INTEGRATING DEPENDENCIES INTO THE TECHNOLOGY PORTFOLIO: A FEED-FORWARD CASE STUDY FOR NEAR-EARTH ASTEROIDS

A Thesis
Presented to
The Academic Faculty

by

Christianna Taylor

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Aerospace Engineering

Georgia Institute of Technology
December 2011

INTEGRATING DEPENDENCIES INTO THE TECHNOLOGY PORTFOLIO: A FEED-FORWARD CASE STUDY FOR NEAR-EARTH ASTEROIDS

Approved by:

Dr. Dimitri Mavris, Committee Chair
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Daniel Schrage
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Simon Briceno
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Christopher Raczynski
GE Energy

Dr. William Lincoln
Jet Propulsion Laboratory

Date Approved: 11 November 2011

To my Mother who has always taught me to be the best in anything that I pursue. I love you.

When I was little and asked what I wanted to be I would reply "A doctor, a lawyer, a brain surgeon, an engineer, sweet, intelligent, little girl". So I'm already an engineer, I'm pretty sweet, and this little girl grew into a grown woman: Finally I'm a doctor. You can't predict everything, but I did pretty well. Thanks Mom!

ACKNOWLEDGEMENTS

I wish to thank my Mother first and foremost for every single sacrifice, encouragement, and demand she put on me. I use to say she was so mean that she could make the sun take a nap (this stems from the fact that I did not want to take a nap and then a cloud went in front of the sun convincing me that she made it take a nap as well); however, I must thank my aggressive ambitious and adventure seeking personality to what I could only see as my Mother being mean when I was little and have grown to understand as an adult. She taught me to find a way to survive no matter what the situation and to never give up. She could not read C's so they were never allowed on my report cards. I was to stand up for myself no matter what the situation as long as I was on the right and be humble when in the wrong. I was to get out of the house immediately and see the world and seek adventure because that is what you should do with life. I have inherited this personality and direction from her who is solely responsible for me getting this far in life.

Next I would like to thank Dr. Mavris for guiding me through these past six and half years. I almost didn't get into graduate school, emailed ASDL, interviewed and the rest is history. He has always provided me with encouragement to pursue opportunities, freedom to accomplish and fail in my endeavors, and guidance throughout this process. Thank you for bringing me into ASDL and allowing me to grow under your guidance.

Dr. Weisbin at the Jet Propulsion Laboratory is next on my list to thank. Thank you for taking a chance on me when not knowing anything about me besides my ability to pursue the NASA GSRP and work in your lab. The START team has been extremely instrumental throughout this entire process. Their comments, meetings, and time are invaluable and I will forever be in debt to their dedication and kindness. Specifically Dr. William Lincoln, Dr. Virgil Adumitroaie, and Hook Hua. Their time commitments to me during the 2009 summer were amazing and most of this could have not have been done without them!

Ms. Wanda Pierson in the Georgia Space Grant Office. Thank you for being such a wonderful mentor to me. I appreciate the lunches, the advice, your time, and your friendship. Thank you for introducing me to NASA, mentors, business partners, and everyone in between. Not everyone is as blessed as I am to have such a wonderful mentor that is knowledgeable in every field and facet of life.

To the Aerodynamic Duo: Michelle Walker - Thank you for not running away when I stalked you to be my first friend in ASDL my first day looking for my research engineer. We went through a lot together! I remember first year bootcamp and you kept telling me it was going to be OK. I remember wanting to go shopping or out without a car and I kept telling you how much you wanted to go so we could go together. Thanks for sticking by me through everything. I love you with all my heart. We're the Aerodynamic Duo! Who could top that?

To the Black Graduate Student Association: We've been through so much! From cookouts, to paintball, to laser tag, to mac'n'cheese offs (here's looking at you K. Reed), to just giving me a home away from home when I was trying to transition from Boston. You all have been there for me. BGSA you have a great responsibility to be everything and anything graduate students need, and lord knows you've done that. I am proud to have been a part of you and I promise to do you even better.

To the Oh girl lunch committee: Shara McClendon, Jessica Abraham, Mela Johnson, Adaora Okwo, Ashley Johnson, - you are my sisters. You are who I look to when I am lost. You are who I will find when you're feeling lost. Tech did that to us and I am forever grateful. Remember when you first get your hair done and it's OH GIRL and a few days later it's OH Girl! Remember when we first came to Tech and Ashley

did the spy thing checking that we were in the PhD club, Adaora had bagels, Jessica had the Waffle House, and Mela was stalking Ludacris, a few years later, and we're all graduating and moving on with our lives. Oh girls, we did it! Love you always! Yes, we need an annual cabin trip.

To my big sister who never wanted me, but got me anyway: Dr. Tammy "Bones" "Diva" McCoy. I love you girl. We ran a half marathon together! I mean if that doesn't seal the deal than NOTHING will. Thank you for always having a space for me in your heart, on your couch, and in your family. Lord knows I've basically moved in as a roommate for part of the year! Your laughter, joy, friendship and love make you my sister. A girl couldn't have a better reluctant sis to take care of her. I love how you carry yourself, look to the future, and have such a wonderful outlook on life. If I should be so lucky to be half the woman that you are, I will have accomplished much in my life.

To my big brothers: Dimitri Hughes, Jabulani Barber, Ousmane Diallo, Davis Balaba, and Keith Reed. What can I really say that we haven't said in jest already? Dimitri thanks for always challenging me on my viewpoints and making me really show how I feel about things. Jabu, you are one of the most eclectic, unique and talented individuals that I know. Never change that sense of exploration that I love about you. To Ousmane and Davis, if ever I had big brothers it would be you two. Davis, I remember sneak attacks my first year when I was trying to get through bootcamp and how kind you were to me when I didn't think I could make it. Ousmane, I remember your drive and spirit. A little sis couldn't feel more protected in your care. Keith Reed, you are a wild spirit with lots of energy. Thanks for always reminding me of the lighter side of things.

To my Best Friend Forever, Sharon Asonganyi. There is not much more to be said than a simple thank you for being there for me. You were one of my first sisters since undergrad and I am grateful that our bond has only grown since we met each

other so long ago in English class freshmen year.

To the National Society of Black Engineers and the Space Special Interest Group. Both of these organizations have inspired me to keep moving on in both my professional career and personal development. I am grateful to all of the opportunities, mentors, scholarships, and examples that you have given both myself and every other African Americans in the STEM disciplines. We all need you and I hope to see NSBE and the Space SIG only grow year after year.

Last but not least, to the love of my life and my future husband Andrew Stamps. I love and cherish you with all my heart. This work is only the ending to a great chapter in my life and I look forward to starting the next chapter with you.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

The Technology Portfolio is an important system tool that enables the user to quantify their qualitative decisions of what technology should be adopted. The use of Technology Portfolios allows a user the power to determine the effectiveness of a technology, the schedule and cost effort and ultimately the development effort associated with integrating that technology. Through this method, large scale projects may evolve into sustainable efforts.

This optimization of the technology portfolio creates large complex design spaces; however, many processes operate on the assumption that their technologies have no dependency on other technologies, because dependencies are not well defined. There are so many types which include functional, systemic, and scheduling variations to name a few of the possibilities faced by the optimization environments. This independence assumption simplifies the process, but also suggests that these environments are missing out on decision power and fidelity. Therefore, this thesis' main research objective is to:

Explain the variations in Portfolio recommendations as a function of adding dependencies.

Each element of the technology portfolio represents the value, cost, schedule, risk, and uncertainty associated with the technology, mission, or capability. In order to converge upon a solution, the process assumes independent technologies are being adopted. The problem with this is that technologies have detriments as well as benefits. They do not contribute to a project individually, but act in accordance with or against each other and existing assets. Different tools have been developed to deal with this type of information at various stages of the process; however, it has not been

widely studied within the technology portfolio domain. Specifically four questions are brought forward to address dependency integration into technology portfolios.

1. What are the dependencies associated with the technology portfolio investment?
2. What is the effect of adding dependencies on the technology portfolio process?
3. What is the effect of reducing the investment time frame?
4. Which has a larger impact on the Portfolio Selection: Input values or Dependencies?

Addressing these research questions requires a technology portfolio framework in which to implement the changes and a sample problem to see their effect on real data. This thesis was done in conjunction with the NASA Jet Propulsion Laboratory in the STrategic Assessment of Risk and Technology (START) lab. START is a technology framework that determines if a technology should be selected, when it should be funded and how much should be funded. The upcoming NASA Near-Earth Asteroid Campaign was studied as a case study. This campaign is the new plan to send humans to an asteroid by 2025 announced by President Obama in April 2010. The campaign involves multiple missions, capabilities, and technologies that must be demonstrated in order to enable deep-space human exploration. Therefore, this thesis capitalized on that intention to show how adopting technology in an earlier mission can act as a feed-forward method to demonstrate technology for future missions by modeling this aspect with dependency integration.

Utilizing this case study, the thesis will show the baseline technology portfolio, integrate dependencies into the process, compare its findings to the baseline case, and ultimately show how adding higher fidelity into the process changes the users decisions. The thesis will conclude with the final methodology, a discussion of new applications for technology portfolios, and suggest future areas of research to further this fidelity to decision makers.

CHAPTER I

INTRODUCTION

Technology Portfolios are essential to the progression of large complex systems. The evolution of cutting edge technology produces cleaner, safer and cheaper possibilities. In an effort to harness this power of new technology, technology portfolios are used to predict the value of integrating the technology into the project. Technologies are evaluated individually during the evaluation of the development effort. The problem is that the technologies have detriments as well as benefits. They do not contribute to a project individually, but act in accordance with or against each other and existing assets. Different tools have been developed to deal with the independence of development efforts at various stages of the process; however, it has not been widely studied within the technology portfolio domain. In an effort to harness this fidelity of dependency between development efforts this thesis investigates the effects of dependencies on the technology portfolio selection.

In order to effectively explain technology portfolios, a few definitions must be presented first. The guidelines on System Engineering for NASA comes from the NASA System engineering handbook. Originally published in 1995 it is now in its 12th edition released in 2007. The following common terms are taken from both the NASA system engineering handbook and the system engineering handbook by Andrew P. Sage.

System - (1) The combination of elements that function together to produce the capability to meet a need. The elements include all hardware, software, equipment, facilities, personnel, processes, and procedures needed for this purpose. (2) The end product (which performs operational functions) and enabling products (which provide life-cycle support services to the operational end products) that make up a system. (NASA System Engineering Handbook)

System Engineering - an interdisciplinary field of engineering that focuses on how complex engineering projects should be designed and managed. Issues such as logistics, the coordination of different teams, and automatic control of machinery become more difficult when dealing with large, complex projects. Systems engineering deals with work-processes and tools to handle such projects, and it overlaps with both technical and human-centered disciplines such as control engineering and project management.[62]

Technology - the organization application, and delivery of scientific and other forms of knowledge for the betterment of a client group. It inherently involves a purposeful human extension of one or more natural processes. [62]

Effectiveness - the power to produce a desired outcome change within some defined metric by the system efficiently and decisively.[62]

Technology Portfolio - a technique in system engineering that looks for the combinations of technologies selected to assist a system in the effectiveness of some objective of the system.[62]

1.1 Technology Portfolio Selection

Technology portfolio selection deals with many aspects of a technology program. Current programs deal with this process in different degrees. Dr. Yu expressed the prospect of Technology portfolio processes best.

"After the decision maker has articulated and assessed the values, identified relevant technology and resource allocation alternatives, and determined the relationships between these alternatives and values in a technology portfolio planning process, he or she will need to find the best resource allocation among the technologies that maximizes the total value of the portfolio. " - Dr. Yu [86]

Allocating resources for a development effort of technologies is essential to the success of the project. Therefore, giving fidelity and power to the decision maker to make an informed decision is essential to the value of that development effort. Adding fidelity to this process requires understanding the figures of merit of the technology portfolio effort. [7, 46, 27]

1.1.1 Technology Portfolio Elements

The development effort for the technology portfolio has several figure of merit associated portfolio elements. As Dr. Yu stated, there is some value assigned to the element and resource allocation. That resource allocation involves the cost and scheduled development of the technology. These aspects must be evaluated and related to one another in a technology portfolio optimization scheme. These three figures of merit, value, cost and schedule, are the first elemental values that the decision maker optimizes. Optimizing this information gives the user the power to know if they should fund a technology, when in the process should they fund it, and finally how much to fund. This information is crucial to the technology portfolio. The user has the option of evaluating and optimizing the risk and uncertainty associated with the development efforts. These five elements of value, cost, schedule, risk and uncertainty are universal for a development effort and add to the fidelity of the decision maker's objective. This is seen in Figure 1.

1.1.2 Technology Portfolio Fidelity

Taking the elemental values of the development effort into account separately gives an optimized portfolio that has some level of fidelity to it. Maximizing and minimizing each element is essential to increasing the fidelity of the portfolio. This thesis seeks to add one more element into this aspect through the introduction of dependencies. The next two sections will define dependencies as well as give examples of their uses and applications.

1.1.2.1 Dependencies

We define **dependencies** as the relationship between two elements wherein one relies on the other in order to function completely or at all. Sample development elements are shown in Figure 2 with the dependency element connected. Dependencies add fidelity into the process by changing the decision makers choice of portfolio. They take into account another level of information that exists between development efforts. The figures of merit described before are individual and essential to the specific elements of the development effort. In contrast, dependencies are essential links that happen between elements. The optimization process is not only seeing the element by itself anymore; it sees the elements as a package that has a different combined score of the five figures of merit.

1.1.2.2 Dependency Applications

There are multiple dependency applications. An example is a competing technology or protocol that the user must decide upon. The system engineer has the option to

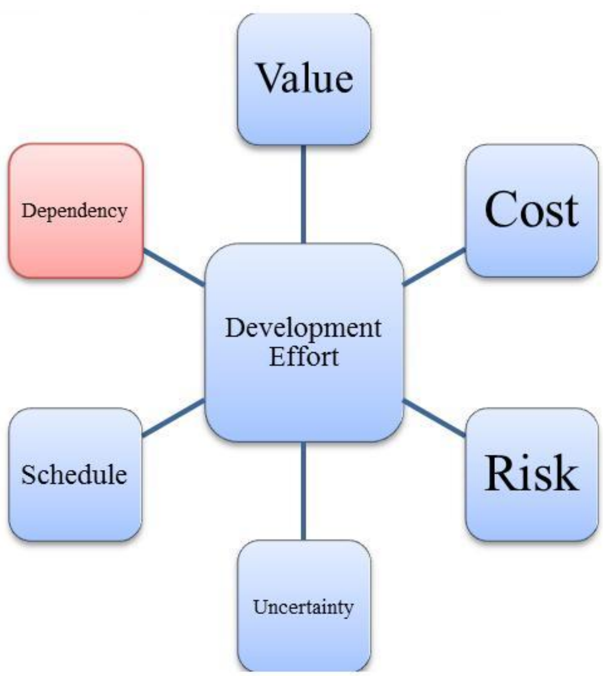


Figure 1: Development effort essential elements

run two scenarios and choose between which portfolio has the largest objective value. Adding a dependency between the two technologies allows the system engineer to run the optimization process only once. The optimizer will decide between the competing technologies dynamically instead of the user statically making the decision after the fact. This process also captures any effects on other development efforts that the static selection may have missed.

Another scenario is for two technologies that are complementary to each other, but can be funded separately. In this case, combining the technologies gives a benefit to the objective value that the optimizer understands. Had this dependency not been included, the optimizer may or may not have chosen the two technologies together. Even if it had chosen them together, it would not have the modified figures of merit benefit that come from including the two together.

1.1.3 Current Technology Portfolio Tools

There are many applications that come from the addition of dependencies into the process. However, this fidelity must be integrated into current technology portfolio

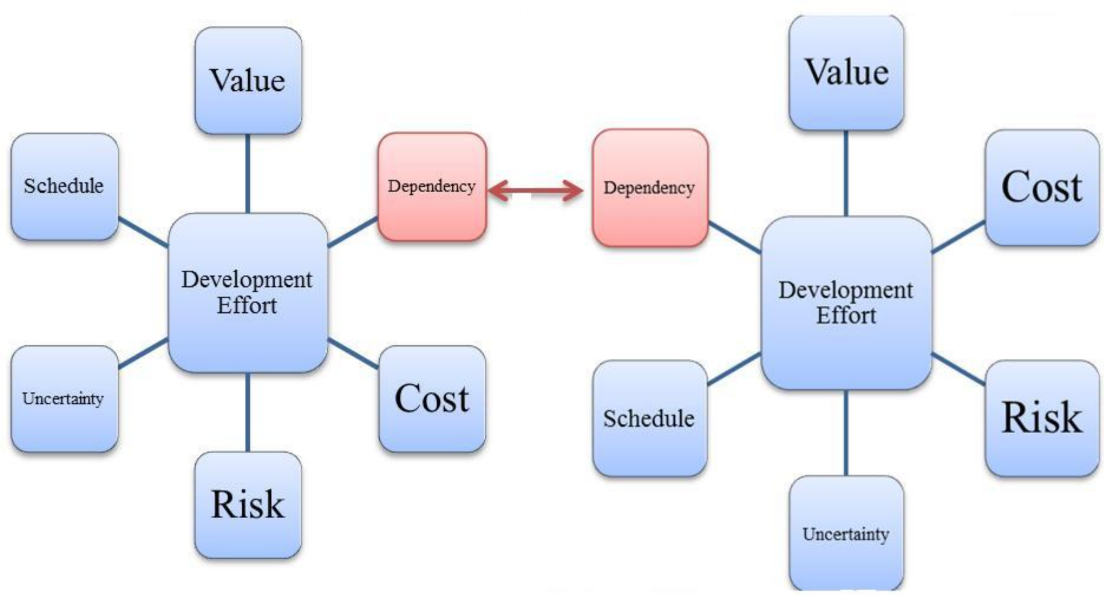


Figure 2: Development effort essential elements

tools. This section will give an overview of a selection of current technology portfolio evaluation tools.

1.1.3.1 Quality Function Deployment (QFD)

Quality Function Deployment, QFD's, is commonly used to prioritize information regarding customers and engineers. A sample QFD is shown in Figure 3. A House of Quality (HOQ) is placed as the roof of the QFD and used to map the technology needs to the requirements and vice versa. This method is quite effective to give the user weights that emphasize the technology's or requirement's ability to be satisfied by the given information. They are highly effective as a decision making tool and have the ability to take some of the dependency information into account in the roof of the HOQ; however, it only takes into consideration the quality of the dependency, not the degree to which the dependency affects the prioritization. It also does not explain the type of dependency or specific relationships between elements created in the HOQ. This may determine if they should consider investing in a technology effort, but it does not provide an actual technology portfolio description. [7, 46, 27]

1.1.3.2 IRMA

IRMA is an interactive mapping of alternatives which encompasses a correlation matrix to handle comprehensibility between dependencies; however, it is static. The IRMA takes into account which technologies will exclude other technologies. This reduces the design space. While IRMA provides a dynamic look at changing the design space, it does not necessarily produce the actual optimized solution. It simply reduces the design space. The IRMA can take into account the example dependency application of inclusive and exclusive technologies; however, it would benefit from the use of partial dependencies.

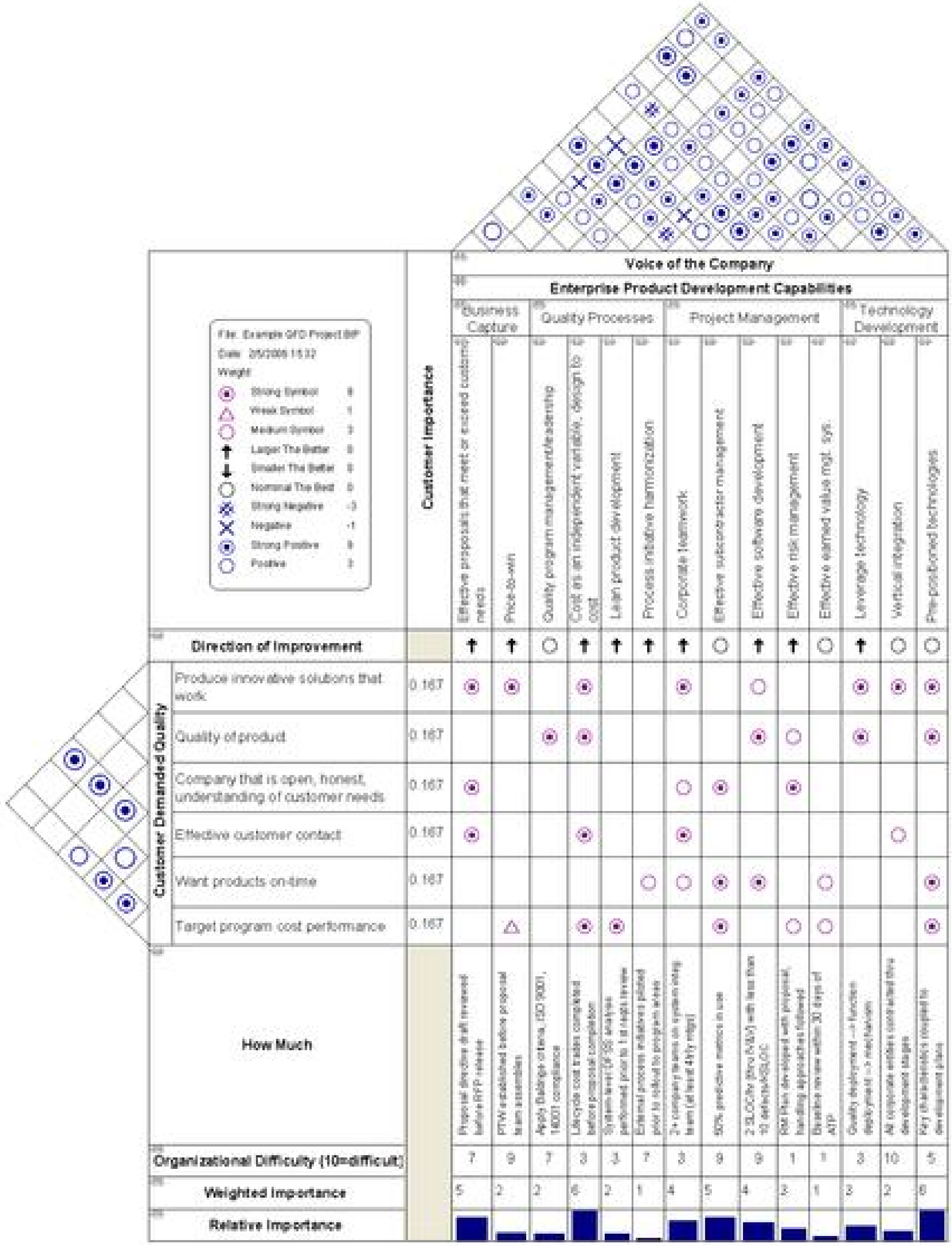


Figure 3: Example QFD

1.1.3.3 Analysis of Variance (ANOVA)

Analysis of Variance is a statistical model to take into account the variance associated with different attributes of a model. It requires that all the elements are independent.

While it does deal with grouping information together and defines the relationship between them, it does not allow for partial dependence. It may define mutual inclusion or exclusion on the response variables, but does not deal with the harder cases which fall outside the simple cases of pure inclusion or exclusion. It is an excellent statistical tool to search through the design space and identify which variables should be included or excluded in the overall process. However, this requires multiple design points in order to perform the statistical calculations to show the importance of different variables. In this case, ANOVA optimizes already developed technologies and then places emphasis on which technologies should be integrated into a program. Althought it does not list the technology portfolio, it does place statistical significance on the elements it evaluates. [11, 13, 45]

1.1.3.4 Technology Identification, Evaluation, and Selection (TIES)

TIES is a combination of most of the techniques explained above. It utilizes a design model and then iterates upon the different variables involved in the model. At a certain step in the process the model has new technology infused into the process. There is a correlation matrix that ”pushes back” on how new technology is included in the model and then uses a design methodology such as Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) or Joint Probabilistic Decision Making (JPDM) in order to deal with the final selection of technologies. TIES provides a process to deal with new technologies, but does not necessarily deal with the specific dependencies associated with the problem. [54] The correlation matrix involved directly addresses the mutually inclusive and exclusive dependencies, not partial dependency scenarios. TIES gives a specific technology portfolio, but does not optimize the scheduling of these elements or give the option to do partial funding. Of the technology portfolio tools presented so far, TIES is the best example of dependencies and gives a technology portfolio, but lacks the scheduling and partial funding fidelity.

1.1.3.5 STrategic Assessment of Risk and Technology

The STrategic Assessment of Risk and Technology (START) is a framework created by the START laboratory at the NASA Jet Propulsion Laboratory in Pasadena, CA. It was created in 2004 as a "system for quantifying the features of each technology, assessing its risk, calculating the probable return-on-investment, and using the results to compare candidates for development. These systems can be invaluable tools for selecting technologies, monitoring and guiding their development, and optimizing mission success." [Weisbin]

This tool takes the multitudes of new technologies associated with any one capability and prioritizes them according to their value, cost and scheduling associated with the intended technology. It is a generally applicable tool that has been applied to a multitude of complex problems ranging from space and aircraft to small business and power grids. START gives a technology portfolio that takes into account the cost and scheduling aspects, but it does not take into account dependencies between the development efforts.

1.1.4 Technology Portfolio Selection Tool

This is a sampling of capabilities associated with technology portfolio selection, but each provide information that is necessary to deal with dependencies. QFD and IRMA both deal with qualitative and mutual inclusion/exclusion of dependencies, but do not deal with non-binary correlations between elements associated in the deployment. ANOVA may determine the degree of a dependency, but has a strict assumption of independent elements. This undermines the study of dependencies. It also requires multiple data points in order to deal with the statistics of the problem. ANOVA can deal with the non-binary dependencies, but has a hard time discerning between the mutually exclusive and inclusive ones.

TIES gives the most complete end to end inclusion of dependencies. However, a working model is not always available when looking at integrating new technology into something that does not have a prototype or working model to integrate. It also has the detriment that it gives an optimized portfolio, but does not optimize the scheduling associated with the portfolio.

START, like TIES, gives a complete end to end technology portfolio selection process. It has the added strength that a working model is not needed as well as its ability to optimize the schedule associated with the development effort. However, START does not take dependencies into account. It has designations that give a certain weighting to an element, but this is not dependent on another element. This is still an independent designation that does not give the fidelity that this thesis seeks to provide.

This thesis is different in that it looks at specific dependencies and not just inclusion/exclusion of dependencies. It hopes to give a broader methodology of working with dependencies in the fact that there are no multiple data points to work out a statics model or a prototype with known relationships to have a working model.

START's general applicability made it a great tool to integrate dependency studies into next generation NASA technology selection techniques. It is at a unique junction of helping current NASA decision makers decide which technologies will be funded starting in Fiscal Year 2011 and how these decisions will have an immediate impact. START is an XML Python based program that allows the user to input a file into the program to be analyzed. It utilizes linear programming with the revised simplex method and branch and bound optimization techniques in order to optimize the objective function created by the program and fueled by the customers.

1.2 Chapter Flow

Chapter 2 will define the research questions and hypotheses associated with investigating the research objective. This will then lead into Chapter 3, which will look at first categorizing dependencies and the specific ones that this project will deal with. This was a milestone in itself, because the design space is so large, that it had to be scoped down to a manageable data set that could let the dependencies be demonstrated as well as documented with real data. Chapter 4 will then go into how to deal with dependencies before Chapter 5 walks through an example. Chapter 6 will combine the findings and present the research plan to move forward with answering the research questions. Chapter 7 through 11 will follow the steps of the dependency methodology to show the result of adding dependencies to the technology portfolio development. Chapter 12 will end with the conclusions to the results presented in Chapter 9-11.

CHAPTER II

RESEARCH OBJECTIVES AND QUESTIONS

2.1 Research Objectives

The main research objective of this thesis is to explain the technology portfolio with respect to dependencies. In order to accomplish this, several research questions were posed to quantify this aspect of adding fidelity into the technology portfolio selection process. These research questions give a first glimpse into the effect of adding fidelity to the process.

Categorizing technologies and the subsequent dependencies associated with them proved to be a daunting task. The first issue that was encountered was the fact that technologies are made to deliver the highest return on investment (ROI). That means the highest value for the lowest cost that can be associated with a specific technology. A user then looks at the technologies' ROI and chooses the highest with respect to the given objective. The problem is that each objective value changes given the scenario; technologies may work in conjunction with other technologies or other capabilities, and may in turn work with other levels of the program. Adding dependencies to this provides different scenarios. This necessitates the need for decision makers to take advantage of this fidelity being lost in the independence assumption process.

Each one of the scenarios suggests that there are different types of dependencies. These are commonly categorized as self-interaction and cross-interaction. Self-interaction is the interaction between elements of the same set. Cross-interaction is the interaction between different sets. Multi-level interactions will be used here to recognize the level of the problems as well as the sources of the dependencies. These are modeled in Figure 4.

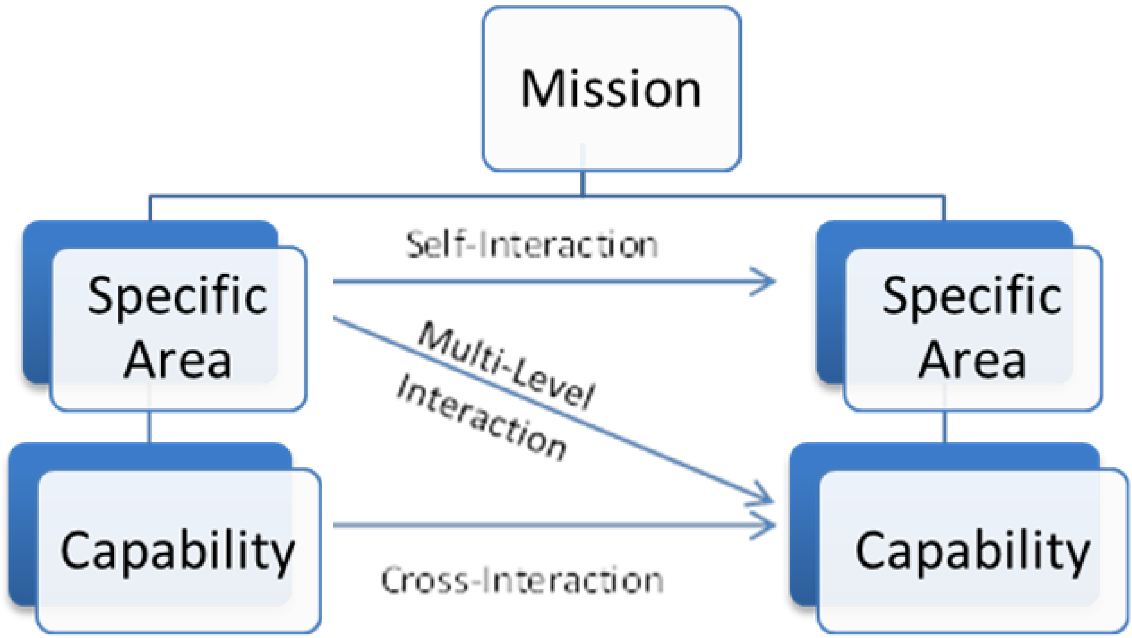


Figure 4: System Engineering Elemental Dependency Structure

Another way to think of the dependencies is the origin of each element. Self-interactions are interactions amongst elements with the same parent, while cross-interactions are interactions of elements with different parent elements. The multilevel interaction captures the parent to child interactions. [62]

2.2 Research Questions

In order to address the main research objective, the first research question comes as:

What are the dependencies associated with Technology Portfolio Selection?

