applicability to quantifying resilience through limited cases of architecture reconfiguration given disruptions [10]. With the exception of the investigation of resilience, previous applications of SODA have kept the systems and inter-system dependencies constant throughout the analysis, only adjusting the self-effectiveness of systems over time to observe the evolution of operability. Previous analysis of space exploration architectures, such as human expeditions to Mars, have been considered as static architectures, with all systems and dependencies present at one time [10, 11]. This approach may miss details inherent in the time-dependent nature of the architecture, such as dependencies that move or change characteristics over the course of the mission.

2.2 Approaches to Evolution and Reconfiguration

As the concept of architecture evolution is central to this thesis, it is important to distinguish evolution from the related concept of reconfiguration. Nielsen et al. describes evolution as changes over the SoS's lifespan in "the functionality delivered, the quality of that functionality, or in the structure and composition of constituent systems" [12]. On the other hand, dynamic reconfiguration is defined as the "capacity of an SoS to undertake changes to its structure and composition, typically without planned intervention" [12]. Evolution and dynamic reconfiguration differ in that evolution refers to planned changes over a longer time scale while reconfiguration refers to the ability of an SoS to change its composition on-the-fly during operation in response to its environment.

Reconfiguration in dynamic SoS architectures has been studied by Moshin et al. with the goal of determining the impact of uncontrolled, stochastic reconfigurations on SoS performance [13]. They consider dynamic architecture changes that fall into four categories:

1. Addition - add new constituent system to the SoS
2. Removal - remove an existing constituent system from the SoS
3. Replacement - remove a constituent system and replace it with a similar one
4. Rearrangement - dissolve the complete architecture and rebuild it in a new arrangement

These categories could also be reasonably used to describe the evolution of an SoS, with planned and deliberate adjustment of constituent systems over time in the four ways described.

While "evolving" can generally be used to describe the kind of planned configuration changes I am considering, I will distinguish between evolving SoS and multi-phase SoS. Evolving SoS refers to an SoS that goes through planned configuration changes over a developmental time frame, whereas a multi-phase SoS goes through planned configuration changes over an operational time frame.

Several authors have explored using dependency analysis methods to evaluate evolving SoS. Francis proposes an ontology for probabilistic analysis of reconfigurable systems with analysis based on FDNA and Bayesian Networks [14]. The ontology includes a link transition matrix and connection possibility frontier, which describes the probability of retaining a link between nodes given a shock and indicates possible nodes a given node may connect to after a shock, respectively. Francis also defines fragility curves to capture the probability of systems failing under a threat. This is similar to the stochastic representation of self-effectiveness in SODA, although the SE distribution is considered constant and is not conditional on which threats or disruptions occur [7].

Mane et al. [15] used Markov chains to represent the developmental dependencies between systems in a network and analyzed the ability of alternative networks to propagate or arrest delays. The results are based on the probability of each system propagating its delay to dependent systems.

Agarwal et al. [16] developed an approach called FILA-SoS for modeling the evolution of the systems and interdependencies that make up an SoS. This allows for the evaluation of alternative acquisition and SoS evolution plans to achieve a desired capability, but it does not consider the impact of operational or developmental disruptions on performance.

3. SODA ENHANCEMENTS FOR EVOLVING AND MULTI-PHASE SOS

This section describes extensions to the SODA methodology and analysis tools that can be used to investigate complex systems and SoS that evolve along planned trajectories through a series of operational phases. Emphasis is placed on planned evolution rather than dynamic reconfiguration because of its general applicability to assisting in the design and development of SoS, as well as the current lack of capability of SODA to address these types of problems.

3.1 SODA Approach and Setup for Multi-phase and Evolving Architectures

Before analysis of an evolving architecture can be conducted, the distinct phases that will be analyzed must be defined. The concept of stable intermediate forms can be helpful for determining which configurations might constitute interesting operational phases. Stable intermediate forms are distinct configurations of a developing SoS that provide a certain level of performance towards the SoS objectives [4]. These forms are meant to provide a structure for incrementally realizing the full performance of the SoS in manageable chunks. A developing SoS may be constantly evolving due to small changes, which will eventually combine into some new capability defined as the next stable intermediate form. To obtain the most value from SODA analysis of evolving architectures, stable intermediate forms that introduce distinct new capabilities should be considered for the evolutionary phases.

To further define configurations of an evolving SoS in the context of the SODA methodology, the four categories of dynamic reconfiguration modes can be adapted from Moshin et al [13]. In addition to the changes to constituent systems described by Moshin et al., the SODA dependency parameters can be adjusted to create a new configuration. So a new configuration can be determined by:

1. Addition - add a node or dependency link to the network

2. Removal - remove a node or dependency link from the network

3. Replacement - alter the stochastic SE distribution of a node or alter the dependency parameters of a link

Rearrangement is not included here because it is unlikely that a SoS will undergo a dramatic reconfiguration during planned evolution across stable intermediate forms.

A basic addition to current SODA implementations is the extension of the code framework to handle multiple dependency adjacency matrices to represent the different phases of the evolving architecture. To keep consistency between all phase representations and allow for easier comparison of disruptions between phases, the adjacency matrix for each phase includes all nodes considered over the full evolution of the architecture. Nodes that are not included in a given phase are grayed out. Nodes are grouped by parent (eg. all subsystems grouped under associated system) to allow for easier tracing of dependencies within an element (along the diagonal) and between elements (off the diagonal). Capability nodes representing the overall performance or objectives of each phase are also included.

It should be noted that multiple matrices are only needed if "internal" nodes or dependencies change between phases. If only the dependencies feeding into leaf nodes (nodes with no outgoing links) change between phases, then multiple versions of the leaf node can be included in a single adjacency matrix. The impact of system disruptions can then be assessed on each leaf node individually.

3.2 Enhancements to Visualization of SODA Results

3.2.1 Failure Impact Range (FIR) Sequence Plots

The FIR plots for deterministic results provide one possible area of extension for suitability to evolving architectures. An example of a FIR plot is shown below in Figure 3.1. The plot shows different bands of operability of the node of interest based on the self-effectiveness (y-axis) of each disrupted node or set of nodes (x-axis). The bars are ordered from left to right with decreasing impact on the nodes of interest. These plots are useful for quickly assessing the amount of impact that disrupting certain nodes to various levels of SE has on the operability of nodes of interest. They can also be used to investigate compound disruptions of node sets that may produce more significant impacts than individual disruptions.

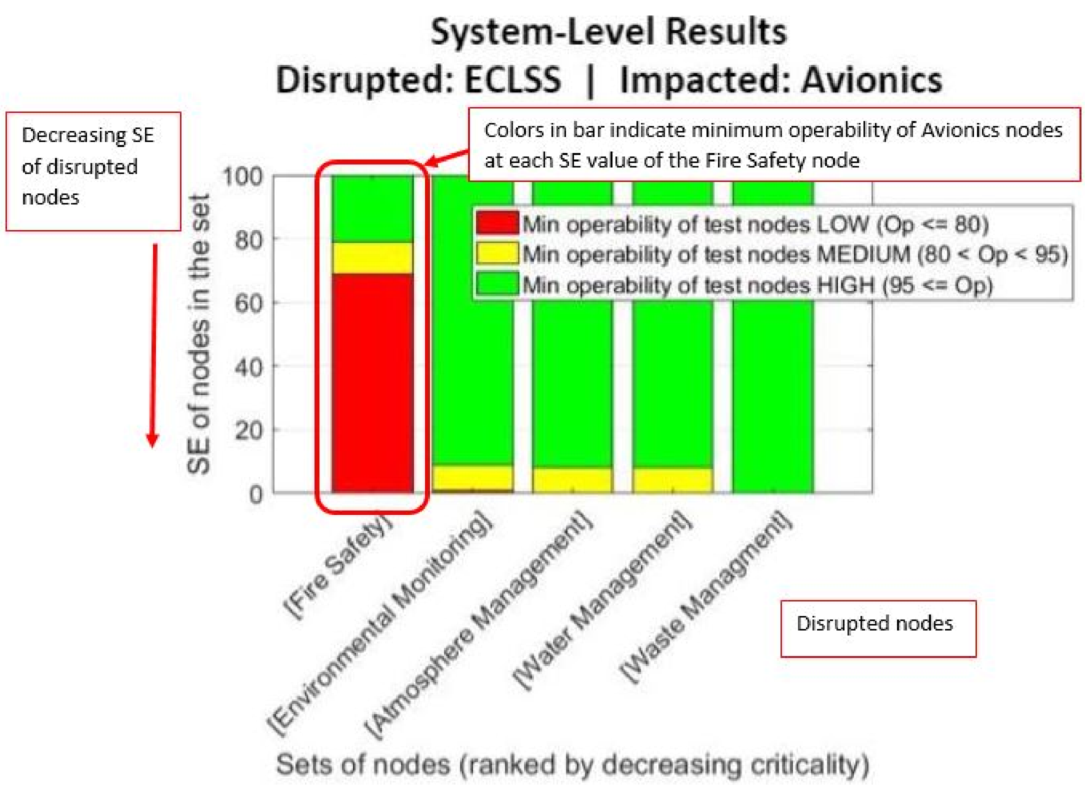

Figure 3.1. FIR plot example from a previous study of the Gateway Habitat module [9]

.

To analyze the impact of disruptions over multiple phases in a way that allows for easy comparison between phases, a sequence of FIR plots can be created. Rather than showing the impact of disruptions on arbitrary nodes of interest, these FIR sequence plots focus on the impact to the capability nodes defined for each phase. Each plot in the sequence shows the impact of disruptions in a group of nodes on the different capability nodes for each phase. The sorting of bars by decreasing impact can be removed for easier comparison of the impact of disrupting a particular node over multiple phases, or the sorting can be kept to show how changing the capability node of interest adjusts the relative criticality of disrupted nodes. For example, the sequence might show that disruptions to System A has a large impact on the capability node for Phase 1, but that the same disruption has a small impact in Phase 2 due to changes to the dependencies between phases. The last plot in the sequence shows a weighted impact of disruptions in each tested node across all phases. The weights given to each phase can be adjusted to provide succinct measures of disruption impact when some phase capabilities are deemed more important than others. Also, impacts of disruptions that occur partway through the evolution of the architecture can be investigated by setting the weights of the previous phases to zero. This can assist architecture designers in recognizing where and when each node is most important so efforts can be taken to minimize the likelihood of disruptions in those conditions. An example FIR sequence plot is shown below in Figure 3.2.

3.2.2 Stochastic Impact Plots

A limitation of the FIR sequence plots is that they show only deterministic results, meaning that all nodes other than those being disrupted are assumed to have SE values of 100. This is likely not the case in real world operations as nodes may not be operating at full potential and may experience random disruptions. To address this, probability distributions can be assigned to the SE of each node and a stochastic Monte Carlo simulation can be run across many instances of the network

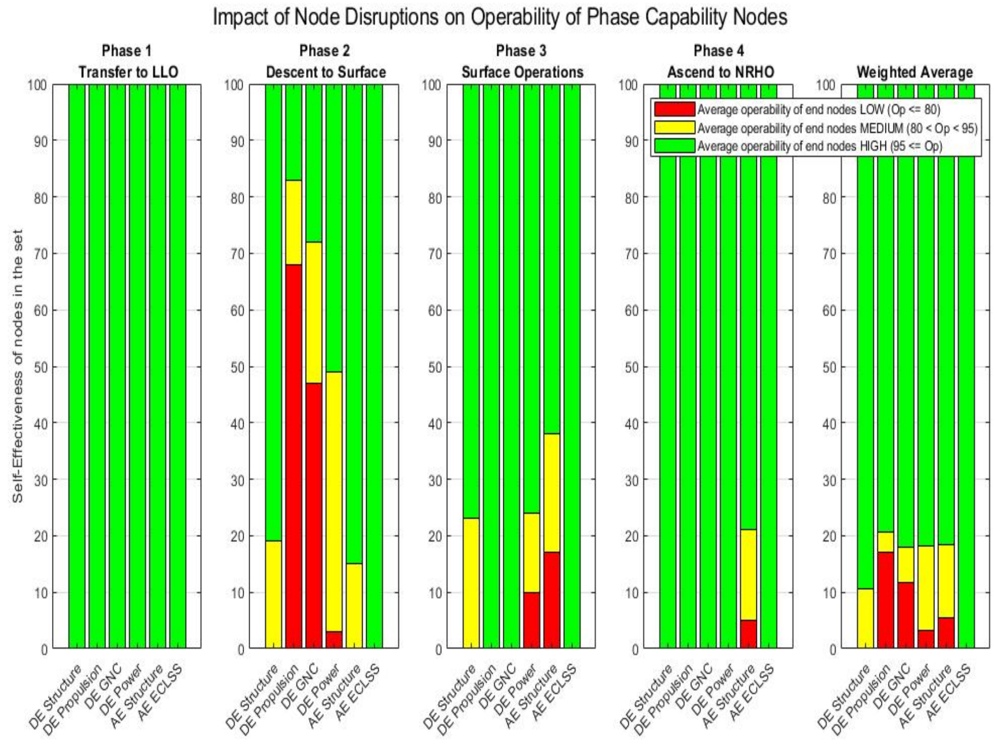

Figure 3.2. FIR Sequence plot example

to determine the expected operability of each node. These results have been used to create operability histograms for all nodes in the architecture, which show how likely each node is to maintain a high level of operability in the face of arbitrary disruptions. However, these histograms can easily become difficult to decipher when many nodes are included, and they provide no information about the impact of disrupting a particular node while other nodes remain stochastic. These results are also not suitable for evolving architectures in which the SE probability distributions given to each node and the interactions between nodes may change across phases of the SoS evolution.

To address the limitations of the current stochastic analysis, a process for determining a single-valued stochastic impact metric is proposed. This can be thought of as a measure of risk, which can be defined as the product of the consequence of a disruption and the likelihood of the disruption occurring. First, a selected node is disrupted deterministically from SE = 100 to SE = 0 while all other nodes remain stochastic. For each deterministic SE value, many instances of the dependency network are computed to produce a mean operability of the node(s) of interest. As with the FIR sequence plots, the nodes of interest here are considered to be the capability nodes associated with each phase, but mean operability for any stochastic node can be considered. Plotting the mean operability as a function of the SE value of the deterministic node produces a line graph that represents the sensitivity of SoS operability (represented by the node(s) of interest) to disruptions in the selected node. Figure 3.3 shows an example of such a plot. The horizontal red line indicates the mean operability of interest when the deterministic node is fully operational, while the vertical arrows represent the impact of disruptions in the node to various SE values. Integrating the area above this curve and below the horizontal red line gives a single value representing the overall impact of disruptions to this node in the context of the stochastic network.

However, this does not consider the likelihood of these SE values occurring in the disrupted node. The impact to mean operability of disrupting the chosen node to a given value of SE is multiplied by the associated probability of that SE value occurring. These products are then summed to give a single value capturing both the impact and likelihood of the SE disruption. The full equation for stochastic impact is given in Equation 3.1.

$$StochasticImpact = \sum_{SE=0}^{100} Impact(SE) * Probability(SE) \tag{3.1}$$

Figure 3.4 shows the probability of a node having an SE value between 79 and 80 based on its probability distribution. Probability distributions can be assigned to nodes based on literature of failure rates for existing similar systems or by consulting with experts. For spacecraft, systems generally have very high reliability and therefore should have SE distributions that are strongly skewed to the right. Systems that are more exposed to the space environment, such as hull and solar arrays, can be

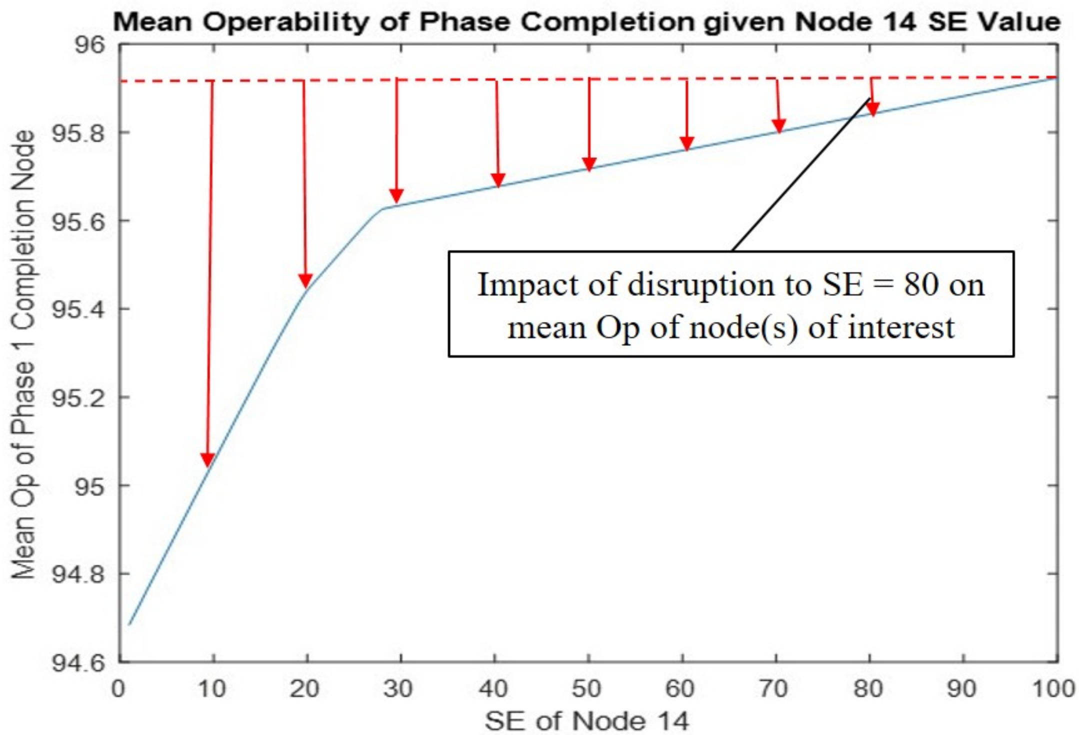

Figure 3.3. Mean Operability vs SE plot example

given SE distributions with slightly lower expected values to represent their increased likelihood of being disrupted due to degradation and other environmental effects. If no information is known about the SE probability distributions, a uniform distribution can be assigned.

Similarly to the results from the FIR sequence plots, these values can be compared between nodes within a phase or the stochastic impact of a node can be compared across phases as node SE distributions or interdependencies change. The difference is that FIR plots only show the individual impact of a disruption, while stochastic impact plots given holistic results on the sensitivity of SoS performance to disruptions in a certain node. One such plot is shown in Figure 3.5. This provides a very simple visualization that can help identify the most impactful nodes on the operability of nodes of interest in the context of the stochastic network. The stochastic impact

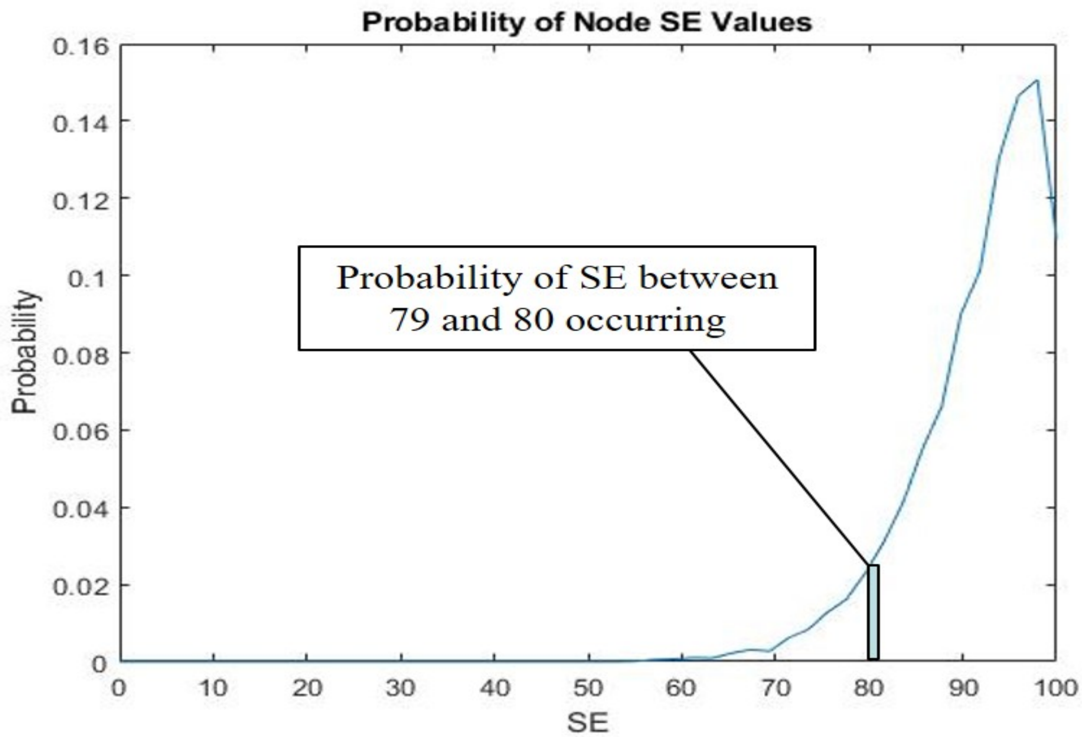

Figure 3.4. SE probability density function example

results can be compared with the FIR plots for the same sets of nodes to determine the different conclusions that are reached by including stochastic effects in the analysis. The stochastic impact values for each phase can also be summed or averaged across all phases to show impact over the full SoS evolution.

Single stochastic impact plots represent a useful addition to stochastic results available from any SODA analysis, regardless if multiple phases are considered or not. They quantify the impacts of node disruptions in the context of node stochasticity to provide a measure of the risk of particular node disruptions that was not available with previous SODA analysis tools.

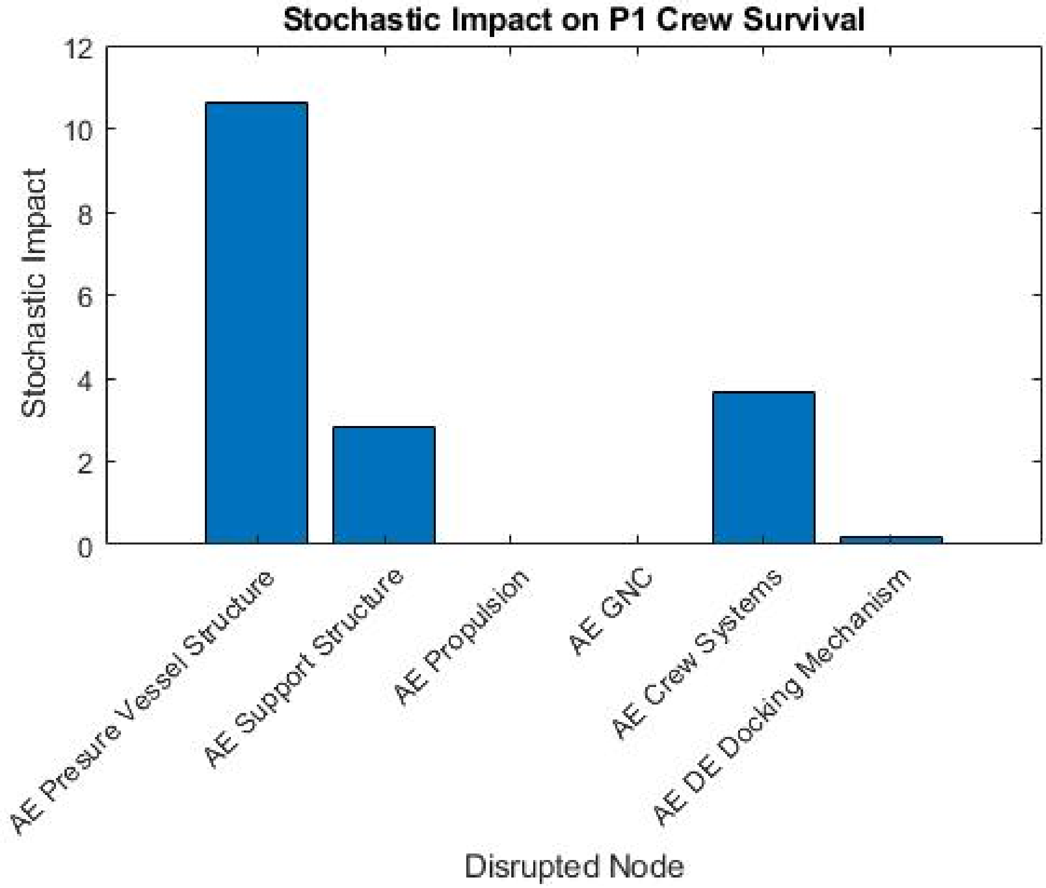

Figure 3.5. Stochastic Impact plot example

3.3 Integration of SODA and SDDA for Evolving Architectures

Consideration of SODA in the context of evolving architectures warrants investigation of opportunities for closer integration with System Developmental Dependency Analysis (SDDA). As the name suggests, SDDA is used to investigate how delays in the development of certain systems may impact the overall schedule of the program based on dependencies dictating the order in which systems can be developed. SDDA was also developed by Guariniello and its consideration of dependencies share similarities with the SODA model [10]. Each system is assigned minimum and maximum independent development times as well as an SE value representing punctuality. Here, decreases in SE indicate a development delay. A punctuality of 100 sets the system's independent development time to its minimum while a punctuality of zero sets the independent development time to its maximum. A system's actual development time depends on its punctuality and the punctuality of other systems on which it depends. SOD and COD values are used to describe the dependencies between systems in the development timeline. SOD indicates how much progress can be made on a system before its predecessor systems are complete. COD indicates the delay threshold in predecessor systems at which development of a successor system will not begin early. SOD is based on developmental relations between systems while COD can be chosen by the user to represent tolerance of risk.

Guariniello presented a simple example of how SDDA can be used with SODA to show how operational capabilities develop over time [10]. He mentions that the analysis could be extended by considering delays or operational disruptions. This idea can be extended further to investigate interactions between operational disruptions and development delays. In addition to only observing the operability of systems and capabilities over time in the face of disruptions or delays, operational disruptions caused by development delays, and vice versa, can be considered. Interactions between the operational and developmental networks can be investigated under different disruption scenarios to produce a more complete analysis of how an architecture evolves. Below is the proposed process for performing an integrated analysis using SODA and SDDA, which is also summarized in Figure 3.6.

1. Define the scope of the integrated analysis. In order to combine SODA and SDDA, there needs to be a common level of abstraction shared between both analyses. For example, if developmental dependencies are considered at the element level, operational dependencies should also be considered at this level at a minimum. Operational dependencies of lower levels like the subsystem level can be included to feed into the operability of higher levels.

2. Define distinct evolutionary phases to represent milestones in the development of the architecture. These can be distinguished by the introduction of a single new system or by the introduction of a group of systems. The most useful delineation would likely be one in which the newly introduced system(s) provide a new operational capability or significantly alter an existing one. This will increase the value of the analysis by removing excess work and ensuring a tighter coupling between the developmental and operational dependencies.

3. Define parameters for SDDA covering the entire planned evolutionary timeline and produce baseline SDDA results. These results represent the planned architecture development independent of any disruptions or delays.

4. Define timescale of each operational phase. Portions of the operational network (eg HLS mission) may not fill the entire length of time between development milestones. There may be downtime between the end of one operational phase and the beginning of the next.

5. Once a system has been completed, apply a degradation and random disruption model to decrease its SE over time. Once all systems required to implement a certain evolutionary phase have been developed, use SODA to determine the operability of nodes of interest over the duration of that evolutionary phase.

6. Introduce developmental delays and/or operational failures and analyze their impacts on the development timeline and system operabilities. Operational failures can come from the degradation model or can be introduced deterministically.

3.3.1 Analysis Options Enabled by Integrated Tools

Integrating SODA and SDDA provides several new avenues of analysis into how developmental and operational dependencies might interact as an SoS is evolving. These analysis options include:

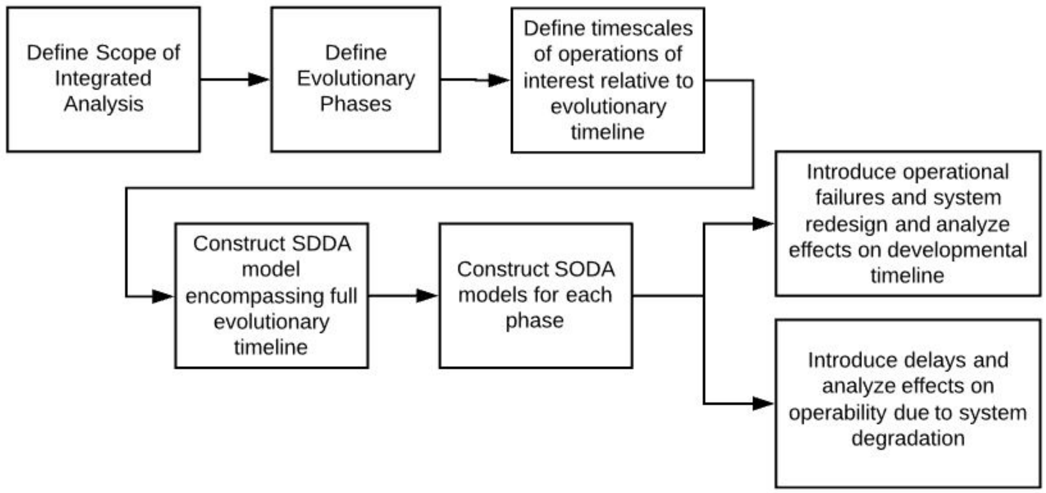

Figure 3.6. Flowchart for integrated analysis using SODA and SDDA

1. What-if scenarios of deterministic or stochastic disruptions and delays in evolution of operational capabilities over time.

2. Impact of delays on operational capabilities when systems are subject to SE degradation and random disruptions after the system is introduced into operations.

3. Impact of operational failures on development schedule due the need for system redesign.

For example, it might be interesting to investigate how delays in the development of certain systems impact the operability of systems given that their SE begins to degrade once they are introduced into service. Delays in a system's development could delay the implementation of the next evolutionary phase without delaying the completion of other systems used in that phase. This may cause some systems to be introduced into operation long before the next phase is fully implemented. When the SE of operational systems is assumed to decrease over time, the longer time span between a system's introduction and implementation of the operational phase in which it is used result in lower SE and lower operability of the operational network when it is evaluated. The important measure here is the relative completion times of systems needed for a particular operational phase, with larger differences in relative completion time corresponding to higher operational impact of development delays. A given delay may cause an operational phase to be implemented later without changing the relative completion times of the required systems. From limited experimentation, increasing the difference between the minimum and maximum independent development times in the delayed system increases the operational impact of a given delay since the same decrease in punctuality corresponds to a larger absolute time difference.

This analysis is accomplished by applying a simple random degradation function to decrease each node's SE by a small random amount at each time step. This idea was introduced by Guariniello [10], where he considered three modes of SE reduction over time:

- Small decrease in SE every time step to represent degradation during operations
- Low chance of a minor disruption to SE
- Very low chance of a major disruption to SE

Starting from the time at which each system is introduced into operations and ending when the operational network is fully implemented, SE values of each system are decreased using this model. The final SE values for all systems included in the next operational phase are then used to calculate the average operability of the phase capability nodes using SODA. Plotting the capability node operability values for different amounts of development delay in a certain systems provides insight into how much of an impact developmental delays can have on operations.

The inverse of this analysis is to determine how an operational disruption might delay the development timeline of the SoS. The most intuitive cause for such a delay is a significant operational disruption of a system requiring construction of a replacement or delays in subsequent development to account for redesign of the failed system and risk reduction. An operational disruption might have one of the following impacts on the development schedule:

1. Delay the completion of systems dependent on the failed system.

2. Delay the implementation of the next operational phase.

3. Increase the minimum and/or maximum independent development time of subsequent versions of the failed system to account for redesign.

An operational disruption that requires reconstruction of a system may have a compound effect on the operability of the SODA network by increasing the amount of time other systems degrade before the capability required for the next operational phase can be regained.

The case study presented in Chapter 5 will demonstrate an application of these ideas.

4. CASE STUDY: HUMAN LANDING SYSTEM

4.1 HLS Summary

As part of NASA's Artemis mission to send the first woman and next man to the Moon by 2024, NASA has solicited proposals from industry partners to develop the Human Lander System (HLS). Several proposals were submitted and in April 2020 three concepts were chosen for further development. The first of the three chosen concepts was proposed by the National Team, which includes Blue Origin, Lockheed Martin, Northrop Grumman, and Draper. The two other chosen concepts were proposed by Dynetics and SpaceX [17]. Each concept offers a unique approach to returning humans to the Moon with varying numbers of elements and steps needed to reach the lunar surface. Each HLS is to launch on a commercial or government heavy lift launch vehicle depending on whether separate elements can be launched independently and rendezvous in orbit, or the integrated HLS needs to be launched on a single vehicle. On the initial mission planned for 2024, the HLS will dock directly with the Orion crew capsule in Near-Rectilinear Halo Orbit (NRHO) near the Moon before taking the astronauts to the lunar surface. In subsequent missions, the expectation is to use the Gateway space station as a staging ground to enable a sustainable presence in lunar orbit. In this configuration, both the HLS and Orion capsule will dock with Gateway, where two astronauts will stay while two others descend to the surface. After reaching the surface, the HLS will support the crew for the duration of their stay (initially 6.5 days) [18] and then return the crew to NRHO where they will board the Orion capsule and return to Earth. The HLS call for proposals laid out general requirements, but was intentionally broad enough to allow companies to propose various different architectures and designs. Architectures based on two proposed HLS concepts will be considered for a case study. These two concepts are the three-element architecture proposed by the National Team and the two-element architecture proposed by Boeing, which was not selected by NASA for further development. The Boeing proposal is used as the second concept because it was the only other confirmed proposal when this research was begun. Future work can evaluate architectures based on the Dynetics and SpaceX proposals still in consideration.

4.1.1 National Team Proposal

The National Team led by Blue Origin proposed a three-element concept similar to NASA's baseline design. This concept consists of a transfer element (TE), descent element (DE) and ascent element (AE). Each of the elements will be launched independently on a commercial launch vehicle (CLV) and will be assembled into the integrated lander in NRHO [2]. A concept image is shown in Figure 4.1. After launching on the SLS Block 1, the Orion crew capsule will dock with HLS or Gateway to allow some or all of the crew to transfer to the HLS. The TE will move the HLS from NHRO to a Low Lunar Orbit (LLO). The TE will then detach from the HLS stack and be discarded. The DE engine is then used to descend from LLO and land on the lunar surface. On the Moon, the AE and DE provide habitable space and access to the surface for the crew. Upon completion of the surface mission, the crew launches in the AE, leaving the DE behind. The AE brings the crew back to NRHO to rendezvous with the Orion capsule or Gateway station. In the initial 2024 mission, all three elements of the HLS will be disposable, but the AE at minimum is expected to be reusable in later missions to improve the sustainability of surface missions [19].

Figure 4.2 shows a concept of operations for the three-element architecture. Note that both the AE and TE are possibilities for reuse, but only reuse of the AE by 2028 is required by NASA.

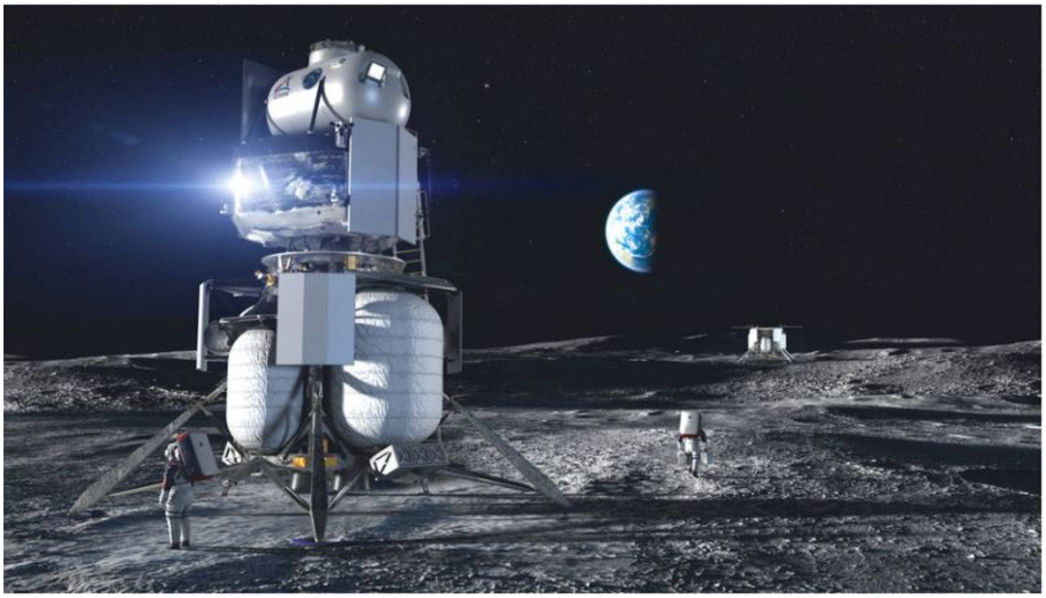

Figure 4.1. Artist's concept of the National Team HLS on the lunar surface [17].

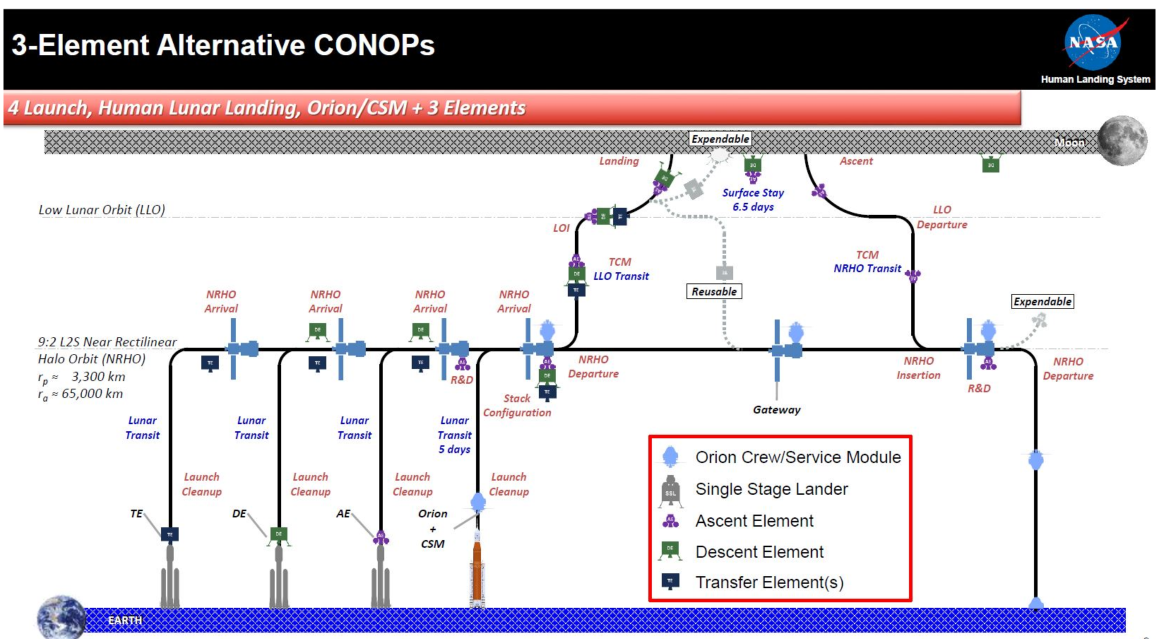

Figure 4.2. Concept of Operations for three-element architecture [20].

4.1.2 Boeing Proposal

Boeing proposed a two-element concept that consists of only a DE and AE. The integrated ascent and descent elements were to be launched together on an SLS Block 1B launch vehicle with an upgraded Exploration Upper Stage (EUS) [3]. This improved upper stage would allow the SLS to launch a greater mass to the NRHO, permitting the use of a larger descent element for both transfer from NRHO to LLO and for the descent to the surface. As in the National Team architecture, the descent and ascent elements support the crew during their surface operations before the crew leaves the DE behind and launches on the AE back to Gateway or Orion. A concept image for this design is shown in Figure 4.3.

Figure 4.3. Artist's concept of the Boeing HLS on the lunar surface. [3]

4.2 Modeling Methodology and Setup

Three steps are required to before analysis can be done using SODA. First, the systems and subsystems under consideration must be defined. Second, the distinction between each evolutionary phase in the HLS mission must be defined. Third, the dependencies between systems must be identified and the SODA parameters for each dependency defined for each phase.

The following terms will be used when referring to the different levels of the HLS concepts:

- Element - free-flying spacecraft module such as the Ascent Element and Descent Element
- Subsystem - One level down from element, major constituent parts of an element such as Structure and Power

4.2.1 HLS System Definition

Both HLS concepts being considered include an Ascent Element and a Descent Element, while one concept also contains a Transfer Element. Subsystems will be defined for each of these elements as they provide a convenient structure on which to construct the dependency model. Many of the subsystems in the Ascent Element and Descent Element are similar in both HLS concepts, so the general subsystem breakdown will be described with added details concerning any clear deviations between the two concepts. NASA NextSTEP-2 Appendix H Human Landing System BAA documents, company press releases, and Apollo Lunar Module configuration documents were used to determine the subsystems included in each element.

4.2.2 HLS Mission Phase Definition

The scope of this case study will consider the lunar mission starting from departure from Orion or Gateway, continuing to the lunar surface, and ending with the return of the HLS to Orion or Gateway. The full mission is expected to take 8 days with a 6.5 day surface stay and 0.5 day transfers between NRHO and LLO [18]. The full lunar sortie operational process can be segmented into a list of twelve general mission phases [18]:

1. Gateway/Orion Undock
2. Vehicle Checkout
3. Gateway/Orion Back Away
4. NRHO Departure
5. Transit to Powered Descent
6. Final Descent
7. Touchdown
8. Surface Mission
9. Initial Ascent
10. Return to Gateway
11. NRHO Insertion
12. Gateway/Orion Rendezvous, Proximity Operations, and Docking (RPOD)

For this study I will simplify the process into four major phases that capture the significant configuration changes the concepts experience. These mission phases represent the stable intermediate forms through which the SoS evolves. These are:

1. **Transfer from NRHO to LLO:** The TE engine (three element concept) or DE engine (two element concept) will fire to move the integrated HLS from its position near Orion or Gateway in NRHO to a circular checkout orbit in LLO.

2. **Descent from LLO to the Lunar Surface:** After detaching the TE (three element concept only), the DE engine will fire to bring the HLS out of LLO and slow its descent on approach to the lunar surface, ending with touchdown on the surface.

3. **Surface Operations and EVAs:** After landing, the crew will live on the surface for the nominal 6.5 day surface stay duration and conduct any scientific or exploration activities.

4. **Ascent from the Lunar Surface to NRHO:** The AE will lift off from the surface, leaving the DE behind, and return to NRHO to rendezvous with Orion or Gateway.

These phases were chosen since they form the core of the lunar surface mission and highlight some of the major differences between the two HLS concepts, such as the inclusion or omission of a dedicated transfer element.

For clarity, the elements included in each mission phase for both concepts are shown in Table 4.1.

Table 4.1. Elements included in each mission phase

Concept	Mission Phase	Ascent Element	Descent Element	Transfer Element
2 Element	NRHO to LLO	✓	✓	×
	LLO to Surface	✓	✓	×
	Surface Ops	✓	✓	×
	Surface to NRHO	✓	×	×
3 Element	NRHO to LLO	✓	✓	✓
	LLO to Surface	✓	✓	×
	Surface Ops	✓	✓	×
	Surface to NRHO	✓	×	×

Ascent Element Decomposition

Similar to the ascent stage of Apollo's Lunar Module, the HLS Ascent Element serves as the living environment for the crew during the lunar mission. It includes all the subsystems needed to keep the crew alive in space as well as flight controls, power subsystems, and communications subsystems needed for the crew to conduct their mission. The AE also contains a propulsion subsystem for departure from the surface as well as a docking port for interfacing with the Orion capsule or Gateway station. The three element AE is largely based on the Orion capsule, using many of it existing subsystems to speed development [21]. The subsystems listed below are generally shared between the two HLS concepts with discrepancies noted.

- **Pressure Vessel Structure:** The pressure vessel structure maintains a pressurized environment and protects the crew from the hazards of space. It also includes windows for external viewing and situational awareness as well a top hatch for exiting into a docked spacecraft. The three element concept's AE has a side hatch for exiting for EVA.

- **Support Structure:** The three-element AE has a support structure that acts as the main bus for many subsystems such as propulsion, thermal control, power, and avionics. It also holds the docking mechanism for connecting with the DE and is connected to the pressure vessel structure [17]. The two-element AE is constructed around the pressure vessel structure and doesn't include a distinct support structure subsystem [3].

- **Propulsion:** Comprised of the Ascent Engine and Propellant Tanks. The Ascent Engine is needed for departure from the lunar surface to NRHO, while the Propellant tanks store the fuel and oxidizer needed for liftoff. The three-element concept has a single ascent engine while the two-element concept has four groups of two engines placed around the edge of the AE.

- **GNC/ADCS:** Includes sensors, electronics, and devices for navigation and attitude control of the spacecraft. Methods for both automatic guidance by flight computers and manual guidance by crew members are included for safety [22].
- **Crew Systems:** Includes equipment needed for crew activities and health such as radiation monitoring, food preparation, hygiene, and EVA equipment.
- **Power:** Includes power generation, energy storage, and power distribution components. Both the AE concepts have a stationary solar panel to generate power and batteries to store energy.
- **Avionics:** Includes flight computers and software, command and data handling, and crew displays and controls necessary for automated and manual control of the spacecraft [23, 24].
- **Thermal Control:** Includes passive and active thermal control measures to maintain operating temperature ranges for other spacecraft subsystems like avionics, power, and propulsion.
- **Communications:** Radio transmitters and receivers for receiving commands from Earth and communicating with Earth and other spacecraft. Also includes equipment to communicate with crew members on EVA and relay their communications to Earth or other spacecraft like Gateway [25].
- **ECLSS:** Includes components to manage atmosphere, water, waste, cabin pressure, and humidity to maintain a habitable and comfortable environment for the crew [26].
- **Orion/GW Docking Mechanism:** IDSS-compliant docking mechanism for docking the AE to Orion or Gateway. Allows for transfer of crew, atmosphere, power, data, and communications between connected elements [27].

- **DE Docking Mechanism:** Docking mechanism for connecting the AE to the DE, as well as for detaching the AE from the DE before ascent from the lunar surface. Allows for transfer of power and data between the AE and DE [27]. In the two-element concept, the mechanism also allows transfer of crew between the AE and the DE's airlock.

- **AE Science and Research:** Equipment and accommodations for conducting scientific experiments on the lunar surface. Each HLS is required to accommodate 100 kg of science experiments, including 35 kg of return mass [19].

Descent Element Decomposition

The DE is used to perform some orbital maneuvers and slow the HLS to a safe landing on the Moon. It provides space for equipment and science experiments to be used by astronauts on the surface. Many of the subsystems in the DE are similar to those in the AE, however there are several distinctions based around the lack of crew habitation and the DE's central role in lunar surface access. The DE is composed of:

- **Main Structure:** The DE main structure consists of the load-bearing structure holding the propulsion subsystem, propellant tanks, avionics, and other subsystems.

- **Landing Structure:** The landing structure includes the landing legs that provide the ability to safely land on the surface and support the DE during surface operations.

- **Propulsion:** Comprised of the descent engine(s) and associated tanks and fluid management components. The DE propulsion subsystem for the three-element concept includes a single Blue Origin BE-7 engine, which uses cryogenic liquid oxygen and liquid hydrogen [28]. The two-element concept uses eight cryogenic liquid oxygen-liquid methane thrusters [3].

- **GNC/ADCS:** Includes sensors and electronics for guiding the HLS to its selected landing site on the Moon and maintaining the correct attitude during the descent.

- **Power:** Includes power generation, energy storage, and power distribution components. The three-element DE uses fuel cells to generate power [28] while the two-element DE uses solar arrays [3].

- **Avionics:** Includes flight computers/software and command and data handling necessary to control the spacecraft and monitor the health of other subsystems.

- **Thermal Control:** Includes passive and active thermal control measures to maintain operating temperature ranges for other spacecraft subsystems like electronics, power, and propulsion.

- **Communications:** Radio transmitters and receivers for receiving commands from Earth and communicating with Earth and other spacecraft.

- **Crew Surface Access:** The three-element concept will use an extendable ladder attached to the DE structure to allow crew members to descend from the AE to the surface. The two-element concept has a small ladder leading from the airlock to the surface.

- **Airlock:** The two-element DE includes a crew airlock that allows for EVAs without depressurizing the entire AE. The airlock is accessed through the AE-DE docking port.

- **AE Docking Mechanism:** Docking mechanism for connecting the DE to the AE, as well as for detaching the AE from the DE before ascent from the lunar surface. Allows for transfer of power and data between the AE and DE.

- **TE Docking Mechanism:** The three-element concept includes a docking mechanism for connecting the DE to the TE, as well as for detaching the TE from the DE after transfer to LLO and before descent to the lunar surface. Allows for transfer of power and data between the TE and DE.

Transfer Element Decomposition

The TE is used only in the three element architecture to take the HLS from Orion or Gateway in NRHO to LLO. After completing the transfer, it is detached from the HLS stack and discarded.

- **Structure:** The TE structure consists of the load-bearing structure holding the propulsion subsystem, propellant tanks, avionics, and other subsystems.

- **Propulsion:** Comprised of the transfer engine and propellant tanks. The transfer engine and tanks are used for bringing the HLS from NRHO to LLO. The transfer element will use a single Blue Origin BE-7 engine for its main propulsion, which uses cryogenic liquid oxygen and liquid hydrogen [28].

- **GNC/ADCS:** Includes sensors and electronics for guiding the HLS on its transfer from NRHO to LLO and controlling the spacecraft's attitude.

- **Power:** Includes power generation, energy storage, and power distribution components. The TE uses two deployable solar arrays to generate power and stores energy in batteries [28, 29].

- **Avionics:** Includes flight computers/software and command and data handling necessary to control the spacecraft and monitor the health of other subsystems [23].

- **Thermal Control:** Includes passive and active thermal control measures to maintain operating temperature ranges for other spacecraft subsystems like electronics, power, and propulsion.

- **Communications:** Radio transmitters and receivers for receiving commands from Earth and communicating with Earth and other spacecraft.