This task was just as large as characterizing the types of parameters associated with the element interactions. The identification of the problem presented a large generic design space. The simple questions of who, what, when and how must be answered in order to make this lattice network. The goal that must be accomplished is the "why" of the problem while the "where" must be pre-defined for the project. This is seen in Figure 5.

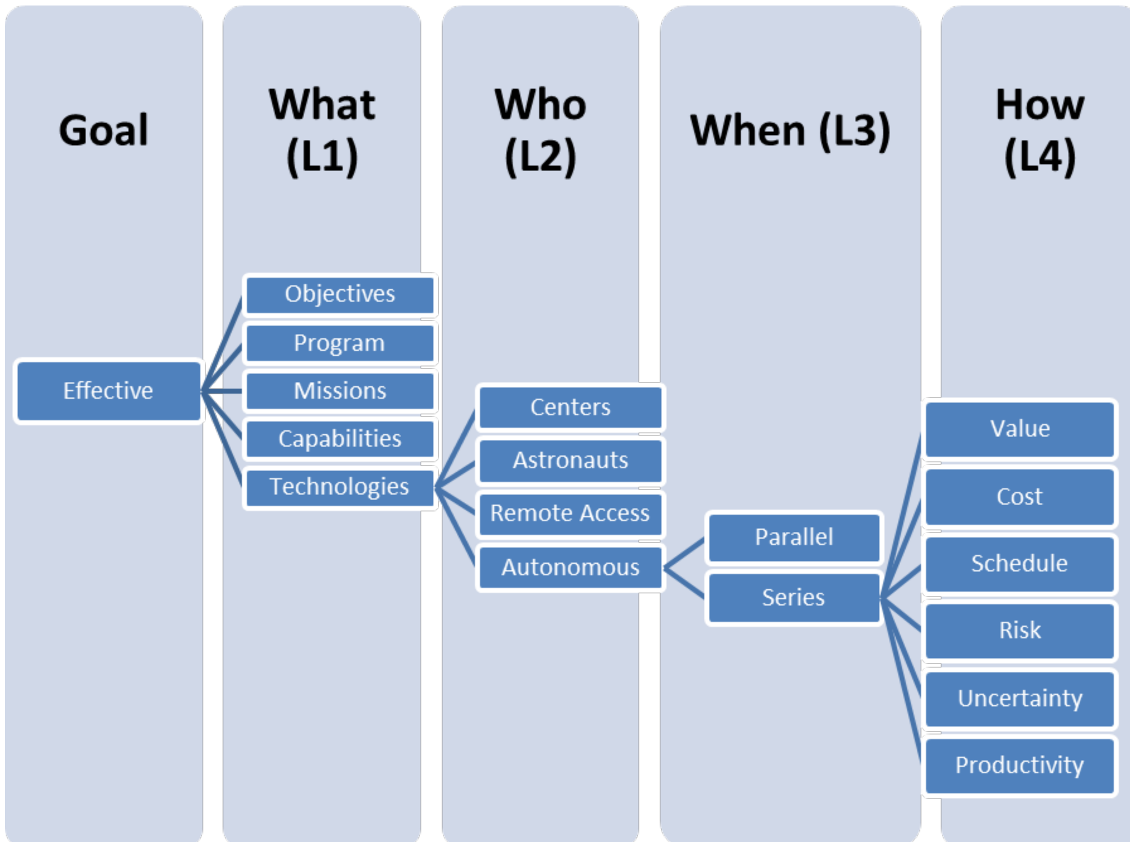


Figure 5: Possible dependencies of a lattice structure

This question was an intermediate, but necessary step, on the way to answering the applicability of this thesis.

What is the effect of adding dependencies on the Technology Portfolio Selection Process?

The actual effects of including dependencies are paramount within this field of work. These additions to the system engineering realm would change the fidelity of the tools associated with technology portfolio selection and constraint analysis. On the other hand, the inclusion of interaction and correlation matrices are to be in addition to the main effects of technology portfolio selection. They are not to be a replacement. Therefore if these effects are too large or robust that they overpower the initial system and conditions specified, they are not necessarily useful as tools to enhance the system engineering field.

The last two questions are more of an academic concern:

How does changing the investment time frame affect the technology portfolio?

This question deals with when higher fidelity is needed. If applying dependencies is too computationally costly without enough of a benefit in the technology integration and planning phases then when should be taken into consideration. Now if the design space is not frozen, then the metrics measured for progress will change. The question is how will it change and is it worth the change? What are the benefits of adding dependencies if it takes away the freedom and increases the cost?

Which has a larger impact on the Portfolio Selection: Input values or the Dependencies?

This question stems from the fact that in a perfect world, the most accurate input values would model the dependencies and give the inherent perfect values. It may be easier to find accurate input values rather than model them with dependencies. It is a long held notion that perfect input values have the largest effect on the portfolio value. However, changing the input values does not necessarily increase the fidelity of the tool. This question will investigate the effect of each one on the value as well as fidelity of the portfolio.

From these research questions, two objectives were investigated below:

O1: Introducing dependencies into the START process changes the recommendations and optimal portfolio analysis associated with Tech Portfolios

O2: Dependencies are dominant to Technology Portfolio selection and Analysis

These two hypotheses will be revisited throughout the thesis to show their relation to the research questions in an effort to answer the main research objective. The first hypothesis investigates the addition of dependencies while the second hypothesis assumes the importance of the fidelity. Success will be measured in the abilities and new attributes given to the process as well as the overall impact of these changes.

CHAPTER III

TECHNOLOGY PORTFOLIO DESIGN SPACE

In order to categorize the design space parameters a sample problem was needed. Constellation was a former NASA program that involved a large complex design space. The objective of the Constellation Program was to carry out a series of human expeditions ranging from Low Earth Orbit to the surface of Mars and beyond for the purposes of conducting human space exploration. It required the next generation of technology portfolios and provided a valuable opportunity to push the technology envelope forward.

Constellation had certain requirements that the program would fulfill in an effort to further human exploration of the solar system. The following was taken from the Constellation Architecture Requirements Document (CARD) presented in 2006. It was intended that the information and technology developed by this program will provide the foundation for broader exploration activities as the operational experience grows. The specific Constellation Capabilities are given below in Table 1.

Table 1: CARD Capability Descriptions [32]

Card Number	Description
CA0001-HQ	The Constellation Architecture shall deliver crew and cargo to the lunar surface and return them safely to Earth.
CA0013-HQ	The Constellation Architecture shall perform Lunar Sortie missions to any designated location on the lunar surface
CA0005-HQ	The Constellation Architecture shall provide the capability to establish and support a permanently habitable outpost on the lunar surface.
CA0014-HQ	Draft The Constellation Architecture shall establish a Lunar Outpost located within 5 degrees latitude of the lunar South Pole (TBR-001-009).
CA0003-HQ	The Constellation Architecture shall provide the capability to perform crewed and robotic activities to further scientific knowledge during lunar missions
CA0004-HQ	The Constellation Architecture shall provide the capability to perform engineering demonstrations and satisfy development test objectives during lunar missions.
CA0006-HQ	The Constellation Architecture shall provide the capability to demonstrate resource extraction and utilization from in situ materials during lunar missions
CA0202-HQ	The Constellation Architecture shall perform lunar surface EVA.
CA0889-HQ	Draft The Constellation Architecture shall deliver crew and cargo to the surface of Mars and return them safely to Earth.
CA0011-HQ	Draft The Constellation Architecture shall provide the capability to perform activities to further scientific knowledge during Mars missions.
CA0404-HQ	Draft The Constellation Architecture shall provide the capability to extract and utilize resources from in situ materials during Mars missions.
CA0074-PO	Draft The Constellation Architecture shall transfer the crew from MTV to the Earth surface in no more than 3 days.

These twelve mission capabilities gave rise to a program architecture approved for the Constellation program seen below in Figure 6.

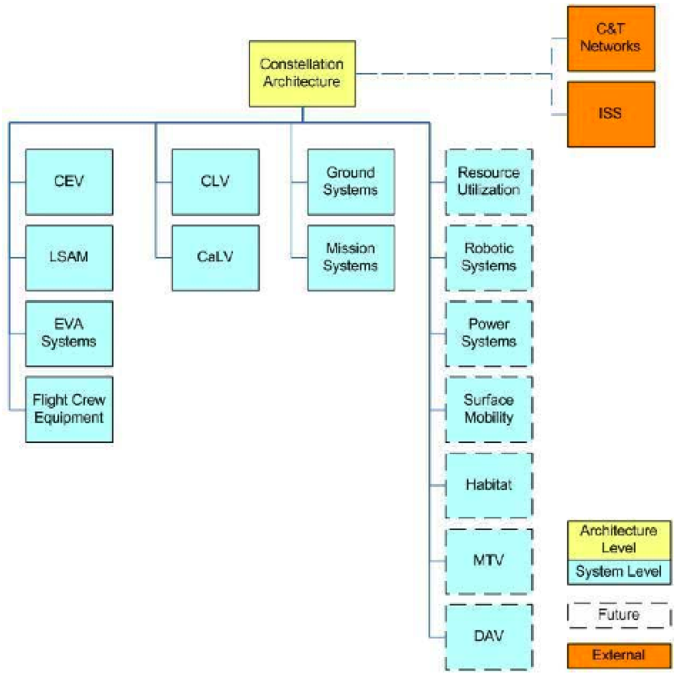


Figure 6: Constellation Architecture Hierarchy from the Constellation Architecture Requirements Document [32]

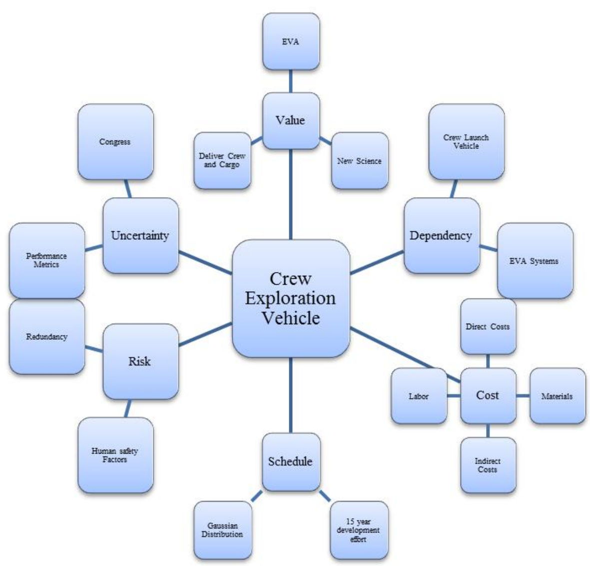


Figure 7: Constellation Development Effort Example

Each box represented in Figure 6, shows a system that is a critical part to the Constellation Architecture. Each box was a complex system with multiple components that required its own technology portfolio. Such a complex and large system brought about many uncertainties as to how to accomplish the required capabilities and work in coordination with both existing and new assets. Figure 7 shows an example of a Crew Exploration Vehicle Development effort.

This program was to be a delicate balance of interactions amongst programs, assets and people. These interactions and dependencies brought a level of complexity that must be examined to move towards higher fidelity design and planning tools. Even at this top level of Constellation, the capabilities shown are NOT independent. The top level capabilities described by CARD gave eight Lunar capabilities in preparation for four Martian Capabilities. This chapter will demonstrate the need for dependencies using Constellation as a sample problem.

3.1 STrategic Assessment of Risk Technology (START)

START is a branch and bound linear program that uses the revised simplex method. It utilizes these methods to determine the optimized development effort for the technology portfolio.

3.1.0.1 Linear Programming

Linear Programming is used to optimize a liner objective function subject to linear equality and inequality constraints. Linear means that it has constant coefficients. START is trying to optimize the performance of a project subject to the budgetary and time constraints. The addition of dependencies to the START model requires the designer to work within the START framework and is mathematically defined in Equations 1 and 2.

Maximize

$$cTx \tag{1}$$

Subject to

$$Ax \leq b, x \geq 0 \tag{2}$$

Where x represents the vector of variables to be determined, while c and b are vectors of known coefficients and A is a known matrix of coefficients. The expression to be maximized or minimized is called the objective function, cTx in this case. The equations Ax=b is the problem that START solves. It translates the objective function into the Ax= b problem and then solves it using the revised simplex method and branch and bound methods. START utilizes the lpsolve Module for its linear programming optimization package. Lpsolve ia freeware open source code that can handle up to 10,000 cases. [26, 8, 23, 35, 53, 10, 65]

3.1.0.2 Revised Simplex Method

The Revised Simplex Method is a way to solve linear programming problems. This is the method that START uses to solve the technology portfolio problem. It is different from the Simplex method in the fact that it modifies its dictionaries at the beginning of each iteration

rather than the end as the original simplex method does. Chv́tal observed the following.

> Each iteration of the revised simplex method may or may not take less time than the corresponding iteration of the standard simplex method. The outcome of this comparison depends not only on the particular implementation of the revised simplex method but also on the nature of the data. ...on the large and sparse [linear programming] problems solved in applications, the revised simplex method works faster than the standard simplex method. This is the reason why modern computer programs for solving [linear programming] problems always use some form of the revised simplex method. -Chv́tal [19]

The Revised Simplex method utilizes the same Ax = b format presented in the linear programming section. Figure 8 shows the steps in both the simplex and revised

Simplex Method	Revised Simplex Method
Determine the current basis, *d*.	$d = B^{-1}b$
Choose x_j to enter the basis based on the greatest cost contribution.	$\tilde{c} = c_V - c_B \cdot B^{-1} \cdot V$ $\left\{ j \middle\| \tilde{c}_j = \min_t(\tilde{c}_t) \right\}$
If x_j cannot decrease the cost, *d* is optimal solution.	If $\tilde{c}_j \geq 0$, *d* is optimal solution.
Determine x_i that first exits the basis (becomes zero) as x_j increases.	$w = B^{-1} \cdot A_j$ $\left\{ i \middle\| \frac{d_i}{w_i} = \min_t\left(\frac{d_t}{w_t}\right), w_t > 0 \right\}$
If x_i can decrease without causing another variable to leave the basis, the solution is unbounded.	If $w_i \leq 0$ for all *i*, the solution is unbounded.
Update dictionary.	Update B^{-1}.

Figure 8: Comparison of the Simplex and Revised Simplex Method[24]

simplex methods. Notice how the basis d, is determined in the beginning as opposed to the last step in the regular simplex method. [24]

3.1.0.3 Branch and Bound

The Branch and Bound method is used to find discrete optimizations. It goes through and systematically evaluates branches of a tree for each variable being considered and determines if the branch is worth searching or not. If a branch is considered useless, the optimization calls are stopped and that branch is cut from the design space. In this way, multiple design combinations are thrown out quickly when a branch is cut off high up on the branch. It throws out branches based on estimates of the boundaries being optimized.

It requires two tools: splitting and branching. During splitting, two smaller subsets are returned that covers the entire domain. Branching defines a tree structure whose nodes are the subsets that the splitting method created. In order to create the boundaries estimates, it uses a method called bounding. The idea is that if the lower bound for a node is greater than the upper bound for a tree node than that branch can be discarded: also called pruning. The recursion process is stopped when there is only one element in the tree left. That is now the optimized answer. [25, 37, 36]

3.1.1 START Architecture

In order to run START the customer must sit down and evaluate each development area on multiple levels. The most important variable inputs to the objective function are the performance, utility and need of the technology. This information is compounded to create the objective function that is optimized in the lpsolve problem. The constraints are given equal consideration which then translates into the Ax=b problem as described in the linear programming section.

3.1.1.1 Performance Probability

The customer must give information on the performance metrics of the technology up for funding. The customer gives a low, high and middle performance value that is taken and translated into a probability function as shown in 9. This performance probability is then used to find the performance value. That value determines the expected utility function which then is optimized in the START equations. Since it is a probability distribution, the area under the curve is equal to one. This is normalized in the coding, so the customer is free to give real values of the expected performance metrics. The performance probability function defines the development costs of the

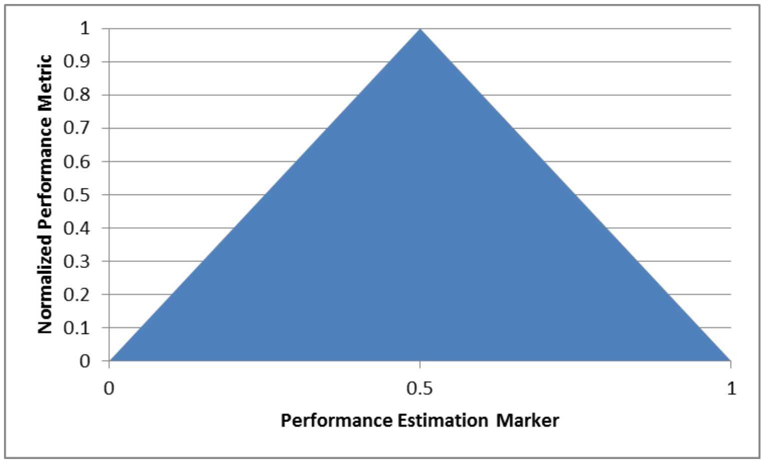


Figure 9: Performance Probability Triangular Distribution

portfolio.

3.1.1.2 Utility Function

The need and utility given by the customer is expressed as a utility function as shown in Figure 11. The x-axis is the need while the y-axis is the utility. This utility function is called upon each time the expected utility for a function is created. The utility function defines the requirements of the portfolio. Figure 11 gives the high and low for utility and need metrics. Once the furthest point is determined the possibilities may continue pas the extreme points given.

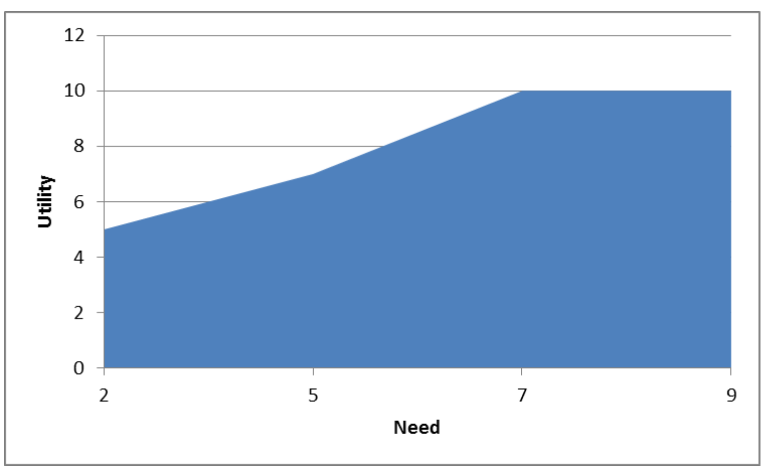


Figure 10: Utility Function of Utility vs. Need graph

3.1.1.3 Expected Utility

The information gathered for the performance probability and utility function are then combined to create the expected utility equation shown below in Equation 3.

$$ExpectedUtility = \int PerformanceProability * UtilityFunction \tag{3}$$

The expected utility is used to determine the objective value of the objective function for each technology.

3.2 START Input File

The customer must input the development effort information into the START input file framework and then ultimately put it into a lattice work of information. The input file looks like Figure 11.

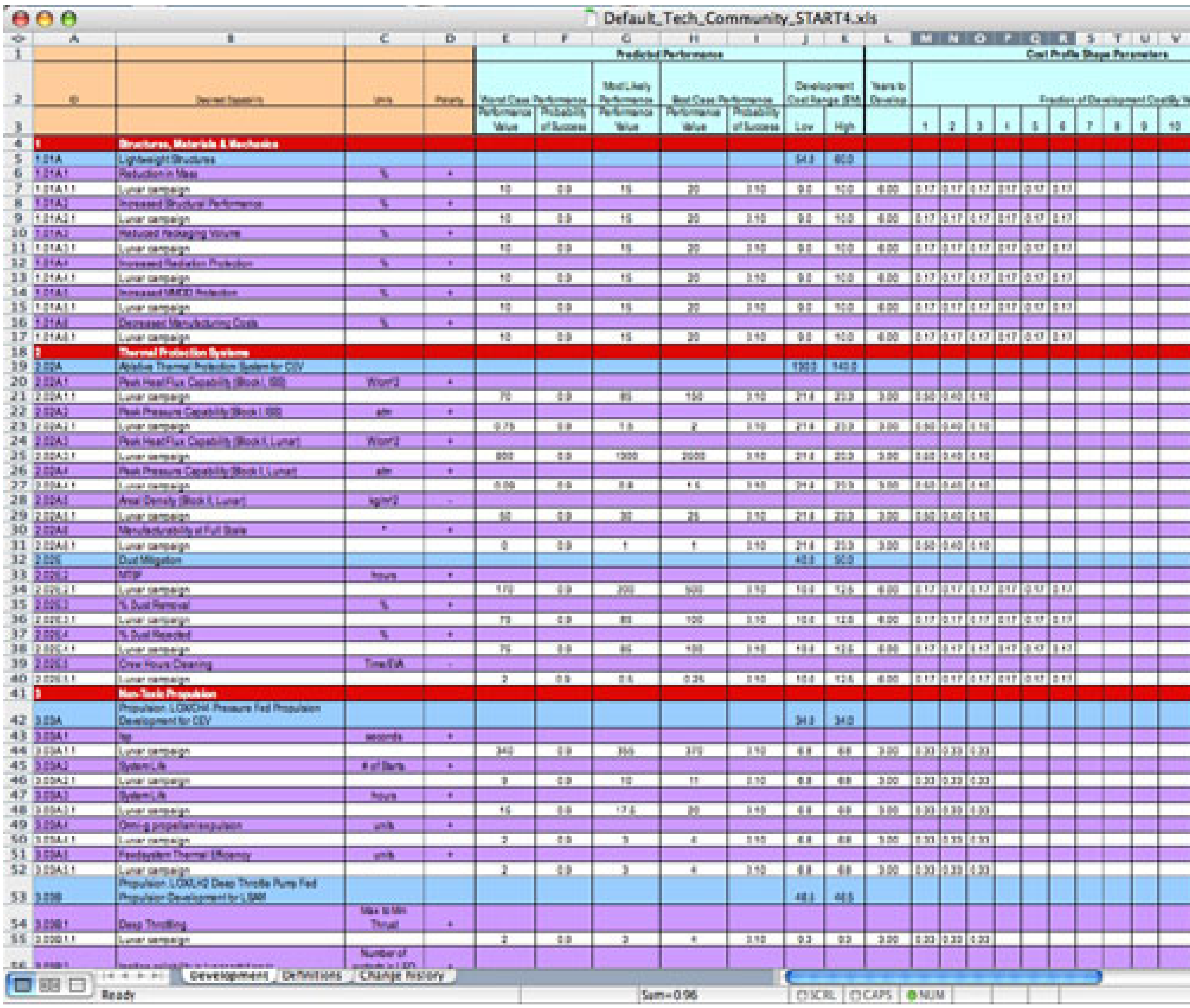


Figure 11: Sample Input file courtesy of the START laboratory

Table 2: Input file color scheme explanations and Example from CARD architecture

Color	Definition	Example
Red	Area	Spacecraft
Blue	Project Code	CEV
Purple	Metric	Crew Module
White	Data	Objective

The categories and technologies are color coded to understand the lattice work of each as shown in Table 2. It shows how a technology would fit into the Constellation lattice structure.

The example given shows how the crew exploration module in relation to the first CARD objective of delivering crew and cargo to the moon would fit into the color coordination. This information is completed for each technology and in this way determines the lattice structure that is inherent in START and integral to the dependency studies presented here.

3.2.1 Optimization Function

Knowing this information from the customer allows START to create the objective function as given below.

Maximize:

$$\sum_i (MissionWeight * \sum_j (ExpectedUtility, j * x_{ij})) \tag{4}$$

Where x is the technology being considered, i is the total number of technologies, and j is the on/off switch.

3.2.2 Baseline Case

In order to start the process, a baseline case was run that included the top level CARD capabilities, the three criteria channels seen in Figure 3 and the specific technologies associated with each section. This information was put into the START input file to create the expected utility return of each one to the overall capability program of constellation. Deciding the input file required the use of an importance scale as shown below in Table 3.

Table 3: Input scale

Value	Description
0	Not Important
25	Slightly Important
50	Important
75	Very Important
100	Extremely Important

Figure 12 shows the baseline case for Constellation based on the eight capabilities that Constellation is required to fulfill. The baseline is a flat graph that shows level

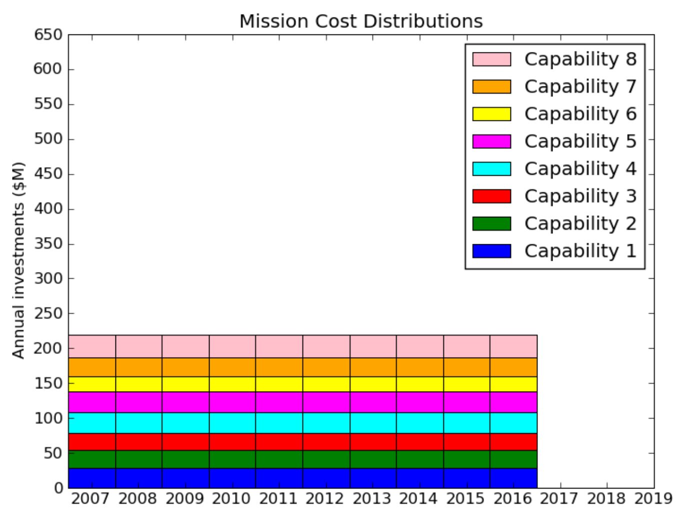


Figure 12: Baseline Output File from START

funding for the entire 10-year process that Constellations is expected to integrate technology. In order to interpret this graph, one must look at the level of funding for each capability to determine the level of importance. This shows that Capability 6 has the least amount of funding, while Capability 5 has the most funding. This would suggest that Capability 6 is the least important when it comes to the given monetary and temporal constraints in START, and that Capability 5 is the most important.

This is not a typical output graph because it only takes into account how much to fund. It does not take into account when to fund, but a further study of the design space shows how this is taken into account with each input parameter.

3.2.3 Design Space

The baseline case was used to investigate the design space for the data set according to Constellation's top level capabilities. There are nine major inputs that affects the output of START: Mission Weights, Reserved Capability, Need, Utility, Performance, Year Desired, Years to Develop and the cost profile.

3.2.3.1 Mission Weights

The mission weights showed a way to investigate START to deal with only one mission at a time. Literally, this is a multiplicative factor that puts emphasis on specific missions. This number can be between 0% and 100%. This parameter allows the user to search specific missions and optimize technologies according to those specific capabilities as well as not choose to fund a capability. It is a static multiplicative factor that the customer sets. It can be used to investigate one mission at a time and then compared to the end result that START provides with each mission is searched against each other.

In order to search this design space, every mission was set to zero, except one was set to equal to 100%. This resulted in only that mission being searched and giving results. A second search was done to set every mission equal to 100% and make only one mission equal to 0. This resulted in everything being chosen except that mission that had a mission weighting of 0. This is an excellent tool to exclude and include missions for a quick comparison. Every capability was chosen in the baseline and thus when one capability received a mission weighting of zero, that was the only mission not chosen. It is not a generic fact that this will always happen. START is an optimization process that searches within a given design space and given constraints.

Therefore, the mission weights is another constraint that is implemented in START. It is not guaranteed that missions will be selected if they have a mission weighting of 1, but it is guaranteed that they will be searched and considered. It is guaranteed that missions with a weighting of 0 will not be considered in the search.

3.2.3.2 Reserved Technologies

Reserving technologies is a part of the development side that tells START that it must choose this technology. It is placed in reserve and the given constraint lines change with each technology that is reserved. Reserving technologies tells START how to change the design space and then search all the other technologies that are not reserved, but need to fit into the given constraints. The reserved technologies determine if technologies are funded first and then allow their value to be applied to each mission.

Reserving technologies could possibly over-constrain the design space. Putting too many technologies in reserve with budgets that go over the budget creates an infeasible design space. However, putting no technologies into reserve searches everything and allows START the most flexibility. While the reserved aspect is an important variable, the initial searches will have nothing in reserve and create the technology portfolio from scratch.

3.2.3.3 Need

The need represents the architects need for that technology. This parameter only says if the technology adds value to the capability being investigated. The original baseline included various needs for each technology as applied to each capability; however, when each need was increased to the next level of importance this changed the dynamic of the problem as shown in Figure 13.

By unanimously increasing the need for every technology, two capabilities were not chosen to be funded. However, every capability was chosen to be funded throughout

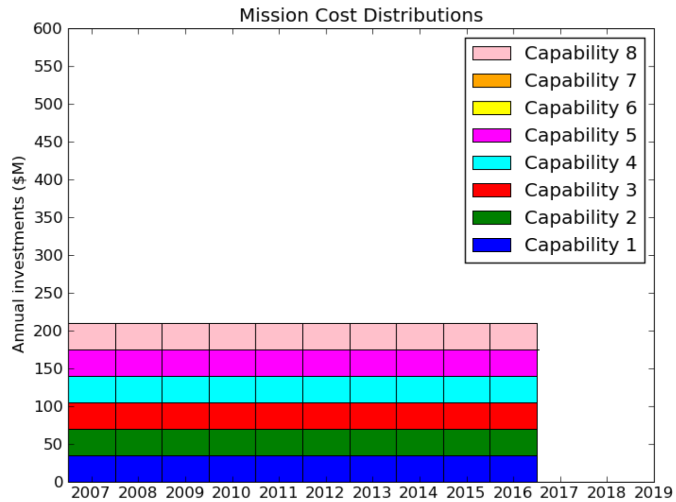


Figure 13: Output file with Increased Need Variables

the entire process in the baseline. Since only the need changed, Figure 13 shows that the Capability 6 and 7 were not selected. Changing the need changes the utility spread by U(x) = U(x+25) in Equation 1, which in turn changes the shape of the distribution and ultimately the process that START will then search and optimize.

3.2.3.4 Utility

The utility like the need changed the distribution as well. Once again, each utility was increased to the next level of importance for all technologies. However, this time START optimized the space and did not select capability 3 and 5. This is seen below in Figure 14.