- **DE Docking Mechanism:** Docking mechanism for connecting the DE to the TE, as well as for detaching the TE from the DE after transfer to LLO and before descent to the lunar surface. Allows for transfer of power and data between the TE and DE [27].

4.2.3 Phase Capability Definition

Each mission phase has distinct objectives that are captured as capability nodes in the SODA analysis. For this study, one capability representing the objective of each phase is added as a node in the respective phase. Crew Survival is also included as a capability node for all phases. The capability nodes included in each phase are given in Table 4.2.

Table 4.2. Capability nodes associated with each mission phase

Mission Phase	**Phase Completion Capability**	**Crew Survival Capability**
NRHO to LLO	Transfer to LLO	Transfer Crew Survival
LLO to Surface	Descent to Surface	Descent Crew Survival
Surface Ops	Surface Operations	Surface Crew Survival
Surface to NRHO	Ascent to NRHO	Ascent Crew Survival

4.2.4 HLS Operational Dependencies Definition

Now that the subsystems, mission phases, and phase capabilities are defined, the operational dependencies between subsystems can be identified and modeled. An adjacency matrix of dependencies was created for each phase in each concept for a total of eight matrices.

The dependencies within each element are largely static across the mission phases since each element is designed to operate independently. The dependencies that change the most are those associated with the changing phase capability nodes and the dependencies between the spacecraft elements.

Dependencies were identified through literature review of interactions between spacecraft subsystems. Sources used for dependency identification include the NASA Appendix H HLS Requirements and associated BAA documents, NASA websites, university lecture material, news releases about each concept, and other sources. Dependencies were first identified for the first operational phase in which all elements are present in both concepts. This represents the case of most possibility of interdependencies between the subsystems. As the phases progress, elements are removed and the dependencies between different elements are evaluated to determine if their removal necessitates the addition or removal of dependencies within the remaining elements.

Once dependencies are identified, the SODA parameters are selected to describe the behavior of each dependency. Information to determine the SODA parameters can be found from literature review, historical data, or subject matter experts [7]. The parameters for this case study were selected based on literature review of each of the subsystems and their interactions within each concept. The intuitive nature of the three SODA parameters allow their values to be determined through a series of questions. These questions are:

1. **SOD:** How much of impact do small disruptions in node i have on the operability of node j?

2. **COD:** What operability of node j is expected given a complete disruption in node i?

3. **IOD:** How much increase in operability of node i is needed to return node j to high operability?

In order to provide more structure to the determination of the parameters, Guariniello introduced a series of nine operability curves that "fill" the parameter space.

SOD, COD, and IOD values are set to either High (H), Medium (M), or Low (L) values to create nine standard LMH operability curves. The specific parameter values for each tier can be adjusted for different scenarios. In this analysis, the LMH structure is used with the values shown in Table 4.3.

Table 4.3. SODA Parameter values corresponding to LMH

Parameter	Low	Medium	High
SOD	0.1	0.4	0.9
COD	20	50	100
IOD	10	40	90

The most intuitive order in which to determine the values of the SODA parameters is to start with COD and then determine SOD and IOD.

4.2.5 Three Element Architecture Dependencies: Phase 1

The subsystem dependencies in the first mission phase (Transfer to LLO) are discussed as a starting point. In the subsequent phases the focus will be on dependencies that change from the previous phase.

Below are several figures showing sections of the adjacency matrix for for first phase of the three-element architecture. The nodes listed along the top are impacted by the nodes listed down the side by dependencies specified in the cell at the intersection of their respective row and column. The adjacency matrix is split into sections showing dependencies feeding into subsystems in the TE, DE, AE, and capability nodes. "Off diagonal" entries in the adjacency matrix indicating dependencies between elements are highlighted. The full matrix can be found in Appendix A.

Figure 4.4 shows the graph representation of the dependencies in the Transfer to LLO phase for the three-element architecture. The arrows show the direction of the dependencies from the feeder node to the receiver node. The subsystems are colored by their associated element.

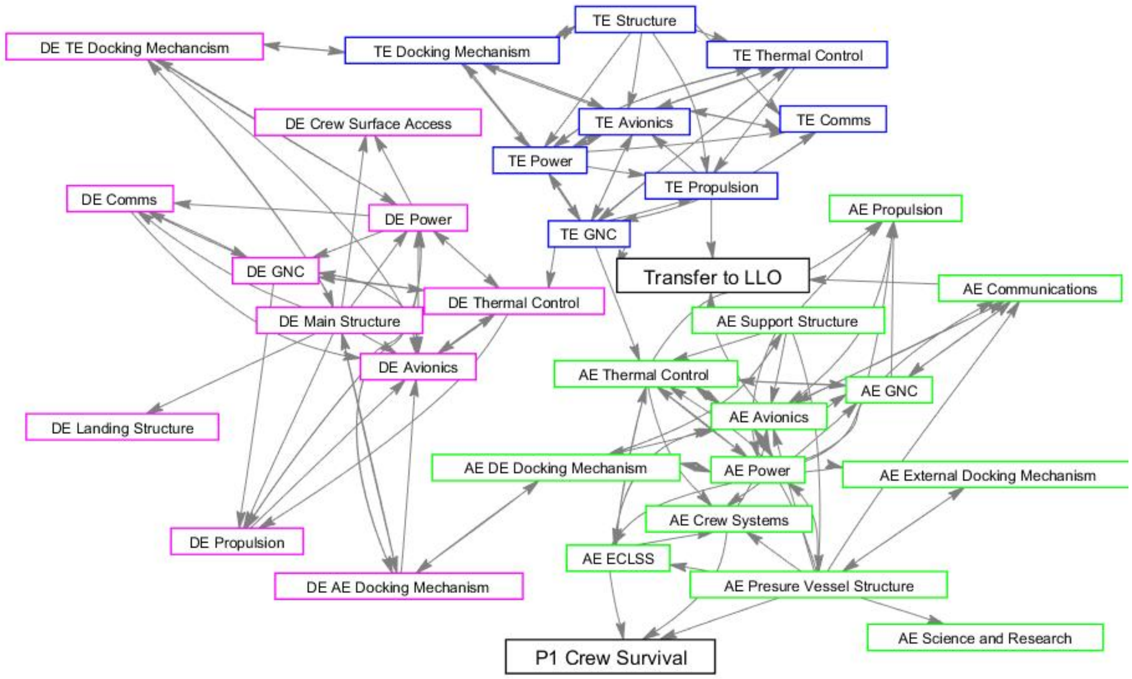

Figure 4.4. Network of subsystem operational dependencies in the Transfer to LLO phase for the three-element concept. Nodes colored by element: Blue = TE, Purple = DE, Green = AE, Black = Capability.

Transfer Element Dependencies: Transfer to LLO

Figure 4.5 shows the TE section of the adjacency matrix. The dependencies between elements are highlighted to distinguish them from the majority of dependencies that are within an element. Most of the dependencies are within the transfer element, however the TE Docking Mechanism has a strong HHH dependency on the DE-TE Docking Mechanism in the descent element. This captures the major interface between the two elements. Each element is connected with a NASA Docking System

(NDS) that allows the separate elements to be autonomously docked together near the Moon. The NDS allows for transfer of power and data between the connected spacecraft [27]. The ability of the docking mechanism to transfer power and data necessitates weak dependencies between the docking mechanism and the power and avionics subsystems. The docking mechanism impacts the TE power and avionics subsystems because they depend partly on power and data from the DE, while the power and avionics impact the docking mechanism to represent the same power and data channel in the opposite direction.

The avionics subsystem has relatively weak dependencies and impacts on several other subsystems due to the avionics' function of relaying commands and data from flight computers to other subsystems and collecting data on the operations of each subsystem [23].

The power subsystem has highly critical impacts on many of the other subsystems since complete loss of electrical power would make the electronic components of these subsystems inoperable.

Descent Element Dependencies: Transfer to LLO

Figure 4.6 shows the adjacency matrix for dependencies feeding into the descent element subsystems. The dependencies internal to the DE are similar to those in the TE. Each of the two docking mechanisms on the DE have strong dependencies on their counterparts on the TE and AE. Again, power and avionics share dependencies with the docking mechanisms to capture the transfer of power and data between elements. Another external dependency is the link between DE thermal control and TE GNC. During the transfer phase, it is assumed the TE GNC will be fully controlling the attitude of the entire HLS. Thermal control partly depends on attitude control to adjust the spacecraft's orientation to maintain favorable heating and cooling conditions [23].

The DE power, propulsion, and thermal control subsystems are highly coupled and critical to one another. The DE utilizes oxygen-hydrogen fuel cells for power,

	Element	Transfer Element							
Element	Subsystems	TE Structure	TE Propulsion	TE GNC	TE Comms	TE Thermal Control	TE Power	TE Avionics	TE Docking Mechanism
Transfer Element	TE Structure		LHL		LHL	LHL	LMH	LMH	HHH
	TE Propulsion							LLM	
	TE GNC		MHM		LML	LMH	LML	LLM	
	TE Comms			LMH				LLM	
	TE Thermal Control		LMH	LMH			LML	LMH	
	TE Power		LHL	HHL	HHL	MMM		HHL	LMH
	TE Avionics			LML	LML	LML	LML		LML
	TE Docking Mechanism	LML					LLM	LLM	
Descent Element	DE Main Structure								
	DE Landing Structure								
	DE Propulsion								
	DE GNC								
	DE Power								
	DE Thermal Control								
	DE Crew Surface Access								
	DE Comms								
	DE Avionics								
	DE TE Docking Mechancism								HHH
	DE AE Docking Mechanism								
Ascent Element	AE Presure Vessel Structure								
	AE Support Structure								
	AE Propulsion								
	AE GNC								
	AE Crew Systems								
	AE DE Docking Mechanism								
	AE External Docking Mechanism								
	AE Power								
	AE Thermal Control								
	AE Avionics								
	AE Communications								
	AE ECLSS								
	AE Science and Research								
Capability	Transfer to LLO								
	Crew Survival								

Figure 4.5. Transfer Element Adjacency Matrix for Phase 1 in three-element concept

which rely on the same fuel used by the propulsion subsystem [28]. This fuel must be maintained at cryogenic temperatures by the thermal control subsystem, which also partly requires power to operate.

Element	Element	Descent Element										
Element	Subsystems	DE Main Structure	DE Landing Structure	DE Propulsion	DE GNC	DE Power	DE Thermal Contro	DE Crew Surface Access	DE Comms	DE Avionics	DE TE Docking Mechanism	DE AE Docking Mechanism
Transfer Element	TE Structure											
	TE Propulsion											
	TE GNC						LMH					
	TE Comms											
	TE Thermal Control											
	TE Power											
	TE Avionics											
	TE Docking Mechanism										HHH	
Descent Element	DE Main Structure		LML	LHL		LMH	LHL	MHM	LHL	LMH	HHH	HHH
	DE Landing Structure											
	DE Propulsion					MHM				LLM		
	DE GNC			MHM			LMH		LML	LLM		
	DE Power			LHL	HHL		MMM	LMH	HHL	HHL	LMH	LMH
	DE Thermal Control			HHH	LMH	LML				LMH		
	DE Crew Surface Access											
	DE Comms				LMH					LLM		
	DE Avionics				LML	LML	LML					
	DE TE Docking Mechancism	LML				LLM				LLM		
	DE AE Docking Mechanism	LML				LLM				LLM		
Ascent Element	AE Presure Vessel Structure											
	AE Support Structure											
	AE Propulsion											
	AE GNC											
	AE Crew Systems											
	AE DE Docking Mechanism											HHH
	AE External Docking Mechanism											
	AE Power											
	AE Thermal Control											
	AE Avionics											
	AE Communications											
	AE ECLSS											
	AE Science and Research											
Capability	Transfer to LLO											
	Crew Survival											

Figure 4.6. Descent Element Adjacency Matrix for Phase 1 in three-element concept

Ascent Element Dependencies: Transfer to LLO

Figure 4.7 shows the adjacency matrix for dependencies feeding into the ascent element subsystems during the Transfer to LLO phase. Once again, the main interaction between the AE and other elements is through the docking mechanism. Many of the dependencies for subsystems like propulsion, power, and avionics are similar to those in the DE. Like the DE thermal control subsystem, the AE thermal control depends on attitude control provided by the TE GNC/ADCS subsystem to help maintain manageable thermal loads. The AE power subsystem generates power using a solar panel, so it is less strongly influenced by the propulsion and thermal control subsystems than the DE power subsystem.

Capability Node Dependencies: Transfer to LLO

Figure 4.8 shows the adjacency matrix for dependencies feeding into the two capability nodes associated with the Transfer to LLO phase. The "Transfer to LLO" capability is strongly dependent on the TE propulsion and GNC subsystems due to their critical functions in physically moving the spacecraft between orbits. Each of these dependencies is assigned a high COD since complete loss of either would make the transfer impossible. AE avionics, specifically the crew displays and controls, also impact this capability due to the need of the crew to be apply to control the spacecraft and monitor its status during the transfer. Finally, communications with the spacecraft are required to be active during all phases of the mission [25], so the AE communications subsystem is also an input to this capability.

The crew survival capability nodes in all phases are dependent on the same set of nodes. The most critical are the AE pressure vessel structure and AE ECLSS since these provide the foundation for a habitable environment in space. Given the range of life-critical functions that the ECLSS subsystem performs, such as atmosphere management, water management, environmental monitoring, and fire protection, the dependency of Crew Survival on ECLSS is given high SOD and COD values [30]. Also

	Element	Ascent Element												
Element	Subsystems	AE Pressure Vessel Structure	AE Support Structure	AE Propulsion	AE GNC	AE Crew Systems	AE DE Docking Mechanism	AE External Docking Mechanism	AE Power	AE Thermal Contro	AE Avionics	AE Communications	AE ECLSS	AE Science and Research
Transfer Element	TE Structure													
	TE Propulsion													
	TE GNC									LMH				
	TE Comms													
	TE Thermal Control													
	TE Power													
	TE Avionics													
	TE Docking Mechanism													
Descent Element	DE Main Structure													
	DE Landing Structure													
	DE Propulsion													
	DE GNC													
	DE Power													
	DE Thermal Control													
	DE Crew Surface Access													
	DE Comms													
	DE Avionics													
	DE TE Docking Mechancism													
	DE AE Docking Mechanism						HHH							
Ascent Element	AE Presure Vessel Structure					MHM		HHH	LMH	LHL	LMH	LML	MHM	LLM
	AE Support Structure	MHM		LHL			HHH		LMH	LHL	MHM			
	AE Propulsion										LLM			
	AE GNC			MHM						LML		LML		
	AE Crew Systems										LLM			
	AE DE Docking Mechanism		LML						LLM		LLM			
	AE External Docking Mechanism	LML												
	AE Power			LHL	HHL	MHM	LMH	LMH		MMM	HHL	HHL	HHH	
	AE Thermal Control			LMH	LMH	LML			LML		LMH		MHM	
	AE Avionics				LML				LML	LML		LML	LML	
	AE Communications				LMH						LLM			
	AE ECLSS					MHM				LML	LLM			
	AE Science and Research													
Capability	Transfer to LLO													
	Crew Survival													

Figure 4.7. Ascent Element Adjacency Matrix for Phase 1 in three-element concept

important is the AE crew systems, which includes equipment for food preparation, health and hygiene, and pressure suits. Since the crew can survive for several days with some disruption in food or hygiene availability, this dependency is only given medium values for all three parameters.

Element	Subsystems	Transfer to LLO	Crew Survival
Transfer Element	TE Structure		
	TE Propulsion	HHH	
	TE GNC	MHH	
	TE Comms		
	TE Thermal Control		
	TE Power		
	TE Avionics		
	TE Docking Mechanism		
Descent Element	DE Main Structure		
	DE Landing Structure		
	DE Propulsion		
	DE GNC		
	DE Power		
	DE Thermal Control		
	DE Crew Surface Access		
	DE Comms		
	DE Avionics		
	DE TE Docking Mechancism		
	DE AE Docking Mechanism		
Ascent Element	AE Presure Vessel Structure		HHL
	AE Support Structure		
	AE Propulsion		
	AE GNC		
	AE Crew Systems		MMM
	AE DE Docking Mechanism		
	AE External Docking Mechanism		
	AE Power		
	AE Thermal Control		
	AE Avionics	MHM	
	AE Communications	MMM	
	AE ECLSS		HHL
	AE Science and Research		
Capability	Transfer to LLO		
	Crew Survival		

Figure 4.8. Capability Adjacency Matrix for Phase 1 in 3 Element Concept

4.2.6 Three Element Architecture Dependencies: Phase 2

The adjacency matrices for phases two to four will not be broken down in as much detail here, rather the focus will be on how dependencies change from one phase to the next. The full adjacency matrix for phase two of the three element architecture can be found Appendix A (A.2). Figure 4.9 shows the graph representation of the dependencies in the Descent to Surface phase for the three-element architecture.

Figure 4.10 shows the changes in the subsystem dependencies from the first phase to the second phase. The most obvious change is the removal of all links to TE subsystems, as the TE is discarded after the first phase is completed. Another focus of changes is the DE-TE Docking Mechanism. Since the docking mechanism is no longer connected to the TE, it no longer transfers power and data to the DE Power and DE Avionics subsystems, respectively. The docking mechanism also is not transferring structural loads between the elements, so the impact on DE Main Structure is removed. Also, AE Thermal Control shifts from being dependent on TE GNC during the transfer phase to being dependent on DE GNC during the descent. Finally, the second phases' capability node "Descend to Surface" is impacted by DE Propulsion, GNC, and Landing Structure rather than the TE Propulsion and GNC that impacted the "Transfer to LLO" capability in the first phase.

Removing or adding redundant subsystems in a SODA model generally changes the SOD and COD values of dependencies on the redundant system. For example, if a sensor depends on electrical power, adding a second power source would make small disruptions in one of the power sources less impactful and would decrease the impact of complete loss of one of the power sources. This would correspond to lower values of SOD and COD in each of the two dependencies than the values used in the original dependency. The opposite situation occurs when the TE is discarded and the HLS loses that source of power. However, this is not reflected as changes to the SODA parameters (black line) in Figure 4.10. This is due to the decomposition level used in the model, where I group the power generation and power distribution components together. At a lower level of decomposition, the power generation component in the DE and the power transfer capability of the DE-TE docking mechanism would both impact the DE power distribution, so removing the TE power transfer would change the parameters of the remaining dependency. At the decomposition level used in the model, however, this change simply results in the removal of the dependency between DE Power and DE-TE Docking Mechanism.

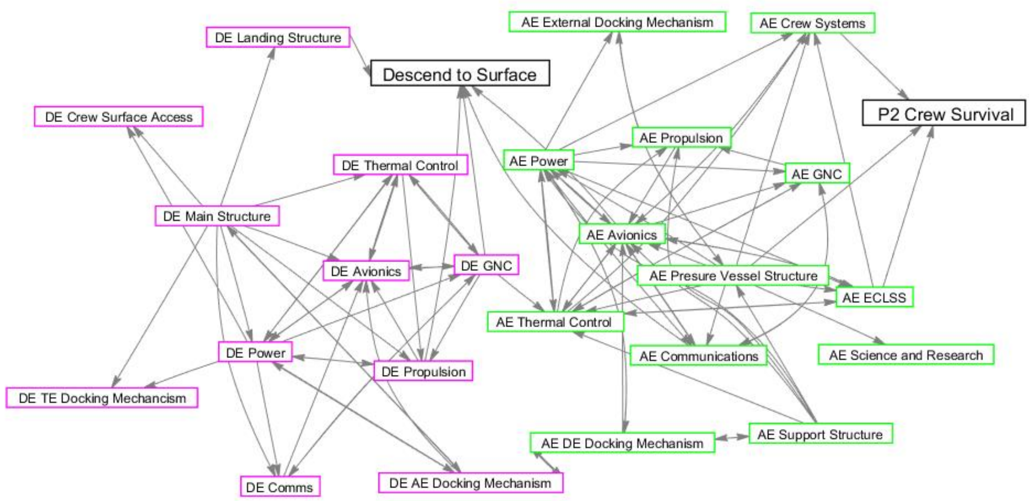

Figure 4.9. Network of subsystem dependencies in the Descent to Surface phase for the three-element concept. Nodes colored by element: Purple = DE, Green = AE, Black = Capability.

4.2.7 Three Element Architecture Dependencies: Phase 3

Figure 4.11 shows the changes in dependencies between the second and third phases. The majority of the changes are due to the shift of the phase completion capability from "Descent to Surface" to "Surface Operations". Once on the lunar

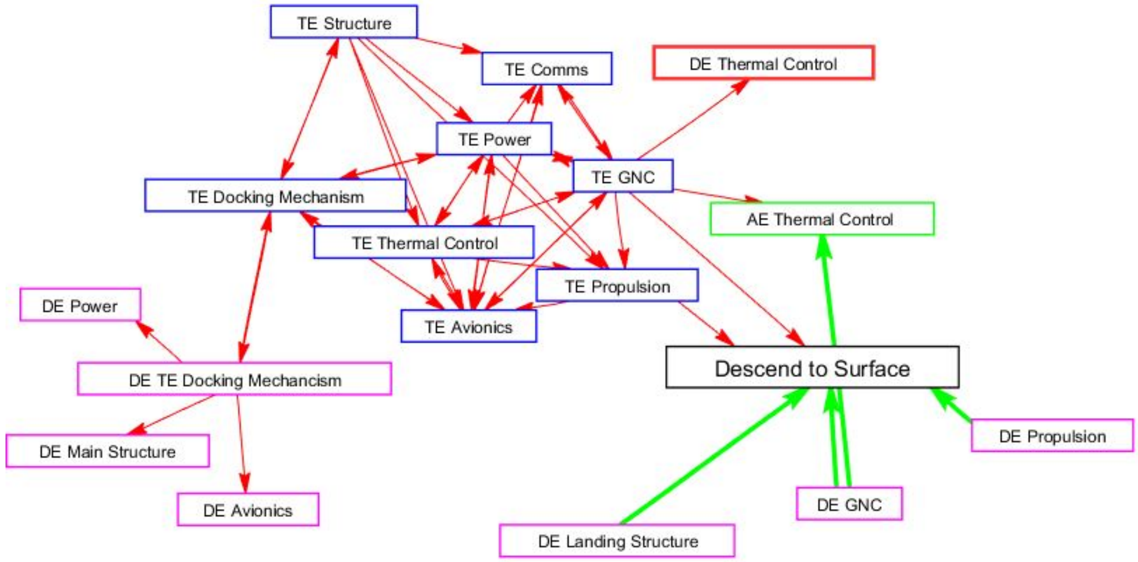

Figure 4.10. Network of changes to subsystem dependencies between the first and second phases for the three-element concept. Nodes colored by element: Blue = TE, Purple = DE, Green = AE, Black = Capability. Links colored by type of change: Green = Added link, Red = Removed link, Black = changed link parameters.

surface, the AE thermal control is no longer dependent on attitude control, so the link from DE GNC is removed. The full adjacency matrix is in Appendix A (A.3).

4.2.8 Three Element Architecture Dependencies: Phase 4

Finally, Figure 4.12 shows the dependency changes between the third and fourth phases. Similar to the transition between the first and second phase, the network shows the removal of all dependencies associated with the DE as well as some dependencies from the AE-DE Docking Mechanism to AE Power, Avionics, and Support Structure subsystems.

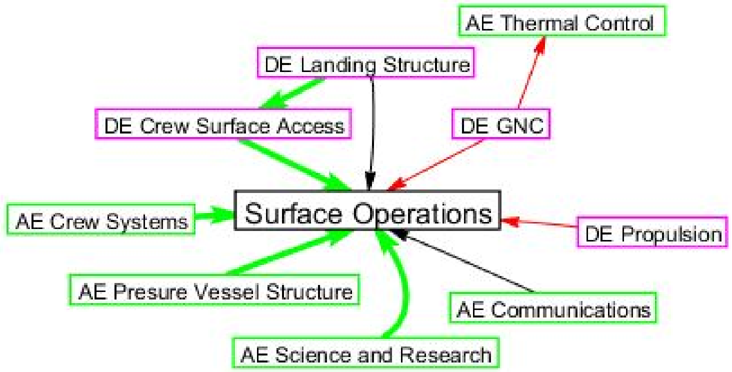

Figure 4.11. Network of changes to subsystem dependencies between the second and third phases for the three-element concept. Nodes colored by element: Purple = DE, Green = AE, Black = Capability. Links colored by type of change: Green = Added link, Red = Removed link, Black = changed link parameters.

4.2.9 Two Element Architecture Dependencies

Figure 4.13 shows the dependency network for the first phase of the two element architecture. Compared to the three element concept, the most obvious difference is the lack of the transfer element. There are also a few subsystem node differences from the three element concept that should be noted. Since there is no transfer element, the DE only has one docking mechanism for connecting with the AE. The DE now includes an airlock to allow crew to enter and exit the spacecraft without depressurizing the entire AE. Also, the AE Support Structure node is removed. Most of the dependencies within the DE and AE are the same as those used in the three element concept, as there was not enough concept-specific information available to justify making them distinct.