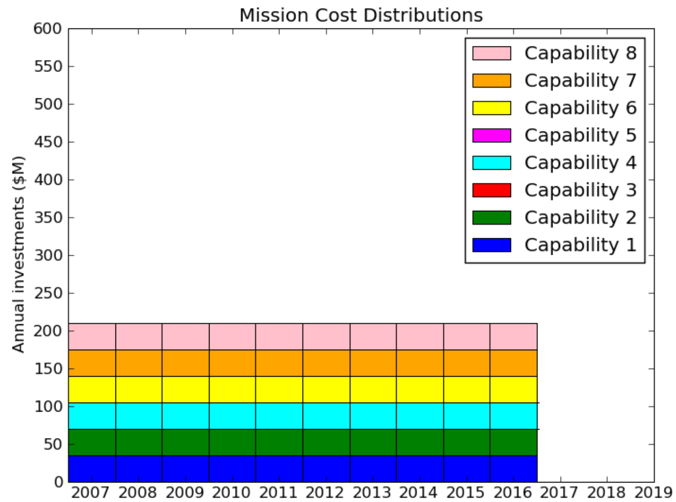


Figure 14: Output file for increased Utility variable

The utility part of the input says that this is the value or expected use that this technology will contribute to the given capability. This changes the distribution again by U(x) = U(x+25). It is important to note here the utility and need both make up the utility function, but did not have the same affect when the importance values were changed. They both choose different capabilities not to fund.

3.2.3.5 Performance

The performance metric is a metric that showed the actual performance of the technology with respect to the capability. The need showed if the technology was applicable, the utility showed how it was used for the capability, and the performance shows how well it will perform. The need and utility are determined by the architect, however the performance is determine by the developer. These three parameters play a key role in whether or not technology is selected.

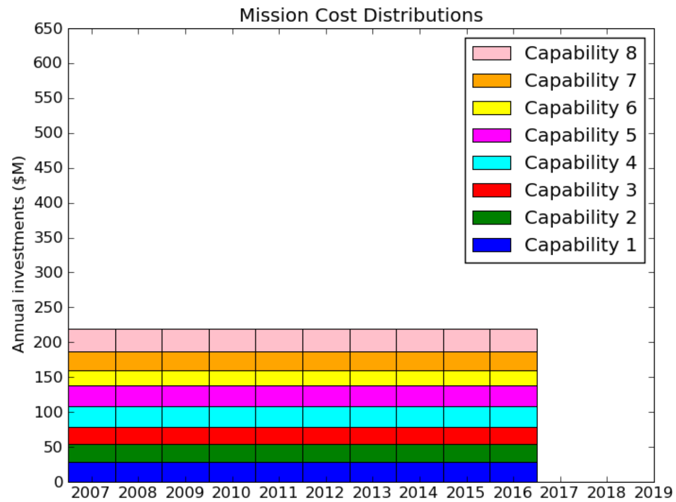


Figure 15: Output file for changed Performance Variable

Figure 15 shows the new portfolio where the performance parameter was increased. Stepping up the performance parameter to the next level showed that Capability 5 still had the least amount of funding, but Capability 8 replace Capability 6 as having the most funding. The performance parameter changed the performance probability distribution of the problem. By increasing the performance of everything, it changed the funding levels, but not which capabilities were funded. This is not a generally applicable rule for how performance works, but it is just as essential as the need and utility variables since it directly affects the expected utility function.

3.2.3.6 Year Desired

Changing the year desired for the various technologies creates a step function as shown in Figure 16. This shows that changing the year desired, allows for the technologies to be optimized each year. The technologies can start different years which do not need to be capped by the first year only. This actually allows a step function similar to the cost profile. Therefore this variable is extremely important to the optimization and design space exploration. This also shows the fact that there is no need to fund a technology past its desired year.

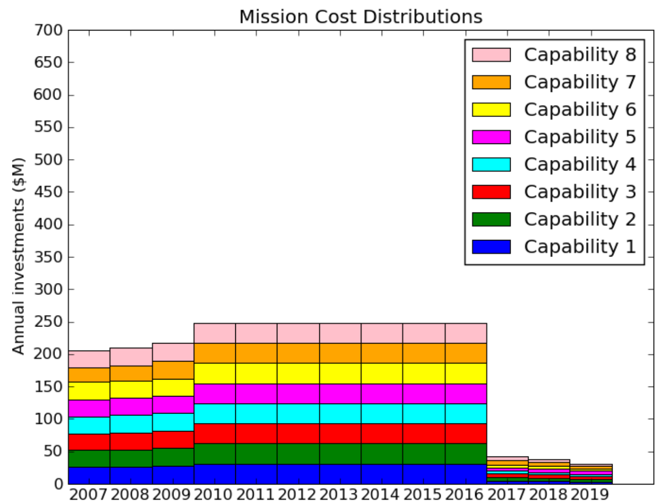


Figure 16: Output file for change in Year Desired Variable

3.2.3.7 Years to Develop and Cost Profile

Changing the years to develop allowed for the capabilities to have different development times. Therefore there is a temporal importance that is captured in this variable. The cost profile was changed to a Gaussian distribution which is a common profile utilized. These variables have the most impact on the profile: temporal, cost, and schedule wise. This is seen above in Figure 17. This is a typical START output profile. The challenge comes when the systems engineer tries to compare the output profiles to each other. [69, 77, 78, 51, 63, 68]

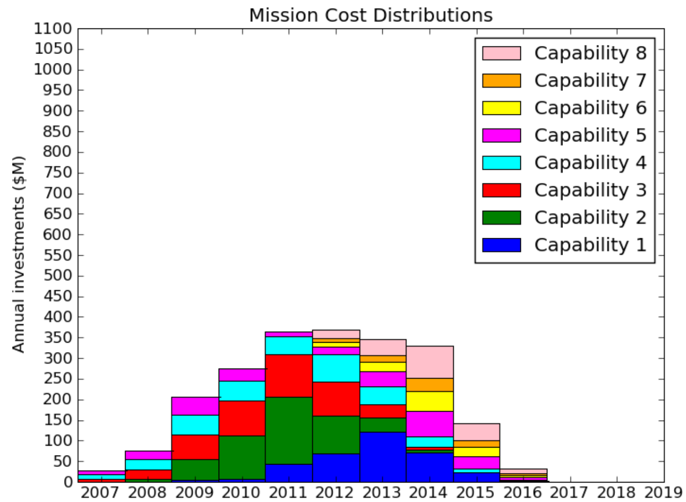


Figure 17: Output file for change in Years to Develop and Cost Profile

3.2.4 Dependency Assumptions

The START example ran the baseline input file. This takes into account the parameters shown and gives the schedule and cost optimized solution. However, START does not take into account the dependencies associated with the technologies and capabilities. Looking at the capabilities provides multiple dependencies between capabilities. Figure 18 shows a sample selection of the dependencies between the elements.

This information suggests a few conclusions of the sample dependencies. Objective 1 and 7 are not dependent on any other objective. They are independent capabilities with respect to the other aspects of Constellation. They actually enable most of the other objectives. On the other hand Capability 4 enables nothing. Capability 2 is both enabled and enhanced by Capabilities 5 and 8. Therefore, these three capabilities must be done in parallel. The user assumes that Capabilities 1 and 7 should be accomplished first since they enable most of the other capabilities.

CARD gives no order associated with the capabilities. Taking into account the dependency between the capabilities, Figure 19 shows the optimized order that would be more efficient if taken one at a time.

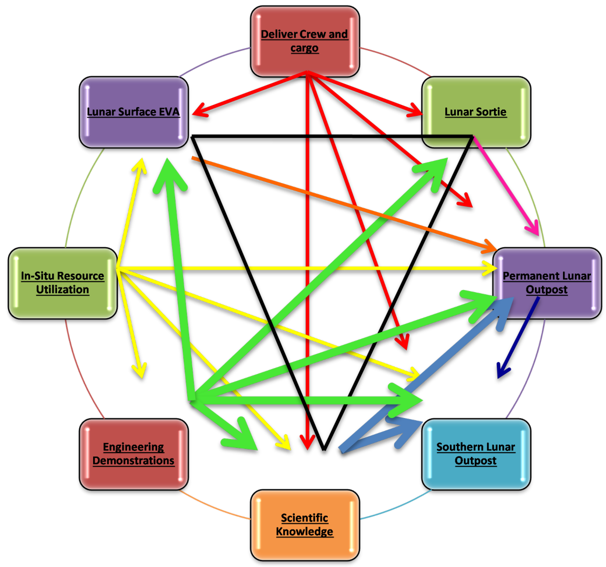


Figure 18: Sample Dependency Categorization amongst CARD top level Capabilities

Comparing the START outputs with the quick dependency studies shows that there was information not taken into account. Looking at the baseline output from START, as shown in Figure 20, shows that there is a completely different order according to funding level than that of only taking into account the dependencies as shown in Figure 19. These two figures show that the independence of capability 1 and 7 are taken into account after capability 4 which is dependent on everything else.

Figures 20 and 19 show that different information is being taken into account at the exclusion of adding dependencies. This simple example establishes the need for integrating dependency information into technology portfolio selection, but it does not explain how to deal with it. Chapter 4 will explain the theoretical context to dealing with dependencies and Chapter 5 will walk through an example from the proposed theoretical context.

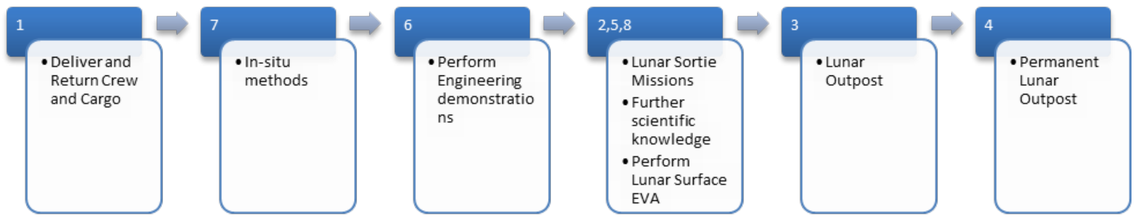


Figure 19: Order of importance taking into account only dependencies between capabilities

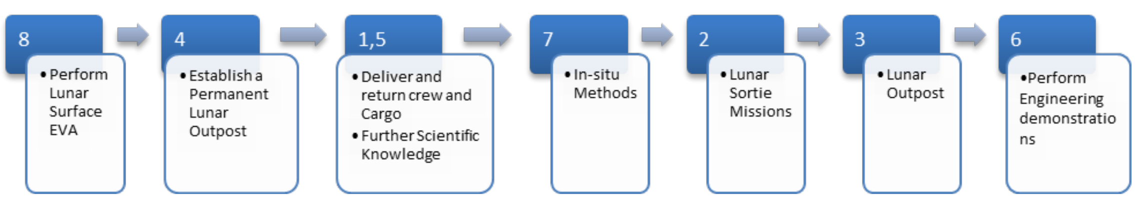


Figure 20: Baseline case from START output without dependencies

CHAPTER IV

DEPENDENCY COEFFICIENTS

Three options are given below in Figure 21 to deal with dependencies: change the input file, output file or the actual process. The fourth possibility is to do a hybrid solution of all three.

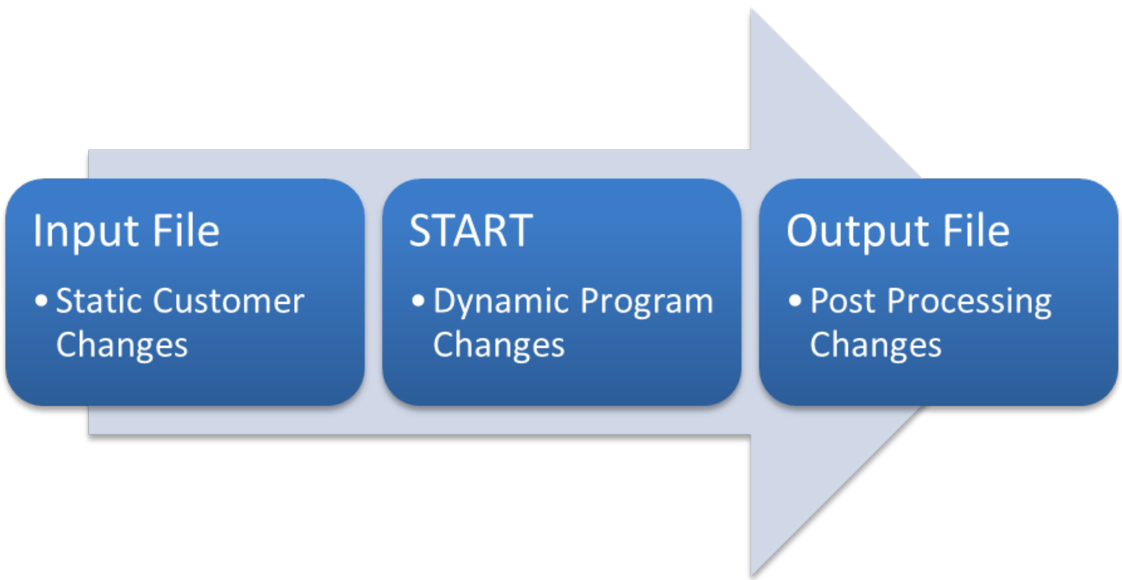


Figure 21: Possible places to address dependencies in the START architecture

These mediums are used to capture dependencies. In order to take them into account a penalty function is assigned. This manifests into the form of coefficients that are assigned to the resulting affected elements which may add or subtract value. This chapter will give an overview of how to change the three areas given in Figure 21, as well as lay the foundation down for how the methodology will progress.

4.1 Input File

The input file is an excel spreadsheet that provides information to the START program. In order to deal with the dependencies here, the Excel sheet was used to put in if-then statements to deal with changes. The if-then statement identifies the coefficient placement. This simple test identifies where to place a coefficient; not necessarily how to apply the coefficient. The actual impact of the number are different coefficient values that will deal with the impacts based on the type of dependency.

The user must deal with dependencies based on the type. There are different coefficient methods to deal with the various technology variables explained in Chapter 3. These can be captured statically in the input file by the user. This gives the customer the advantage of having more input and a chance to really capture their desired effects when it comes to technology portfolio recommendations.

4.2 Output File

Changing the output file is once again a static process, but here the optimization has already occurred. At first glance this does not seem to be the best solution in order to show technology portfolio solutions. Post-processing the solution may result in a non-optimal result since the process was dynamically optimized during the process. However, the physical meaning of changing the output file corresponds to missions becoming precursor to future missions.

The same principles as the input file are applied where the process changes the objective value of the specified technology and then filters out the lowest value to change the post-processing. The issue here is that statically filtering out a dynamic problem does not guarantee that the overall portfolio value increases.

The solution presented suggests that instead of using a filter to eliminate solutions, post processing dependencies may be used to increase funding for specific technologies that would enhance or enable other missions. This makes the output dependency coefficient a means to ensure an element that has already been selected for a specific project may enhance an additional project not taken into account by START. This output dependency phenomena may be dynamically captured within the process by integrating dependencies into the process.

4.3 Penalty Functions

In order to implement possible dependencies, the equivalent of a penalty function is needed to create the effect of both static and dynamic dependencies. A closer look shows that there are various ways to apply penalty functions. The interior, external, and interior-exterior are the most common. If the problem being solved is of the form f(x) then adding a penalty function gives T(x) = f(x)+r P(x) where P(x) is the degree of the penalty and r is called the penalty parameter.

4.3.1 Interior and Exterior Penalty Functions

Both the interior and exterior penalty functions work by making unconstrained minimizations. An interior penalty function is a benefit to a development effort, while an exterior penalty function is a detriment. In both applications the direction of the dependency must be taken into account.

4.3.2 Interior-Exterior Penalty Functions

This is a combination of the two methods to deal with both benefits and detriments. Combining the penalty functions means modeling multiple relationships in the input and output files. Creating multiple additions and subtractions from multiple places. Physically modeling the penalties is an issue due to the two implementation possibilities. The first could be seen to be a certain percentage increase in a metrics value. The other could be a finite addition with no percentage that would matter on the current value of the technology. [41, 42, 24]

4.4 Dynamic Programming

Penalty functions work for static changes where the user has decided that there is a penalty that changes the input or output files. However, it is not effective when the penalty is a selective change. Penalty functions are expressed here as static changes, but do not address the relationship of two elements being chosen together or separately. Therefore in order to deal with the dynamic programming changes, a new proposal is suggested. This proposal includes dynamically programming dependencies into the START framework. Dynamically dealing with the dependencies requires a different approach to both the programming aspects as well as the dependency implementation.

4.4.1 Programming Possibilities

There are two suggestions to deal with the dependencies in the programming architecture: grouping and separate. Both will be explored in this section.

4.4.1.1 Grouping

Grouping the technologies means creating a hierarchy within the START framework. In this architecture, technologies are grouped together and made into a higher group and then compete with the rest of the technology portfolio. This can be seen below in Figure 22.

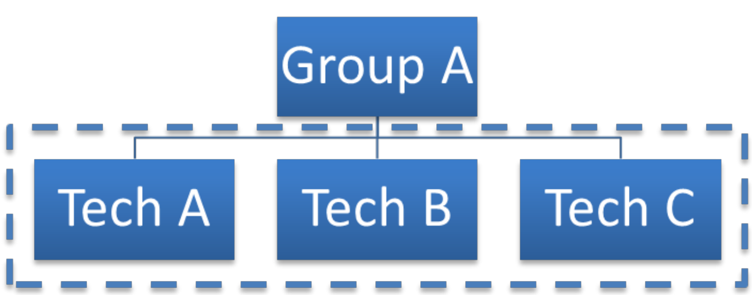


Figure 22: Grouping architecture for programming possibilities

Whenever START sees this group, it will have the attributes that Tech A, Tech B and Tech C provides at some elevated value since they are working together. This requires the user to determine both how to group technologies together as well as the elevated metrics within the groupings. START must then be used to check for each metric separately if they are chosen together as well as use the elevated one.

This type of grouping means creating a subroutine that groups the technologies and then inputs them into START. It does not change the START programming or framework, just a separate subroutine. This is a simpler solution to the programming architecture. However, the problems arise from the issue of how to combine technologies. A technology is part of a development effort, which has a schedule and cost associated with the process. Therefore determining the best group development effort is a challenge to the systems engineer when placing this type of "group" into the input file.

4.4.1.2 Separate

Making a separate correlation coefficient for each technology requires a direct connection. In this case, each technology must have a direct correlation coefficient applied to it. Each technology competes for inclusion in a portfolio just as before, but each technology is elevated when a dependency is applied during the selection process. This is seen below in Figure 23.

Figure 23: Separate architecture for programming possibilities

The issue here is that the START programming must be changed each time to apply the dependency as well as check to see that the dependency is chosen. This way each benefit of a dependency is not applied regardless of if it is warranted. This is a harder solution than the grouping architecture, but it allows for the program to apply dependencies directly to the technologies. It also does not deal with changing the established input file development efforts. For these reasons, the separate programming environment was implemented in this model.

4.4.2 START Programming

Changing the START programming provides the dynamic solution to including dependencies. START's input file must indicate dependencies are present as well as define them. Therefore this requires some pre-processing and the addition of new information. The true challenge is to implement the correct dependency coefficient function for the chosen dependency. This chapter will later touch upon the different types of dependency coefficients used after going through the architecture behind implementing them.

4.4.3 Correlation Coefficients

A correlation coefficient is traditionally defined as Equation 5

$$\rho = \sum((x_i - x)(y_i - y))/(n-1))\sigma_x * \sigma_y \tag{5}$$

Where x and y are two mean values of the elements and σ is the standard deviation of the element. This is the traditional statistical determination that requires multiple sets of data. The customer is adopting new technology, so there may not be statistical data to draw upon. There may be times that the company adopts upgrades to current technology resulting in learning curves that may extrapolate the values, cost and scheduling of new technology. The customer may also give the extrapolated values in the input file and then this application would be an overestimation.

4.4.4 Multiple Input Multiple Output

The Multiple Input Multiple Output (MIMO) method comes from signal processing arrays. This is found in many different applications from wireless signals to satellite signal receivers. This is a well-documented approach that has appropriate applications with correlation matrices. In the MIMO model, each user generates many independent multi-path signals arriving to the adaptive array within $\pm\ \Delta$ off the mean angle of arrival (AOA) φ. This is taken from Loyka et al. and Figure 24 shows the geometry that multiple path signals take and deals with the correlation coefficients.

The AOA probability density function is assumed to be uniform and all elements are assumed to be statistically independent and similar. This is given by Equation 6.

$$R_{ik} = 1/2\Delta \int {}^{\sigma-\Delta}_{\sigma+\Delta} \exp|jz(i-k)\sin\beta|d\beta \tag{6}$$

Where z = 2 d/λ, d is the inter-element distance, λ is the wavelength, 2Δ is the angle spread of the incoming multi-paths, φ is the average angle of arrival and j is the imaginary unit. For the signal array processing, they assume there is no loss of generality for λ to be assumed unity. For $\delta = \pi$, Equation 6 reduces to the classical expression found in Equation 7.

$$R_{ik} = Jo[z(i-k)] \tag{7}$$

Where J_0 is the zeroth-order Bessel function of the first kind. For $\Delta < \pi$ a Bessel series expansion was derived in Loyka et al. However, for small Δ this converges extremely slowly and requires many terms to estimate R_{ik}. A simple approximation for Equation 5 for small Δ and $\varphi = 0$ can be derived using sin(β)=$beta$ for small β, and performing integration on Equation 5:

$$R_{ik} = [sinz(i-k)\Delta]/[z(i-k)\Delta] \tag{8}$$

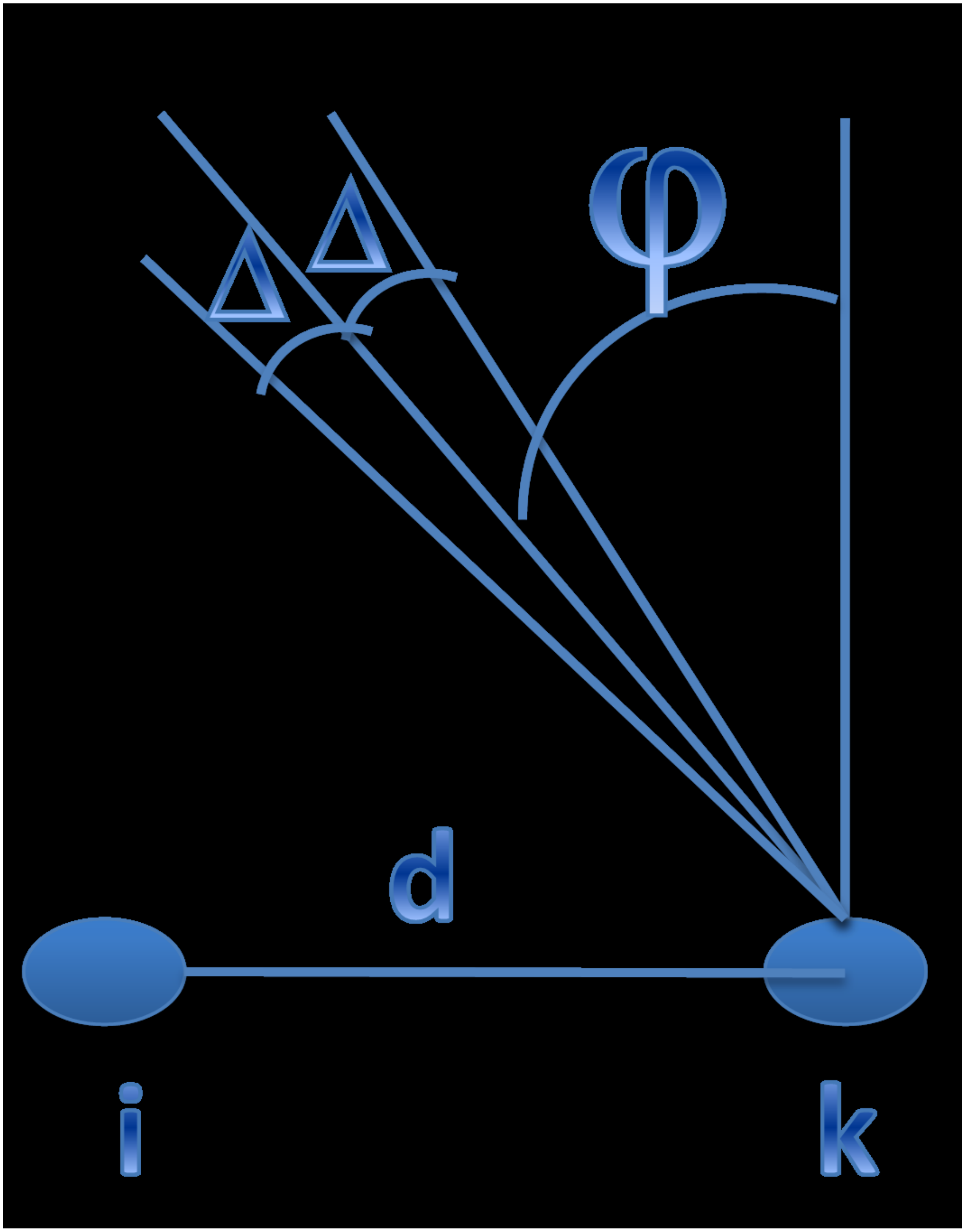


Figure 24: Current MIMO geometry for MIMO equations

The smaller Δ , the better the accuracy with an upper bound around $\pi/4$. This is how the traditional correlation coefficient is calculated for signal processing arrays. The reason that the MIMO solution for correlation coefficients is suggested is that there are similarities with having multiple inputs and outputs. The multiple inputs being modeled here are the dependencies from different technologies. The multiple outputs are the same type of value given to the overall portfolio. MIMO was also suggested because it gives a closed form solution to the dependency matrix R_{ik}. However, in order to implement the MIMO solution a few changes must be made in order to deal with the lattice network created in START. Therefore Figure 19 is changed to look like Figure 20 below.

The first thing that is changed is the element location from side by side to source and sink of the dependencies. The angle defines where in the lattice structure the source of the dependencies starts. The Δ is the confidence in the dependency that should be modeled. The value d is almost like the dependency cloud radius found around the element whose values are being changed for the dependencies.

4.4.4.1 Equations

Knowing these differences, the MIMO equations may be adjusted for the START framework. The most important is the fact that the correlation coefficient matrix is dependent on two things: the type of dependency and the source of the dependency. The source of the dependency is inherent in the lattice structure of the development effort. The type of dependency is the hardest to find.

Problems arise for the satellite calculations when $\lambda \neq 1$ and $\varphi \neq 0$. Therefore START cannot use the Bessel equation. Starting from Equation 5, assuming that $\sin(\beta) = \beta$ and substituting A = z(i-k) gives

$$R_{ik} = (1/2\Delta) \int_{\varphi+\Delta}^{\varphi-\Delta} \exp[jA\beta] d\beta \tag{9}$$

$$R_{ik} = (1/2\Delta)(1/jA) \exp[jA\beta]|_{\beta=\varphi+\Delta}^{\beta=\varphi-\Delta} \tag{10}$$

$$R_{ik} = (1/A\Delta)(1/2j)(\exp[jA(\varphi + \Delta)] - \exp[jA(\varphi - \Delta)]) \tag{11}$$

$$R_{ik} = (1/A\Delta)(1/2j)(\exp(\varphi) \exp[jA(\Delta)] - \exp[jA(-\Delta)]) \tag{12}$$

$$R_{ik} = (1/A\Delta)(\exp(\varphi) \sin(A\Delta) \tag{13}$$

$$R_{ik} = (1/(z(i - k))\Delta)(\exp(\varphi) \sin((z(i - k))\Delta) \tag{14}$$

Table 4: Legend for sources of dependencies within the START lattice framework

Source of the Dependency	value
Multilevel	φ=0
Self-Interaction	$\varphi=\pi/2$
Cross Level Interactions	$0 < \varphi < \pi\ /2$

Where z = 2π*d/λ

λ =1/ρ

d =1/[2Δ cos φ]

Where ρ is the value of the dependency modeled. In order to get around the issue of cos = 0 issue, a Taylor's series of cos is used instead. Inserting this into the d-equation turns into the equation below.

$$d = 1/(2\Delta(1 - \varphi^2/4)) \tag{15}$$

And thus turns the z-equation into

$$z = \pi\rho/\Delta(1 - \varphi^2/4) \tag{16}$$

Adding Equation 15 into Equation 13 gives

$$R_{ik} = (1/(\pi\rho/\Delta(1 - \varphi^2/4)(i - k))\Delta)(\exp(\varphi)\sin((\pi\rho/\Delta(1 - \varphi^2/4)(i - k))\Delta) \tag{17}$$

Which simplifies to

$$R_{ik} = (1/(\pi\rho/(1 - \varphi^2/4)(i - k)))(\exp(\varphi)\sin((\pi\rho/(1 - \varphi^2/4)(i - k))) \tag{18}$$

Equation 18 is the MIMO equation used for the START lattice structure. It is only dependent on two variables φ and ρ. The Δ term canceled itself out which represents the uncertainty of uncertainty. This is a term that was controversial since a user cannot know what they do not know. These are the equations used for the MIMO correlation coefficients and the lattice structure presented.

These are correlated to the system engineering book's definition of possible interactions. This information is created in the input file. Therefore when considering the dependencies and their origin, this will be inherit in how the information is gathered and stored. [50] This was seen in Figure 4

4.4.4.2 Issues

Dealing with MIMO matrices has some issues. The first is that the matrices are not the inverse of each other; meaning that the direction matters. The correlation matrix of Technology A to Technology B is not the same as the correlation matrix of Technology B to Technology A. The matrix must be calculated for each dependency direction and therefore is not symmetric. The bottleneck for dependency implementation is suspected to be at the lpsolve modeling problem. Therefore adding too many constraints will be a problem. It is easy to over constrain the problem and make it unsolvable. The system engineers may need to determine the correct input that does not need every constraint dynamically programmed into the model.

The next issue is to have good housekeeping throughout the programming problem. It is easy to overestimate the role of a dependency. The type of programming used to implement the dependencies will determine the value of the dependency to the model.

The last issue of the MIMO method is that finding documented correlations are not easy. This is not necessarily something that is studied or even documented. However, these suggestions are exactly what must be found in order to study this phenomenon. This will be explored in more detail in the future works study, but for demonstration purposes the dependencies are considered nominal representations of elemental changes in the START program.

4.5 Programming Architecture

The procedure used comes from communication architecture for multiple input multiple output satellite connections. This seemed appropriate since the technologies and dependencies are layers that make an intricate lattice to produce an optimal result.

Satellite communication architecture deals with lattice work every day when receiving multiple signals from multiple sources in space. This structure shows the origin of the dependency, and the type of dependency. This is information that must be taken into account when it comes to implementing the dependencies.

In order to understand this process and apply it to the START frame work the user must first understand the optimization problem. As explained in Chapter 3, START works with the customer giving inputs as to what the most important aspects of the project are. However, the actual optimization process requires an objective function and ultimately constrains the problem.