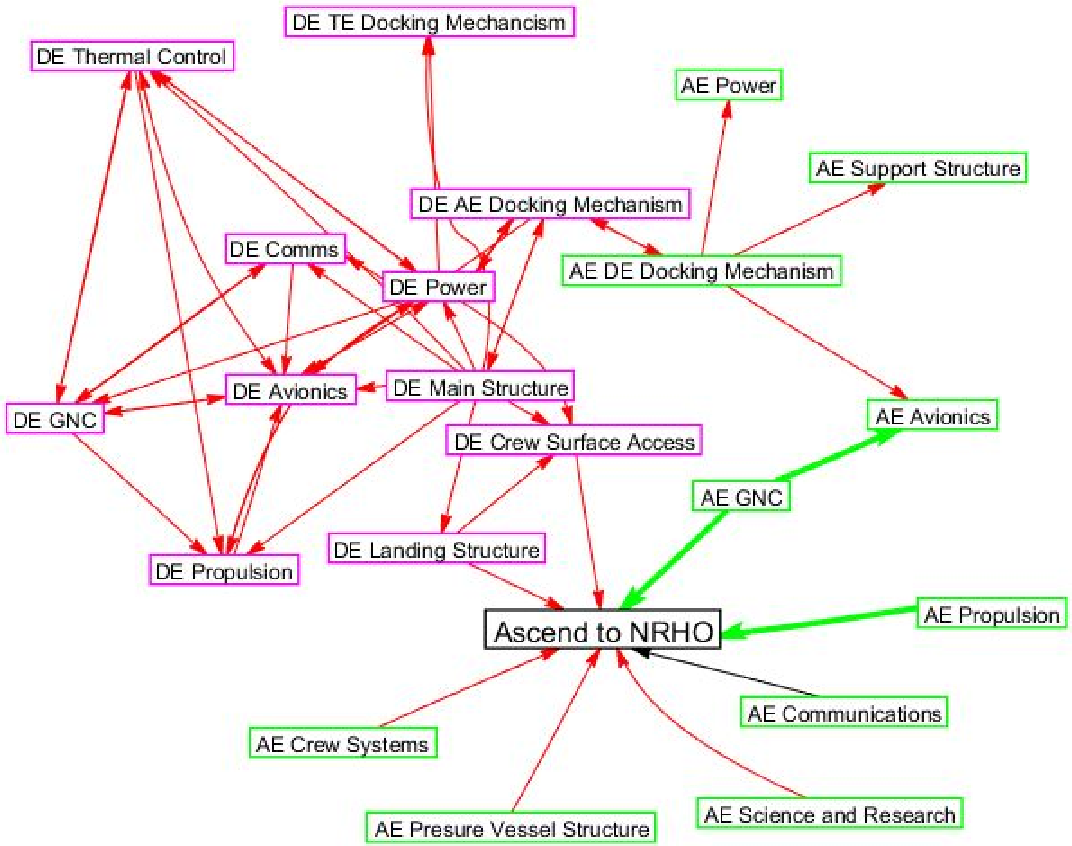

Figure 4.12. Network of changes to subsystem dependencies between the third and fourth phases for the three-element concept. Nodes colored by element: Purple = DE, Green = AE, Black = Capability. Links colored by type of change: Green = Added link, Red = Removed link, Black = changed link parameters.

4.3 Results for HLS Case Study

4.3.1 Analysis using FIR Sequence Plots

FIR sequence plots with the disrupted nodes sorted by impact provide a good overview of how the most critical subsystems change when considering different phase capability nodes. The capability nodes representing phase completion are generally dependent on different subsystems in each phase, so changes in the order of most impactful subsystems can be due to both the different dependencies feeding the ca-

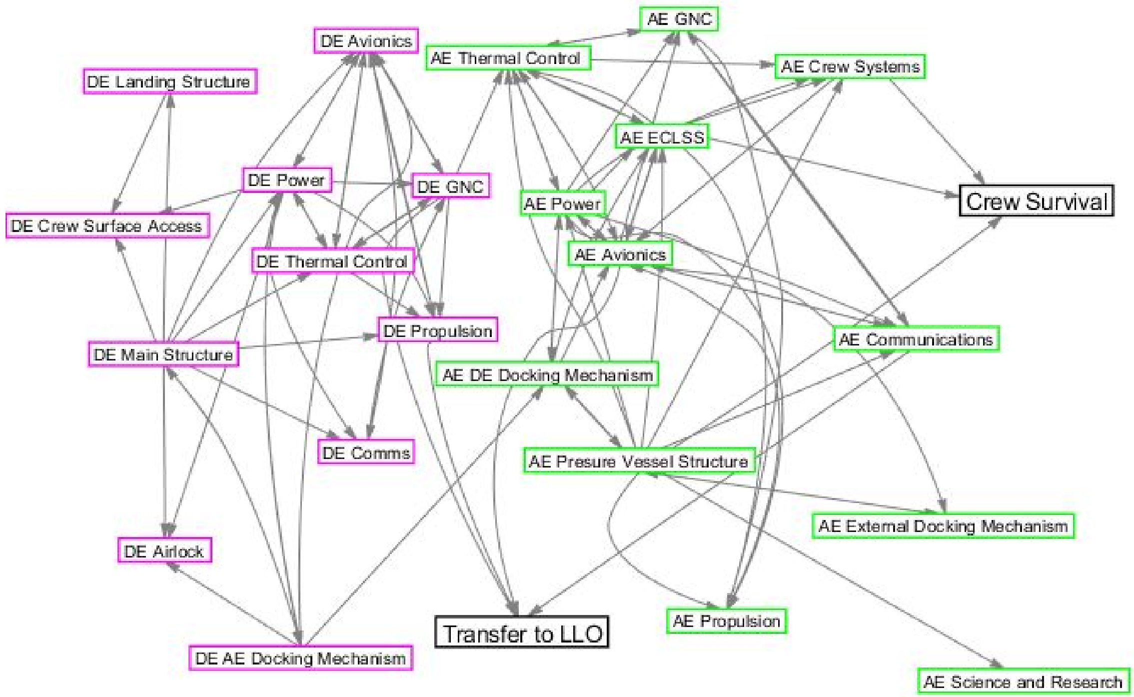

Figure 4.13. Network of subsystem dependencies for the first phase of the two-element concept. Nodes colored by element: Purple = DE, Green = AE, Black = Capability.

pability node and any changes in the internal dependencies between subsystems. On the other hand, each crew survival capability node is dependent on the same subsystems, so the changes in disruption impact will only be due to changes in the internal subsystem dependencies, such as the removal of an element.

Figure 4.14 shows the top nine most impactful subsystems on the phase completion capabilities for the first two mission phases in each of the two HLS concepts. Colored blocks are added under each disrupted subsystem to indicate which element that subsystem corresponds to. This is helpful for quickly identifying changes in the most important elements across the mission phases. Blue block indicates TE, purple indicates DE, and green indicates AE. Some boxes are white to indicate that there is no impact on that node, but it is listed because there are fewer than nine nodes that have a significant impact on that capability. In the three-element concept, an obvious observation is the removal of TE subsystems from the set of most impactful nodes when moving from phase one to phase two. This is expected since the TE is discarded after the first phase and therefore should not impact capabilities in the second phase. Since both phase completion capabilities are associated with propulsive burns and maneuvers, it makes sense that the propulsion and GNC subsystems in the TE and DE are among the most critical subsystems in the first and second phases, respectively. AE Avionics, Support Structure, and Communications are also impactful to both phases due to their necessity for crew control of the spacecraft. Considering the two-element concept, the impactful subsystems do not change much between the two phases, but the order of impact changes slightly. Since both phases mainly require a propulsion burn from the DE propulsion for completion, the DE propulsion, GNC, and thermal control subsystems are most important in both phases. The impacts of other subsystems such as AE avionics and communications change slightly due to changing interdependencies across the phases.

Figure 4.15 shows the most impactful subsystems on the phase completion capabilities for the third and fourth mission phases. A major difference here is the larger number of critical subsystems in phase 3 for the two-element concept. These stronger impacts arise because of the inclusion of the airlock in the DE, which causes the AE and DE to be more strongly interdependent during crew surface operations. The impacts of subsystems in the fourth phase are largely the same in both concepts, with some small deviations due to differing dependencies within each concept.

Figures 4.16 and 4.17 show the nine most impactful subsystems for the crew survival capabilities in each phase. In both concepts, AE subsystems are the most critical with AE power and pressure vessel structure subsystems occupying the top two spots. The AE support structure is also critical in the three-element concept. Across all four phases in each of the concepts, the subsystems that most impact the crew survival capabilities are essentially static. This could be considered a positive result by designers, as a few key subsystems can be focused on to ensure high operability of crew survival for the entire mission. Surprisingly, disruptions to the ECLSS subsystem do not result in low operability (red bar) of the crew survival capability in any of the phases. This is a result of the fact that only one subsystem is disrupted at a time in these results, so even though crew survival is strongly and directly dependent on ECLSS, the high operabilities of other subsystems feeding the crew survival node keep its operability in the high and medium zones. This is also because ECLSS does not impact other subsystems that are needed for crew survival. AE power and structural subsystems impact many other subsystems so disruptions in them cascade throughout the network to cause a larger overall effect.

Looking at disruptions in each element independently is useful for understanding how the impact of particular subsystems on capability nodes change for different phases. Considering the TE subsystems in this way is not particularly interesting because those subsystems are only present during the first phase. Looking at the DE or AE is more insightful since those subsystems are present in most or all of the phases. Figure 4.18 shows the impacts of disrupting subsystems in the DE on phase completion capabilities. Here I can easily see that the Descend to Surface and Surface Operations capability nodes are most impacted by disruptions in a different set of DE subsystems. Disruptions in the propulsion, GNC, and thermal control subsystems have the most impact in phase two while main structure and crew surface access have the most impact in phase three. None of the DE subsystems impact the fourth phase capability node, which is expected since the DE is not included in the fourth phase, but none the subsystems have an impact on the transfer to LLO capability either. The dependencies between the DE and TE subsystems through the docking mechanism are not significant enough to completion of the transfer phase to show any impact when any of the DE subsystems are disrupted.

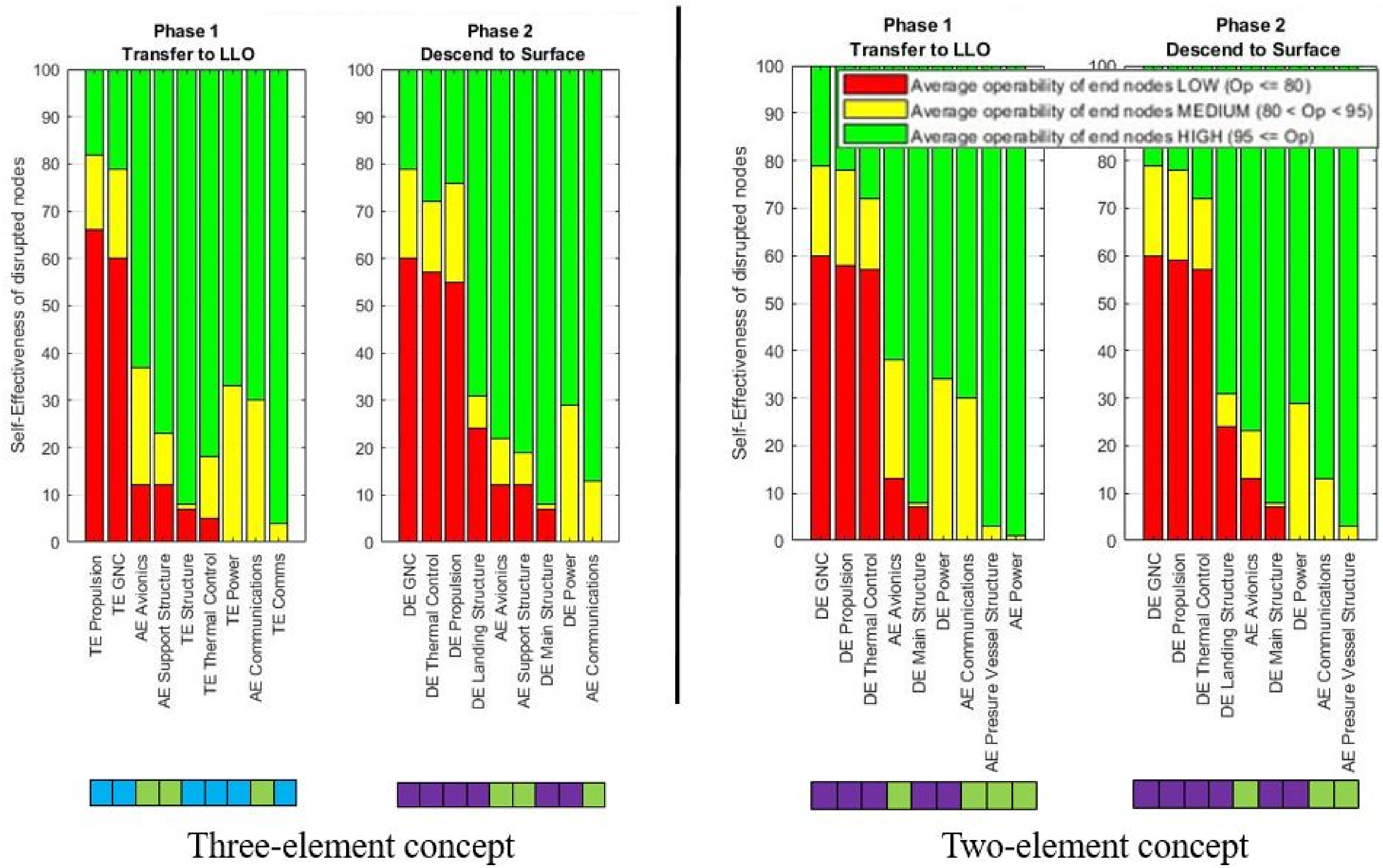

Figure 4.14. Comparison of most impactful node disruptions on the phase completion capabilities in phases one and two between the two concepts

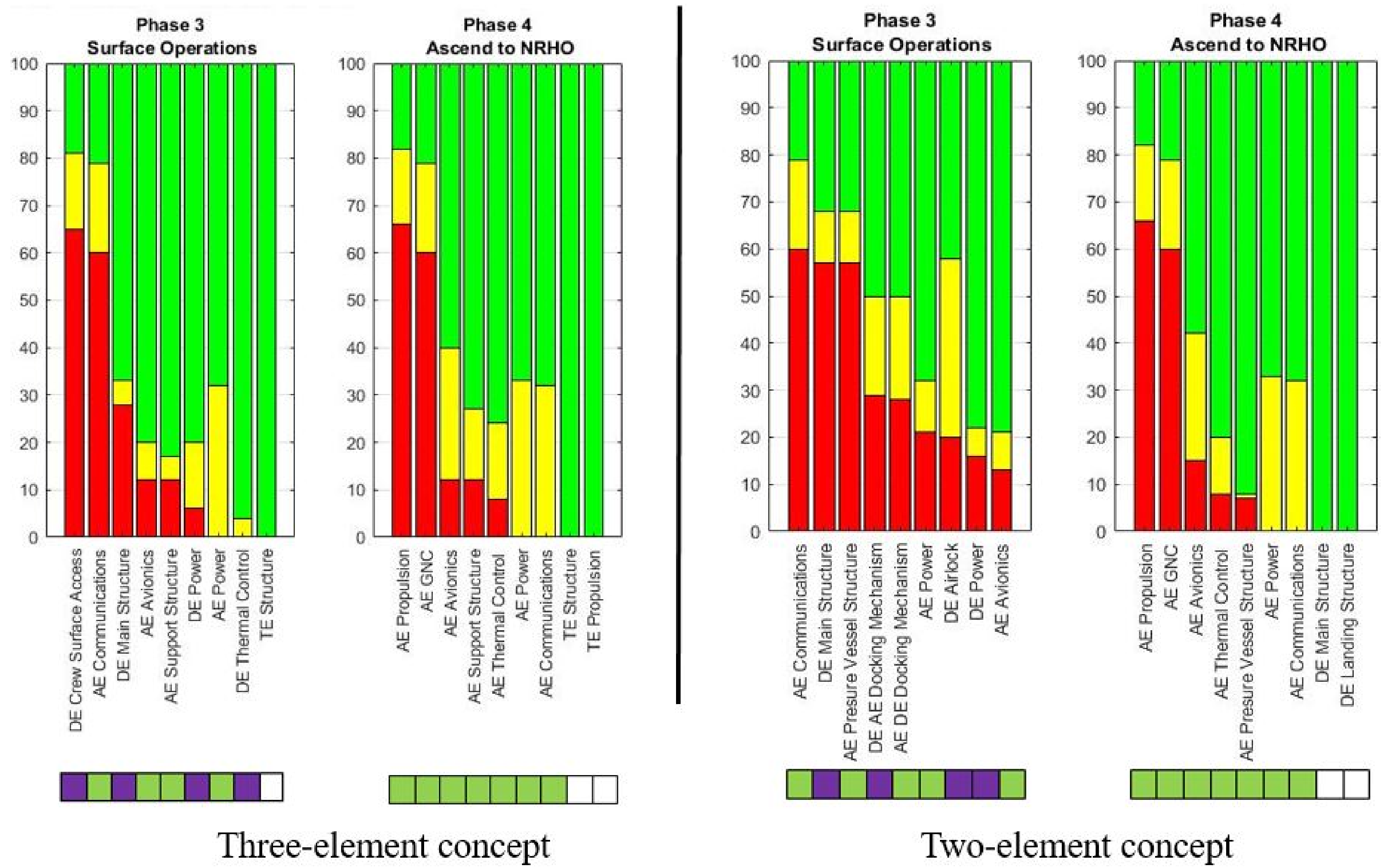

Figure 4.15. Comparison of most impactful node disruptions on the phase completion capabilities in phases three and four between the two concepts

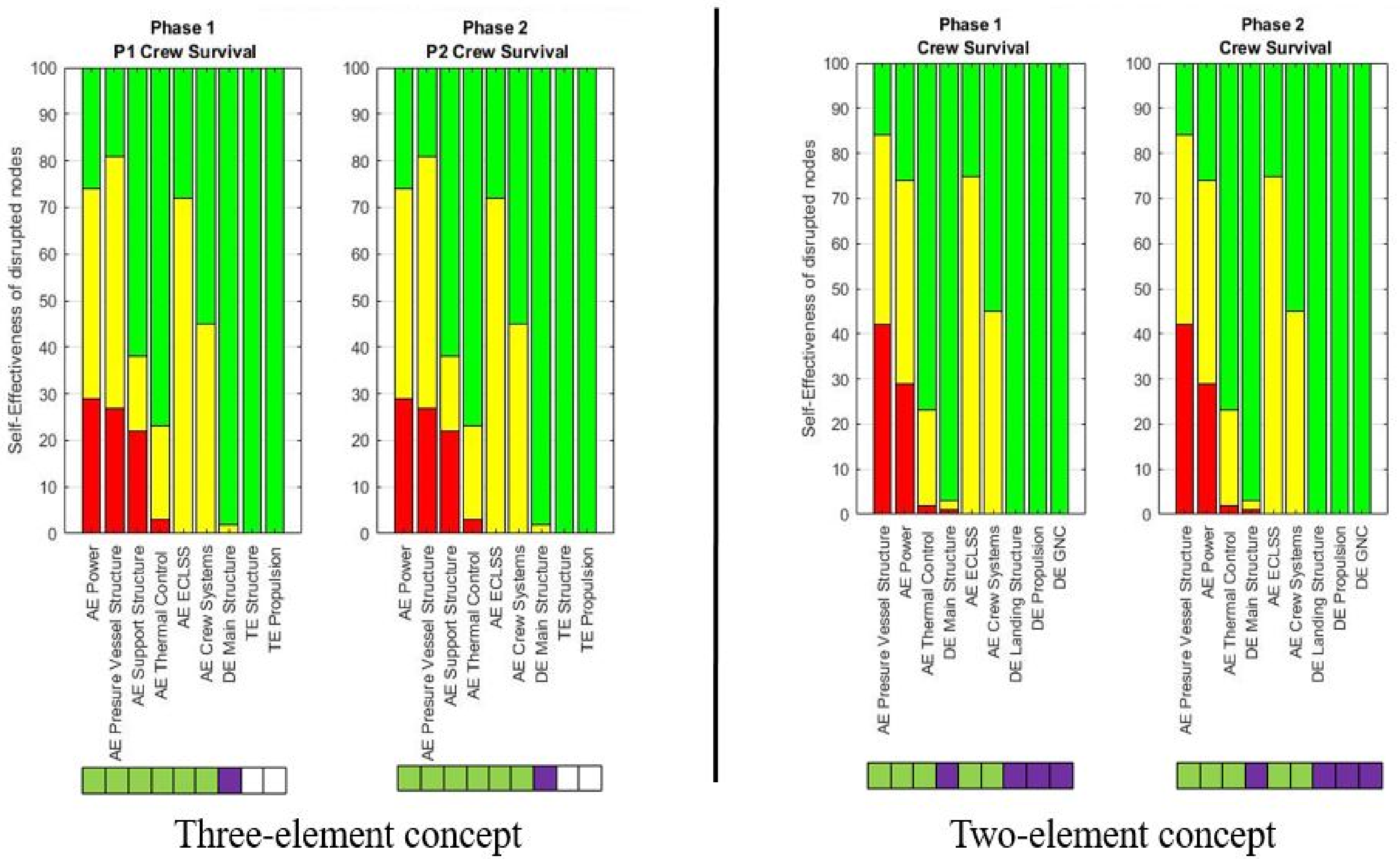

Figure 4.16. Comparison of most impactful node disruptions on the crew survival capabilities in phases one and two between the two concepts

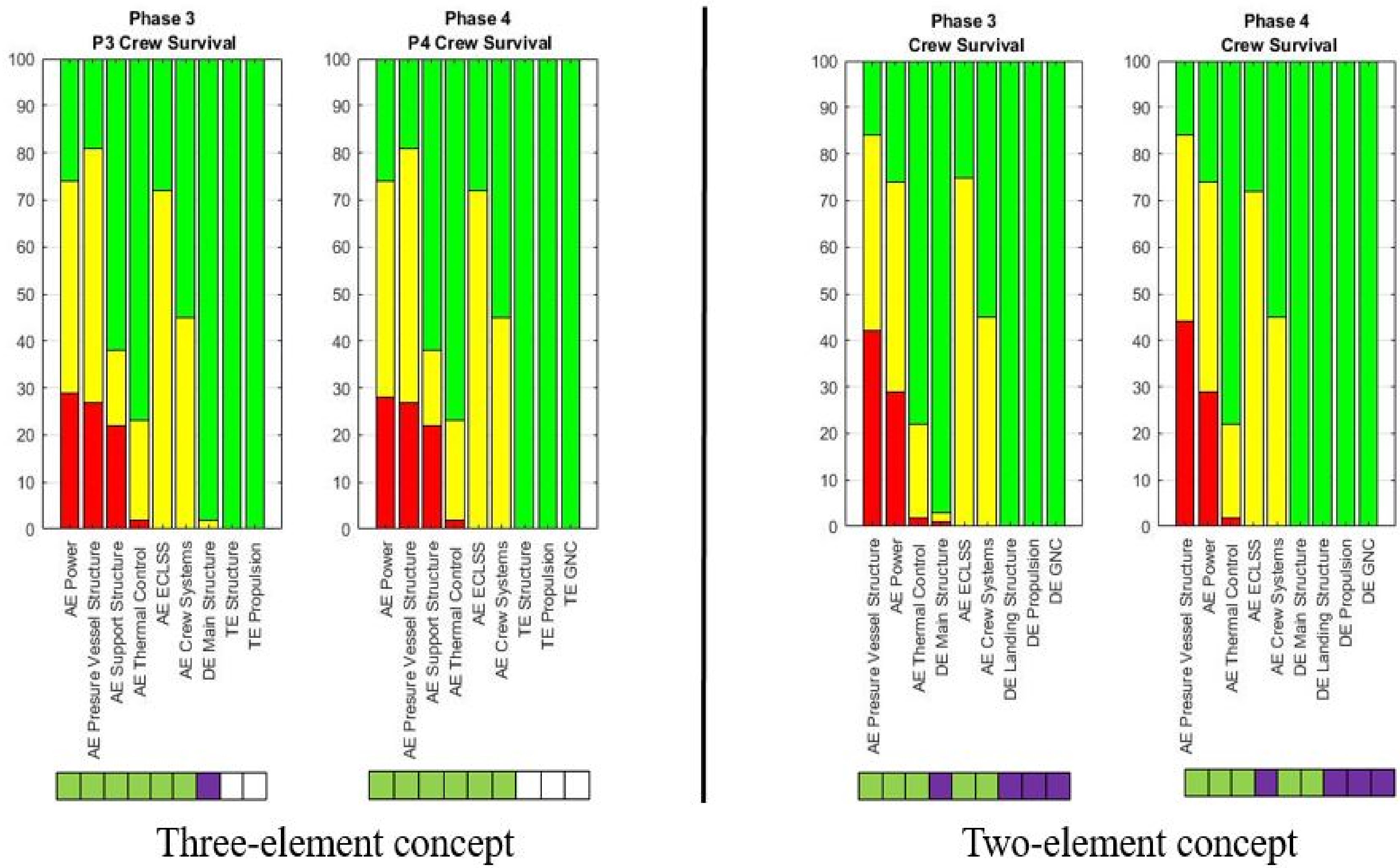

Figure 4.17. Comparison of most impactful node disruptions on the crew survival capabilities in phases three and four between the two concepts

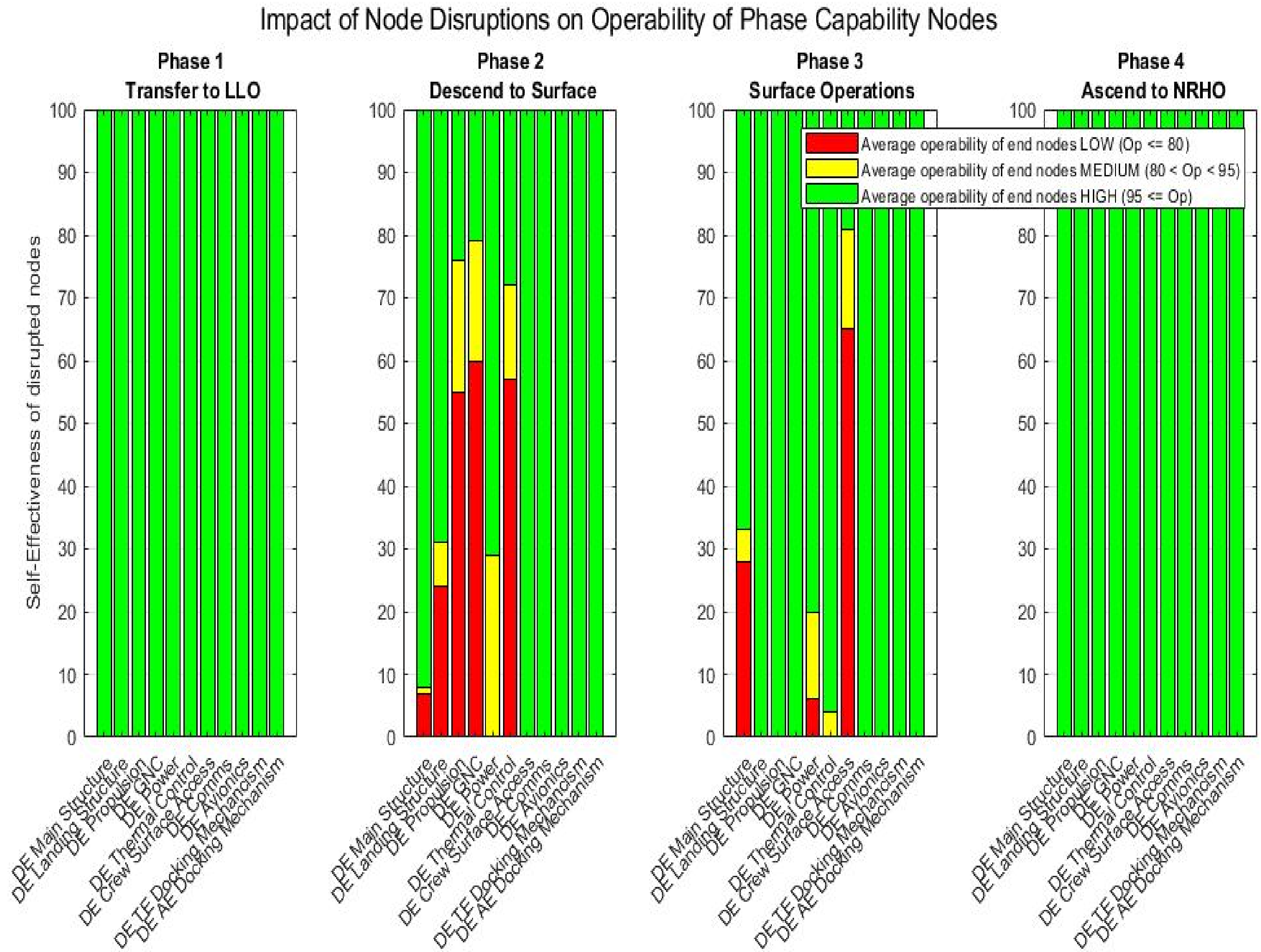

Figure 4.18. Impact of DE subsystem disruptions on the phase completion capabilities for each phase in the three element architecture.

4.3.2 Node Self-Effectiveness Probability Distributions

The probability distributions assigned to each node are based on the reliability of each subsystem. Various distributions such as uniform distribution and beta distribution can be assigned to each subsystem. For the HLS subsystems, beta distributions were used with two general shapes. Subsystems that degrade such as Power, ECLSS, and structure are given beta distributions that are heavily skewed to the right to give high expected values for SE. Other systems that are more susceptible to binary failures than degradation are given "bathtub" shaped distributions that have a large tail on the right and a smaller tail on the left to represent the small probability of a complete failure.

For previous applications of SODA to space exploration SoS's, the systems are presumed to be active over quite a long period of time. In these cases it can be expected that systems may degrade over time due to exposure to the space environment and operational wear. In stochastic analysis, the SE values of such systems are assigned beta distributions to approximate the reliability of the system over that time period. These distributions generally have high expected values in the 80s or 90s. Since the HLS mission phases take place over a relatively short timescale and subsystems are made to be very reliable and often redundant, it is reasonable to assume that there will not be mush loss of SE during the mission. One of NASA's requirements is that the HLS hardware have a reliability of 0.975 for a lunar sortie [19]. Sustained missions must have per mission reliability of at least 0.98 from Gateway separation to Gateway return [19]. Therefore, accurate SE distributions for all subsystems would have very high expected values in the upper nineties. For demonstration purposes I will assign each subsystem SE values following the $beta(50, 1.5)$ distribution in the first phase and decrease the value of the first parameter to 30, 20, and 15 in the subsequent phases. This will decrease the expected value of the distribution across the phases to represent degradation of subsystems over time.

4.3.3 Analysis using Stochastic Impact Plots

Using the SE distributions defined above, the stochastic impact of each node can be computed in each phase following the process outlined in section 3.2.2. Results for some of the AE systems from the three-element concept are shown in Figure 4.19.

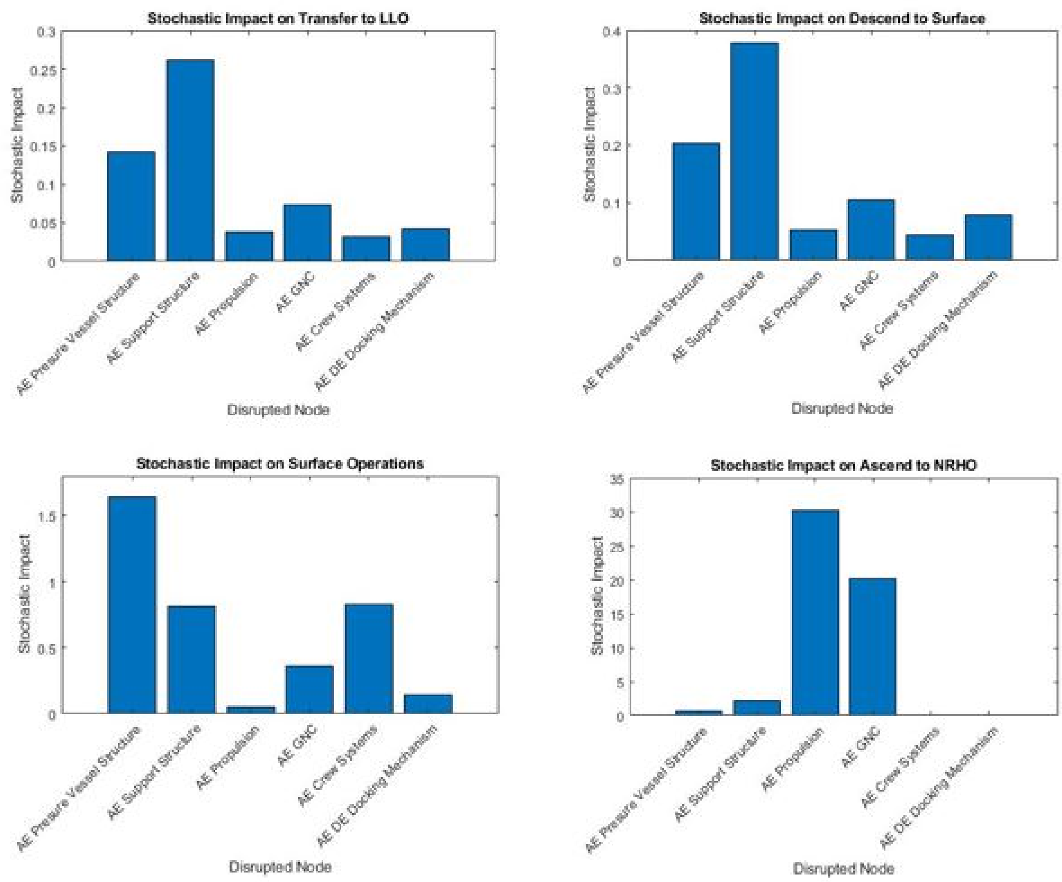

Figure 4.19. Stochastic Impact Plots for AE subsystems in the three-element concept

While the FIR sequence plots show the impacts of individual disruptions on operability of a capability node, the Stochastic Impact plots provide a holistic representation of the impacts of subsystem disruptions in the context of the stochastic network. As can be seen from the plots, the relative stochastic impact of subsystem disruptions changes across the phases. In addition, the increasing scale of the y-axis as the phases progress indicates that the largest impacts in each phase are growing. This makes sense because in each phase the subsystems where assigned SE distributions with lower expected values, which should correspond to higher expected impacts.

4.3.4 Calculation of Overall Mission Success

The multi-phase operations of some complex systems like the HLS require that one phase is completed at a certain level of performance before the next phase can occur. Failure of one phase will most likely prevent the subsequent phases from happening, and might trigger some contingency process instead. For example, a major failure preventing completion of the transfer maneuver from NRHO to LLO would likely result in aborting the rest of the mission and returning the crew to the Gateway station or Orion capsule. When analyzing alternative SoS using SODA, a simple way to compare the overall quality of alternatives is to compare the expected values of nodes of interest based on stochastic analysis. For multi-phase architectures that must follow a sequence of phases, rather than comparing the expected values of the various phase capabilities achieved by alternative architectures, the overall process represented by the capability nodes can be considered. A simple approach is to represent the process as a sequence of phase capabilities with capability operability thresholds at the end of each phase to decide whether to proceed to the next phase. For relatively short phases, such as those considered for the HLS mission, the operabilities can be computed using the standard stochastic analysis. For longer phases, system degradation and random disruptions can also be considered and the operabilities of capability nodes at the end of the phase duration will be compared to the thresholds. Proceeding through the sequence of phases for many instances of the stochastic model provides a percentage of successful missions for each alternative architecture, which can be used as a metric for comparing the quality of each alternative.

This approach can also be used to determine if the process often fails at a particular phase. Once a problematic phase is known, further analysis can be done

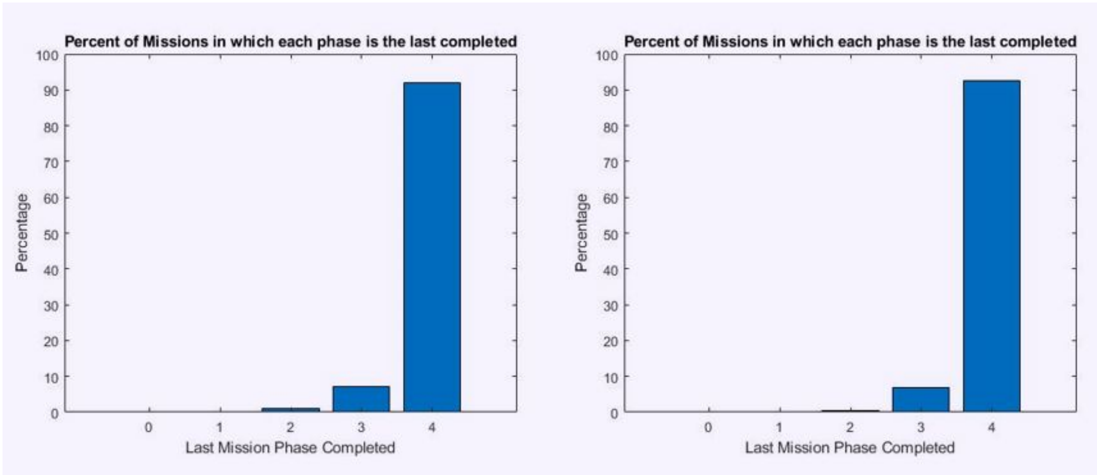

Figure 4.20. Percent of missions in which each phase is the last completed in the three-element concept (left) and two-element concept (right). Capability operability threshold for phase completion is 90.

with FIR sequence plots and Stochastic Impact plots to determine which nodes and dependencies might be causing the problem.

Figure 4.20 shows the percentage of stochastic runs in which each phase is the last completed for both HLS concepts. A phase is successfully completed if both the phase completion and crew survival capability nodes have operabilities above a certain threshold. The threshold was set to 90 for these runs. About 92% of the runs for each concept complete the fourth phase, meaning they successfully completed the full mission. About 10% completed the third phase but not the fourth. A very small portion completed phase two but not phase three. The similarity between the plots suggest that there is not a significant difference in the overall quality of either concept in terms of achieving mission success. It is important to remember that the purpose of this work is not to truly evaluate the competing architectures, but rather to use them to demonstrate new analysis tools. The scope of the four-phase HLS mission considered in this work makes simplifying assumptions about the operational process and leaves out some important differentiating factors between the two concepts such as launch and automated docking complexities. The models were created using publicly available data, so more accurate evaluation of concept differences requires more detailed information about each design.

4.3.5 Usefulness of Results

The two concepts considered represent alternative designs in which either more systems must be built but only used once (e.g. TE), or fewer systems must be built but used for multiple critical operations (e.g. DE in two-element concept). The FIR sequence plots quantify the impacts of disruptions in each of these two concepts across multiple mission phases so that designers can make better decisions about which alternative may be better. This data can be compared to information regarding the cost of developing new systems vs. making other systems more robust to disruptions. For example, if it is known that it will be inexpensive to make the most impactful DE subsystems (GNC, propulsion, thermal control) more robust to disruptions, it may be beneficial to choose the two-element concept as it eliminates the extra TE. On the other hand, if making an expendable TE that has high reliability for a shorter lifespan is cheaper, then these results can help focus effort on the most impactful subsystems to ensure success. One aspect this analysis doesn't consider is the actual transition process between phases, such as docking and undocking operations. The ease of transition between phases is an important consideration when choosing between multiple alternatives.

The second version of the FIR sequence plot, in which impacts of disruptions in the same group of nodes is compared across evolutionary phases, can be used by designers to help understand when particular systems are most important. When planning the developmental evolution of an SoS, this can be helpful for determining how many resources need to be allocated to each system at different phases of development. For example, if a system can be upgraded or replaced over time and it is required, but not critical, for an earlier mission phase, less emphasis can be placed on making that system robust initially. Later on, resources can be dedicating to making the system more robust for the phases in which it is most critical through hardware enhancements/replacement or software updates. The ability of designers to make these kinds of upgrades obviously depends on the application. Some subsystems in a persistent habitat like the Gateway station can likely be upgraded through EVA's if the station will be entering a new mission phase that more heavily relies on certain subsystems. Also, the US Air Force is leading an effort to develop reprogrammable satellites to meet evolving mission needs [31]. In this scenario, the impact of disruptions in software functions could be assessed for different cases of mission evolution to help determine when certain software must be operating with high reliability to ensure mission success.

The stochastic impact results can be used in similar ways to the deterministic results, but they can be used to measure the overall risk of a concept within an evolutionary phase or across multiple phases. The stochastic impacts of each node in the network can be compiled through summation or averaging across all evolutionary phases to provide a single overall value of risk for that node. The overall stochastic impacts of key nodes can be compared between alternative concepts to see if one offers a clear advantage in expected overall performance. The stochastic impact of all nodes in a concept can also be compiled to provide a single value representing the risk of that concept, which can easily be compared with other concepts.

5. CASE STUDY: ARTEMIS MISSION SEQUENCE DEVELOPMENT

5.1 Artemis Program Definition

NASA is planning a set of crewed Artemis missions to make multiple landings on the lunar surface and establish a permanent habitat on the Moon and in lunar orbit. While the Gateway was moved off the critical path for the initial Artemis landing planned for 2024, it will be an integral part of later missions and establishing a sustainable presence at the Moon. The initial version of Gateway will consist of a Power and Propulsion Element (PPE) and a Habitation and Logistics Outpost (HALO). All elements of the first HLS will be discarded after use, but NASA plans to reuse the ascent element from later missions to increase the sustainability of lunar surface access.

Here I will consider a sequence of three Artemis landing missions to demonstrate the integration of SODA and SDDA. The first mission will utilize all disposable HLS elements and one Orion capsule. The second mission will use all new HLS elements, but the AE will not be discarded afterwards. This mission will also utilize the Gateway station to transfer crew from a new Orion capsule to the HLS. The third mission will use the AE still docked to Gateway from the second mission along with a new DE, TE, and Orion. The system configuration required for each landing mission is considered a phase in the evolution of the Artemis SoS.

5.2 Description of SODA Model

For this study, SODA will be used to assess the operability of phase capability nodes in the HLS mission at the time when the landing mission begins. Each subsystem in the HLS begins to degrade and experience random failures once it is launched. The degradation model used can decrease a system's SE in three different ways and is applied at time steps of one month [10]:

- Small decrease in SE every time step to represent degradation during operations (e.g. decrease in solar panel efficiency over time)
- Low chance of a minor disruption to SE (e.g. small disruptions in electronics or data from radiation single-event effects)
- Very low chance of a major disruption to SE (e.g. severe damage to hull from micrometeoroid impact)

Given the short duration of the actual lunar landing mission, degradation is only applied to HLS subsystems up to the beginning time of the mission, not during the landing mission itself. The SE values of the HLS subsystems at the time when the landing mission begins are used to compute the operability of all the phase capability nodes.

Developmental delays may impact the operability of capability nodes by changing the length of time between completion of some systems and the beginning of the mission in which that system is used. Increasing this time span increases the duration over which the system can degrade or experience random failures before the landing mission begins. This generally corresponds to lower node SE at the start of the mission and therefore lower operabilities for capability nodes.

For situations in which the evolutionary phases occur over a longer time scale, degradation can be continued during the phases and the operability of phase capability nodes can be computed across their respective phases. In these cases, developmental delays would affect the beginning SE values of newly added systems in each phase while existing systems will continue to degrade across the phases.

5.3 Description of SDDA Model

The SDDA model is built based on NASA's Artemis program timeline and plan for sustainable lunar presence. Developmental dependencies are based on the common technology shared between systems or on the planned operational interactions that require partial development of one system to determine interfacing.

Figure 5.1 shows the SDDA network for the portion of the Artemis program under consideration. The nodes labeled with 'Dev' indicate the design phase of development for that node while the corresponding nodes without 'Dev' refer to the construction and testing of that node. It should also be noted that I am considering Artemis missions starting with Artemis-3 (the first crewed lunar lander mission), so the Orion 1 node refers to the Orion capsule used for the Artemis-3 mission rather than the first Orion capsule built for Artemis-1.

Developmental dependencies are based on technology needs of systems or interfaces and interactions between systems that impose a developmental dependency between two nodes. The minimum and maximum independent development times are based on NASA's current schedule for the Artemis program. I use January 2020 as the beginning of the timeline. The stated planned launch date for each system is assigned as the minimum development time and a 50% buffer is added on for the maximum development time. The HALO module is expected to reach PDR by the end of 2020 [32]. Based on NASA's Project Life Cycle diagram [33], PDR indicates the project is near the end of design and fabrication can begin. Therefore, the HALO Dev node is given a MINIT of 1 year and the HALO node is made dependent on HALO Dev with an SOD value of 0.8 to indicate that construction can begin before design is completely finalized. The PPE and HALO modules are planned to be launched integrated on a single vehicle by the end of 2023 [32]. To match this timeline, the HALO node is assigned a MINIT of 3 years. Since the PPE needs to be on approximately the same schedule as the HALO for an integrated launch, the

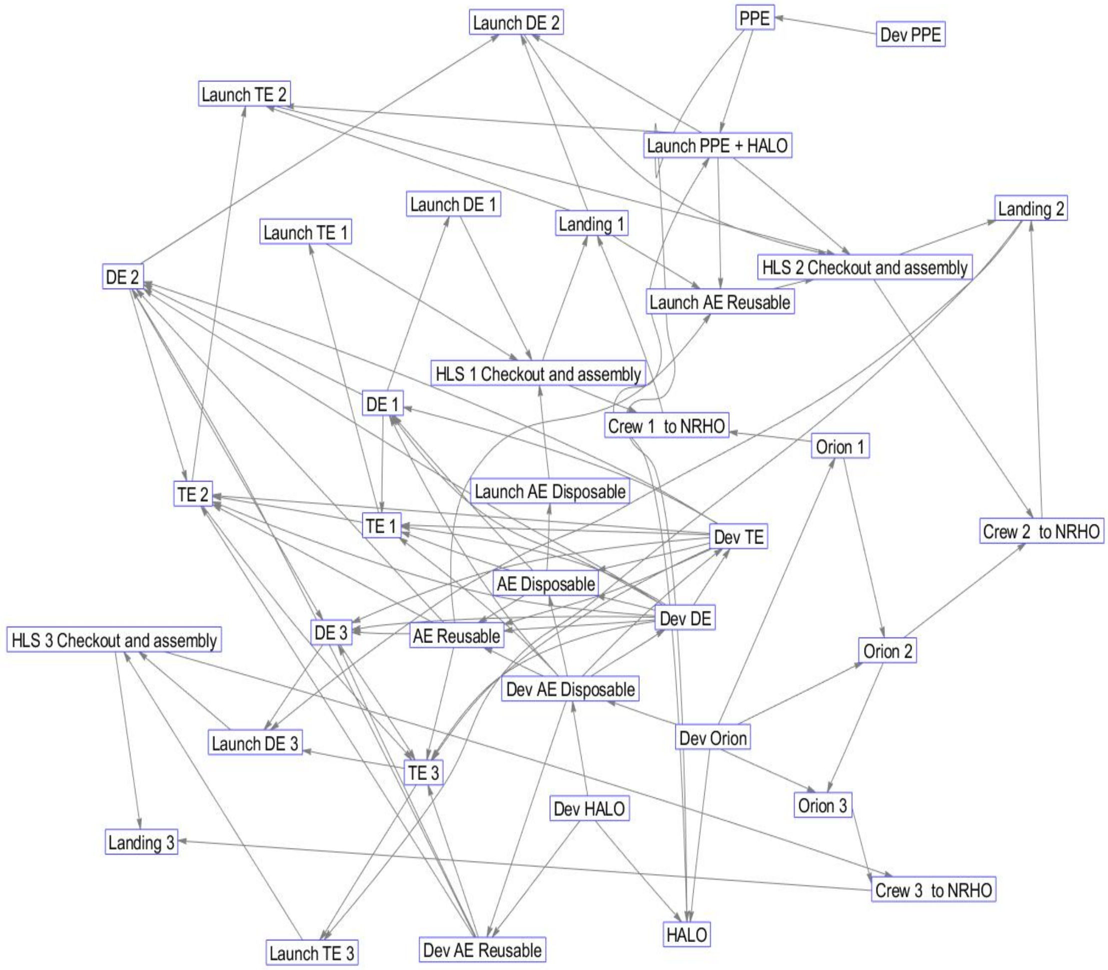

Figure 5.1. Artemis developmental dependency network

PPE Dev and PPE nodes are given the same development times as HALO Dev and HALO, respectively.

The SDDA dependency matrix must be acyclic, meaning that two nodes cannot be mutually dependent on one another. For systems that are largely developed concurrently but are also highly interdependent, such as the HLS elements, an order of development must be prescribed. I chose this order to be AE, DE, TE for all three HLS missions considered. Each DE is dependent on its corresponding AE, and each

TE is dependent on its corresponding DE. Small values of SOD are used for these dependencies to allow them to be developed largely at the same time. All iterations of the DE, for example, are dependent on DE Dev, but the SOD value is decreased for subsequent iterations. Similarly, each DE iteration is dependent on the previous iteration.

5.4 Integrated SODA-SDDA Implementation and Results

Following the process laid out in section 3.3, a model of the evolution of the SoS and operability of key capability nodes over time can be analysed with disruptions and delays. Two classes of analyses are performed. The first considers the impact of developmental delays on the operability of HLS capability nodes for each lunar mission. The second investigates the developmental impact of operational failures in HLS elements that require redesign and reconstruction.

5.4.1 Impact of Developmental Delays On HLS Capabilities

Figure 5.2 shows the baseline Artemis HLS development timeline with no delays. The timeline proceeds from designing and building each system to incrementally launching sets of HLS elements and Orion capsules to complete three lunar lander missions. Each HLS mission is indicated in the baseline schedule.

To demonstrate how developmental delays can impact the operability of HLS capability nodes, I will go through an example considering delays in the AE Disposable node. The driver of operational impacts in this situation is the degradation over time of HLS elements from the time they are launched until the start of the lunar landing mission. Figure 5.3 shows an example of the decrease in average HLS operability due to the decrease in HLS subsystem SE values. Note that these operability values are always decreasing over the time span of 10 months. The longer this time span is, the lower the operabilities will be when the landing mission begins.