4.5.1 Optimization Problem

The problem that lpsolve solves is stated as follows:

Maximize:

$$\sum_i (MissionWeight * \sum_j (ExpectedUtility_i, j * x_{ij})) \tag{19}$$

Subject to:

$$\sum_i (\sum_j (\sum_q (y_{i,j,q} * c(t)_{i,j,q} \leq budget(t))) \tag{20}$$

for all years t

$$\sum_i (\sum_j (\sum_q (y_{i,j,q}))) = x_{i,j} \tag{21}$$

for all missions i and capabilities j

if capability i,j is reserved then x i,j = 1

if capability i,j is enabling then x i,j $\leq$ x n,m

for all capabilities n,m that are reserved.

if capability i,j is enhancing then x i,j $\leq$ x a,b

for all capabilities a,b that are enabling.

Therefore in order to deal with the dependency in this framework, one must understand where each dependency fits. The objective function will contain more static changes while the constraint additions will be more dynamic and expand the problem. The objective function mainly deals with value additions while the constraints deal with the time and cost dependencies. The differences here are subtle to the type of linear programming. The constraint functions are additions to the A matrix in the Ax = b problem, while the objective function shows the penalty and coefficient changes in the b vector. Therefore in order to deal with dependencies they must be applied according to the type of dependency.

4.5.2 START Programming Framework with Dependency Application

This information is presented with the fact that there are two different dependency studies that must be looked at which deals with the domain of correlation coefficients. The first problem deals with binary correlation coefficients or constraint dependencies. The second deals with non-binary correlation coefficients or value dependencies. These two programming dependencies must be incorporated into the START framework in order to explore the design space. The user may either change the objective function or change the constraints.

4.5.2.1 Current START Architecture

The current START Architecture holds true to the Ax = b linear programming problem. This is seen in Figure 26. It is not drawn to scale, but presented here as the initial phase of the problem.

The first problem is presented as the binary correlation coefficients or defined here as Constraint Dependencies. In order to add constraints, the A matrix must add rows to itself and add values to the b vector. The X-vector does not change because no new variables are introduced into the problem. Binary correlations coefficient changes the START frame work as seen in Figure 27.

Moving on to the non-binary correlation coefficients shows the other types of dependencies that exist. Figure 28 shows the changes in objective functions as well as the two different programming architectures discussed earlier. Dealing with the orange and green sections first shows the same dependencies shown here as the Constraining Dependency additions. Moving onward to the purple sections shows the grouping programming architecture. This adds constraints, and does not change the objective function.

In this grouping scenario, new variables G_A and G_B must be added to the x-vector and then the appropriate values must be added to the A-matrix. This adds columns to the A matrix instead of rows. The x-vector must be changed in this case to account for the new group variables. Whenever START encounters this variable, it will see a variable with each metric addition as the group allows from the composite of the variables. Once again, it is unknown how to create G_A and G_B due to the fact that there is no simple way to combine two elements.

In order to do the separate programming there is no need to put in new variables. This programming architecture is shown in yellow in Figure 28. An example will be given in the next chapter, but this shows the separate programming architecture. It only assumes that the values in the A matrix will be changed based on another technology when that technology is chosen. Thus, the separate programming method only changes the b-vector value according to the MIMO changes when the two elements are chosen together.

The last dependency change shown here is the fact that there may be times when the program becomes non-linear. In order to deal with this, a crude way has been developed. It is possible to think of the START framework as a module. This module may have different A-matrix or b-vector inputs. The example shown here has different b-vectors. The top level architecture changes the b-vector input, runs START, and holds the objective function. From there, the entire process is run again with the new b-vector input. The output objective functions are compared and the highest is kept as the best solution. This has the potential to slow down the process, although it does allow for START to run without dealing with quadratic or non-linear properties. The bottle neck is suspected to be at the lpsolve model when doing this. This method will not be investigated, but is recognized as a possibility. There are non-linear solvers that are a better approach to utilizing a non-linear application.

4.6 Programming Conclusions

Both problems involve extensive programming and application considerations. Constraining Dependencies deals with the binary correlation coefficient programming. This solves the problem of alternative technology selection,as well as mutually inclusive and exclusive constraints. This is primarily done by adding constraints to the A-matrix. The Value Dependencies deals with the non-binary correlation coefficients that require the MIMO correlation matrix. This deals with the separate programming application; however, some dependencies have the ability to make the problem non-linear which requires clever consideration of the programming modules. This thesis will only focus on the separate linear program problem, and leave the quadratic and grouping scenario programming for future work.

The next chapter will go into an example of applying these dependencies from the input to the output file as shown in Figure 28.

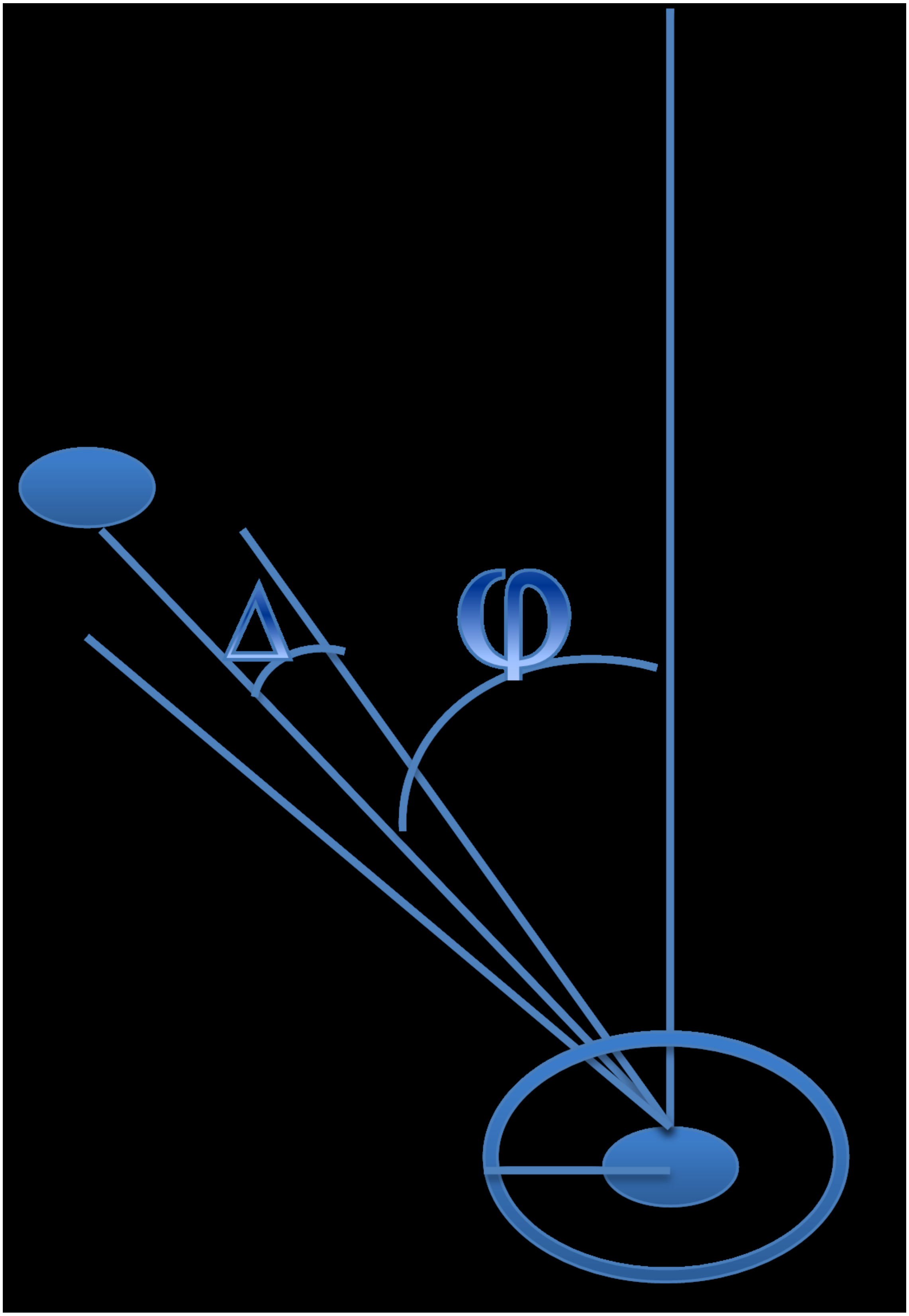


Figure 25: Adjusted MIMO geometry for START applications

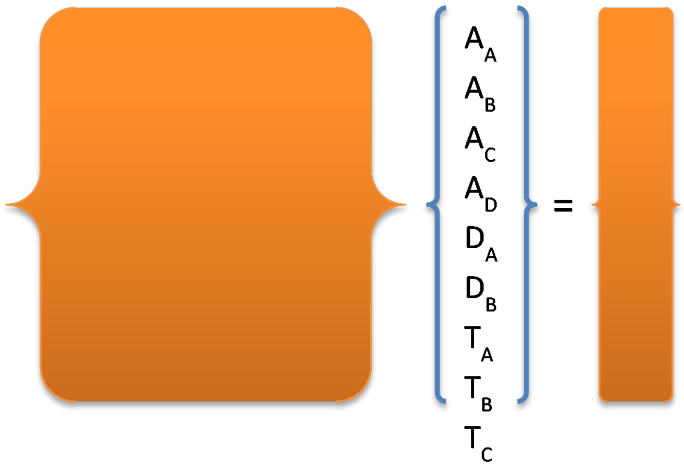


Figure 26: Current START framework for lpsolve architecture

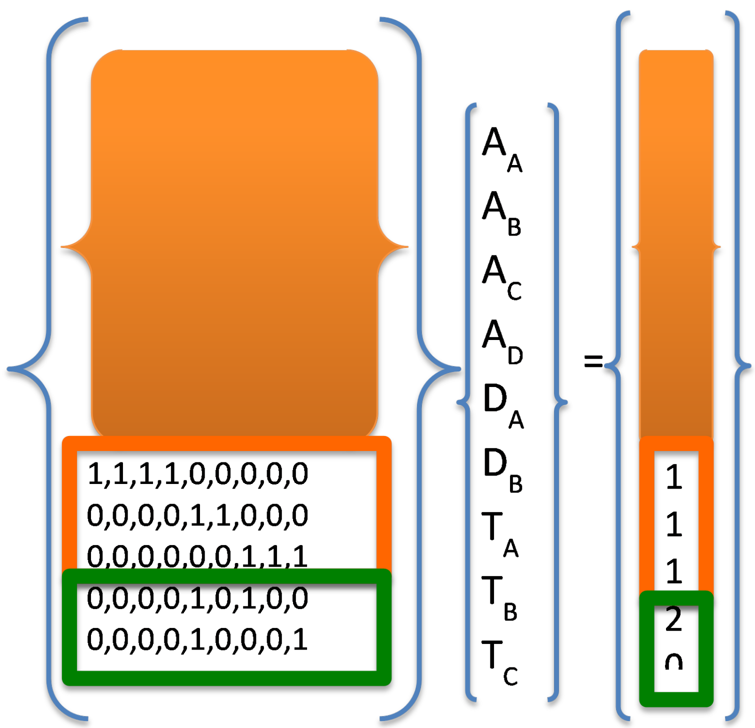


Figure 27: START Architecture adjusted to deal with separate dependencies while adding constraints to the equation.

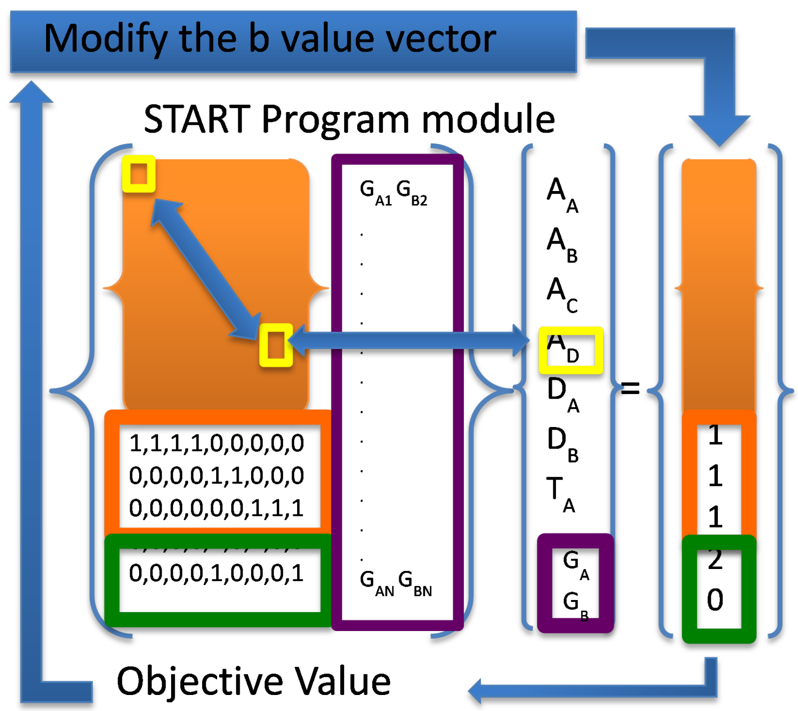


Figure 28: START architecture adjusted to add grouping and separate programming framework into the equations

CHAPTER V

BENCHMARK EXAMPLES

Constellation showed the need for dependency inclusion. The example problem for the input and output file coefficients shows the changes for each technology from the CARD selection used in the design space exploration in the beginning. This chapter will implement the static dependencies and finish with an example of the MIMO process suggested in Chapter 4.

5.1 Input files

Changing the input files presents a problem as to how to deal with the input file dependencies. Using the information about the lattice structure shows the four types of dependencies to show their affects: the self-interactions, cross-interactions, self-cross interactions and multi-level interactions as shown below in Figure 29. Now the issue with this is that the changes are static. If a value is changed in the input file, it has the same output objective value regardless of where the dependency came from. This is seen in Table 5

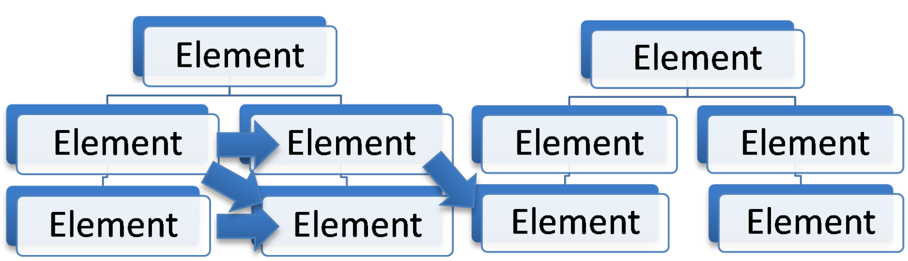


Figure 29: Example of the four input file interactions possibilities.

It is not an efficient way to deal with dependencies. The reason is that the effects of the dependencies are not dependent on other technologies. It sees the benefits or detriments of an assumed dependency, but does not do a check to make sure that

Table 5: Input scale

	Need	Performance	Time to Develop	Utility
Self	134.97	99.98	134.97	148.47
Self-Cross	134.97	99.98	134.97	161.97
Cross	134.97	99.98	134.97	148.47
Multi-level	134.97	99.98	134.97	168.72

the technology causing the dependency to occur is actually chosen in the resulting portfolio. Therefore the dependencies cannot be modeled fully by only input file dependencies only.

5.2 Output Files

Changing the output files gives the advantage of having an optimized portfolio. This is the best scenario for the precursor mission scenario. There are two missions that must be funded with technologies. If the first portfolio finds an optimized portfolio, then the process should not change the selections. Funding technologies that were not part of the optimized portfolio is not recommended because then the solution is sub-optimal. Changing the output files has the same static disadvantages of changing the input files.

Rather than increase technologies not previously funded, the better solution is to increase funding for a technology already chosen that will be used for the second technology. In this case the users would do better to take those technologies out of the second portfolio and let the others compete without the technologies in portfolio A.

This is done naturally through politics. An example for this is that Constellation is to use Lunar missions as precursor missions for Martian missions. Therefore anything created for the lunar missions would be capable for the Martian missions as technology demonstrations. The Output file scenario is best to solve the precursor mission scenario.

5.3 Programming

In order to explain the programming architecture a new example problem must be shown, because CARD is too large to show one elemental change. The Top level constellation problem was too large to effectively show how the programming architecture will be applied. This section will present a smaller sample problem.

5.3.1 Dependency Sample Problem

Suppose the objective is to deliver a low-temperature motor. In order to accomplish this, there is money to invest in three different areas: actuators, thermal control, and dust mitigation. The problem is set up in Figure 30.

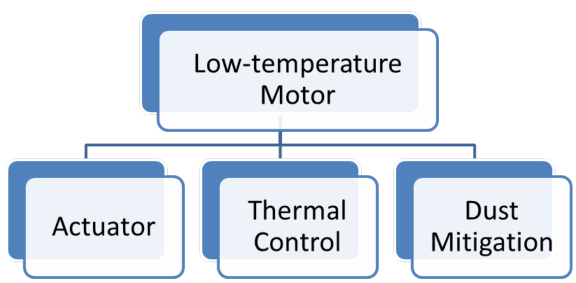


Figure 30: Programming example for a technology portfolio for a low-temperature motor selection.

A technology portfolio would determine where to invest the money in order to deliver this low-temperature motor. Programs such as START would take the inputs of each technology's metrics and decide which one to invest in and add to the project. There are multiple problems that come from this prospect. This is combined in two different problems as described here. Suppose there are four actuators, three thermal controls, and two dust mitigation technologies to choose from. Therefore the problem looks something like Table 6.

If the portfolio is not careful, it may choose three actuators and a dust mitigation

Table 6: Example Problem set up for a low-temperature Motor

Technology	Possibilities
Actuators	A_A, A_B, A_C, A_D
Thermal Control	T_A, T_B, T_C
Dust Mitigation	D_A, D_B

system. There are certain constraints that can prevent that from happening. There are also cases when certain technologies work well together or do not work with others. START can only take into account what is the best combination given cost and time constraints. These two scenarios can be modeled with dependencies. The problems are shown Table 6 in the example of binary and non-binary correlation coefficients.

5.4 Problem 1: Constraint Dependencies

This problem deals with binary correlation coefficients. This is the mutually inclusive or exclusive problem. In this case there is a constraint that states that at most one of each technology is chosen or none at all. This deals with the fact that the program needs one type of each technology for the overall goal. Constraints are added, such as the constraint of mutually exclusive as shown in Equation 22. Since START is a binary linear programming system, the technologies are either 0 or 1. This effectively deals with the multiple alternative issue.

$$T_A + T_B + T_C = 1$$

$$D_A + D_B = 1$$

$$A_A + A_B + A_C + A_D = 1 \quad (22)$$

The user may include technologies that work well together. For example, if Dust mitigation A and Thermal Control A work well together and only want to be chosen

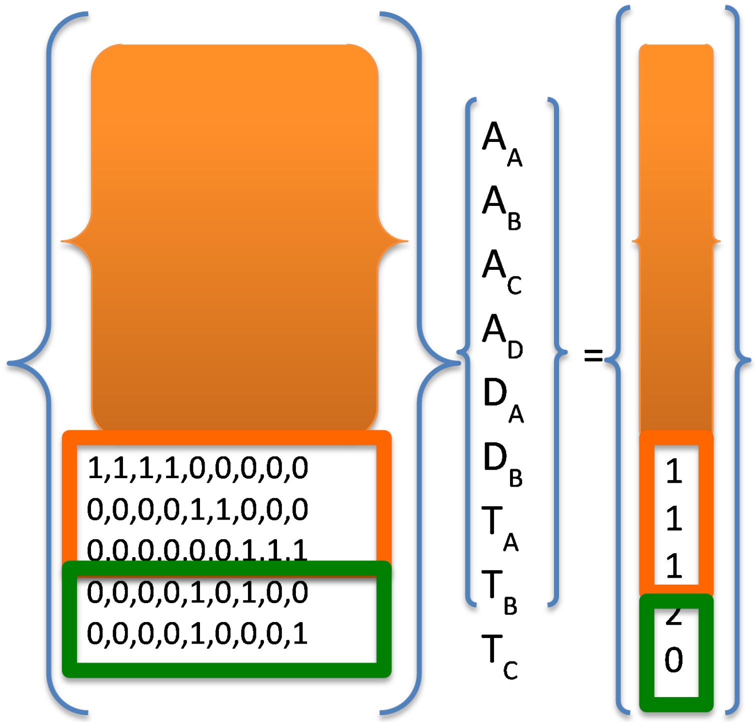


Figure 31: Inputting the constraint equations of the problem into the motor example

together, then a constraining dependency may model this. This is shown in Equation 23

$$T_A + D_A = 2 \tag{23}$$

Conversely if Thermal Control A will never be able to work with Dust Mitigation B then this may also be designed as a constraint as shown in Equation 24.

$$T_A + D_B = 0 \tag{24}$$

These constraining dependency relationships may be added to the row section of the A matrix and b-vector. This seen in Figure 31 taken from Chapter 4.

The orange boxes incorporate the alternative scenarios while the green one shows the grouping scenarios. This brings up a point about grouping scenarios. If they can just be added as constraints, is there a need to actually have a programming architecture for it? The answer is that while adding constraints deal with the binary correlation coefficients, it may over constrain the problem and ultimately not allow START to converge on a solution.

5.5 Problem 2: Value Dependencies

Moving from the binary correlation coefficients to the non-binary ones shows the need for MIMO. As explained in Chapter 4, binary correlation coefficients are not enough to deal with these types of coefficients. If the solutions are not mutually inclusive or exclusive, then something else must be done. The reason is that the user must take into account the source of the dependency as well as whether or not it is turned on or off as shown in Figure 32.

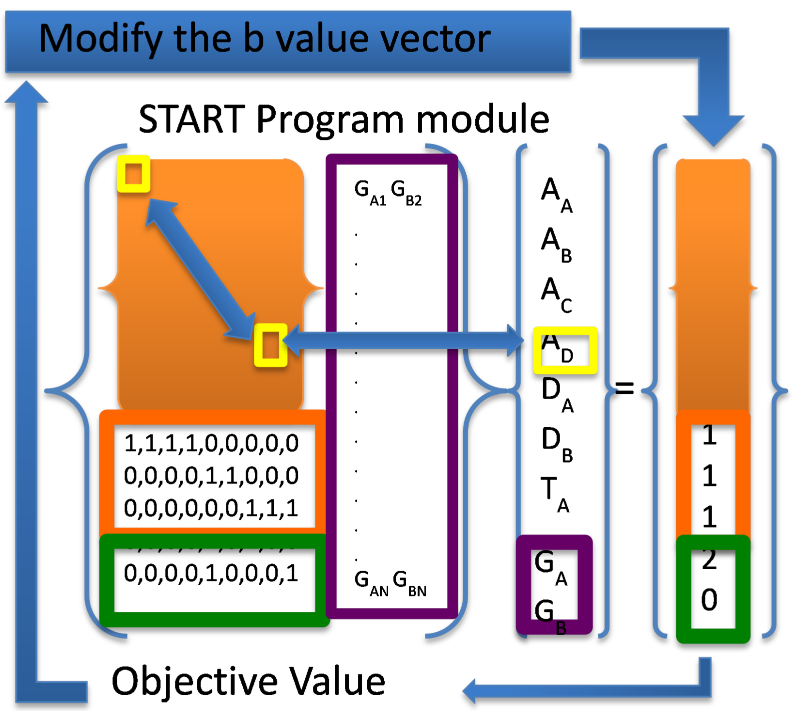


Figure 32: Inputting the Grouping scenarios into the START example

Suppose the dependency shown is the one in the yellow box from Figure 32. The assumption is that there is a dependency that Actuator A has on Dust Mitigation System A if Actuator A is chosen. Therefore the dependency is modeled using MIMO first, and then a check is done to see if Dust Mitigation system A is chosen or not.

5.5.1 MIMO Calculation

Suppose the Actuator has a dependency on the Dust Mitigation. The dependency source in the lattice structure and the actual dependency must be modeled.

For this example, assume that φ is 1.57 for the dependency source and ρ is 0.85 dependency. Utilizing Equation 25 for the MIMO equations, as shown below, gives a z-value of 6.96 for a performance to performance dependency.

$$
\begin{gathered}
R_{TD} = (\rho, \varphi) \\
R_{ik} = (\exp \varphi \sin((\pi\rho(i-k))/((1-\varphi^2/4))))/(\pi\rho(i-k)/((1-\varphi^2/4))) \\
z = \pi\rho/[\Delta(1-\varphi^2/4)] \\
Z = \pi * 0.85/((1-(\pi/2)^2) \\
z = 6.96
\end{gathered} \tag{25}
$$

Moving on to the actual correlation coefficient shows a 43% increase in the value here if Actuator A and Dust Mitigation Control A is chosen.

$$
\begin{gathered}
R_{TD} = \exp(3.14) * \sin((6.97-0))/((6.97)) \\
R_{TD} = 0.432
\end{gathered} \tag{26}
$$

Therefore START would multiply the Performance factor of the Thermal Control by 1.432 and use that value whenever Dust Mitigation A was chosen in accordance with Thermal Mitigation Control A. This example suggests that the performance of the thermal coefficient depends on the dust mitigation by 43% with an 85% dependency strength and parent-child origin of the dependency. In this way the entire correlation coefficient matrix may be created as shown in Table 7.

Table 7: Example Problem set up for a low-temperature Motor

Thermal to Dust	Mitigation	Performance	Need	Utility
φ	Performance	1.57	0	0
	Need	0	1.57	0
	Utility	0	0	1.57
ρ	Performance	0.85	0.85	0.85
	Need	0.85	0	0
	Utility	0.85	0	0
z	Performance	6.958117386	2.6703515	2.6703515
	Need	2.6703515	0	0
	Utility	2.6703515	0	0
R_{TD}	Performance	1.431640702	1.170012266	1.170012266
	Need	1.170012266	1	1
	Utility	1.170012266	1	1

The importance here is that this is only the multipliers for Thermal Control A, NOT the multipliers for Dust Mitigation A which is the source of the dependency. The MIMO took into account the dependency only based on the fact that the dependent solution was chosen, rather than the penalty function explained for the Input files which penalty was based on the actual value of the dependent variable. Knowing this information will allow for a thorough look at the research questions presented.

CHAPTER VI

RESEARCH PLAN

Adding dependencies to the technology portfolio process changes the investment decisions by changing the optimized portfolio. In order to investigate these phenomena, four research questions were presented.

6.1 Research Questions:

1. What types of dependencies are involved with technology portfolio selection?
2. What is the sensitivity of the technology portfolio selection?
3. When in the process should dependency studies be included in technology portfolio selection?
4. What is the effect of adding dependencies rather than improving the input file accuracy?

The four research questions were integrated into this process by adding the constraints into the linear program module.

6.2 Searching the Design Space

The original proposal suggested to group elements together. By grouping elements prematurely, START loses flexibility because, while technologies may interact with each other, they are planned separately. This means they have their own cost, value, and scheduling constraints accompanying them. This is an interesting topic because the real situation that is being discussed is does the addition of element A and element B really equal a new element C, represented below in Equation 27.

$$A + B = C \tag{27}$$

If elements really are grouped according to Equation 27 then this can initially be modeled by changing the input file to only include element C. However, this loses flexibility. The temporal aspect is optimized in a technology portfolio as well as the value aspect. If the cost profile looks like Figure 33 and Figure 34 for element A and element B then what is the correct combined profile for the two?

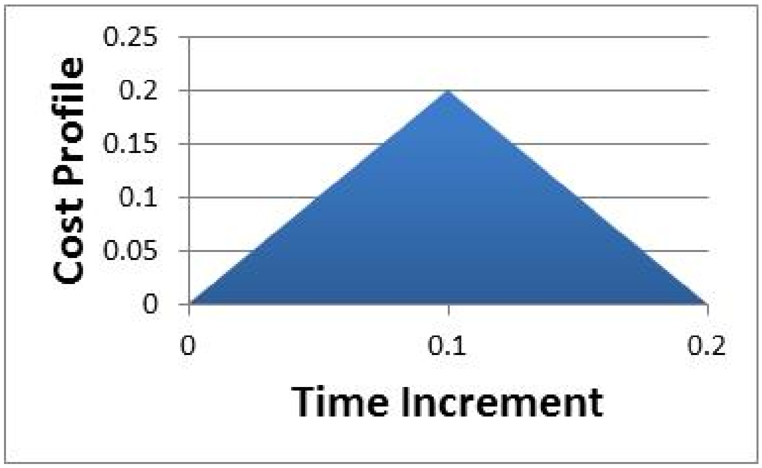


Figure 33: Element A Property Distribution

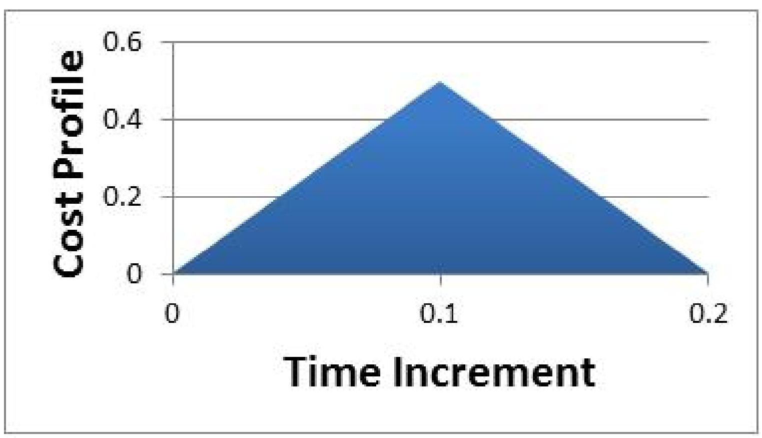


Figure 34: Element B Property Distribution

Figure 35 shows that Element C can be placed at any of the starting times as shown. If both element A and B are chosen, START would dynamically go through all the options shown in Figure 35 to decide where to start investing in technology B. By combining these cases, as shown in Figure 36 - Figure 38, the user loses the flexibility to check every case given in Figure 35.