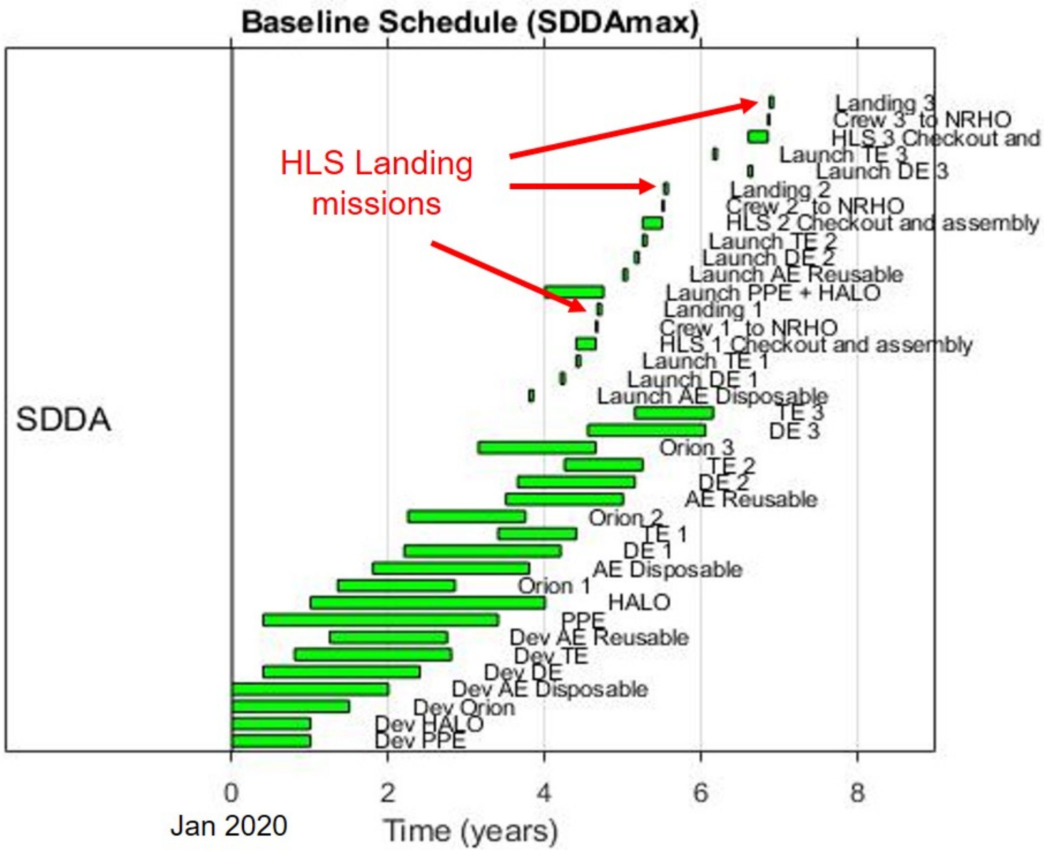

Figure 5.2. Gantt chart of Artemis development timeline with no delays

Delaying the AE Disposable element impacts the time span between the introduction of each HLS element and the beginning of the mission as shown in Figure 5.4. The time span for AE Disposable increases until a delay of 60 where it flattens out. This is because of the COD values in the dependency of DE 1 on AE Disposable. For a node j that is developmentally dependent on node i, recall that COD_{ij} specifies the delay in node i at which development of node j will no longer start before node i has finished. If $SE_i < COD_{ij}$, then the beginning time (BT) for node j gradually increases as SE_i decreases. This continues until the beginning time of node j equals the completion time of node i ($BT_j = CT_i$), which corresponds to $SE_i = COD_{ij}$. Once this point is reached, BT_j will no longer increase as the delay in node i in-

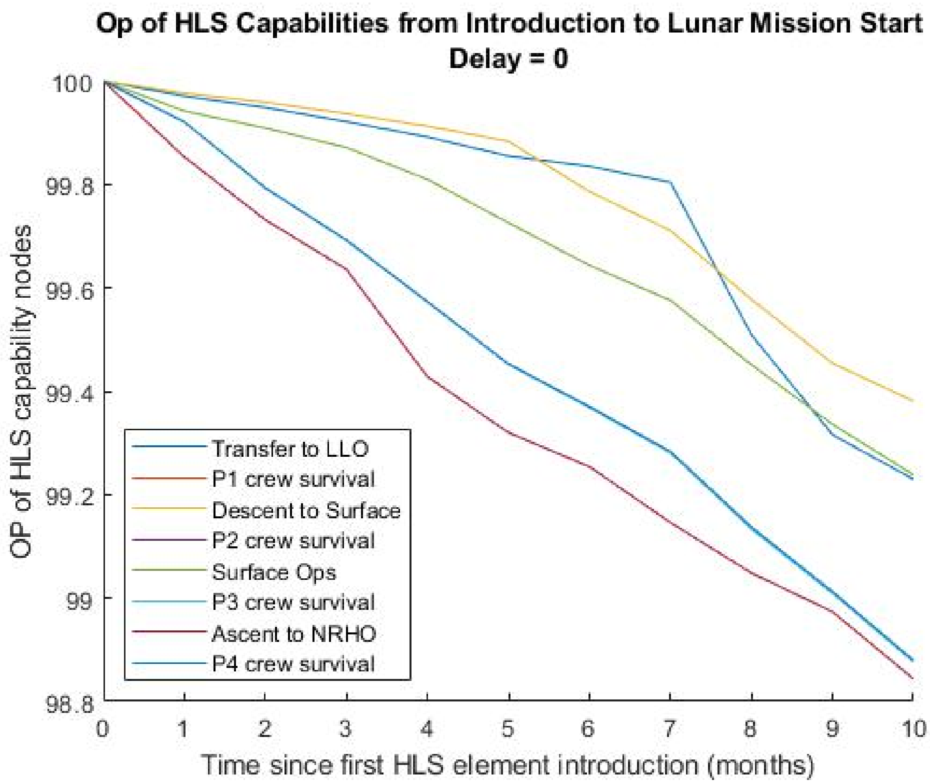

Figure 5.3. Decrease of HLS capability operability over time

creases. The development times for DE 1 and TE 1 are not impacted by the delay in AE Disposable, so the time between AE Disposable completion and the beginning of Landing 1 only depends on how late DE 1 development starts compared to when AE Disposable begins. Delaying AE Disposable causes DE 1 to begin later and later until DE 1 starts at the same time that AE Disposable completes.

Figure 5.5 shows the average impact of delaying AE Disposable on the operability of HLS capability nodes. Note that unlike in Figure 5.3, the curves don't always decrease. In this plot the x-axis represents delay rather than time, and while longer delay often means more time for system degradation, Figure 5.4 shows that this isn't always the case. The operabilities generally decrease as the time span increases, but at a delay of 60, the time spans become constant and the operabilities appear to level

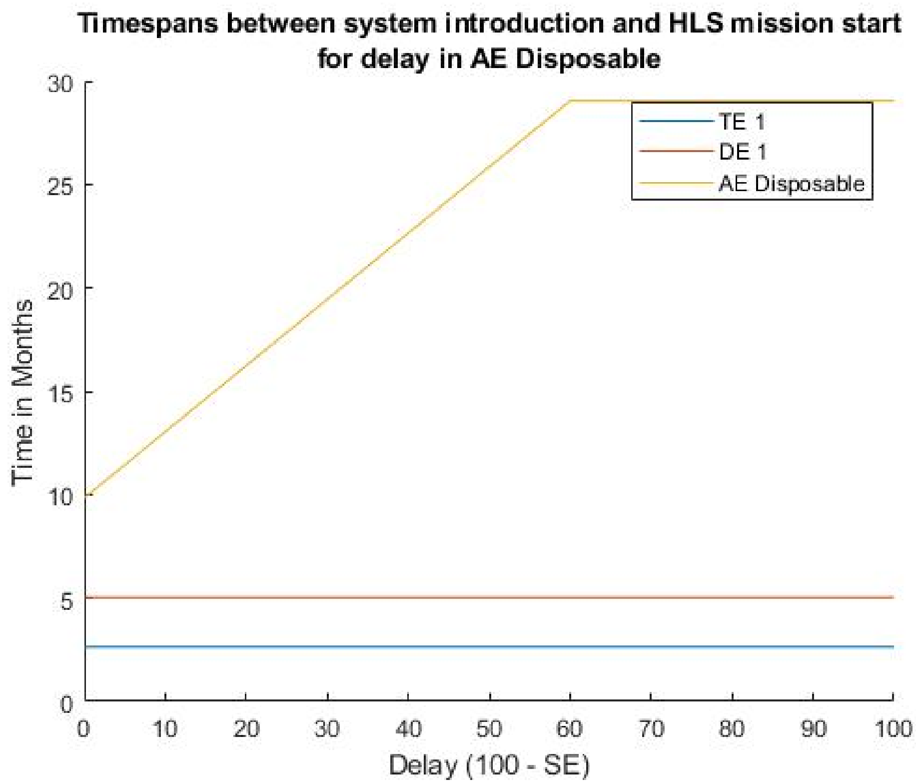

Figure 5.4. Impact of delays in AE Disposable on time spans between HLS element introduction and lunar mission start

off as well. The slight jumps in the operability curves are a result of the stochastic nature of the degradation model used.

As done previously, the operabilities of the HLS capability nodes can be compared against a threshold at each phase to determine if the full mission is completed successfully. Figure 5.6 shows the impact of delays in AE Disposable and DE 1 on HLS mission completion percentage. Once again, the curve generally decreases as delay increases until "leveling off" after delay = 60. Again, the curve isn't perfectly smooth because of the stochastic degradation model.

This same kind of analysis is repeated to determine the impacts of several different systems on the capabilities of each HLS mission.

Table 5.1 shows the delay in lunar mission start time that results from delays in several of the developed systems. As expected, longer delays in development corre-

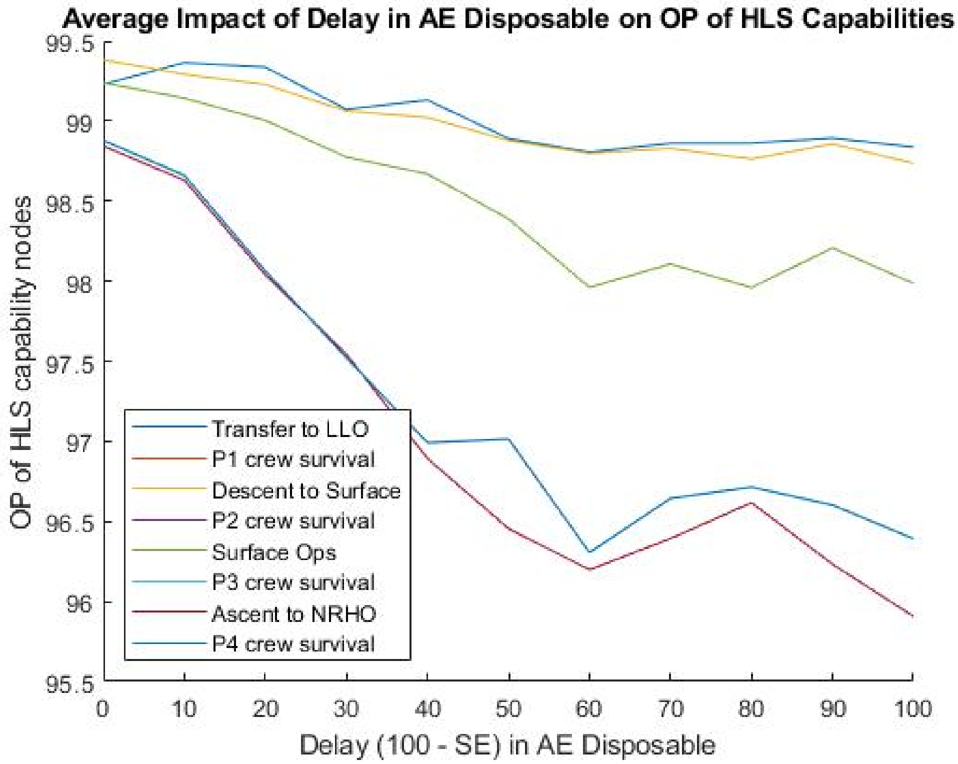

Figure 5.5. Impact of delays in AE Disposable on average operability of HLS capability nodes

spond to longer delays in mission start times, but not all missions may be affected equally by a delay in a given system. For example, delaying TE 1 has a larger impact on the start time of the first mission than on the second and third missions. This is because some of the initial delay is absorbed within the schedule before the later missions begin. Interestingly, delays in Orion 1 are completely absorbed for all missions. Delays in the HALO also don't have much impact, with relatively small delays on the start time of the second mission and no impact on the third mission due to delay absorption by other systems. In general, delays in a system have the most impact on the start time of the mission during which that system is used or introduced.

Figure 5.7 shows the impacts of developmental delays in different systems on mission completion percentage for the first HLS mission. The PPE and HALO are not required for the first mission so delays in these systems have no impact on mission

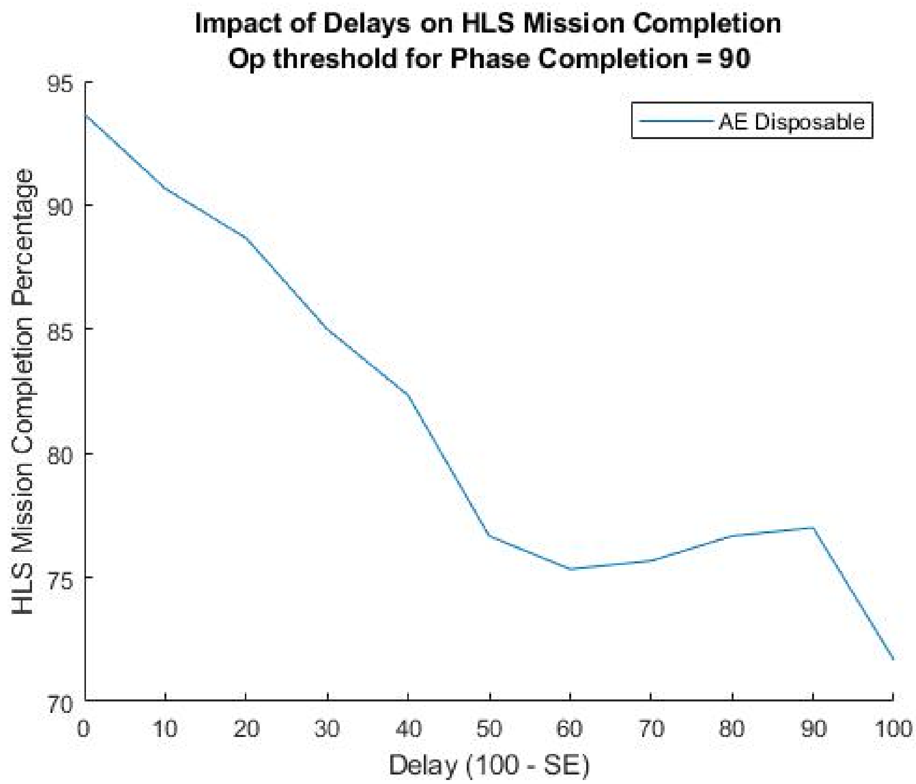

Figure 5.6. Impact of delays in AE Disposable on HLS Mission 1 Completion Percent

completion. Orion 1 also does not have an impact since it is completed far enough ahead of the other HLS elements that even its maximum delay does not push back the mission. The AE, DE, and TE all have growing impacts on mission completion likelihood as they are delayed more. A delay in any of these elements after one or more of the others have already launched creates significant time spans over which the deployed elements can degrade, lowering the likelihood of mission completion.

Figure 5.8 shows results of a similar analysis for the second HLS mission. Here, delays in the PPE have by far the most impact, but delay sin HALO don't have as significant an effect. This result is somewhat counter intuitive since the PPE and HALO launch together and one might expect delays in either to have a similar effect on the launch time. The difference is that the HALO is dependent on the PPE but no other systems HLS elements are directly dependent on the HALO. This means

Table 5.1. Impacts of Developmental Delays on HLS Mission Start Time

Delayed System	Delay (100-SE)	Mission 1 Start Delay (years)	Mission 2 Start Delay (years)	Mission 3 Start Delay (years)
AE Disposable	20	0.933	0.883	0.883
	50	2.333	2.283	2.283
	100	3.6	3.55	3.55
DE 1	20	0.667	0.6929	0.6929
	50	1.667	1.55	1.55
	100	2.8	2.55	2.55
TE 1	20	0.4	0.379	0.079
	50	1	1.15	0.85
	100	2	2.15	1.85
Orion 1	20	0	0	0
	50	0	0	0
	100	0	0	0
PPE	20	0	0.6	0
	50	0	2.25	1.22
	100	0	3.4	2.37
HALO	20	0	0	0
	50	0	0.25	0
	100	0	1	0
AE Reusable	20	0	0.75	0.75
	50	0	1.875	1.875
	100	0	2.85	2.85
DE 2	20	0	0.6	0.643
	50	0	1.5	1.35
	100	0	2.4	2.1

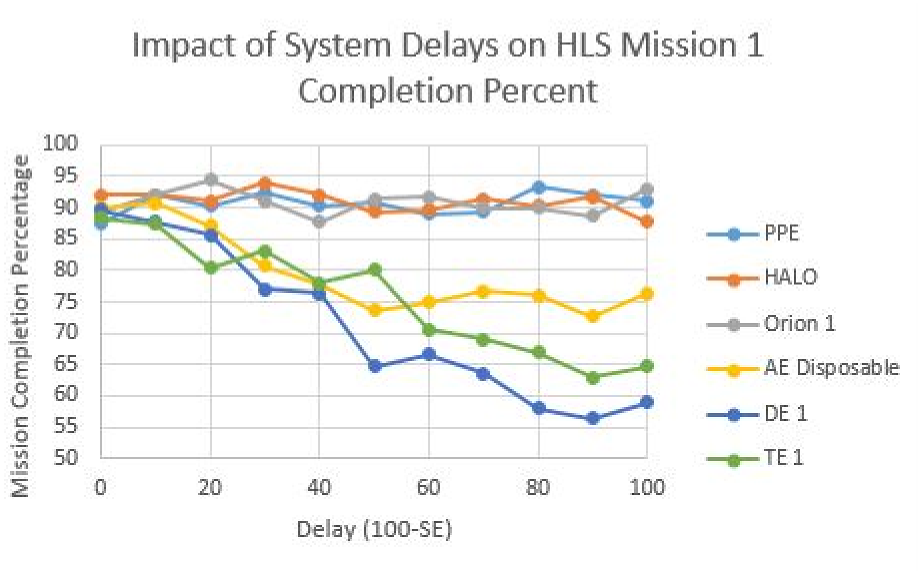

Figure 5.7. Impact of System Delays on HLS Mission 1 Completion Percent

that delaying the PPE will gradually delay the start time of HALO (as discussed previously) and create a compounded delay, while delaying the HALO will add on a relatively short amount of development time. Since none of the other HLS elements depend on the HALO, they each launch once they are completed and wait in orbit through any delays in the PPE or HALO. It should be noted that in a real scenario, mission planners would probably delay launching any of the HLS elements required for a given mission until all the systems are ready to go, essentially eliminating any extra degradation time. However, the HLS elements are assumed to launch once they are completed in this study to demonstrate the types of results that can be observed in such a case when the introduction of systems into operation cannot be as closely controlled.

The impacts of delays on completion of the third HLS mission are shown in Figure 5.9. The overall mission completion percentage is lower than for the first two missions because of the 'AE Reusable' element's continuous degradation since the beginning of the second mission. Similar to the second mission, delays in the PPE have a

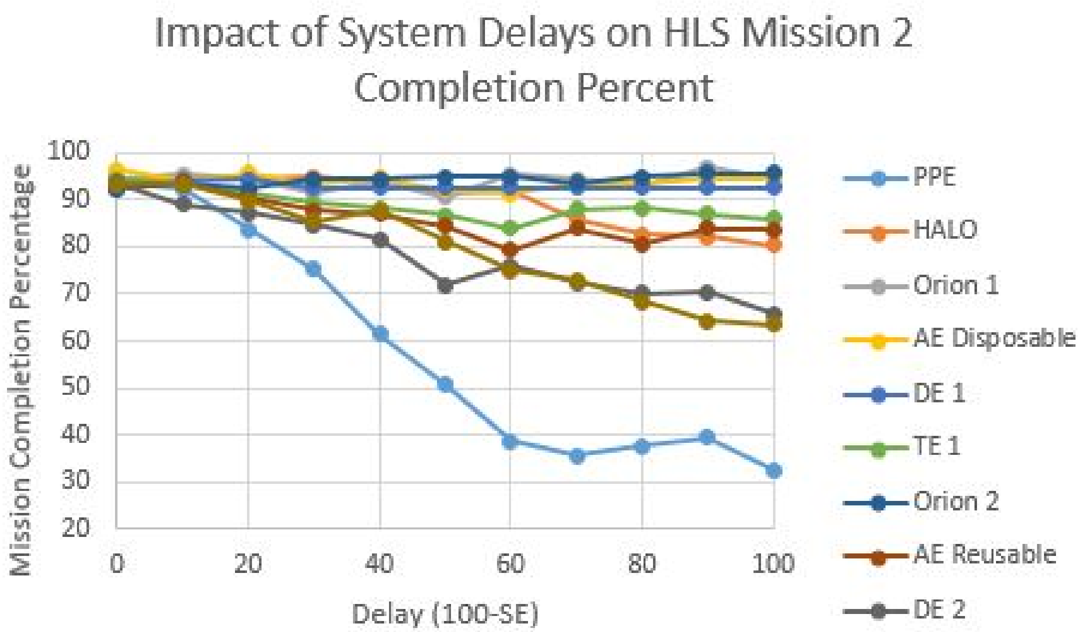

Figure 5.8. Impact of System Delays on HLS Mission 2 Completion Percent

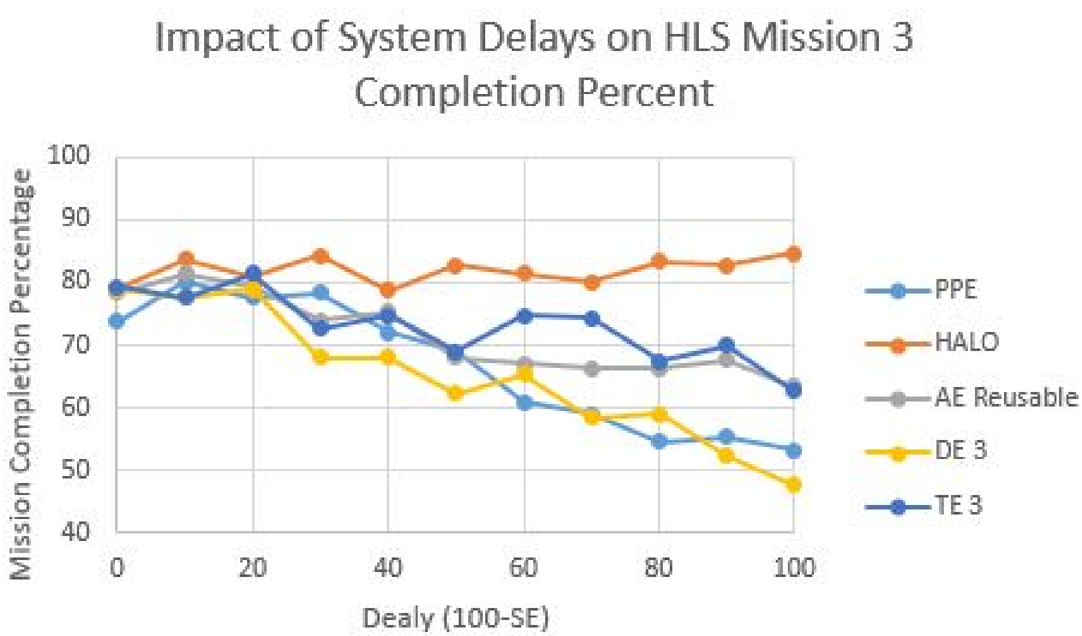

Figure 5.9. Impact of System Delays on HLS Mission 3 Completion Percent

significant effect on mission completion but delays in HALO have no impact on the likelihood of mission success.

5.4.2 Impact of Operational Failures On Development Schedule

It is possible that a deployed system could fail before all the systems required to implement a lunar mission are ready. This could require a redesign of the failed system and reconstruction so that the planned lunar mission can carry on at a later date. Introducing this redesign into the development schedule obviously could impact the overall development timeline. Given the formulation of SDDA, the time at which a certain deployed system fails is significant to how the development schedule reacts. If, for example, the AE Disposable fails during its flight to the Moon before DE 1 and TE 1 are completed, then AE Disposable will be redesigned while the development times of DE 1 and TE 1 are extended due their developmental dependence on AE Disposable. If the AE Disposable fails after DE 1 and/or TE 1 are completed and launched, then AE Disposable will go through redesign while the DE and TE wait in lunar orbit. Looking back on the previous analysis, the waiting period for the DE and TE will also cause them to degrade and reduce the operability of HLS capabilities at the mission start time.

For this analysis, systems are deterministically failed at specified times after their introduction into operation. When a system fails, a new development node is added to the timeline at time of failure to represent the redesign and reconstruction of the failed system. The redesign is given a development time equal to 80% of the original system's development time, but this percent can be varied for different situations. The systems that were developmentally dependent on the failed system are made dependent on the system redesign.

Figure 5.10 shows the adjusted development timeline when the PPE fails at 20% of its life time. The life time is defined as the time duration from which the system is introduced into operations to when it is discarded or no longer used. So the life time of AE Disposable is from the launch of AE Disposable to the end of Landing 1. The life cycle of the PPE is the time from launch of the PPE to the end of Landing 3 (since that is the last landing considered in this study). A new development node is added for redesign of the PPE, which causes the second and third landing missions to be delayed until a new PPE is constructed.

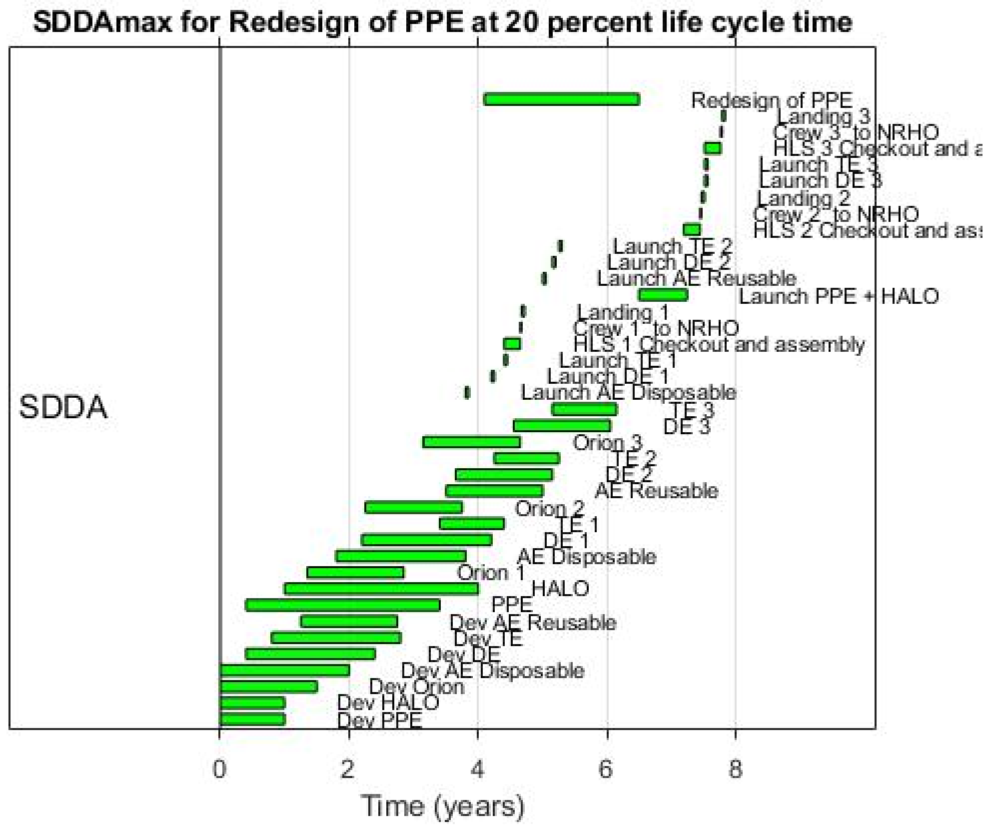

Figure 5.10. Development timeline with redesign of PPE after failure at 20% of its life time

The impacts of operational failures on the delay in start time of each of the three HLS landing missions are considered separately. Figure 5.11 shows the impact of operational failures and redesign of certain systems on the delay of the first HLS landing mission. The delay in mission start time generally increases as the failure time increases. Failures in AE Disposable and DE 1 impose the longest delays. The jagged curves are due to the development of systems that are dependent on the failed system. For example, the impact of a failure in AE Disposable increases as it fails later up to a failure time of 40. Between 40 and 50, the DE 1 that was dependent on AE disposable complete development and enters operation. Before this time, the redesign of AE Disposable would also delay DE 1 and result in a larger overall delay on the mission schedule. After this time, however, the AE redesign does not delay the DE (since it is already complete) so the overall delay impact is less. An early failure of Orion 1 has no impact on the mission start time because it is completed early enough that the redesign can be finished before the baseline start time of the mission. Failures in the PPE and HALO do not delay the first HLS mission since they are not required for the first landing.

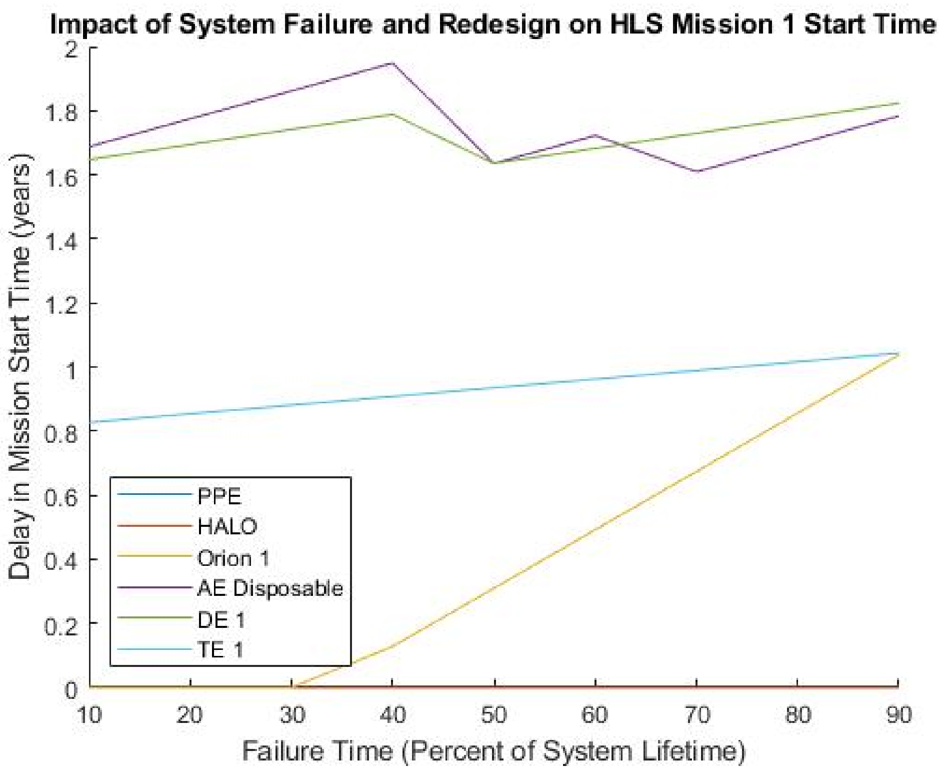

Figure 5.11. Impact of operational failures on delay of the first HLS landing mission

Figure 5.12 shows the delay of the second HLS mission caused by system failures and redesign. Here, the DE, TE, and Orion all have increasing impact as they are failed later, as expected. Later failures in the PPE and HALO also cause increasing delays up to a cutoff point. This is the point in the PPE and HALO life time at which the second landing mission occurs, so failures in these systems after this point will not delay the second mission. A similar result can be seen for AE Reusable.

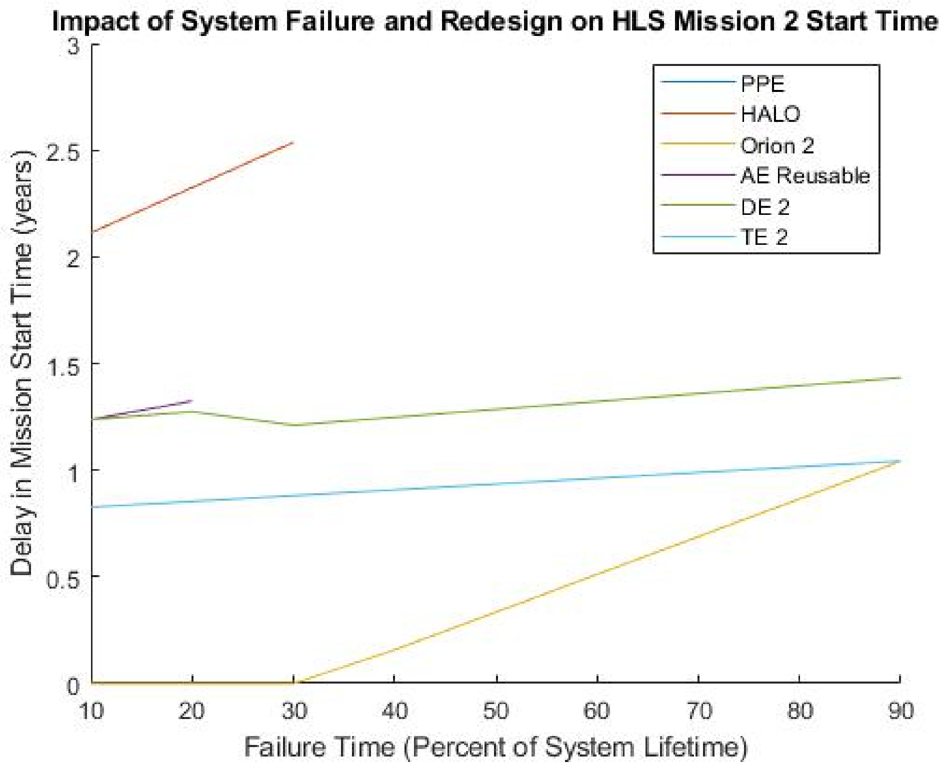

Figure 5.12. Impact of operational failures on delay of the second HLS landing mission. PPE and HALO both shown by orange line.

Results for the third HLS mission are shown in Figure 5.13. The curves for the AE, DE, TE are once again jagged because of the dependent systems that their redesigns can delay. Failures in the PPE or HALO impose the longest delays on the start of the third mission. Since they have longer independent development times, the assigned redesign time is longer as well.

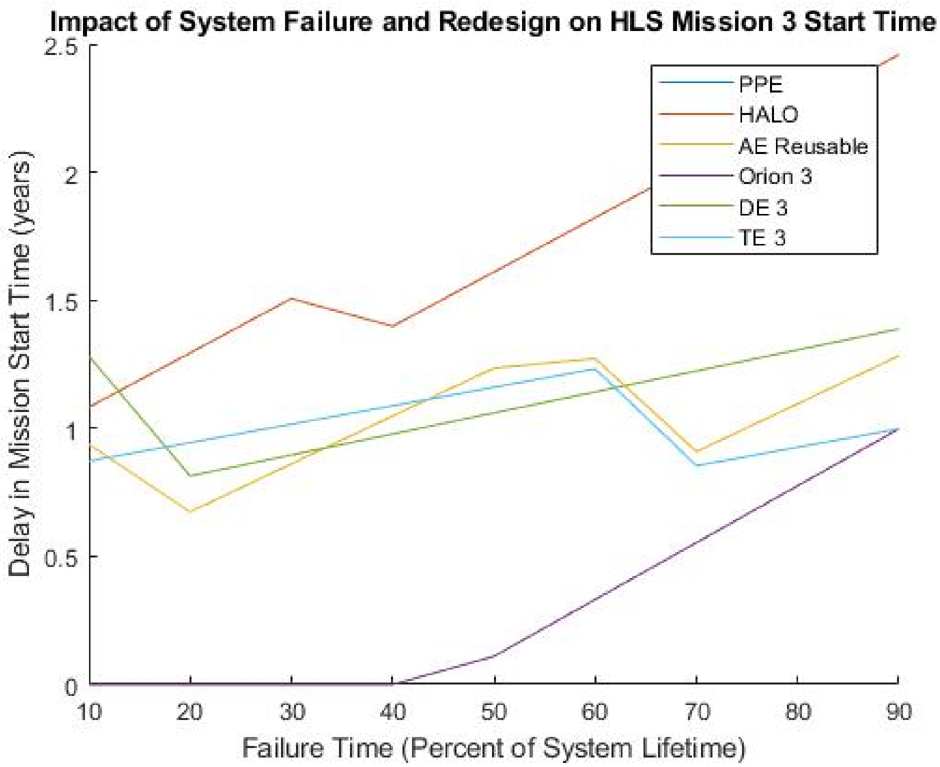

Figure 5.13. Impact of operational failures on delay of the third HLS landing mission. PPE and HALO both shown by orange line.

5.4.3 Usefulness of Results

The results from integrated SODA and SDDA can generally be used to help managers plan schedules and allocation of development resources to minimize the impacts of delays and operational failures on successful implementation of a program throughout its development.

Analysis of the impact of delays on system degradation and operations can be used to inform when systems should be introduced into operation. To decrease the degradation of systems before they are utilized it is best to introduce all systems as late as possible before they are required. In real development this may not be possible due to budget and resource constraints. The launch schedule of systems for space exploration programs may be restricted by availability of launch vehicles and facilities as well as launch windows imposed by the relative positions of planets. If a certain system is very robust and unlikely to degrade or fail in operations, it can be introduced early while systems with short lifespans should be introduced later. Quantification of the impact of delays on system degradation can be used to plan contingencies for certain delay scenarios.

This could also be used to evaluate tradeoffs between possibly having to redesign a single complicated system or developing multiple less robust systems that are faster to develop. While a failure in a highly reliable monolithic system may be unlikely, a failure might severely delay implementation of the next evolutionary phase. Development of a few less robust systems that can fulfill the same role may initially increase the development time but provide an ability to "absorb" the failure of one of the systems without requiring a redesign to regain the capability. This would effectively increase the robustness of the schedule to operational failures. SODA tools such as stochastic impact plots can be used to asses the likelihood of a system being disrupted enough that a replacement is required.

5.5 Relationship between SDDA and Integrated Master Schedules

Integrated Master Schedules (IMS) are used by program managers to define the development schedule of a program through its major milestones. In order for SDDA results to be useful, the relationship between these two tools must be discussed. An example IMS for the Orion program is shown in Figure 5.14. PERT is often used in IMS development to model the developmental dependencies between tasks and events. A major difference between PERT and SDDA is that SDDA can model partial dependencies whereas PERT assumes that a dependent system cannot start development until a predecessor is finished [10]. SDDA allows for system development to start before all its predecessors are complete based on the values assigned to SOD and COD parameters.

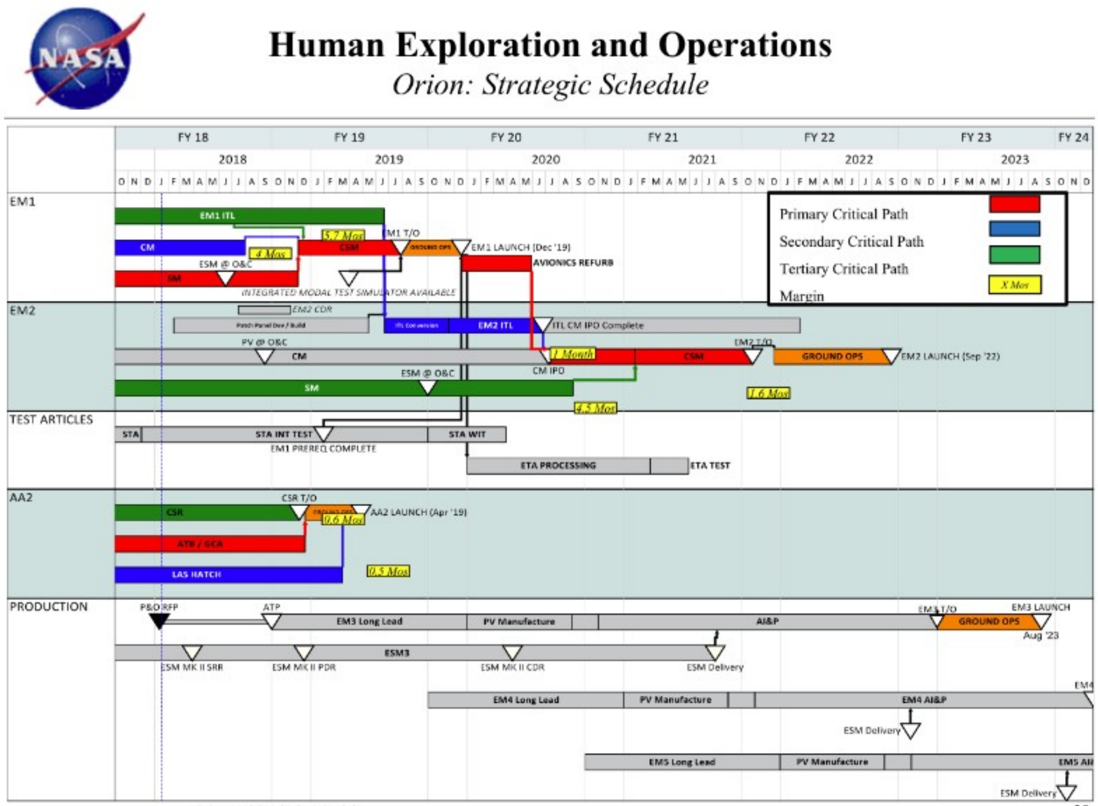

Figure 5.14. Integrated Master Schedule for Orion [34]

Two key characteristics of the IMS are critical paths and margins. The critical path can be seen in the Gantt chart outputs from SDDA by determining which sequence of developments determines the end time of the overall development. The critical path from SDDA may be different in timespan and/or development sequence than the one derived from PERT due to consideration of partial dependencies. SDDA can be used to evaluate the impact of delays on the critical path. Delaying a system development not initially on the critical path could alter the critical path to include that development, or the delay could be absorbed through partial dependencies before impacting the overall timeline. COD values for developmental dependencies can be chosen by managers to represent their risk tolerance. Higher COD values will prevent tasks from starting too early and going through extended development times, but will also decrease how much a delay can be absorbed by pushing back the start times of dependent systems.

Planned schedule margins can be included in the SDDA model by creating new development ”tasks” to represent these margins. The margins can be made partially or fully dependent on the completion of other development tasks to locate them properly in the schedule. To assess the impact of development delays on a particular margin, the start time of the margin ”task” can be compared between scenarios with different delays. The resulting change in margin start time indicates how much of the margin is used up by that delay. If the change in the start time is longer than the margin, then the delay will eliminate the margin and cause a delay in the development timeline. One disadvantage of SDDA is that the completion time of key milestones or developments cannot be set to then work backwards to form the schedule. All development completions times in SDDA are based on the dependencies and independent development times assigned from the start of development.

The NASA Scheduling Management Handbook [35] defines several task sequencing relationships that can be included in an IMS. The most common relationship is finish-to-start, meaning a successor task cannot begin until its predecessor is finished. This can be handled in SDDA by assigning SOD = 1 for that dependency, but again, an advantage of SDDA is its ability to model partial dependencies. Start-to-start and finish-to-finish relationships can be captured easily in SDDA using partial dependencies. Dependencies with low SOD values will allow both tasks to start near the same time while the existence of a dependency in SDDA will force a successor to finish after its predecessors. Lag time between the completion of one task and start of another can be handled by adding an intermediate ”task” between with the desired duration. Lead time is naturally handled by decreasing the SOD parameter to get a desired lead time duration.

The handbook also mentions constraints that can generally be divided into constraints on start times and constraints on end times. In SDDA, constraints can be placed only in the sense that start times can be artificially delayed to occur at a later time than what is originally output from the SDDA model. Development start and completion cannot be constrained to occur before a certain time in SDDA.

6. CONCLUSION

The objectives of this thesis were to answer the following research questions:

1. What enhancements to SODA's methodology and analysis tools are needed to consider the changing configurations of evolving SoS architectures and provide new insights and value to SoS designers and managers?

2. Do these enhancements provide new useful results for SoS decision makers when applied to a Artemis HLS case study?

6.1 SODA Enhancements

The SODA methodology was enhanced for application to evolving SoS architectures first through the definition of an approach to modeling reconfigurable SODA networks and through new data visualization methods. Failure Impact Range sequence plots take advantage of the large amount of deterministic impact information a FIR plot can portray and allows for easy comparison of disruptions across different evolutionary phases. Impacts of sets of system disruptions can be used to investigate how gradually disrupting more systems as the phases progress impacts the operability of capabilities of interest. FIR sequence plots represent a useful addition to deterministic data visualization in that they succinctly show the impact of node disruptions on different phase capability nodes, highlighting the effects of changing network configurations across the phases.

Stochastic Impact plots are an enhancement to stochastic analysis for both static and evolving architectures. As opposed to previous stochastic analysis methods used with SODA, the Stochastic Impact plots quantify the impact of disrupting a certain node while considering the stochastic nature of other nodes in the network. The plots provide a measure of the risk (both consequence and likelihood) of disruptions in a certain node, whereas analysis previously relied on the disconnected use of deterministic analysis to determine the consequence of disruptions and stochastic analysis to determine the likelihood of those disruptions occurring. Stochastic Impact plots can capture effects from configuration changes in node interdependencies as well as changes in node SE distributions across evolutionary phases. Combined, the FIR sequence plots and stochastic impact plots can be used by decision makers to evaluate the risk and impact of disruptions as an SoS evolves to help determine where and when resources should be allocated to ensure proper operations of key systems.

Integration of SODA and SDDA provides a framework for investigating the interaction of operational disruptions and developmental delays during the evolution of an SoS. These types of interactions have not been addressed by other studies and offer several new avenues of analysis for evolving SoS architectures. Two basic kinds of interactions were introduced and studied: impacts of developmental delays on operabilities of interest when systems undergo degradation, and impacts of operational failures on the development timeline when the failed systems are redesigned and rebuilt. These cases provide a foundation on which more complicated or recursive interactions can be studied. This integrated analysis can be used by decision makers to help plan the development of systems in a way that maximizes operational success. Tradeoffs between development of complicated monolithic systems and development of multiple less robust systems can be assessed based on the likelihood of catastrophic failures in each compared to the impact of those failures on the development schedule. Consideration of system degradation can be used to help determine the best timing for system introduction under schedule and resource constraints.

6.2 HLS Results

These enhancements were used to produce some interesting results for the two HLS concepts considered. FIR sequence plots highlighted that the node disruptions that have the most impact on phase completion capabilities (eg Transfer to LLO) are highly dependent on the phase being considered, while the disruptions most impactful to the crew survival capabilities are essentially constant throughout all phases. The AE pressure vessel structure, AE power, and ECLSS subsystems are among the most important for maintaining the crew survival capability in all phases.

The likelihood of mission success was computed for each concept by assigning capability operability thresholds to determine if each mission phase was completed or not. The operabilities of the phase capability nodes were computed stochastically based on a set of SE distributions assigned to nodes in each phase to approximate subsystem degradation across the phases. The mission success percent was near 90% for both the three-element and two-element concepts. While this suggests the two concepts have similar performance in the HLS mission, it was noted that some important distinguishing factors between the concepts may have been missed due to lack of detailed concept information or neglect of certain aspects of the HLS mission (like launch and docking operations), which, when included, might produce more distinct results.

Integration of SDDA and SODA models was used to address the broader evolution of the Artemis program through a series of HLS missions. Impacts of development delays on the mission completion likelihood for each of the three missions were analyzed. This revealed that the extent of operational impact due to system degradation is driven by the resulting time spans between system introduction and the beginning of each mission, rather than by the length of the delay itself. Development delays that delayed the introduction of all HLS elements by a similar amount had little effect on mission completion likelihood. Even small delays that shift the relative introduction times of HLS elements and cause some elements to loiter longer than normal can significantly reduce the probability of mission success.

The process used to develop and analyze schedules using SDDA was connected with the development of Integrated Master Schedules commonly used by program managers in the space industry. A significant difference between SDDA and PERT, which is often used in IMS construction, is SDDA's modeling of partial developmental dependencies that allow successors to start before predecessors are complete. Key aspects of an IMS, critical path and schedule margins, can be easily captured and analysed in SDDA. A disadvantage of SDDA is its inability to preset completion times of key tasks to then work backward to determine the development schedule.

6.3 Future Work

The main limitation of the analysis presented here is the lack of detailed information on the HLS concepts. Revision of the SODA models to better match specifics of the design of the three-element concept would provide more accurate results that could be useful for support concept choices in the future. Also, the amount of phases considered in the HLS mission can be expanded to reach more accurate and insightful results, but again this requires more detailed knowledge of a concept's design.

When performing integrated analysis with SODA and SDDA, a more complete SODA model can be used to investigate the impacts of delays on operational capabilities other than those directly related to the lunar landing missions. For example, the PPE and HALO modules can be decomposed into subsystems and incorporated into the SODA model so the impact of delays and degradation on broader capabilities such as maintaining a sustainable presence near the Moon can be analyzed.

REFERENCES

REFERENCES

[1] Brian Dunbar. Moon to mars overview, Jun 2018.

[2] Stephen Clark. Companies release new details on human rated lunar lander concepts, April 2020.

[3] Stephen Clark. Boeing proposes sls-launched lunar lander, November 2019.

[4] Mark W. Maier. Architecting principles for systems-of-systems. *Research in Engineering Design*, 28:53–69, 2017.

[5] Judith Dahmann et al. An implementer's view of engineering systems of systems. *IEEE Aerospace and Electronic Systems Magazine*, August 2011.

[6] Guariniello et al. Tool suite to support model based systems engineering-enabled system of system analysis. 2018.

[7] Cesare Guariniello and Daniel DeLaurentis. Supporting design via the system operational dependency analysis methodology. *Research in Engineering Design*, 28:53–69, 2017.

[8] Paul R. Garvey and Cesar Ariel Pinto. Introduction to functional dependency network analysis. 2010.

[9] Guariniello et al. Quantifying the impact of systems interdependencies in space systems architectures. In *70th International Astronautical Congress (IAC) 2019*, October 2019.

[10] Cesare Guariniello. *Supporting Space System Design Via System Dependency Analysis Methodology*. PhD thesis, Purdue University, 2016.

[11] Steven Zusack, Cesare Guariniello, and Daniel DeLaurentis. Operational dependency analysis of a human mars architecture based on the soda methodology. In *2018 IEEE Aerospace Conference*, 2018.