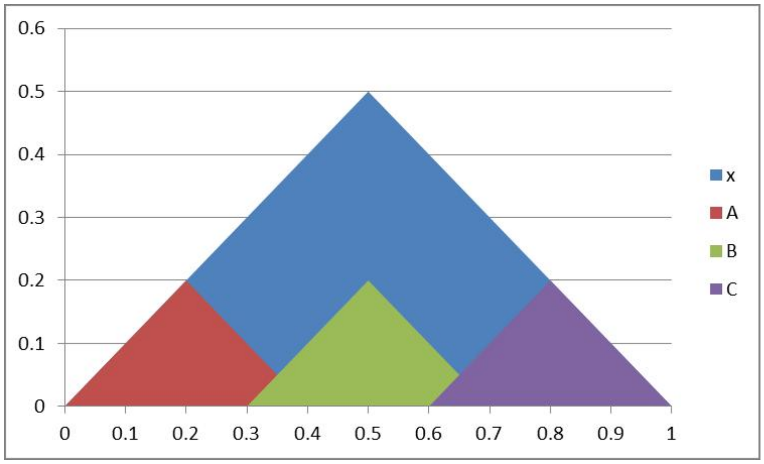


Figure 35: Inputting the Grouping scenarios into the START example

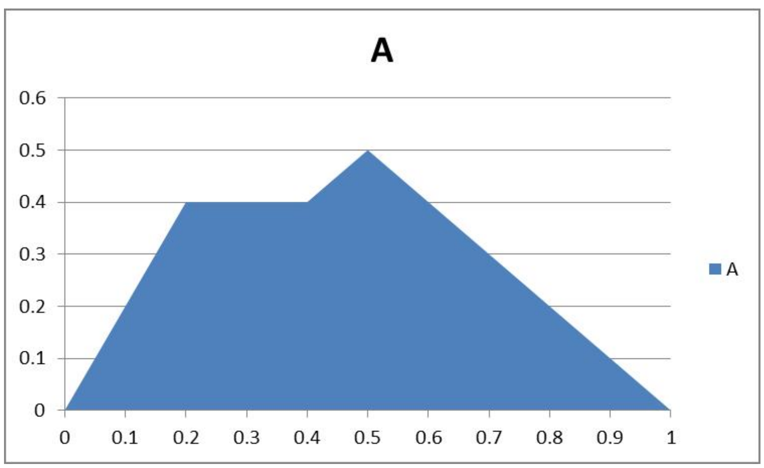


Figure 36: Inputting the Grouping scenarios into the START example

If this is decided in the input file, the question becomes which profile is the best selection. Since the methodology chooses to add fidelity to the technology selection process without losing flexibility this is better optimized dynamically instead of predetermined by the user. The next challenge comes when Equation 27 is not the correct way to combine the profiles; two more solutions are possible as seen in Equation 28 and Equation 29.

$$A + B < C \tag{28}$$

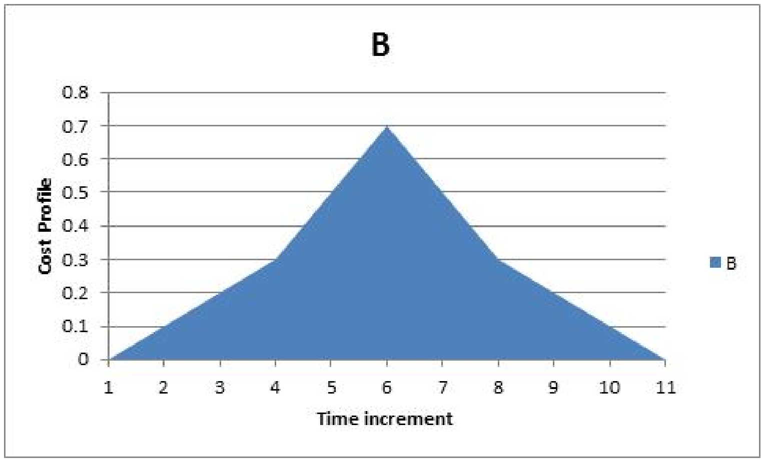


Figure 37: Inputting the Grouping scenarios into the START example

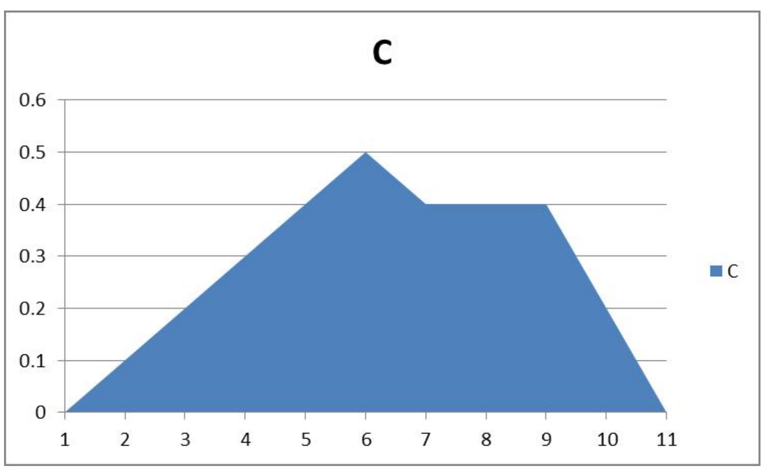


Figure 38: Inputting the Grouping scenarios into the START example

$$A + B > C \tag{29}$$

Once again if Equation 28 and Equation 29 are possible, the same problem arises of how to combine the two portfolios proposed in Figure 35 above. For the moment it is assumed that Equation 27 is the correct solution. If we further assume that the two can be combined in the manner of Figure 29 above, then the solution to include element C into the project would be to add this as an input dependency or rather different group instead of dynamically check this type of investment decision. That is why the grouping scenario was not included in this process.

6.3 Combining Elements

The optimization process chosen is a linear branch and bound system. It takes an input that creates the optimized portfolio with respect to the given elements. If a user were omniscient then they could put the correct dependent elements into the input file. Therefore the dependencies would not be needed. However, the issue is that the user loses fidelity this way since they do not know how to exactly combine two technologies without losing the flexibility. START has a few built in functions that exist to model a few dependency cases: reserved, enabling, and enhancing.

Reserved means that the technology is selected regardless of the optimization process. Mathematically it means that the mission value is zero ($\mathrm{M}_i = 0$) and the constraint on the element is one ($\mathrm{C}_i = 1$). The user cannot place a dependency on this type of element. If an element is truly reserved than it is exempt from a dependency.

Enabling means that a technology enables a mission. The solution only looks for a mission that has all the enabling technologies selected. This is the same as reserved on a mission level. Mathematically this means that the mission constrain is negative 1 (M_i = -1) and the constraint on the element is zero ($\mathrm{C}_i = 0$). There are no dependencies on enabling technologies. If an OR dependency is placed between elements, then it is effectively not choosing the mission. A mission will not be chosen

Table 8: START Logic Gates and cases

A	B	1 Currently	2 A Needs B	3 B Needs A	4 A AND B	5 A OR B	7 Not A	8 Not B
0	0	Y	Y	Y	Y	Y	Y	Y
1	0	Y	N	Y	N	Y	N	Y
0	1	Y	Y	N	N	Y	Y	N
1	1	Y	Y	Y	Y	N	N	N

if all of the enabling technologies are not chosen. If the A Needs B is put on the dependency then it is redundant since both must be selected anyway to enable the mission.

Enhancing means that a technology is in addition to the core technology requirements. Enhancing elements are the only ones that make logical sense to create dependencies upon. An enhancing element can have a dependency with any other type of element, but it cannot be the other way around.

These terms are specific to START, but the concepts are universal to technology portfolios. Enhancing technologies are the only types of elements that involve dependencies. This gives a smaller subset of elements; however, there is a logic that dictates the possible dependency cases.

6.3.1 Logic Gates

The next step concerned which dependencies should be included in the dependency selection independent of the problem or platform. This is purely what type of scenarios are possible to model. The first rule was that if the relationship was not modeled then the solution should still be possible. All possibilities were investigated shown in Table 8. Currently START operates as the first case in Table 8. The first two columns represent element A and element B. The 0 and 1's underneath represent if the element was chosen or not. The Y represents a valid case while the N represents and invalid case.

The importance is that if both elements are not chosen, this case should be allowed. If nothing is chosen or violated then START should run, and not violate the constraints. Cases 2-5 where the only ones modeled. The Not A and Not B are not interesting cases to model. If the user does not want to have element A or element B included, then they would not include it in the input file. There is no reason to model this dynamically in the environment.

6.3.1.1 Modeling Constraint Dependencies

A Needs B and B Needs A

This relationship states that one element depends on another element. This can be used for technology that interacts with or enhances a different technology. This is represented by cases 2 and 3 above. **A AND B**

This case can be modeled as two dependencies with both dependencies enabled as A Needs B and B Needs A. This is case 4. Originally the AND case was included in the list of needed dependency studies, but using A Needs B gives higher fidelity and flexibility into the system and can be modeled with a combination dependencies.

A OR B

This relationship represents competing technologies or capabilities. For example there could be two different technologies or companies for the same solution. This could also happen when there are two different scenarios, such as different types of EVA. This is case number 5.

These logic gates are independent of the actual problem being modeled and are applied to the mission and technology levels of the optimizing environment for constraint dependencies.

6.3.1.2 Modeling Value Dependencies

Value Dependencies deals with utilizing MIMO as described in the Multiple Input Multiple Output section. MIMO uses the ideal that there are multiple output and inputs that affect the element individually. MIMO is used to change the objective function of the element because it is affected by the element it is dependent upon.

If there is a MIMO dependency between Element A and Element B, the MIMO coefficient is calculated and applied to element A. Problem 1 dependencies must now be applied. START must only choose Element A when Element B is chosen so an Element A needs Element B is implemented. Problem 2 changes the actual value that is optimized after it is calculated due to the dependency associated with the process. Effectively, Problem 2 uses MIMO to change the optimization coefficient in the linear problem, and apply only one dependency constraint of A Needs B to the overall matrix.

6.3.2 Invalid START Logic Gate

There is another logic gate that would have the first row with all N's. A second logic gate that has the first row with all N's would not be valid because if the elements are required to be modeled, then the elements would be reserved as explained earlier. This case is given in Table 9. This will be discussed further in the future work chapter.

Table 9: START Logic Gates and cases

A	B	1 Currently	2 A Needs B	3 B Needs A	4 A AND B	5 A OR B	7 Not A	8 Not B
0	0	N	N	N	N	N	N	N
1	0	Y	N	Y	N	Y	N	Y
0	1	Y	Y	N	N	Y	Y	N
1	1	Y	Y	Y	Y	N	N	N

6.4 Technique for Order Preference to Similar Ideal Solution (TOPSIS)

In order to differentiate between portfolios the relative difference between portfolios was used to rate the changes that adding dependencies produced. The Technique for Order Preference to Similar Ideal Solutions (TOPSIS) was used for this purpose. It is part of the multiple objective decision making tool series (MODM) which take multiple metrics of a complex problem, normalizes them, and eventually compares them based on the users' preference. TOPSIS assumes the best solution is closest to the ideal solution so that the lowest cost and highest value is obtained. This can be seen in Figure 39.

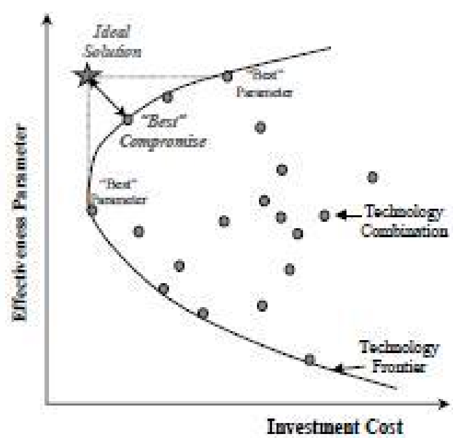


Figure 39: TOPSIS Ideal Solution

The TOPSIS methodology will measure the fidelity of the portfolios. The different alternatives can be identified by the different categories. The fidelity categories concerned here are the objective value, cost, start year, dependency level, and partial funding. These are values that contribute to the relative value of the portfolios with respect to each other. This sort of scheme allows for the fidelity to be measured.

6.4.1 TOPSIS Methodology

The TOPSIS methodology has six steps given below.

6.4.1.1 Step 1: Normalize the metrics

$$x_i j = r_i j / \sqrt{\sum_{i=1}^{m} (r_{ij}^2)} \tag{30}$$

6.4.1.2 Step 2: Construct the Weighted Normalized DM

$$v = \begin{pmatrix} v_{1,1} & v_{1,2} & \cdots & v_{1,n} \\ v_{2,1} & v_{2,2} & \cdots & v_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ v_{m,1} & v_{m,2} & \cdots & v_{m,n} \end{pmatrix} = \begin{pmatrix} w_1 * x_{1,1} & w_2 * x_{1,2} & \cdots & w_n * x_{1,n} \\ w_1 * x_{2,1} & w_2 * x_{2,2} & \cdots & w_n * x_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ w_1 * x_{m,1} & w_2 * x_{m,2} & \cdots & w_n * x_{m,n} \end{pmatrix} \tag{31}$$

This step brings in the weighting of the attributes as according to the importance to the customer

6.4.1.3 Step 3: Determine the ideal and negative-ideal solutions

Each Metric must be determined to be a "Cost" or "Benefit" to the portfolio This determines if one takes the minimum or Maximum value of the weighted normalized values Determine the Positive and Negative Ideal Solution according to the equation:

$$A^* = \{(max_i v_{ij} | j \in J)(min_i v_i j | j \in J) i = 1, 2, \cdots, m\} = v_1^*, v_2^*, \cdots, v_j^*, \cdots, v_n^* \tag{32}$$

6.4.1.4 Step 4: Calculate the separation measure

The Separation Measurement will show how far away from the best and worst possible values the given Alternatives are

6.4.1.5 Step 5: Calculate the relative closeness to ideal solution

$$S_{i+} = \sqrt{\sum_{j=1}^{n}((v_{ij} - v_j^+)^2)}, i = 1, 2, \cdots, m$$

$$S_{i-} = \sqrt{\sum_{j=1}^{n}((v_{ij} - v_j^-)^2)}, i = 1, 2, \cdots, m$$

$$C_{i+} = S_{i-}/(S_{i+} + S_{i-}), 0 < C_{i+} < 1, i = 1, 2, \cdots, m \tag{33}$$

6.4.1.6 Step 6: Rank the preference order

Then rank the order preference according to which has the highest Closeness Value. The highest closeness valued Alternative is the "Best" solution according to the criteria that is being judged

Each portfolio will receive a TOPSIS value based on the given criteria and ultimately show the relative difference between portfolio changes with respect to the dependencies implemented.

6.4.2 Visualization Tool

In order to determine the effect on the relationship on the portfolio, a visualization tool was created to compare the value of the portfolios and their subsequent changes. The information taken was the overall objective value of the portfolio to show the changes in the information.

TOPSIS was then performed with weighings metrics on the objective function. The TOPSIS values were then graphed into a contour plot. The weights can be changed which would change the investment decisions of the decision maker. TOPSIS took into account the objective function, total cost, start year, partial funding and flexibility with a bias of 30%, 40%, 20%, 10%, 10% respectively. Figure 40 give sample outputs from the visualization tool.

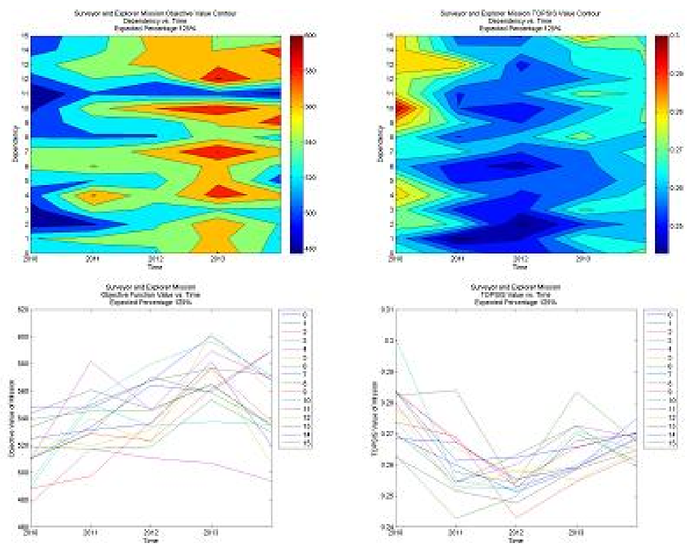


Figure 40: Sample output from the visualization tool

The left graphs are the given output value from START for the portfolio. The right graph are the TOPSIS procedure applied to this process. It is presented this way in order to see how phenomena translate to decisions made relative to each other. **Terms: Flexibility** is the ability for a technology, capability, or mission to be used for a different aspect of a program. Any capabilities that can be used for one mission and then used for another have a higher flexibility rating. The flexibility value was evaluated from experts.

Extensibility is the ability for a technology, capability, or mission to be used only for that particular mission. The extensibility is literally the inverse value of the flexibility value.

This information was used to make decisions based upon the given information for the relative difference between portfolios.

6.4.3 Scalability

The issue of scalability came up during the course of this thesis. Scalability is an issue that deals with whether or not the modeled relationships will be able to "scale up" or make an impact in problems that are larger instead of the test cases used to model. Looking at scalability as how many elements are involved in the optimization process is modeled as a 10^N problem where N is the power level

N = 0 is an irrelevant case

N = 1 is for testing problem to show that it can be done

N = 2 is a good sized sampling problem

N = 3 is a Normal Problem range

N = 4 is the upper limits of the lpsolve function capability

N = 5 is outside the limits of lpsolve

The Real question here is at what level do you do a hierarchy of problems? Problems were N $\geq$ 5 suggest a hierarchy approach. This would entail breaking the problem down into N = 3 or N = 4 problems, optimizing these sub-problems then optimize on a different level working up. However, the largest data set that has been tested using START is an N = 3 set which would be sufficient to show important decision making strategies in the early mission concept design analysis. The data tested here was a N = 2 sample problem for early analysis. As a program progresses, a problem can increase the variables, and scale to larger problems.

An interesting question would be at what point in time does a program's technology portfolio start changing N values? This will be discussed in the future section in Chapter 13 of this thesis.

6.5 Running the Design Space

In order to investigate the research questions three different axes of information were investigated. The axes are expected input percentage, dependency, and temporal. This is depicted in Figure 41.

This design space means that the portfolio changed the starting investment year for the temporal changes and the expected percentage for uncertainty in the input

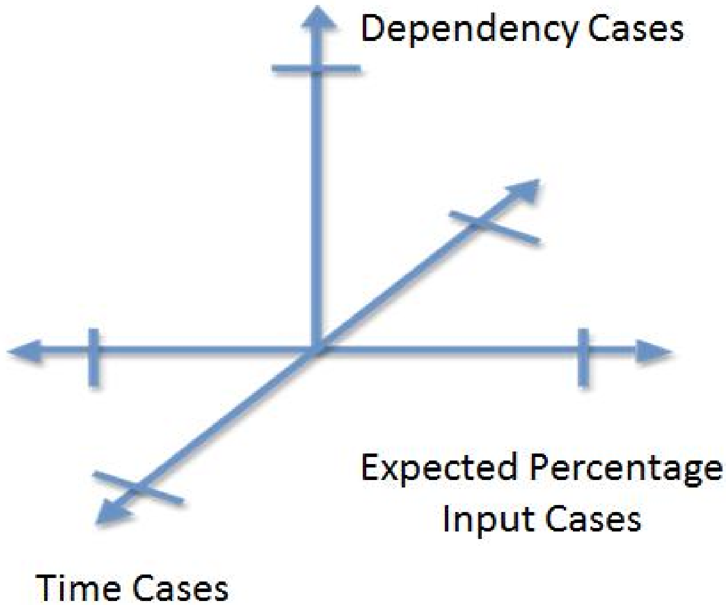


Figure 41: START Data Study Axes

values. The expected percentage was changed only on the value parameters of need, utility, and performance. These are values that determine the expected utility that goes into the objective value and ultimately is optimized.

Adding dependencies into the process shows how the portfolios change. Dependencies are not continuous changes. They are discrete changes. There is no way to add a partial dependency. Therefore, the dependency axis is discrete and represented as levels of dependencies which can be singular dependencies as well as multiple dependencies.

6.5.0.1 *Research Question 2: What is the effect of adding dependencies on the Technology Portfolio Selection Process?*

By looking at the cross section of Dependency vs. Accuracy, the second research question will show trends. Looking at this information can show how including the dependencies will actually affect the objective function and subsequent TOPSIS value.

The original hypothesis was that by adding dependencies into the process the portfolio would change. By freezing a start year and varying the expected input percentage and the dependencies, one can see how the addition of dependencies changes the investment decision.

It is expected that by adding more constraints into the problem, the solution will have a lower objective value. There is a possibility that there is no change in objective function for certain dependencies. The reason is that the relationship may not be chosen which is represented in Table 8 by the top row which shows that both element A and element B were not chosen. If this is the case, then the user may look at the relationship to see that if there were no dependencies in the process or that the relationship was either not chosen or already chosen without adding the fidelity of dependencies.

6.5.0.2 Research Question 3: How does changing the investment time frame affect the technology portfolio?

In order to answer the third research question, the temporal axis is investigated by changing the starting investment year. In this case, the start date of the portfolio was changed, effectively changing the investment time period available, and ultimately money available. Including this information will first allow a comparison of the highest objective function each year. Then the user can investigate which core technologies were always chosen as well as which were never chosen. Conclusions may be gathered from the changes in investment strategies as the start date and time frame changes.

The hypothesis for this question is that the earlier the start year the higher the objective value the portfolio will have. It is expected that the more time the portfolio has to develop technologies, the more technologies will be selected. Now this may be true for the objective values, but the real interesting changes come in the TOPSIS selection aspect. Just because a portfolio has the largest objective function does not make it the most desirable selection. By looking at a specific expected percentage input change and letting the start year and dependencies vary, the user can see how changing the start year affects their decisions.

6.5.0.3 *Research Question 4: Which has a larger impact on the Portfolio Selection: Input values or the Dependencies?*

Finally changing the values of the input values by some percentage will offer a change in the investment decisions. This question comes up when looking at how including dependency relationships result in the same objective function as having a different value input variable into the START environment. It comes down to the cross section of the change in Expected Percentage vs. Dependency axes. The information can be seen in a simplified way by looking at the objective function and TOPSIS value vs. the axis that is being studied. The interesting phenomena happens when adding dependencies is the same as changing an input value by a certain percentage. The question is does this happen, and if it does why?

The last research question deals with the search for truth. The fourth question is which has a larger impact on the portfolio recommendations: Accurate Input values or Dependency additions? This thesis assumes that there is some sort of optimized "Truth" recommendation. The optimization process that START uses looks for this truth and gives the highest portfolio according to the constraints given. The goal is a little different. The decision maker is not only looking for the highest objective value, they are also looking for the "truth". The answer is that the user should look for the highest objective function that is truthful.

This question looks to see if it is worth the effort to figure out the "best" values or use "estimates" and place more constraints on the problem. By creating information on three different planes, the user can slice the information into contour plots. The research question comes down to looking at the relationship between these axes with respect to their effects on the objective function.

Assume there is some sort of truth that the user seeks then they can also assume that there is a cushion around that truth that is good enough. This can be seen in 42.

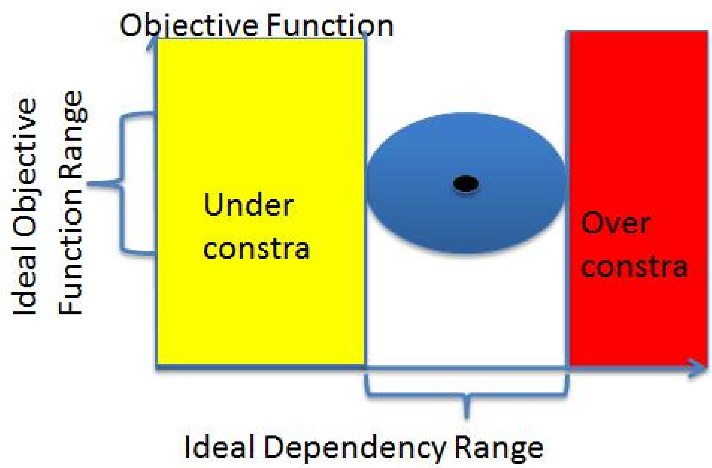


Figure 42: Searching for Truth in an Optimized Setting

The black dot represents the "truth", or optimized portfolio with all dependencies included and the best input values associated with the element. The blue circle around the truth is a cushion that the user will accept as a good representation of the best investment portfolio.

Here there is a line with which the problem is over constrained with too many dependencies as well as under constrained with not enough. Both lines shown encompass the ideal as well as the true portfolio. The user is looking for the truth, but will accept anything within the limits of the ideal cushion solution. The decision maker wants to be on the under constrained side so that START can actually solve the problem, while still giving a good representation of it.

Changing the focus to the expected input percentages, shows in Figure 43 the same 'truth' dot and cushion. In this case the yellow represents not enough information included in the portfolio recommendation.

The line is the switch over from inaccurate values to accurate values, but one can assume it is a range/region that encompasses the "truth cushion region". The user does not want inaccurate values optimized in START, but they do not want too optimistic values that give inaccurate portfolios as well. The user aims to be within a certain cushion that has a truth inherent in it, as well as a cushion that is good

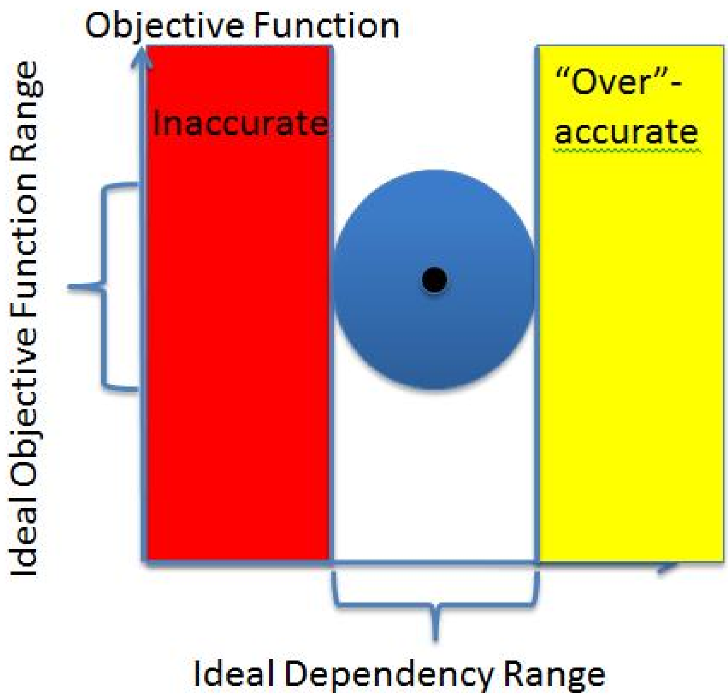


Figure 43: Searching for Truth in an Optimized Setting using constraints

enough for the user to accept the values.

The desired region is everything to the right of the inaccurate line in the truth cushion range. The acceptable region is the desired region as well as the "over"-accurate yellow section as well. In this case the yellow part represented "wasted effort" to get better values. Ideally the user wants to avoid being over constrained by dependencies and having inaccurate data values.

The idea of Question 4 is to find out how that is possible through the addition of dependencies or working "harder" to get more accurate data. Something that comes out of this is a top down approach. By changing the accuracy of input technologies this would give new technologies for goals. Therefore if the current technology is A, the analysis suggest that B is needed to find some threshold objective function, then everyone can go back to the drawing board to see if B is possible. If B is not possible, the idea is to see what is the highest objective function using A that can be made by taking into account dependencies.

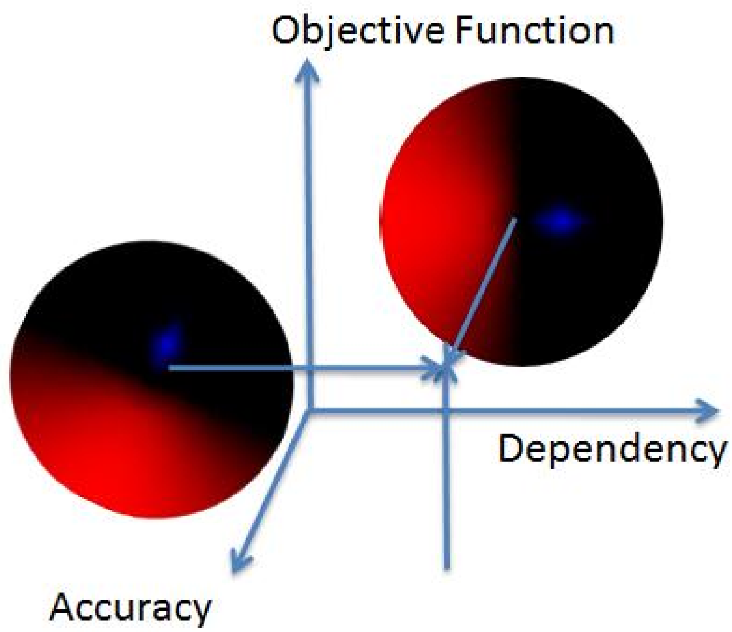


Figure 44: Sphere of influence around the 'truth' portfolio being sought.

Combining Figure 42 and Figure 43 gives a sphere of possibilities around the "truth". If this is continued to project the desired under constrained and accurate viewpoints we cut out a sliver of the sphere of desired portfolios. This can be seen in Figure 44.

These desired portfolios are accurate and precise with respect to dependencies. Landing in the other two accurate but not precise or precise but not accurate are better solutions than landing in the not precise AND not accurate slivers. This is the fidelity term the thesis is trying to measure.

6.5.0.4 Measuring Fidelity

Fidelity is the measure to which a model represents the real world. (Need a citation here!) Adding dependencies is one way to add fidelity into the process. Other ways include changing the accuracy and precision of the input values or the time period with which the process will optimize. The fidelity can be seen as a function of the expected input percentage and dependency values of the elements.

fidelity= $\int$(Expected Input Percentage,Dependency)

The fidelity is measured through looking at the various values put into the TOPSIS tool. These must be taken into account when looking at the accuracy and precision of the portfolios.

6.5.0.5 Accuracy and Precision

Looking closer at accuracy and precision shows a visual representation in Figure 45. Ideally the portfolios should be accurate and precise as shown in Figure 45c. The TOPSIS fidelity value output will cloister around the perceived "truth".

(a) Accurate but not Precise (b) Precise but not Accurate (c) Accurate and Precise

Figure 45: Accuracy and Precision Relationship

Ultimately the addition of dependencies and changing the expected input value is a way of changing the accuracy and precision. This reiterated in Table 10. As the expected input value increases, the accuracy increases. Meanwhile as the dependencies and fidelity increases the precision increases. This is analogous to the target cloister in Figure 45c, moving around the bulls eye as in Figure 45b. Table 10 shows that the four cases are different combinations of moving around the targeted or "truthful" technology portfolio. By changing the input aspects, the model tests the changing target scenarios and shows how the fidelity changes as it moves from one block to the next.

Table 10: Accuracy and Precision dispositions

		Precision/Dependency	
		-	+
Accuracy/ Expected Input value	-	Low Accurate but not precise	Moderate Precise but not accurate
	+	Moderate Precise but not accurate	High Precise and Accurate

6.5.0.6 Realities

The reality is that the user does not know the "truth", nor do they know what the cushion around the truth looks like (e.g. it could be square or some other shape). There are other factors of fidelity that are not included here, but can warp the "truth" value. Looking at the contour plots of the objective function versus Dependency and Accuracy will give an idea of where the "truest" and "highest" objective function value lies. The decision maker can take the derivative of the values surrounding this function of "truth" and "highest" and compare these two values. Comparing the derivatives will give an idea of the "ease" of putting higher fidelity into the model. It will be interesting if the "truth" and "highest value" portfolios are in the same region, completely opposite of each other, or right next to each other. At some point adding dependencies and higher accuracy must yield the same value and that's where the interesting design space regions lie and should be analyzed.

6.5.0.7 Partial Funding

Partial funding is a setting that START has. Partial funding means that the entire elemental settings as put in the process will not be funded. However, some partial fraction of the element will be funded. This has the same effect on the objective value as well as the funding level. It does not affect the schedule the same way. The funding period will not change, but the subsequent cost and value will change according to the partial funding level specified by the user.

Table 11: Floor Branch Selection Setting

	Default			Greed			No Greed		
Dependency	Obj. Func	Cost	ROI	Obj. Func	Cost	ROI	Obj. Func	Cost	ROI
0	259.05	5201.64	0.0498	259.05	5201.64	0.0498	259.05	5201.64	0.0498
1	241.64	6282.06	0.0385	241.64	6282.06	0.0385	249.74	6445.96	0.0387
2	242.31	4640.53	0.0522	242.31	4640.53	0.0522	243.51	5058.03	0.0481
3	242.06	4866.76	0.0497	242.05	4866.76	0.0497	242.99	5172.08	0.0470

6.5.0.8 LP-Solve Settings: Greed vs. No Greed

The LP Solve method has a setting called the greedy mode. Greedy mode means that the solution looks for the local optimum. This method quickly finds the optimum, by cutting off the other branch from the branch and bound method. Once it finds a local optimum, it cuts the problem in half. When this happens, the global optimum can be lost. Greedy algorithms are acceptable for some types of problem. However, in order to find the characteristics of adding dependencies the greedy algorithm was taken out of the lp-solve selection. This caused longer run times, but gave optimal results. A comparison of the results can be seen in Table 11. The highest objective value was with the No Greed Mode.

6.5.0.9 Branch Selection

The LP Solve function is a branch and bound selection algorithm. There are three settings for which branch the lp-solve function will choose to look at first. The three selections are the floor, ceiling and automatic. Since the selection is not using the greedy algorithm, it now has the choice to look at both branches. When looking at the floor setting, lp-solve looks at the lower branch first as seen in Table 11. The automatic setting can be seen in Table 12. The difference here is quite substantial. Utilizing the automatic branch selection gives higher values with all of the dependencies included in the process. Comparing the automatic selection method with the floor method, shows that the baseline of the floor selection setting is lower than including all of the dependency settings of the automatic selection criteria.

Table 12: Automatic Branch Selection Setting

	Default			Greed			No Greed		
Dependency	Obj. Func	Cost	ROI	Obj. Func	Cost	ROI	Obj. Func	Cost	ROI
0	303.27	7369.74	0.041	303.27	7369.74	0.041	303.20	7429.01	0.041
1	283.54	7386.30	0.038	283.54	7386.30	0.038	281.77	6882.97	0.041
2	300.68	7114.84	0.042	300.68	7114.84	0.042	298.79	7107.34	0.042
3	268.56	7347.12	0.037	268.56	7347.12	0.037	282.63	7530.59	0.038

6.6 Sample Problem

In order to go through the full methodology a sample problem will be shown to show the changes that the dependencies and MIMO will implement. The START input file will have two columns. The first column will consist of the strings of values that will be involved with the dependency that will be applied. If newer types of dependencies are found it will be easy to implement them into the program source. There are two types of information in addition to the actual dependency values that are being equated. The first is what type of dependency: Constraint or Value. Once this is determined the next step is to determine where it is being applied: value, time, or cost. At this point on value dependencies were implemented. There are other classes of dependencies that are possible such as cost and scheduling dependencies. Once this is determined the actual values that are then used in the input values. This is shown in Figure 46.

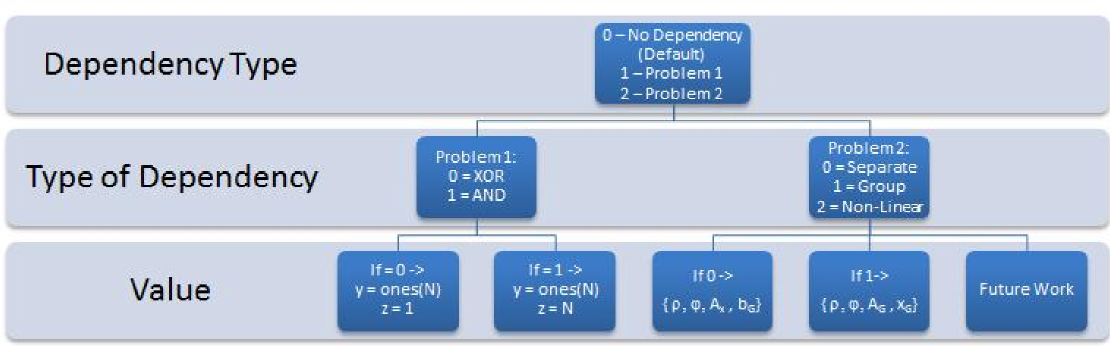


Figure 46: Input Value Format

Table 13: Sample Problem Element Setup

Element	Number of Elements
Mission	3
Capability	6
Technology	18

This formatting will allow for multiple inputs for the program so that the fidelity may be increased in a variety of ways later on.

6.7 Sample Results

The user can run START with dependencies and it may not choose to activate the dependency. Meaning that both elements may not be chosen and this does not violate the constraints set forth in the dependency case table above. Going through a sample problem will show the effect of the dependencies on the portfolio. The sample problem is set up as described in Table 13.

This brings us to a total 18 total technologies for the three missions. Everything has generic numbers and values and running this information as a baseline gives Table 14. This translates into a strategic area graph as shown in Figure 47. Looking on the mission level shows Figure 48. This shows that Mission 1 is chosen. The objective function is 59.38. All of the capabilities for Mission 1 and Mission 2 are chosen in the baseline portfolio. This is a perfect way to put dependencies on Mission 3's mission and technology level.

Table 14: Baseline Data Results for Generic Problem

Strategic Area	Capability Area	Metric	Mission	Selection	Objective Value	Partial Funding
Capability A	Technology A.1	Metric A.1.1	Mission 1	Selected	5.872499	0.8
Capability A	Technology A.2	Metric A.2.1	Mission 1	Selected	5.872499	0.8
Capability B	Technology B.1	Metric B.1.1	Mission 1	Selected	5.872499	0.8
Capability B	Technology B.2	Metric B.2.1	Mission 1	Selected	5.872499	0.8
Capability C	Technology C.1	Metric C.1.1	Mission 1	Selected	5.872499	0.8
Capability C	Technology C.2	Metric C.2.1	Mission 1	Selected	5.872499	0.8
Capability A	Technology A.1	Metric A.1.1	Mission 2	Selected	3.138186	0.7
Capability A	Technology A.2	Metric A.2.1	Mission 2	Selected	3.138186	0.7
Capability B	Technology B.1	Metric B.1.1	Mission 2	Selected	3.138186	0.7
Capability B	Technology B.2	Metric B.2.1	Mission 2	Selected	3.138186	0.7
Capability C	Technology C.1	Metric C.1.1	Mission 2	Selected	3.138186	0.7
Capability C	Technology C.2	Metric C.2.1	Mission 2	Selected	3.138186	0.7
Capability A	Technology A.1	Metric A.1.1	Mission 3	Selected	2.110255	0.6
Capability A	Technology A.2	Metric A.2.1	Mission 3	Selected	2.110255	0.6
Capability B	Technology B.1	Metric B.1.1	Mission 3	Selected	2.110255	0.6
Capability B	Technology B.2	Metric B.2.1	Mission 3	Selected	2.110255	0.6
Capability C	Technology C.1	Metric C.1.1	Mission 3	Selected	2.110255	0.6

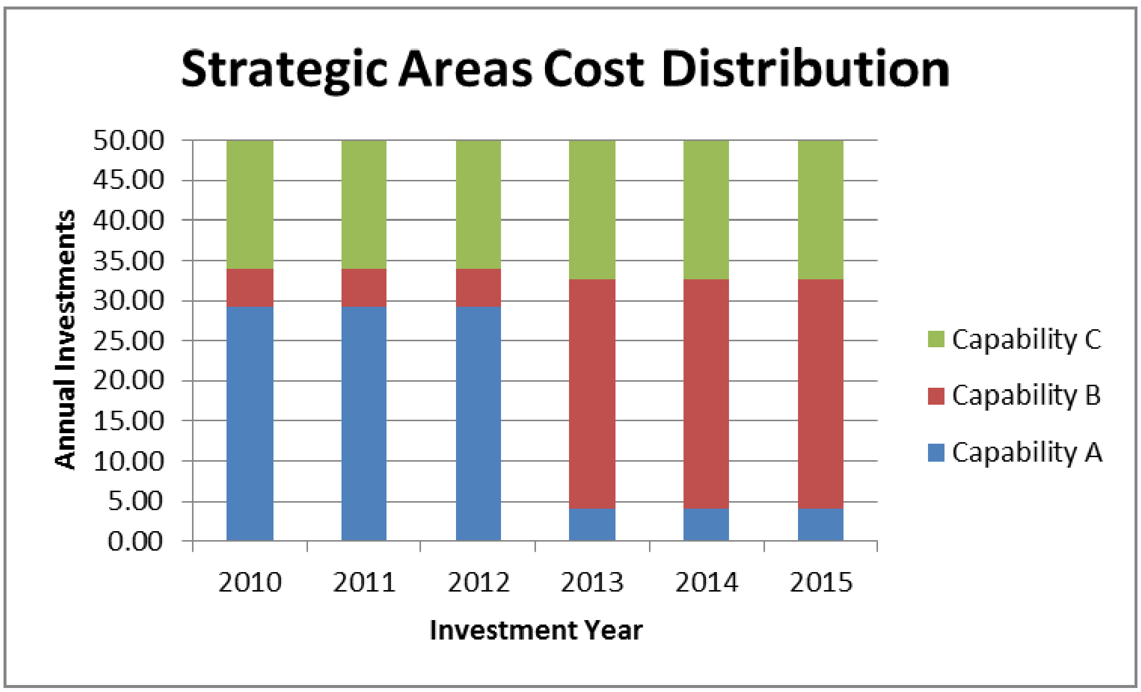


Figure 47: Sample Problem Baseline Strategic Area Cost Distribution

6.7.1 Mission Dependency

This section will go through the changes when mission dependencies are added. Mission 1 needs Mission 3 and Mission 1 or Mission 2.

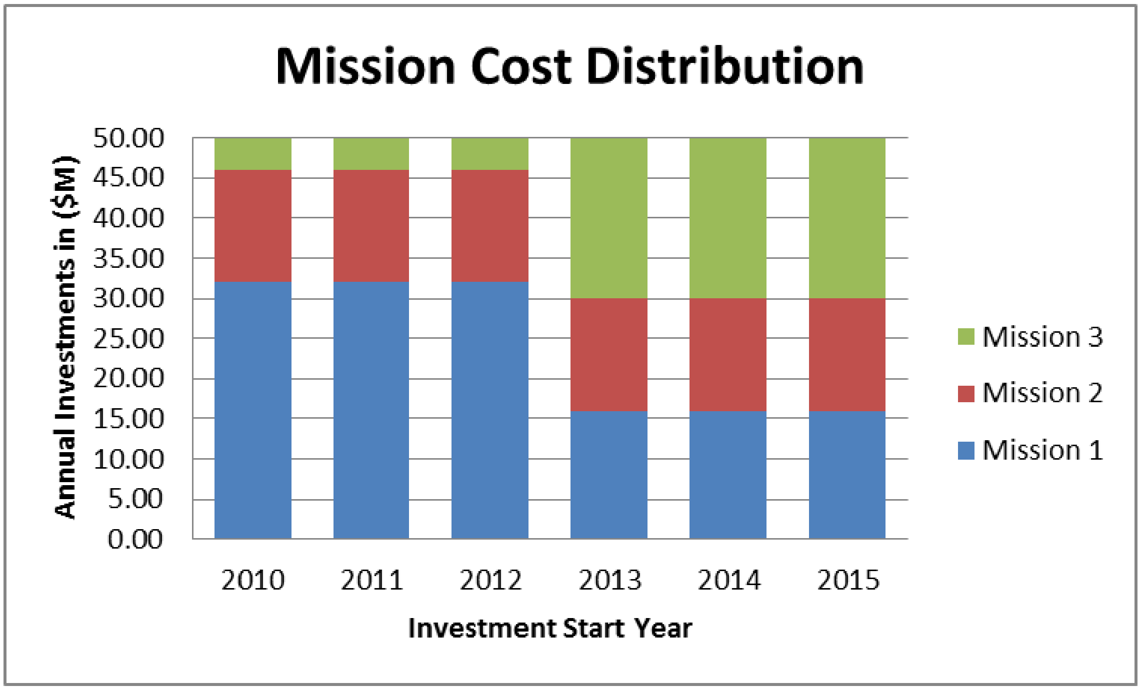


Figure 48: Baseline Mission Cost Distributions

6.7.1.1 *Mission 1 needs Mission 3*

If we put a dependency of Mission 1 needs Mission 3 in order to be valid. Placing this dependency shows the figure given below. The objective function is 59.39 which is a 0% change from the baseline figure above. This case is strange. Literally the Mission 3 component was chosen, but no capabilities were funded with it.

In this case Mission 2 was completely not chosen throughout the optimization.

Table 15: Baseline Data Results for Generic Problem

Strategic Area	Capability Area	Metric	Mission	Selection	Objective Value	Partial Funding
Capability A	Technology A.1	Metric A.1.1	Mission 1	Selected	5.872499	0.8
Capability A	Technology A.2	Metric A.2.1	Mission 1	Selected	5.872499	0.8
Capability B	Technology B.1	Metric B.1.1	Mission 1	Selected	5.872499	0.8
Capability B	Technology B.2	Metric B.2.1	Mission 1	Selected	5.872499	0.8
Capability C	Technology C.1	Metric C.1.1	Mission 1	Selected	5.872499	0.8
Capability C	Technology C.2	Metric C.2.1	Mission 1	Selected	5.872499	0.8
Capability A	Technology A.1	Metric A.1.1	Mission 2	Selected	3.138186	0.7
Capability A	Technology A.2	Metric A.2.1	Mission 2	Selected	3.138186	0.7
Capability B	Technology B.1	Metric B.1.1	Mission 2	Selected	3.138186	0.7
Capability B	Technology B.2	Metric B.2.1	Mission 2	Selected	3.138186	0.7
Capability C	Technology C.1	Metric C.1.1	Mission 2	Selected	3.138186	0.7
Capability C	Technology C.2	Metric C.2.1	Mission 2	Selected	3.138186	0.7
Capability A	Technology A.1	Metric A.1.1	Mission 3	Selected	2.110255	0.6
Capability A	Technology A.2	Metric A.2.1	Mission 3	Selected	2.110255	0.6
Capability B	Technology B.1	Metric B.1.1	Mission 3	Selected	2.110255	0.6
Capability B	Technology B.2	Metric B.2.1	Mission 3	Selected	2.110255	0.6
Capability C	Technology C.1	Metric C.1.1	Mission 3	Selected	2.110255	0.6
Capability C	Technology C.2	Metric C.2.1	Mission 3	Selected	2.110255	0.6

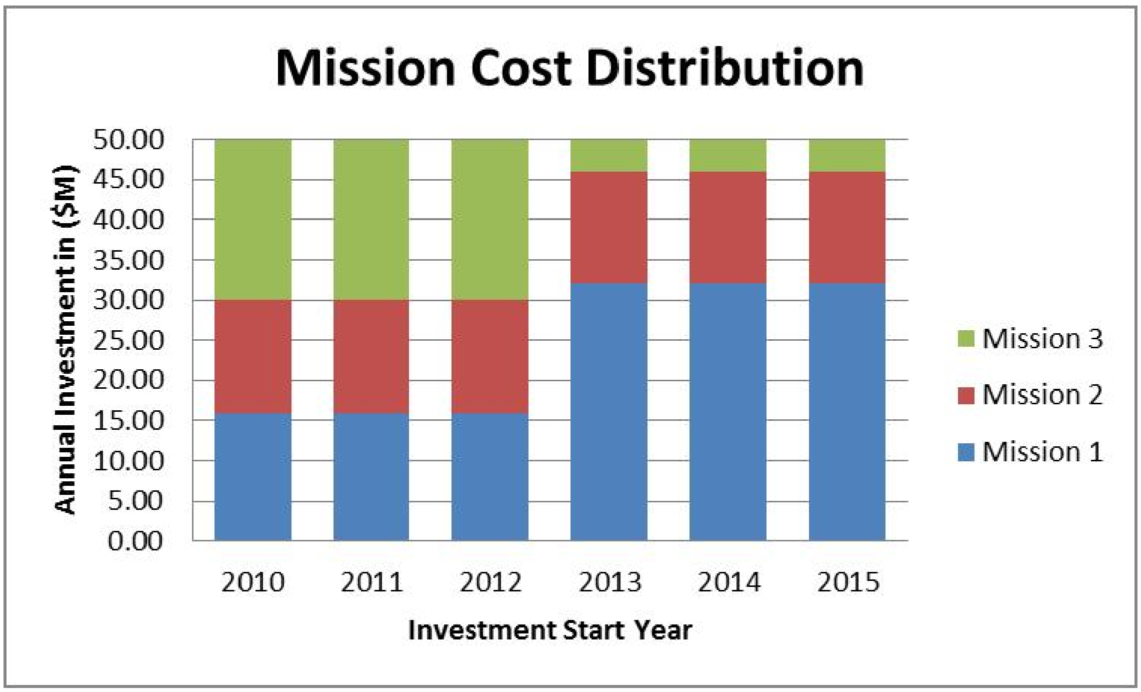


Figure 49: Mission Cost Distribution Mission 1 Needs Mission 3 Example

6.7.1.2 *Mission 1 or Mission 2*

Mission 1 and Mission 2 were chosen for the baseline. By putting an OR dependency between Mission 1 and Mission 2, this completely changes the dynamic of the problem. This time it shows Figure 51. This change chose Mission 1 and Mission 3. The

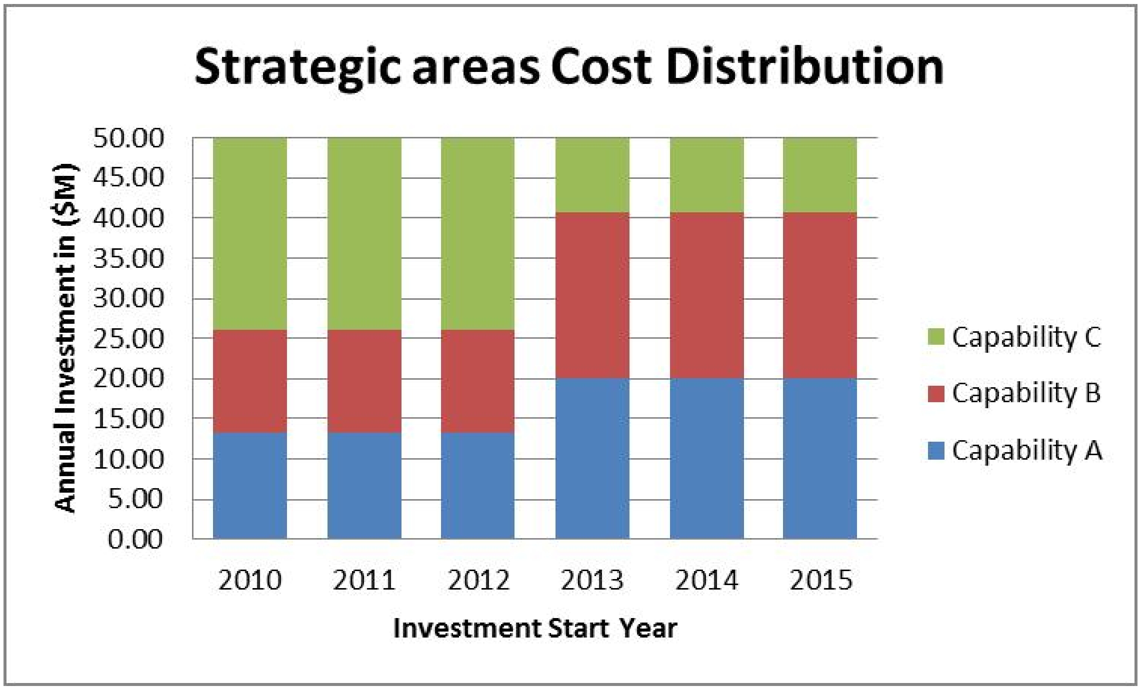


Figure 50: Strategic Area Cost Distribution Mission 1 Needs Mission 3 Example

objective function was 53.99 which is a 9.1% decrease from the baseline figure shown above. By just accepting the baseline strategy, the user would have chosen Mission 2 instead of Mission 3 because they did not model the OR dependency. This results in a decrease of objective function by around 10%, but it does result in an accurate relationship modeling. The user knows to choose Mission 1 over Mission 2.

Table 16: Generic Baseline Portfolio Selection

Strategic Area	Capability Area	Metric	Mission	Selection	Objective Value	Partial Funding
Capability A	Technology A.1	Metric A.1.1	Mission 1	Selected	6.298719	1
Capability A	Technology A.2	Metric A.2.1	Mission 1	Selected	6.298719	1
Capability B	Technology B.1	Metric B.1.1	Mission 1	Selected	6.298719	1
Capability B	Technology B.2	Metric B.2.1	Mission 1	Selected	6.298719	1
Capability C	Technology C.1	Metric C.1.1	Mission 1	Selected	6.298719	1
Capability C	Technology C.2	Metric C.2.1	Mission 1	Selected	6.298719	1
Capability A	Technology A.1	Metric A.1.1	Mission 2	NOT Selected		
Capability A	Technology A.2	Metric A.2.1	Mission 2	NOT Selected		
Capability B	Technology B.1	Metric B.1.1	Mission 2	NOT Selected		
Capability B	Technology B.2	Metric B.2.1	Mission 2	NOT Selected		
Capability C	Technology C.1	Metric C.1.1	Mission 2	NOT Selected		
Capability C	Technology C.2	Metric C.2.1	Mission 2	NOT Selected		
Capability A	Technology A.1	Metric A.1.1	Mission 3	Selected	2.699451	1
Capability A	Technology A.2	Metric A.2.1	Mission 3	Selected	2.699451	1
Capability B	Technology B.1	Metric B.1.1	Mission 3	Selected	2.699451	1
Capability B	Technology B.2	Metric B.2.1	Mission 3	Selected	2.699451	1
Capability C	Technology C.1	Metric C.1.1	Mission 3	Selected	2.699451	1
Capability C	Technology C.2	Metric C.2.1	Mission 3	Selected	2.699451	1

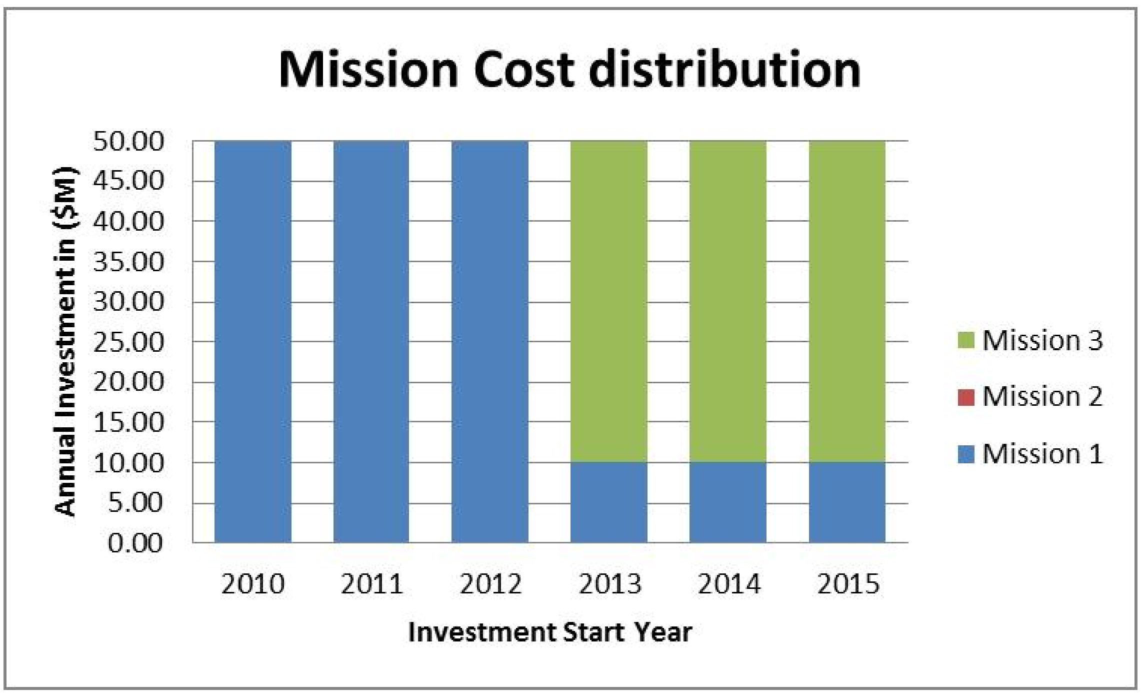


Figure 51: A OR B Generic Problem Demonstration

6.7.2 Capability Dependency

6.7.2.1 Technology 1 needs Technology 2

In this case Technology 1 from Mission 1 needs Technology 2 from Mission 3. As you can see from the figures below, all three capabilities are chosen. However, you can see

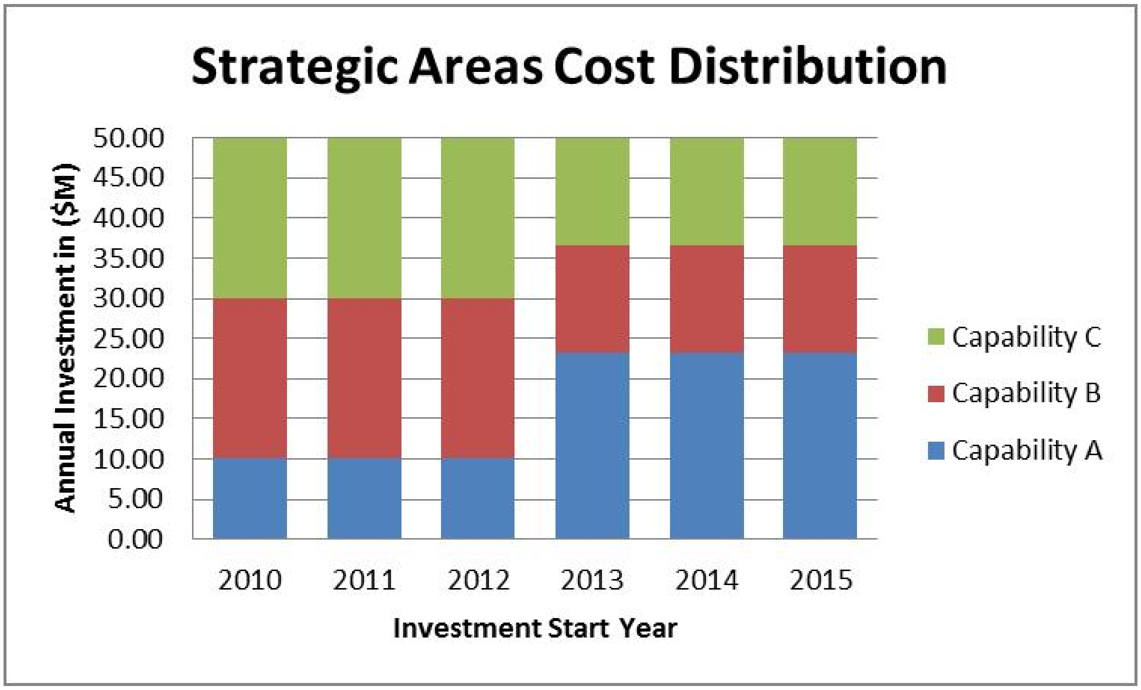


Figure 52: A OR B Generic Dependency Demonstration Mission cost Distributions

from the Mission Cost Distribution graph in , that Technology 2 from Mission 3 was chosen which is why there is a Mission 3 funded part in this graph. The objective function here is 55.79, which is a 6.1% decrease from the original baseline function. In this case, by modeling this relationship, there is only a 6% penalty compared to the 9% penalty from the mission dependency modeled above. However, had this not been modeled, Technology 1 from Mission 1 would have been chosen in he baseline, but not Technology 2 from Mission 3 which it depends upon. Modeling this relationship gave an accurate account of the requirements needed.