[12] Claus Ballegaard Nielsen, Peter Gorm Larsen, John Fitzgerald, Jim Woodcock, and Jan Peleska. Systems of systems engineering: Basic concepts, model-based techniques, and research directions. *ACM Comput. Surv.*, 48(2), September 2015.

[13] A. Mohsin, N. K. Janjua, S. M. S. Islam, and V. V. Graciano Neto. Modeling approaches for system-of-systems dynamic architecture: Overview, taxonomy and future prospects. In *2019 14th Annual Conference System of Systems Engineering (SoSE)*, pages 49–56, 2019.

[14] Royce A. Francis. Simulation-based analysis of reconfigurable system of system network topologies for resilience using bayesian networks. 2015.

[15] Mane et al. A markov perspective on development interdependencies in networks of systems. *ASME Journal of Mechanical Design*, October 2011.

[16] Agarwal et al. Flexible and intelligent learning architectures for sos (fila-sos): Architectural evolution in systems-of-systems. In *2015 Conference on Systems Engineering Research*, 2015.

[17] Gina Anderson, Matthew Rydin, and Jena Rowe. Nasa names companies to develop human landers for artemis moon missions, April 2020.

[18] NASA. *Attachment A14: Surface Mission Duration Sensitivity White Paper*, August 2019.

[19] NASA MSFC. *Human Landing System Requirements Document*, October 2019.

[20] NASA. Hls: Architecture analysis of alternatives, August 2019.

[21] Lockheed Martin. Space exploration: Crewed lunar lander, 2020.

[22] NASA. *Attachment A17: HLS White Paper for Manual Control Systems*, August 2019.

[23] David Akin. Spacecraft avionics, November 2019.

[24] Sally Cole. Orion spacecraft's avionics designed for reliability in deep space, March 2015.

[25] NASA. *Attachment A08: HLS Communications Information*, September 2019.

[26] L. Peterson. Environmental control and life support system: System engineering workshop, 2009.

[27] Justin McFatter, Karl Keiser, and Timothy Rupp. Nasa docking system block 1: Nasa's new direct electric docking system supporting iss and future human space exploration. *44th Aerospace Mechanisms Symposium*, May 2018.

[28] Blue Origin. Blue moon: National team hls, 2020.

[29] Spaceflight 101. Cygnus spacecraft, 2020.

[30] NASA MSFC. Environmental control and life support system (eclss), August 2017.

[31] Nathan Strout. Hypergiant is building a reprogrammable satellite constellation with the air force, July 2020.

[32] Gina Anderson, Isidro Reyna, and Rachel Kraft. Nasa awards northrop grumman artemis contract for gateway crew cabin, June 2020.

[33] Garrett Shea. Nasa program/project lifecycle, December 2019.

[34] Philip Sloss. Crewed orion spacecraft passes critical design review, December 2018.

[35] NASA. *NASA Schedule Management Handbook*, January 2010.

APPENDICES

A. SODA ADJACENCY MATRICES

	Element	Transfer Element								Descent Element											Ascent Element													Capability	
Element	Subsystems	TE Structure	TE Propulsion	TE GNC	TE Comms	TE Thermal Control	TE Power	TE Avionics	TE Docking Mechanism	DE Main Structure	DE Landing Structure	DE Propulsion	DE GNC	DE Power	DE Thermal Contro	DE Crew Surface Access	DE Comms	DE Avionics	DE TE Docking Mechanism	DE AE Docking Mechanism	AE Pressure Vessel Structure	AE Support Structure	AE Propulsion	AE GNC	AE Crew Systems	AE DE Docking Mechanism	AE External Docking Mechanism	AE Power	AE Thermal Contro	AE Avionics	AE Communications	AE ECLSS	AE Science and Research	Transfer to LLO	Crew Survival
Transfer Element	TE Structure		LHL		LHL	LHL	LMH	LMH	HHH																										
	TE Propulsion							LLM																										HHH	
	TE GNC		MHM		LML	LMH	LML	LLM							LMH														LMH					MHH	
	TE Comms			LMH				LLM																											
	TE Thermal Control		LMH	LMH			LML	LMH																											
	TE Power		LHL	HHL	HHL	MMM		HHL	LMH																										
	TE Avionics			LML	LML	LML	LML		LML																										
	TE Docking Mechanism	LML					LLM	LLM											HHH																
Descent Element	DE Main Structure										LML	LHL		LMH	LHL	MHM	LHL	LMH	HHH	HHH															
	DE Landing Structure																																		
	DE Propulsion													MHM				LLM																	
	DE GNC											MHM			LMH		LML	LLM																	
	DE Power											LHL	HHL		MMM	LMH	HHL	HHL	LMH	LMH															
	DE Thermal Control											HHH	LMH	LML				LMH																	
	DE Crew Surface Access																																		
	DE Comms												LMH					LLM																	
	DE Avionics												LML	LML	LML																				
	DE TE Docking Mechancism								HHH	LML				LLM				LLM																	
	DE AE Docking Mechanism									LML				LLM				LLM								HHH									
Ascent Element	AE Presure Vessel Structure																								MHM		HHH	LMH	LHL	LMH	LML	MHM	LLM		HHL
	AE Support Structure																				MHM		LHL			HHH		LMH	LHL	MHM					
	AE Propulsion																													LLM					
	AE GNC																						MHM						LML		LML				
	AE Crew Systems																													LLM					MMM
	AE DE Docking Mechanism																			HHH		LML						LLM		LLM					
	AE External Docking Mechanism																				LML														
	AE Power																						LHL	HHL	MHM	LMH	LMH		MMM	HHL	HHL	HHH			
	AE Thermal Control																						LMH	LMH	LML			LML		LMH		MHM			
	AE Avionics																							LML				LML	LML		LML	LML		MHM	
	AE Communications																							LMH						LLM				MMM	
	AE ECLSS																								MHM				LML	LLM					HHL
	AE Science and Research																																		
Capability	Transfer to LLO																																		
	Crew Survival																																		

Figure A.1. Phase 1 Adjacency Matrix for 3 Element Concept

	Element	Transfer Element								Descent Element											Ascent Element													Capability	
Element	Subsystems	TE Structure	TE Propulsion	TE GNC	TE Comms	TE Thermal Control	TE Power	TE Avionics	TE Docking Mechanism	DE Main Structure	DE Landing Structure	DE Propulsion	DE GNC	DE Power	DE Thermal Control	DE Crew Surface Access	DE Comms	DE Avionics	DE TE Docking Mechanism	DE AE Docking Mechanism	AE Pressure Vessel Structure	AE Support Structure	AE Propulsion	AE GNC	AE Crew Systems	AE DE Docking Mechanism	AE External Docking Mechanism	AE Power	AE Thermal Control	AE Avionics	AE Communications	AE ECLSS	AE Science and Research	Descend to Surface	Crew Survival
Transfer Element	TE Structure																																		
	TE Propulsion																																		
	TE GNC																																		
	TE Comms																																		
	TE Thermal Control																																		
	TE Power																																		
	TE Avionics																																		
	TE Docking Mechanism																																		
Descent Element	DE Main Structure										LML	LHL		LMH	LHL	MHM	LHL	LMH	HHH	HHH															
	DE Landing Structure																																	MHM	
	DE Propulsion													MHM				LLM																HHH	
	DE GNC											MHM			LMH		LML	LLM											LMH					MHH	
	DE Power											LHL	HHL		MMM	LMH	HHL	HHL	LMH	LMH															
	DE Thermal Control											HHH	LMH	LML				LMH																	
	DE Crew Surface Access																																		
	DE Comms												LMH					LLM																	
	DE Avionics												LML	LML	LML																				
	DE TE Docking Mechancism																																		
	DE AE Docking Mechanism									LML				LLM				LLM								HHH									
Ascent Element	AE Presure Vessel Structure																								MHM		HHH	LMH	LHL	LMH	LML	MHM	LLM		HHL
	AE Support Structure																				MHM		LHL			HHH		LMH	LHL	MHM					
	AE Propulsion																													LLM					
	AE GNC																						MHM						LML		LML				
	AE Crew Systems																													LLM					MMM
	AE DE Docking Mechanism																			HHH		LML						LLM		LLM					
	AE External Docking Mechanism																				LML														
	AE Power																						LHL	HHL	MHM	LMH	LMH		MMM	HHL	HHL	HHH			
	AE Thermal Control																						LMH	LMH	LML			LML		LMH		MHM			
	AE Avionics																							LML				LML	LML		LML	LML		MHM	
	AE Communications																							LMH						LLM				MMM	
	AE ECLSS																								MHM				LML	LLM					HHL
	AE Science and Research																																		
Capability	Descend to Surface																																		
	Crew Survival																																		

Figure A.2. Phase 2 Adjacency Matrix for 3 Element Concept

	Element	Transfer Element								Descent Element											Ascent Element													Capability	
Element	Subsystems	TE Structure	TE Propulsion	TE GNC	TE Comms	TE Thermal Control	TE Power	TE Avionics	TE Docking Mechanism	DE Main Structure	DE Landing Structure	DE Propulsion	DE GNC	DE Power	DE Thermal Control	DE Crew Surface Access	DE Comms	DE Avionics	DE TE Docking Mechanism	DE AE Docking Mechanism	AE Pressure Vessel Structure	AE Support Structure	AE Propulsion	AE GNC	AE Crew Systems	AE DE Docking Mechanism	AE External Docking Mechanism	AE Power	AE Thermal Control	AE Avionics	AE Communications	AE ECLSS	AE Science and Research	Surface Operations	Crew Survival
Transfer Element	TE Structure																																		
	TE Propulsion																																		
	TE GNC																																		
	TE Comms																																		
	TE Thermal Control																																		
	TE Power																																		
	TE Avionics																																		
	TE Docking Mechanism																																		
Descent Element	DE Main Structure										LML	LHL		LMH	LHL	MHM	LHL	LMH	HHH	HHH															
	DE Landing Structure															LHL																		LHL	
	DE Propulsion													MHM				LLM																	
	DE GNC											MHM			LMH		LML	LLM																	
	DE Power											LHL	HHL		MMM	LMH	HHL	HHL	LMH	LMH															
	DE Thermal Control											HHH	LMH	LML				LMH																	
	DE Crew Surface Access																																	HHH	
	DE Comms												LMH					LLM																	
	DE Avionics												LML	LML	LML																				
	DE TE Docking Mechancism																																		
	DE AE Docking Mechanism									LML				LLM				LLM								HHH									
Ascent Element	AE Presure Vessel Structure																								MHM		HHH	LMH	LHL	LMH	LML	MHM	LLM	LLM	HHL
	AE Support Structure																				MHM		LHL			HHH		LMH	LHL	MHM					
	AE Propulsion																													LLM					
	AE GNC																						MHM						LML		LML				
	AE Crew Systems																													LLM				LML	MMM
	AE DE Docking Mechanism																			HHH		LML						LLM		LLM					
	AE External Docking Mechanism																				LML														
	AE Power																						LHL	HHL	MHM	LMH	LMH		MMM	HHL	HHL	HHH			
	AE Thermal Control																						LMH	LMH	LML			LML		LMH		MHM			
	AE Avionics																							LML				LML	LML		LML	LML		MHM	
	AE Communications																							LMH						LLM				MHH	
	AE ECLSS																								MHM				LML	LLM					HHL
	AE Science and Research																																	LML	
Capability	Surface Operations																																		
	Crew Survival																																		

Figure A.3. Phase 3 Adjacency Matrix for 3 Element Concept

	Element	Transfer Element								Descent Element											Ascent Element													Capability	
Element	Subsystems	TE Structure	TE Propulsion	TE GNC	TE Comms	TE Thermal Control	TE Power	TE Avionics	TE Docking Mechanism	DE Main Structure	DE Landing Structure	DE Propulsion	DE GNC	DE Power	DE Thermal Control	DE Crew Surface Access	DE Comms	DE Avionics	DE TE Docking Mechanism	DE AE Docking Mechanism	AE Pressure Vessel Structure	AE Support Structure	AE Propulsion	AE GNC	AE Crew Systems	AE DE Docking Mechanism	AE External Docking Mechanism	AE Power	AE Thermal Control	AE Avionics	AE Communications	AE ECLSS	AE Science and Research	Ascend to NRHO	Crew Survival
Transfer Element	TE Structure																																		
	TE Propulsion																																		
	TE GNC																																		
	TE Comms																																		
	TE Thermal Control																																		
	TE Power																																		
	TE Avionics																																		
	TE Docking Mechanism																																		
Descent Element	DE Main Structure																																		
	DE Landing Structure																																		
	DE Propulsion																																		
	DE GNC																																		
	DE Power																																		
	DE Thermal Control																																		
	DE Crew Surface Access																																		
	DE Comms																																		
	DE Avionics																																		
	DE TE Docking Mechancism																																		
	DE AE Docking Mechanism																																		
Ascent Element	AE Presure Vessel Structure																								MHM		HHH	LMH	LHL	LMH	LML	MHM	LLM		HHL
	AE Support Structure																				MHM		LHL			HHH		LMH	LHL	MHM					
	AE Propulsion																													LLM				HHH	
	AE GNC																						MHM						LML	LLM	LML			MHH	
	AE Crew Systems																													LLM					MMM
	AE DE Docking Mechanism																																		
	AE External Docking Mechanism																				LML														
	AE Power																						LHL	HHL	MHM	LMH	LMH		MMM	HHL	HHL	HHH			
	AE Thermal Control																						LMH	LMH	LML			LML		LMH		MHM			
	AE Avionics																							LML				LML	LML		LML	LML		MHM	
	AE Communications																							LMH						LLM				MMM	
	AE ECLSS																								MHM				LML	LLM					HHL
	AE Science and Research																																		
Capability	Ascend to NRHO																																		
	Crew Survival																																		

Figure A.4. Phase 4 Adjacency Matrix for 3 Element Concept

	Element	Descent Element											Ascent Element													Capability	
Element	**Subsystems**	DE Main Structure	DE Landing Structure	DE Propulsion	DE GNC	DE Power	DE Thermal Control	DE Crew Surface Access	DE Airlock	DE Comms	DE Avionics	DE AE Docking Mechanism	AE Pressure Vessel Structure	AE Propulsion	AE GNC	AE Crew Systems	AE DE Docking Mechanism	AE External Docking Mechanism	AE Power	AE Thermal Control	AE Avionics	AE Communications	AE ECLSS	AE Science and Research	Transfer to LLO	Crew Survival
Descent Element	DE Main Structure		LML	LHL		LMH	LHL	MHM	MHM	LHL	LMH	HHH														
	DE Landing Structure							LHL																		
	DE Propulsion										LLM														MHH	
	DE GNC			MHM			LMH			LML	LLM									LMH					MHH	
	DE Power			LHL	HHL		MMM	LMH	MHM	HHL	HHL	LMH														
	DE Thermal Control			HHH	LMH	LML					LMH															
	DE Crew Surface Access																									
	DE Airlock																									
	DE Comms				LMH						LLM															
	DE Avionics			LML	LML	LML	LML																			
	DE AE Docking Mechanism	LML				LLM			HHH		LLM						HHH									
Ascent Element	AE Presure Vessel Structure													LHL		MHM	HHH	HHH	LMH	LHL	MMH	LML	MHM	LLM		HHL
	AE Propulsion																				LLM					
	AE GNC													MHM						LML		LML				
	AE Crew Systems																				LLM					MMM
	AE DE Docking Mechanism											HHH	LML						LLM		LLM		LML			
	AE External Docking Mechanism												LML													
	AE Power													LHL	HHL	MHM	LMH	LMH		MMM	HHL	HHL	HHH			
	AE Thermal Control													LMH	LMH	LML			LML		LMH		MHM			
	AE Avionics														LML				LML	LML		LML	LML		MHM	
	AE Communications														LMH						LLM				MMM	
	AE ECLSS								LHM							MHM				LML	LLM					HHL
	AE Science and Research																									
Capability	Transfer to LLO																									
	Crew Survival																									

Figure A.5. Phase 1 Adjacency Matrix for 2 Element Concept

Element	Element	Descent Element											Ascent Element												Capability	
Element	Subsystems	DE Main Structure	DE Landing Structure	DE Propulsion	DE GNC	DE Power	DE Thermal Control	DE Crew Surface Access	DE Airlock	DE Comms	DE Avionics	DE AE Docking Mechanism	AE Pressure Vessel Structure	AE Propulsion	AE GNC	AE Crew Systems	AE DE Docking Mechanism	AE External Docking Mechanism	AE Power	AE Thermal Control	AE Avionics	AE Communications	AE ECLSS	AE Science and Research	Descend to Surface	Crew Survival
Descent Element	DE Main Structure		LML	LHL		LMH	LHL	MHM	MHM	LHL	LMH	HHH														
	DE Landing Structure							LHL																	MHM	
	DE Propulsion					MHM					LLM														MHH	
	DE GNC			MHM			LMH			LML	LLM									LMH					MHH	
	DE Power			LHL	HHL		MMM	LMH	MHM	HHL	HHL	LMH														
	DE Thermal Control			HHH	LMH	LML					LMH															
	DE Crew Surface Access																									
	DE Airlock																									
	DE Comms				LMH						LLM															
	DE Avionics			LML	LML	LML	LML																			
	DE AE Docking Mechanism	LML				LLM			HHH		LLM						HHH									
Ascent Element	AE Presure Vessel Structure													LHL		MHM	HHH	HHH	LMH	LHL	MMH	LML	MHM	LLM		HHL
	AE Propulsion																				LLM					
	AE GNC													MHM						LML		LML				
	AE Crew Systems																				LLM					MMM
	AE DE Docking Mechanism											HHH	LML						LLM		LLM		LML			
	AE External Docking Mechanism												LML													
	AE Power													LHL	HHL	MHM	LMH	LMH		MMM	HHL	HHL	HHH			
	AE Thermal Control													LMH	LMH	LML			LML		LMH		MHM			
	AE Avionics														LML				LML	LML		LML	LML		MHM	
	AE Communications														LMH						LLM				MMM	
	AE ECLSS								LHM							MHM				LML	LLM					HHL
	AE Science and Research																									
Capability	Descend to Surface																									
	Crew Survival																									

Figure A.6. Phase 2 Adjacency Matrix for 2 Element Concept

Element	Element Subsystems	Descent Element											Ascent Element												Capability	
Element	Subsystems	DE Main Structure	DE Landing Structure	DE Propulsion	DE GNC	DE Power	DE Thermal Control	DE Crew Surface Access	DE Airlock	DE Comms	DE Avionics	DE AE Docking Mechanism	AE Pressure Vessel Structure	AE Propulsion	AE GNC	AE Crew Systems	AE DE Docking Mechanism	AE External Docking Mechanism	AE Power	AE Thermal Control	AE Avionics	AE Communications	AE ECLSS	AE Science and Research	Surface Operations	Crew Survival
Descent Element	DE Main Structure		LML	LHL		LMH	LHL	MHM	MHM	LHL	LMH	HHH														
	DE Landing Structure							LHL																	LHL	
	DE Propulsion					MHM					LLM															
	DE GNC			MHM			LMH			LML	LLM															
	DE Power			LHL	HHL		MMM	LMH	MHM	HHL	HHL	LMH														
	DE Thermal Control			HHH	LMH	LML					LMH															
	DE Crew Surface Access																								MMM	
	DE Airlock																								HHH	
	DE Comms				LMH						LLM															
	DE Avionics			LML	LML	LML	LML																			
	DE AE Docking Mechanism	LML				LLM			HHH		LLM						HHH									
Ascent Element	AE Presure Vessel Structure													LHL		MHM	HHH	HHH	LMH	LHL	MMH	LML	MHM	LLM	LLM	HHL
	AE Propulsion																				LLM					
	AE GNC													MHM						LML		LML				
	AE Crew Systems																				LLM				LML	MMM
	AE DE Docking Mechanism								HHH			HHH	LML						LLM		LLM		LML			
	AE External Docking Mechanism												LML													
	AE Power													LHL	HHL	MHM	LMH	LMH		MMM	HHL	HHL	HHH			
	AE Thermal Control													LMH	LMH	LML			LML		LMH		MHM			
	AE Avionics														LML				LML	LML		LML	LML		MHM	
	AE Communications														LMH						LLM				MHH	
	AE ECLSS								LHM							MHM				LML	LLM				MHM	HHL
	AE Science and Research																								LML	
Capability	Surface Operations																									
	Crew Survival																									

Figure A.7. Phase 3 Adjacency Matrix for 2 Element Concept

Element	Element	Descent Element											Ascent Element												Capability	
Element	Subsystems	DE Main Structure	DE Landing Structure	DE Propulsion	DE GNC	DE Power	DE Thermal Control	DE Crew Surface Access	DE Airlock	DE Comms	DE Avionics	DE AE Docking Mechanism	AE Pressure Vessel Structure	AE Propulsion	AE GNC	AE Crew Systems	AE DE Docking Mechanism	AE External Docking Mechanism	AE Power	AE Thermal Control	AE Avionics	AE Communications	AE ECLSS	AE Science and Research	Ascend to NRHO	Crew Survival
Descent Element	DE Main Structure																									
	DE Landing Structure																									
	DE Propulsion																									
	DE GNC																									
	DE Power																									
	DE Thermal Control																									
	DE Crew Surface Access																									
	DE Airlock																									
	DE Comms																									
	DE Avionics																									
	DE AE Docking Mechanism																									
Ascent Element	AE Presure Vessel Structure													LHL		MHM	HHH	HHH	LMH	LHL	MMH	LML	MHM	LLM		HHL
	AE Propulsion																				LLM				MHH	
	AE GNC													MHM						LML	LLM	LML			MHH	
	AE Crew Systems																				LLM					MMM
	AE DE Docking Mechanism												LML								LLM		MHM			
	AE External Docking Mechanism												LML													
	AE Power													LHL	HHL	MHM	LMH	LMH		MMM	HHL	HHL	HHH			
	AE Thermal Control													LMH	LMH	LML			LML		LMH		MHM			
	AE Avionics														LML				LML	LML		LML	LML		MHM	
	AE Communications														LMH						LLM				MMM	
	AE ECLSS															MHM				LML	LLM					HHL
	AE Science and Research																									
Capability	Ascend to NRHO																									
	Crew Survival																									

Figure A.8. Phase 4 Adjacency Matrix for 2 Element Concept

B. SDDA ADJACENCY MATRIX

	Dev Orion	Dev AE Disposable	Dev DE	Dev TE	Dev AE Reusable	PPE	HALO	Orion 1	AE Disposable	DE 1	TE 1	Orion 2	AE Reusable	DE 2	TE 2	Orion 3	DE 3	TE 3	Launch AE Disposable	Launch DE 1	Launch TE 1	HLS 1 Checkout and Assemb	Crew 1 to NRHO	Landing 1	Launch PPE + HALO	Launch AE Reusable	Launch DE 2	Launch TE 2	HLS 2 Checkout and assemb	Crew 2 to NRHO	Landing 2	Launch DE 3	Launch TE 3	HLS 3 Checkout and assemb	Crew 3 to NRHO	Landing 3
Dev PPE	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Dev HALO	0	0.1	0	0	0.2	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Dev Orion	0	0.2	0	0	0	0	0.3	0.9	0	0	0	0.9	0	0	0	0.85	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Dev AE Disposable	0	0	0.2	0.2	0.5	0	0	0	0.9	0.3	0.3	0	0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Dev DE	0	0	0	0.2	0	0	0	0	0.3	0.9	0.3	0	0.1	0.9	0.3	0	0.8	0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Dev TE	0	0	0	0	0	0	0	0	0.3	0.3	0.9	0	0.1	0.1	0.9	0	0.1	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Dev AE Reusable	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1	0.1	0	0.1	0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PPE	0	0	0	0	0	0	0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0
HALO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0
Orion 1	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
AE Disposable	0	0	0	0	0	0	0	0	0	0.2	0.2	0	0.8	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DE 1	0	0	0	0	0	0	0	0	0	0	0.2	0	0	0.6	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TE 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Orion 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
AE Reusable	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1	0.1	0	0.1	0.1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0
DE 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1	0	0.6	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
TE 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0
Orion 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
DE 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
TE 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	1	1	0	0
Launch AE Disposable	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Launch DE 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Launch TE 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HLS 1 Checkout and Assembly	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0

Figure B.1. First half of SDDA adjacency matrix

	Dev Orion	Dev AE Disposable	Dev DE	Dev TE	Dev AE Reusable	PPE	HALO	Orion 1	AE Disposable	DE 1	TE 1	Orion 2	AE Reusable	DE 2	TE 2	Orion 3	DE 3	TE 3	Launch AE Disposable	Launch DE 1	Launch TE 1	HLS 1 Checkout and Assemb	Crew 1 to NRHO	Landing 1	Launch PPE + HALO	Launch AE Reusable	Launch DE 2	Launch TE 2	HLS 2 Checkout and assemb	Crew 2 to NRHO	Landing 2	Launch DE 3	Launch TE 3	HLS 3 Checkout and assemb	Crew 3 to NRHO	Landing 3
Crew 1 to NRHO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Landing 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
Launch PPE + HALO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0
Launch AE Reusable	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0
Launch DE 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0
Launch TE 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0
HLS 2 Checkout and assembly	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
Crew 2 to NRHO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Landing 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
Launch DE 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0
Launch TE 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0
HLS 3 Checkout and assembly	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
Crew 3 to NRHO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Landing 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure B.2. Second half of SDDA adjacency matrix