Table 17: Capability Dependency Technology 1 needs Technology 2 example problem

Strategic Area	Capability Area	Metric	Mission	Selection	Objective Value	Partial Funding
Capability A	Technology A.1	Metric A.1.1	Mission 1	Selected	5.872499	0.8
Capability A	Technology A.2	Metric A.2.1	Mission 1	Selected	5.491826	0.7
Capability B	Technology B.1	Metric B.1.1	Mission 1	Selected	5.872499	0.8
Capability B	Technology B.2	Metric B.2.1	Mission 1	Selected	5.491826	0.7
Capability C	Technology C.1	Metric C.1.1	Mission 1	Selected	5.872499	0.8
Capability C	Technology C.2	Metric C.2.1	Mission 1	Selected	5.872499	0.8
Capability A	Technology A.1	Metric A.1.1	Mission 2	Selected	3.138186	0.7
Capability A	Technology A.2	Metric A.2.1	Mission 2	Selected	3.138186	0.7
Capability B	Technology B.1	Metric B.1.1	Mission 2	Selected	3.138186	0.7
Capability B	Technology B.2	Metric B.2.1	Mission 2	Selected	3.355714	0.8
Capability C	Technology C.1	Metric C.1.1	Mission 2	Selected	3.138186	0.7
Capability C	Technology C.2	Metric C.2.1	Mission 2	Selected	3.138186	0.7
Capability A	Technology A.1	Metric A.1.1	Mission 3	Selected	2.35364	0.7
Capability A	Technology A.2	Metric A.2.1	Mission 3	Selected	2.35364	0.7
Capability B	Technology B.1	Metric B.1.1	Mission 3	Selected	2.110255	0.6
Capability B	Technology B.2	Metric B.2.1	Mission 3	Selected	2.110255	0.6
Capability C	Technology C.1	Metric C.1.1	Mission 3	Selected	2.110255	0.6
Capability C	Technology C.2	Metric C.2.1	Mission 3	Selected	2.110255	0.6

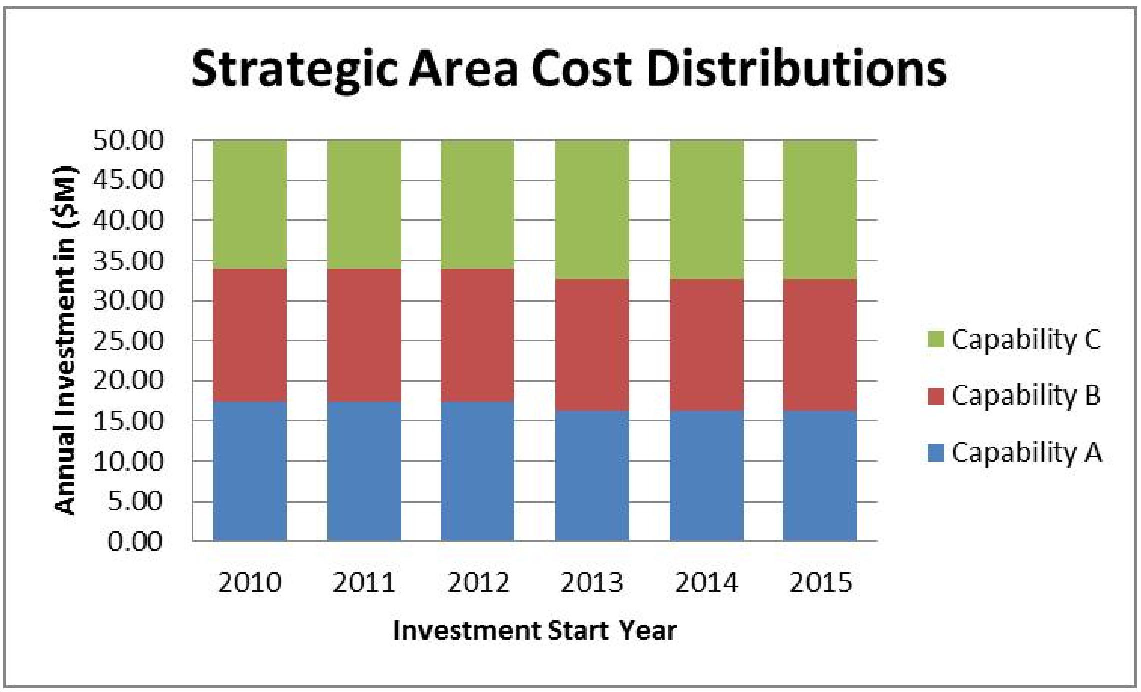


Figure 53: Capability Dependency Demonstration for Strategic area Cost distribution

6.7.2.2 *Technology 1 or Technology 2*

This last case models that we must choose Technology 1 or Technology 2 from the same capability. In this case, you cannot see the direct affect from the graphs given below. The outcome is that Technology 2 was not selected. The table below comes

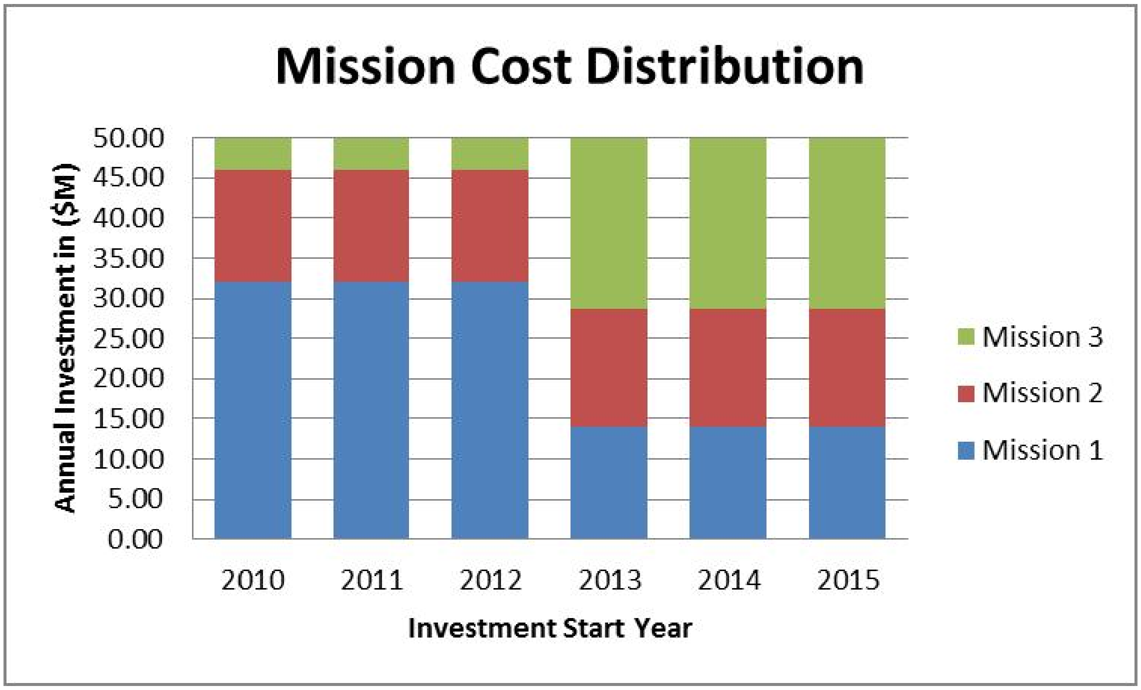


Figure 54: Capability Dependency Demonstration for Mission Cost Distributions

from the output excel file of the solution. This shows that Capability A from Mission 2 was not chosen, compared to the baseline where it was chosen.

The objective function is 55.79 which is a 6.1% decrease from the baseline. In this case it was the same decrease in objective function as the above case where Technology 1 needs Technology 3. Now this is not going to always be the case. This is an example just to show that the dependencies will change the investment decision. However, there will be times where a different dependency relationship modeling will give the same change in objective function. It is basically saying that some relationships are equivalent to others being modeled in the process.

Table 18: Capability A or B Dependency demonstration data collection

Strategic Area	Capability Area	Metric	Mission	Selection	Objective Value	Partial Funding
Capability A	Technology A.1	Metric A.1.1	Mission 1	Selected	5.872499	0.8
Capability A	Technology A.2	Metric A.2.1	Mission 1	Selected	5.872499	0.8
Capability B	Technology B.1	Metric B.1.1	Mission 1	Selected	5.872499	0.8
Capability B	Technology B.2	Metric B.2.1	Mission 1	Selected	5.872499	0.8
Capability C	Technology C.1	Metric C.1.1	Mission 1	Selected	5.872499	0.8
Capability C	Technology C.2	Metric C.2.1	Mission 1	Selected	5.872499	0.8
Capability A	Technology A.1	Metric A.1.1	Mission 2	Selected	3.355714	0.8
Capability A	Technology A.2	Metric A.2.1	Mission 2	Selected	3.138186	0.7
Capability B	Technology B.1	Metric B.1.1	Mission 2	Selected	3.138186	0.7
Capability B	Technology B.2	Metric B.2.1	Mission 2	NOT Selected		
Capability C	Technology C.1	Metric C.1.1	Mission 2	Selected	3.138186	0.7
Capability C	Technology C.2	Metric C.2.1	Mission 2	Selected	3.138186	0.7
Capability A	Technology A.1	Metric A.1.1	Mission 3	Selected	2.35364	0.7
Capability A	Technology A.2	Metric A.2.1	Mission 3	Selected	2.35364	0.7
Capability B	Technology B.1	Metric B.1.1	Mission 3	Selected	2.35364	0.7
Capability B	Technology B.2	Metric B.2.1	Mission 3	Selected	2.35364	0.7
Capability C	Technology C.1	Metric C.1.1	Mission 3	Selected	2.35364	0.7
Capability C	Technology C.2	Metric C.2.1	Mission 3	Selected	2.35364	0.7

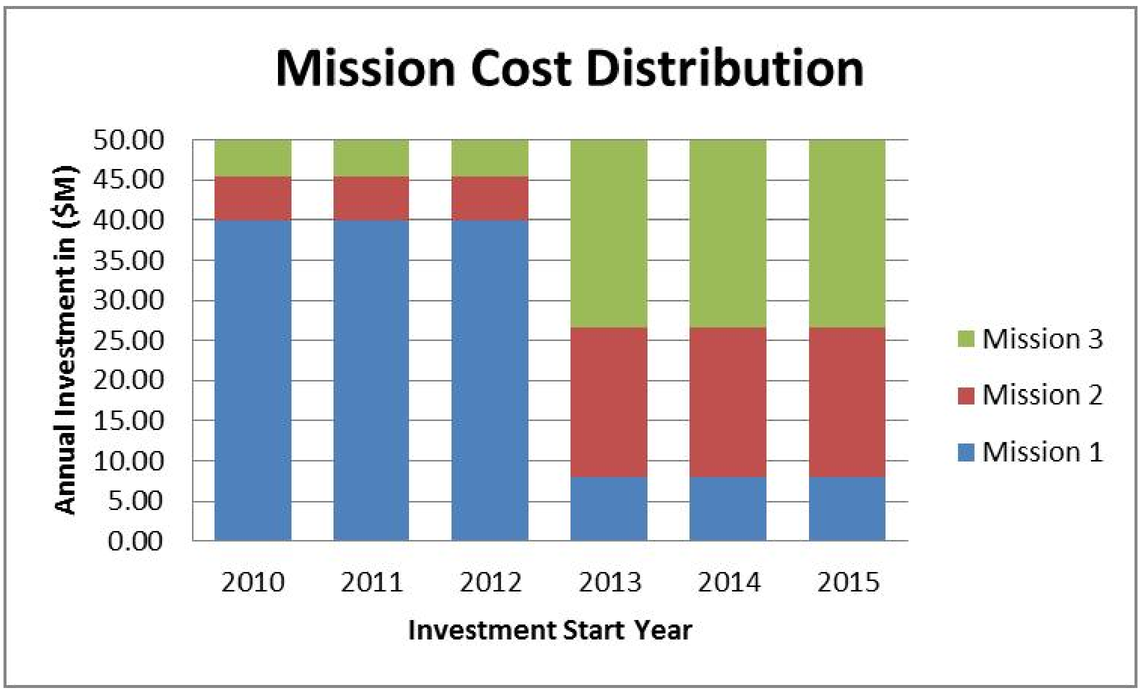


Figure 55: Capability A OR B Mission Cost Distributions

6.7.3 MIMO Dependency

MIMO only works for technologies for the moment, but may be used for capabilities eventually. There are two values that must be included for MIMO: psi and rho. For this example, $\varphi = 2$ and $\rho = 20$.

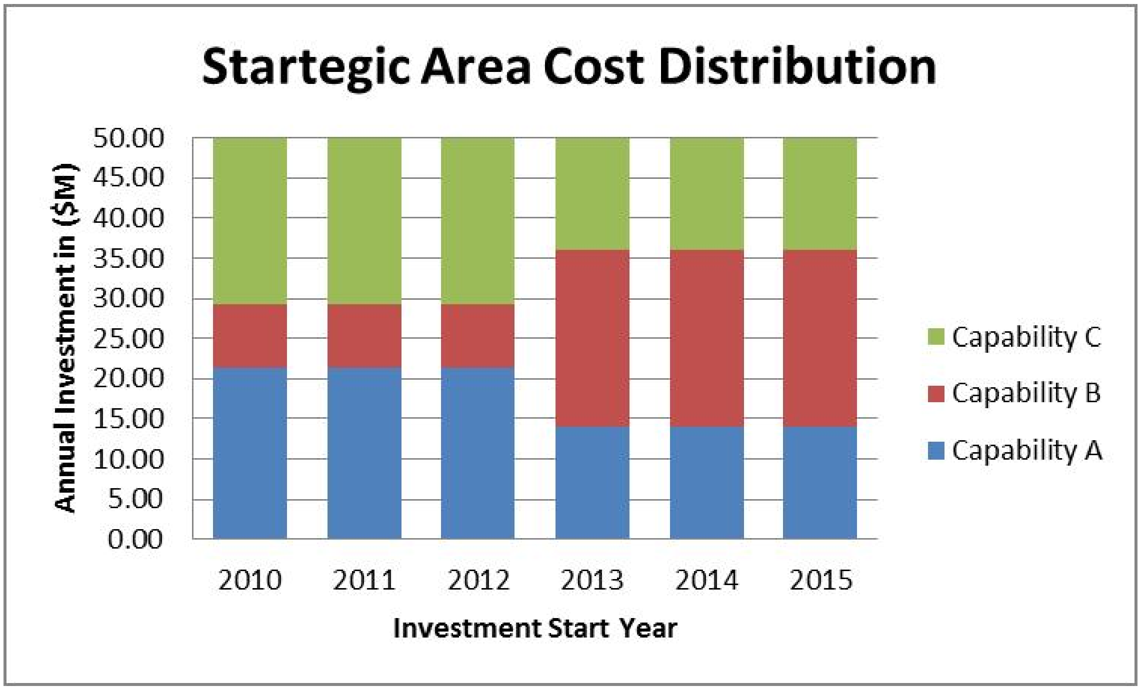


Figure 56: Capability A OR B

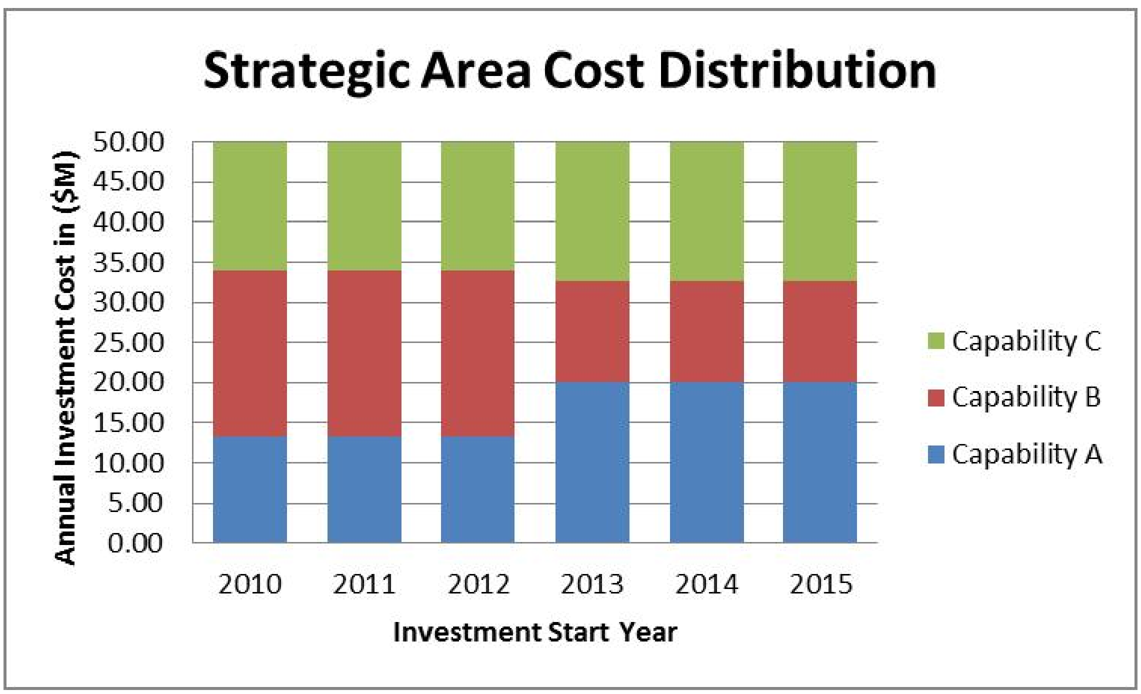


Figure 57: Strategic Area Cost Distribution for MIMO Dependency Example

6.8 Final Methodology:

Bringing all of these elements together to answer the research questions result in a methodology that includes dependencies, gives information regarding the optimized

Table 19: MIMO Dependency Data Collection

Strategic Area	Capability Area	Metric	Mission	Selection	Obj. Value	Partial Funding
Capability A	Technology A.1	Metric A.1.1	Mission 1	Selected	5.872671	0.8
Capability A	Technology A.2	Metric A.2.1	Mission 1	Selected	5.872499	0.8
Capability B	Technology B.1	Metric B.1.1	Mission 1	Selected	5.872499	0.8
Capability B	Technology B.2	Metric B.2.1	Mission 1	Selected	5.872499	0.8
Capability C	Technology C.1	Metric C.1.1	Mission 1	Selected	5.872499	0.8
Capability C	Technology C.2	Metric C.2.1	Mission 1	Selected	5.872499	0.8
Capability A	Technology A.1	Metric A.1.1	Mission 2	Selected	3.138186	0.7
Capability A	Technology A.2	Metric A.2.1	Mission 2	Selected	3.138186	0.7
Capability B	Technology B.1	Metric B.1.1	Mission 2	Selected	3.138186	0.7
Capability B	Technology B.2	Metric B.2.1	Mission 2	Selected	3.138186	0.7
Capability C	Technology C.1	Metric C.1.1	Mission 2	Selected	3.138186	0.7
Capability C	Technology C.2	Metric C.2.1	Mission 2	Selected	3.138186	0.7
Capability A	Technology A.1	Metric A.1.1	Mission 3	Selected	2.110255	0.6
Capability A	Technology A.2	Metric A.2.1	Mission 3	Selected	2.110255	0.6
Capability B	Technology B.1	Metric B.1.1	Mission 3	Selected	2.110255	0.6
Capability B	Technology B.2	Metric B.2.1	Mission 3	Selected	2.110255	0.6
Capability C	Technology C.1	Metric C.1.1	Mission 3	Selected	2.110255	0.6
Capability C	Technology C.2	Metric C.2.1	Mission 3	Selected	2.110255	0.6

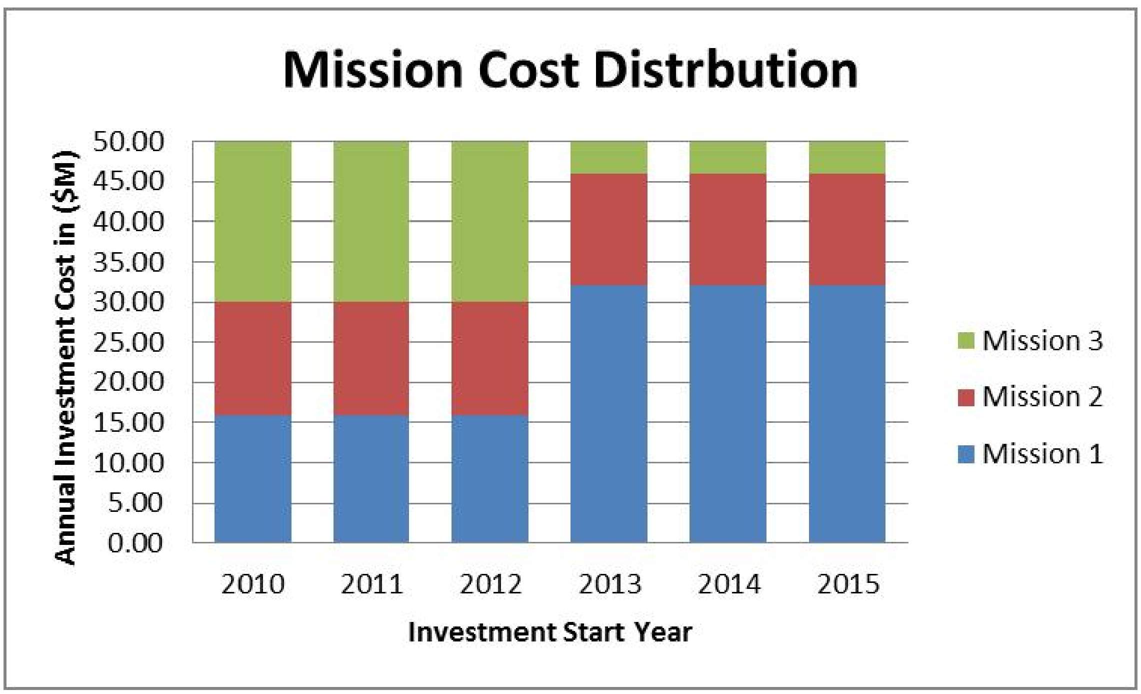


Figure 58: Mission Cost Distribution for MIMO Dependency Example

solution given different conditions of expected input value percentages, the best starting year to invest in technologies and higher fidelity as to what dependencies to include in the process. The final methodology is for the design space and ultimately analyze it.

Step 1: Background Literature review

The user must first start with a background literature review to see what technologies are available. This information must cover the essential input values as well as any possible dependencies and interactions the user would like to model.

Step 2: Infuse Flexibility and Extensibility into the input file

Once a background literature review has been done, the user must start to build the database. This must include the input information as well as the fidelity and extensibility terms populated from industry experience. Adding the flexibility into the project gives the technology another metric to measure in the fidelity process. These terms are used later for the Visualization TOPSIS tool.

Step 3: Integrate dependencies into the file

Two dependencies have been introduced in this thesis. The user must determine the level of the dependency (Mission vs. Capability) and the type of dependency. If the dependency is a Constraint Dependency, then the use must add appropriate constraints between the elemental aspects in their technology portfolio process. These constraints must be in line with the logic table given in Table **??**. If the dependencies are Value Dependencies, then the user has the option to use MIMO between capabilities. This requires that the user needs to know the origin of the dependency element and the performance metric change. Once the MIMO coefficient is performed, the user may now change the objective value coefficient. In the future if other dependency categories are identified and modeled, then the coefficient change may not necessarily be in the objective value. It may be that other categories change the constraint coefficients. Step 3 requires that the user apply the desired dependency categories in preparation to run and evaluate the design space.

Step 4: Run the baseline and any dependency sets

The Technology Portfolio process has been set up in Steps 1-3. In Step 4, the baseline is first run. The baseline is the portfolio results without any dependencies; however, it does include the flexibility aspect inherent in the input file from Step 2. The baseline may not be the final decision or highest fidelity portfolio, but it does give a reference design space. The next run should include the dependency design space. For the purpose of this research, multiple axes are run to study the impact of dependencies and run to through the Visualization tool. It may be the case that the user will only have two portfolios to compare (Baseline and Dependency) where they would not need the visualization tool in this case. In that case, they would need TOPSIS to compare the two technologies. This is useful when looking at how to value dependencies. Constraint Dependencies are straight forward to their value. Value Dependencies are not necessarily as apparent. Measuring their objective value as well as their TOPSIS value is a valuable tool to quantify the quality of the dependencies are being considered.

Step 5: Analyze the Results

The User has a variety of analysis that they may conduct to explore the design space created. The analysis given below are possible design space analysis available to the user.

Bottom Up

A Bottom Up analysis will be used to look at the highest possible outcome of any particular dependency inclusion set to see what the highest value possible using the current values. This type of analysis shows the user where they will be in a given time frame for a given portfolio. A bottom up can be combined with a top down analysis to determine the gap between where the user expects to be and where they will end up due to their current trajectory.

Top Down

The Top down analysis will be used to look at highest possible objective function and see what change in input percentages result in this value. The top down analysis will help determine the necessary technology portfolio now to get a perceived change. The top down analysis is performed by running the baseline and the dependency space to see the outcome from the bottom up analysis. From there, the user can change the inputs based on a Pareto Frontier to determine the highest contributing development efforts. Once these inputs are changed, a design of experiments should be run to systematically determine the best technology portfolio investments to arrive at the desired future point.

Temporal

The temporal analysis will look at the change in start times for the problem. Looking at the start time vs. the change in input percentages will give the optimized start time that will allow the user to get the best use of their investment techniques.

Feed Forward Gap Analysis

The feed forward gap analysis will be done to see what percentage of technology demonstration is best in earlier missions to benefit the last mission. The use of Constraint dependencies based on the flexibility ratings can give both a Feed-Forward and Feed-Backward analysis. This will not be looked at in this portfolio; however, it is a possible outcome from the inclusion of dependencies. **Step 6: Conclusions**
The user will take the analysis they performed in Step 5 to answer their desired inquires. For this thesis this last step concludes the results as to answer the four questions directly through the ten cases studies presented through the research and the different analysis sections presented in the past four steps.

6.9 Chapter Layout

Chapter 7 will take care of Step 1: Background literature review of the NEA Problem. Chapter 8 will go through building the database, run START for the entire design space and visualize the work with the START Visualization Tool. Chapter 9-12 will show 5 different NEA campaign cases to exhibit result to announce the research questions. Chapter 13 will make conclusions for this methodology as well as go into potential future works that could be included for follow up research.

CHAPTER VII

NEA BACKGROUND AND LITERATURE REVIEW

7.1 Introduction

Near Earth Asteroids (NEAs) are earth threats that concern all space and defense agencies. The threats are real and serve as a current initiative and possible NASA program. A nominal campaign would consists of three different missions over a twenty year period that would first send spacecraft to survey NEAs in the near vicinity, then send a rover to collect more data and ultimately send a human exploration mission to an asteroid. The NEA Campaign is seen as a stepping stone to Mars since it would demonstrate deep space human capabilities. This campaign is in its infancy and presents multiple high level dependencies that may be leveraged in the early stages to demonstrate this methodology.

This relationship of utilizing previous missions to set up future missions to accomplish a campaign is a perfect case study of dependencies in action. Each mission will adopt technology to build upon the information obtained from the previous mission. The technology selection for each mission is information that may be programmed into the inputs and ran utilizing dependencies. This approach puts flexibility into the programs utilizing the greater fidelity of the tool instead of limiting it to the inputs and users only.

The top level dependencies can be shown here since it is in its infancy. This does not provide a comprehensive list of all dependencies but it does provide valid ones for the beginning of the process.

7.1.1 NEA Campaign Origins

In 2006 NASA announced the launch of Constellation.[21] Originally this thesis was going to work on Constellation dependency data when proposed in 2009. In 2009 the Augustine Commission stated that "[the] U.S. human spaceflight program appears to be on an unsustainable trajectory." [1] It recommended a "flexible path" with "multiple destinations." Constellation was then canceled to that avail in August of 2010.

In response to this NASA headed in a new direction that adhered to that commission by suggesting the Near Earth Asteroid Program.

> If the Constellation program for lunar flights is terminated, in its place NASA would strengthen science and technology research and study options for human flights to asteroids and the moons of Mars. The new direction was made official on April 15, 2010 when President Obama spoke about his vision for space exploration, calling specifically for a human visit to a NEA by 2025. [58]

The new approach was implemented in the President's proposed NASA budget for FY11, which would (if approved by Congress) substitute a NEA for the Moon as the next target for human exploration. [1] With a new direction towards a NEA program instead of Constellation, NASA set upon designing a campaign that would accomplish the President's vision.

Each mission would have specific objectives that it must accomplish in order to contribute to the overall campaign. These specific objectives are summed up in Table 20. These objectives are used as a basis to decipher dependencies between missions. Each mission must accomplish these specific objectives in order to enable the next mission to launch. It is notable that the number of objects changes from mission to mission which is a function of the fact that initially there are no known suitable

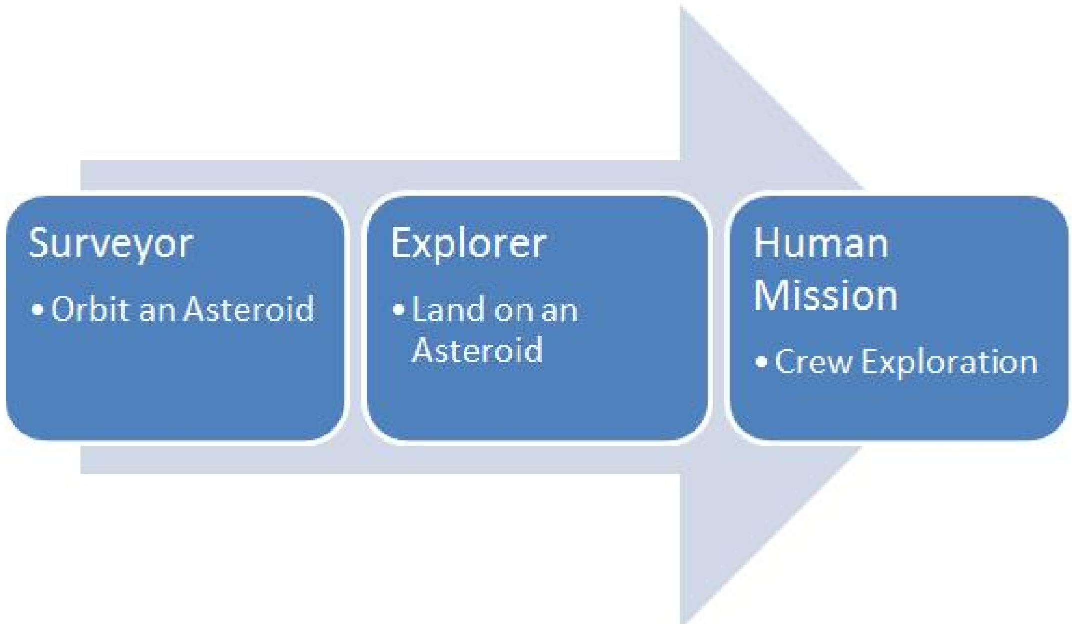


Figure 59: NEA Program overview

Table 20: Objectives correlated with each of the mission sections

Objective	NEA Surveyor	NEA Explorer	NEA Crew
Physical parameters - size, shape, gravity, density, roughness, visible/thermal albedos	√	√	√
Surface composition, resources, organic, bound water and other volatiles	√	√	√
Secondary objects and orbiting debris	√	√	√
Deep space radiation environment	√	√	√
Small body proximity operations	√	√	√
Surface and orbiting dust hazard analysis		√	√
Surface strength, cohesion, friability		√	√
Internal object structure, cohesion		√	√
Surface attachment, grappling approaches		√	√
Sample/resources acquisition, analysis, return			√
Object deflection / threat mitigation			√
Human first deep space venture – Crew system capability demonstration		√	√

asteroids for human landing in the beginning. As the campaign progresses, NEA asteroid knowledge will increase and identify suitable human mission targets for the last stage of the campaign.

Looking deeper at the objectives associated with the mission can show the overlap of the objective sections with each mission as shown in Table 20. This type of information overlap is a perfect example of where dependencies may be utilized to show how technology may address this intersection of objectives.

7.1.2 NEA Operations

Each mission brings new challenges and operation scenarios that contribute to the overall campaign. It is uncertain what experiments will be demonstrated during the NEA human mission; however, there are multiple possibilities for human asteroid interactions. Part of determining the technology portfolio related to each mission is to keep in mind the actual objectives that may be accomplished. Asteroid mining, grappling surfaces and extravehicular activities will be determined far in advanced, but can be speculated upon earlier to show technology demonstrate possibilities in a NEA Campaign.

7.1.3 NEA Target Information

What makes a successful target? The asteroid must be quite large with a slow rotation and an orbit that is close enough to get there for all three missions. Current observation techniques supply a few possible asteroids targets. There have also been quite a few missions involving asteroids that could be possible threats.

There is an estimated 1.1-1.7 million objects in the Solar System. Of these, most are in the asteroid belt between Mars and Jupiter. The Congressional act of 2005 led to increased funding for ground based surveys to detect, track, catalogue and characterize 90% of NEAs > 140m by 2020. Of these over 5000+ new NEAs discovered in the past few years, of these new ones approximately 1130 are in close Earth proximity and classified as Potentially Hazardous Asteroids (PHA). It is expected that 100,000 more will be detected in the coming years with an estimated 20% or >20,000 (>140m) as potentially hazardous. [60, 73]

Figure 60 and Figure 61 show the number of discovered NEA's and large asteroids defined as a kilometer or greater since 1995 respectively. There are multiple possibilities to investigate for this campaign; however, not every one of these asteroids are suitable targets.

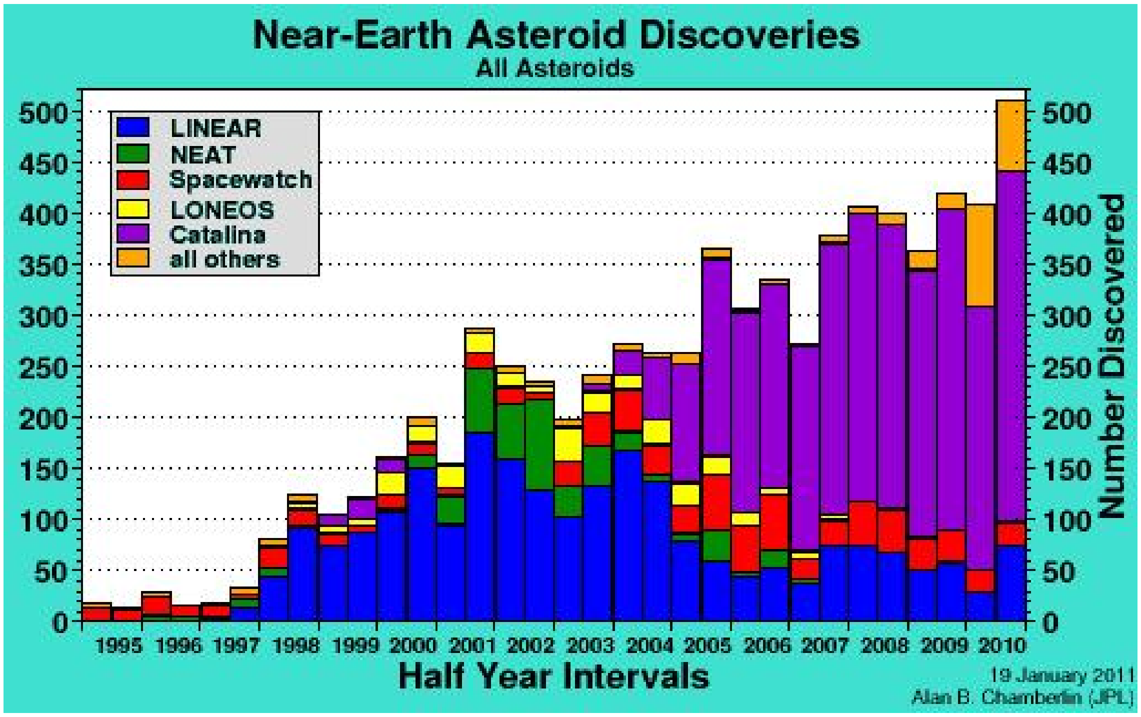


Figure 60: Number of all asteroids discovered

[71]

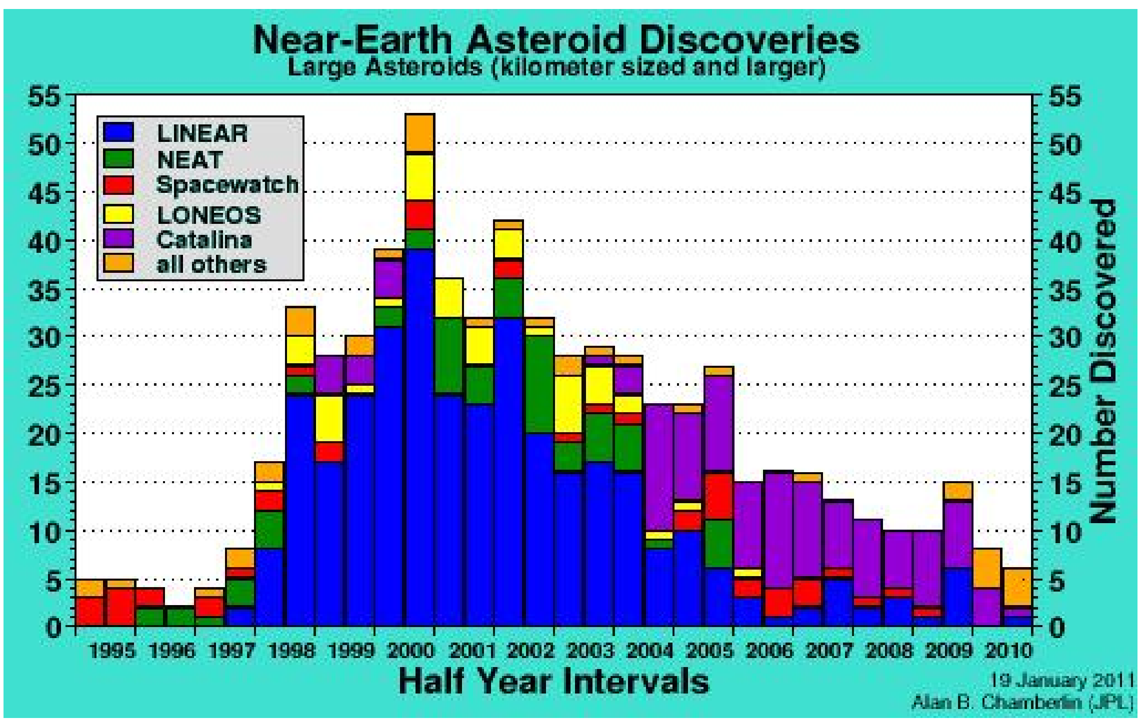


Figure 61: Number of Large Asteroids discovered

[71]

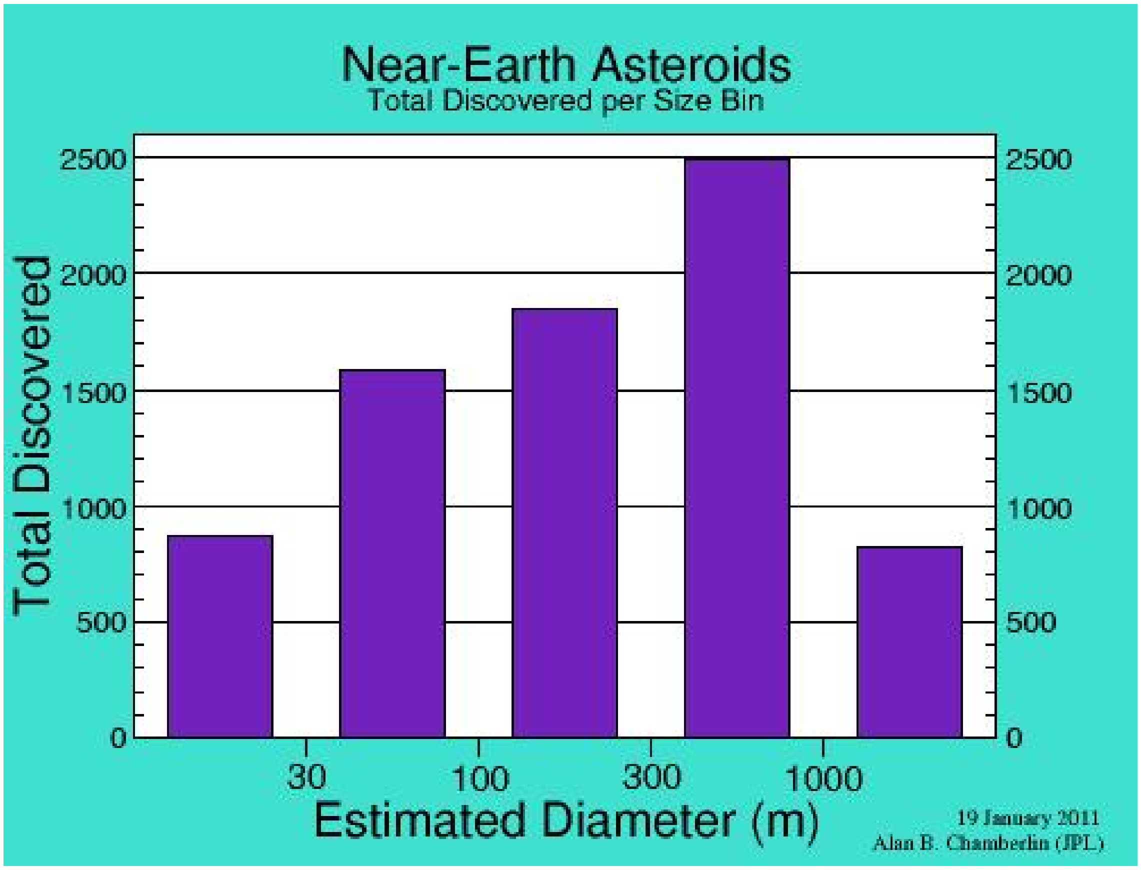


Figure 62: NEA histogram of asteroid diameter

[71]

Figure 62 is a histogram that represents the data to show the estimated diameters of the asteroids found in Figure 60 and Figure 61. Ideally astronauts would only want to go to asteroids that are above 1 km, so this puts the range around 800 asteroids.

7.1.4 NEA Classification

These NEA's can be classified into various categories. For example the orbits can be classified as seen in Table 21.

Physical characteristics such as the absolute magnitude, diameter, and rotation period are other requirements that must be taken into account to determine the type of environment that is involved with the NEAs. This information can be found by

Table 21: Orbit Explanations

Orbit Type	Definition
ATEN	Inside Earth Approach
Apollo	Outside Earth Approach

Table 22: Possible NEA Targets

Name	Classification	Absolute Magnitude	Mag sigma	Diameter	Rotation Period	Pole direction/ Sigma	SMASSI Spectral type
1991 JW Apollo	(NEO, PHA)	19.12	0.6054				
1999 RA2 Apollo	(NEO)	20.909	0.98442				
Apophis	Aten (NEO,PHA)	19.7	0.4	0.27km 0.06	30.4 h 77.6/-82.3 degrees 10		Sq
2008 EV5 2	Aten(NEO, PHA)	20	0.42285 .45km : 0.40 σ	3.7 25 h σ .001	77.6/-82.3 degrees :σ 10		
1989 UQ	Aten(NEO, PHA)	1931	0.67108	319.7998626	77 33 h		B
2001 CC21	Apollo(NEO)	18.372	.93807	383.1568422	5.017 h		L

spectroscopy and other methods. Table 22 provide possible NEA Targets with class information to show the diversity of possible NEAs being considered.

The two main groups are C- and S-. The C-group consists of the carbonaceous objects while the S-group is more of a stony nature. These types of environments are critical to the selection of NEA targets for the Surveyor and Explorer missions of the campaign.

7.2 NEA Campaign questions answered with Dependencies

The NEA campaign is a complex problem with multiple elements that interact with each other on multiple levels. It is a perfect case study to demonstrate the change in the technology portfolio selection versus modeling relationships between elements. Modeling dependencies early in process shows the largest possible impact of technology selection for human deep space missions.

For example adding dependencies between capabilities within an earlier mission can show the adoption of new technology that will affect a later mission. Similarly adding dependencies between missions can show the impact of precursor missions on the final human mission. Some interesting questions come out such as: can the same information be accomplished in only one precursor mission? Are two missions enough or does NEA exploration demand three missions? Are the correct missions being sent?

Another interesting question deals with putting later mission constraints on earlier missions. If the Surveyor must be as safe for a NEA as an Explorer mission does this affect the technology portfolio? Does the Surveyor even need to be as safe as the Explorer mission in order to survey the environment to make it safe for the Explorer mission? If the NEA Human mission, which have extremely high safety constraints, are placed upon the Explorer mission technology portfolio will this enable both missions or make all missions infeasible?

These are just a sampling of the questions that may be addressed in the technology portfolio selection process by adding dependencies into the process. There are multiple scenarios that may be investigated, but they must allow the elements to remain flexible in choosing technologies while maintaining a high level of fidelity so that the user can believe the results.

Chapter 8 will give the example input file to show the possible changes without technical data. It will focus on the changes associated with putting the fidelity into the technology decision making process through research question 1.

CHAPTER VIII

NEA DESIGN SPACE EXPLORATION

The methodology requires that the input file models the appropriate technology relationships to change the investment decisions. Once the elements are defined, specific dependencies may be included in the process. Multiple cases were determined to demonstrate the thesis questions as well as answer specific questions about the NEA campaign. This chapter will first define the dependencies associated with the campaign, then go into the specific input file generation and dependencies modeled. It will finally show all the cases expected to be run.

8.1 Dependency Defined

Going through the possible dependencies stay in line with Chapter 1, but here the focus is on the different levels of dependencies as given in Figure 63. This problem allows the categorization of the science/value dependencies as well as the programmatic dependencies between missions and objectives.

It was suggested that for the precursor mission scenarios to only change the output files. While this works well for projects that are taking information from past missions in order to deal with the current mission, the suggestion here is to program the dependency into the project with the human mission in mind as the main mission. This is not to say the user is not taking the precursor missions seriously, it is saying that the ultimate goal or weighted possibility is the last mission, instead of the two precursor missions with spacecraft and rovers.

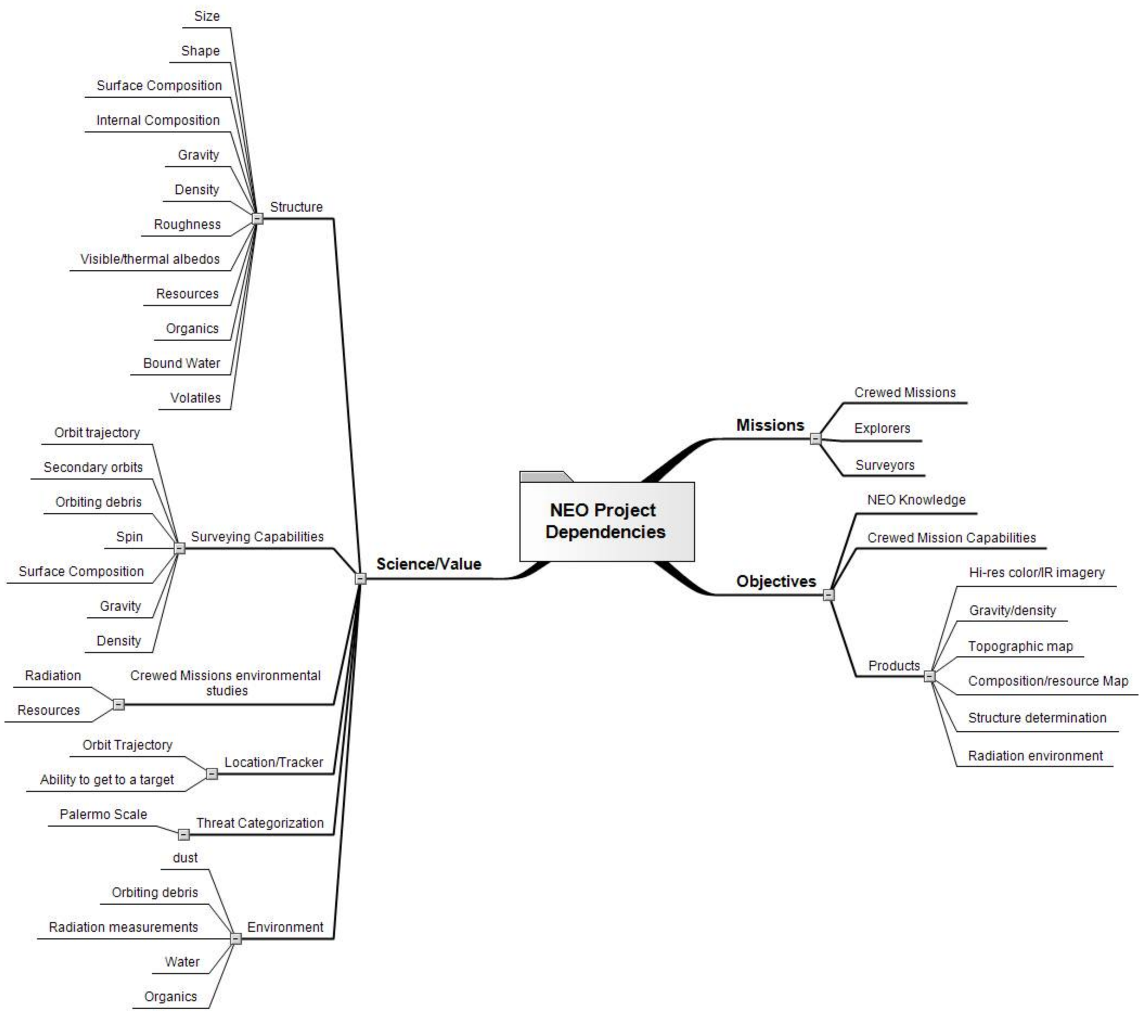


Figure 63: NEO Dependency Possibilities

8.1.1 Figures of Merit

The figures of merit evaluated using TOPSIS will be the value, cost, schedule, partial funding, flexibility, and dependency. The value will be the objective function that is given. The cost will be the total cost of the recommended portfolio as well as individual technology selections from the partial funding values. The scheduling will be decided by the first year of development, the year of delivery and the duration of the program or technology. The partial funding is an output of the optimized portfolio. The flexibility will be an expert opinion option that is defined from field experience. Each are outputs, but more importantly markers that allow the user to compare one portfolio to another.

8.2 Input file

8.2.0.1 Strategic Areas

The strategic areas were taken from the Mars Design Review Manual.[30] These major areas were used to create a general deep space capabilities input file that START could run. Any deep space human mission could fall from these general strategic areas; for example, the Constellation profile that was used originally for the Thesis Proposal. This general input file for START was used for the data runs for capability areas to follow in its footsteps. Table 23 shows the strategic areas used for the generic input file for deep space human mission.

8.2.0.2 Capability Areas

The capability areas come from the NEA committee. These were specific to the NEA mission. This is an example for NEA; however, Lunar, Martian, or ISS missions can be created that are specific to the deep space human mission. Table 24 shows the NEA mission specific capabilities for this mission. [58]

8.2.0.3 Specific Technology

The specific technology was kept the same as the capability areas. At this point in the campaign, there are no specific technologies to compare. However, the methodology can be used to decipher between the technologies as well as the capabilities. Each capability can have multiple technologies that are associated with the capabilities. For example, a capability could be EVA technology, while specific technologies could be multiple instruments used for EVA, or even multiple procedures for EVA capabilities.

This aspect of increasing the specific technologies follows the line of scalability presented earlier. At this point, the problem is a $N = 2$ level; however, adding the specific technologies associated with each capability would move the problem to a $N=3$ or $N=4$ problem.

Table 23: Strategic Areas taken from the Mars DRM [30]

Strategic Area
Advanced EVA for Mars
Advanced Habitation Concepts for Mars
Advanced Life Support
Advanced Life Support - ISRU Synergism
Advanced LO2/CH4 Propulsion
Advanced TPS for Earth Entry
Bio Safety (Planetary Protection)
Communications and Navigation
Cryogenic Fluid Management
Entry, Descent, and Landing
Fission Surface Power System
Hardware Scavenging and Recycling
Maintenance and Repair, In-Situ Fabrication and Prototyping
Mars In-Situ Resource Utilization
Medical Care
Nuclear Thermal Propulsion
Radiation Protection
Scientific Systems. I.e. Subsurface Access
Surface-Based Diagnostics, Test and Verification
Sample gathering - scientific systems
Advanced EVA for Mars
Advanced Habitation Concepts for Mars
Advanced Life Support
Advanced Life Support - ISRU Synergism
Advanced LO2/CH4 Propulsion
Advanced TPS for Earth Entry
Bio Safety (Planetary Protection)
Communications and Navigation
Cryogenic Fluid Management
Entry, Descent, and Landing
Fission Surface Power System
Hardware Scavenging and Recycling
Maintenance and Repair, In-Situ Fabrication and Prototyping
Mars In-Situ Resource Utilization
Medical Care
Nuclear Thermal Propulsion
Radiation Protection
Scientific Systems. I.e. Subsurface Access
Surface-Based Diagnostics, Test and Verification
Sample gathering - scientific systems

Table 24: Capability Areas taken from [58]

Capability Areas
Mobility
EVA Technology
Sensor Development
Surface Mobility
Autonomous Systems
Automated Rendezvous and Docking
Life Support and Habitation
Advanced Thermal Control & Protection Systems
Life Support and Habitation
Microgravity Effects
In-Space Chemical Propulsion
High Speed Earth re-entry (¿ 11.0 km/s)
Environment Mitigation (e.g. Dust)
Advanced Nav/Comm
Cryogenic Fluid Management (e.g. zero boil off)
Cryogenic Fluid Transfer
Aeroshell & Aerocapture
Precision Landing
Sensor Development
High Power Space Electrical Power Generation
High Efficiency Space Power Storage
Supportability & Logistics
Lightweight Materials & Structures
Supportability & Logistics
In-Situ Resource Utilization
Exploration Medical Capability
Human Health and Countermeasures
Behavioral Health and Performance
Space Human Factors & Habitability
Heavy Lift Propulsion Technology
High Power Electric Propulsion
Space Radiation Protection
Solar Observations: Particles, Wind and Flames
Human Exploration Telerobotics
Autonomous Systems
Human Robotic Systems
Sensor Development
Sample Gathering
Advanced Avionics/Software
Autonomous Systems

8.2.1 Creating the Input file

In order to incorporate the information given from the Mars Design Review Manual, a system had to be created in order to back out the cost and scheduling aspects from the larger system to smaller system. This was done by assuming that there is some cost and schedule associated with a Mars Capability B. A subsequent NEA capability A that is similar to Mars Capability B must have some relationship to the Mars capability. This is depicted below in Figure 64.

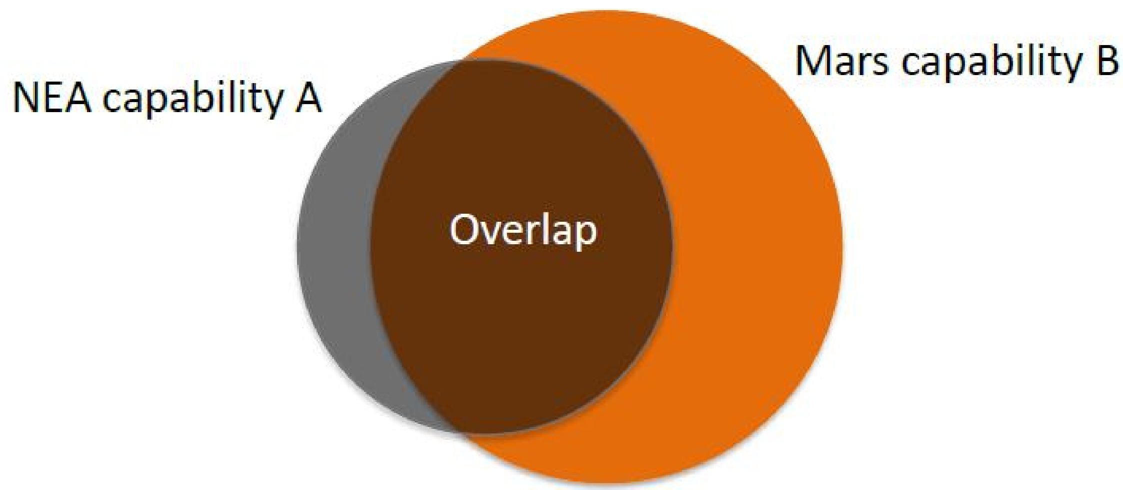


Figure 64: Mars and NEA capability overlap

The information associated with the Mars capability is known from the Mars DRM. Therefore, the size of the NEA capability can be backed out from this information utilizing two variables. These variables are flexibility and extensibility. The overlap part of the two elements would be the flexibility. Flexibility is the ability of the technology to be used for multiple purposes. Extensibility on the other hand is the ability for the technology to only be used for the current technology. In this case that value is the non-overlap part of the Mars Capability B. Two parameters are therefore defined as X and Y Where:

X = Fraction of NEA capability A relevant to Mars capability B

= Overlap / Total NEA capability A.

Y = Fraction of Mars capability B demonstrated by NEA capability A

= Overlap / Total Mars capability B.

The X and Y values may be estimated by engineering assessment of common elements. Using estimates for X and Y, and the ROM costs for total Mars capability ($M) to compute rough estimate of costs (assumes some overlap, i.e. X not equal 0 and Y not equal 0). This means more information can be taken from this procedure.

$$\text{Overlap cost} = \$M * Y$$

$$\text{Mars-specific cost} = \$M * (1\text{-}Y)$$

$$\text{NEA capability cost} = \$M * Y/X$$

$$\text{NEA-specific cost} = \$M * Y * (1/X - 1)$$

Each capability from the Mars DRM may now be backed out into the NEA domain to see what technologies that NEA may demonstrate. The assumption is that there is some sort of overlap and that the cost and scheduling is accordingly adjusted in order to get a smaller percentage of the original capability. The same procedure can be done in order to back the capabilities for Surveyor and Explorer as seen in Figure 65.

In the case of the precursor missions, they were backed out directly from the NEA capability rather than sequentially (i.e. Explorer was based on NEA and Surveyor was based on Explorer). The reason is that if one of the precursor missions is not chosen then the capability is not sequentially applied to the total technology portfolio selection. In order to determine the values for the Explorer and Surveyor the equations are extended as seen below.

$$\text{NEA Total} = Z = Y/X$$

$$A = \text{Fraction of NEA Surveyor relevant to NEA Human}$$

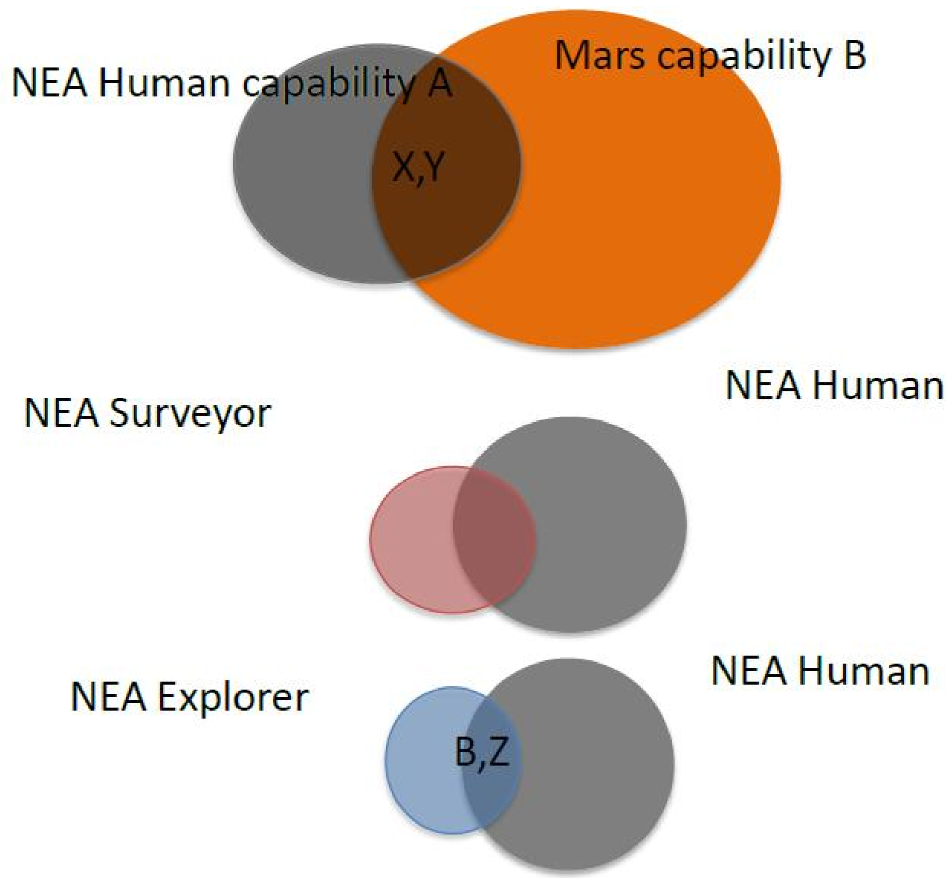


Figure 65: Precursor Capability Overlap

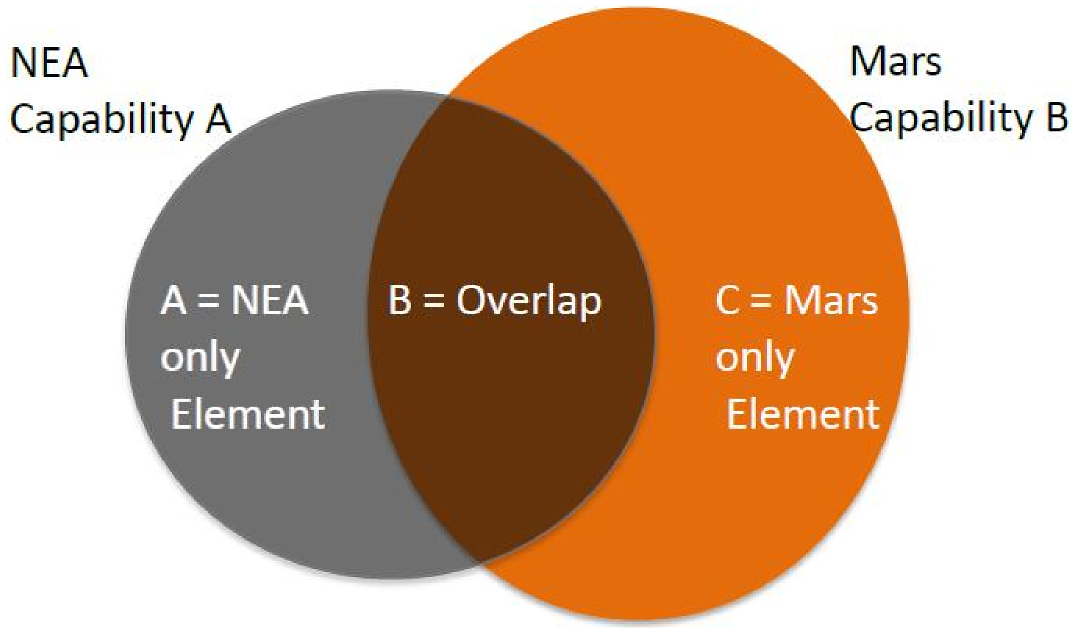


Figure 66: Feed Forward Relationships

B = Fraction of NEA Human relevant to NEA Surveyor

C = Fraction of NEA Explorer relevant to NEA Human

D = Fraction of NEA Human relevant to NEA Explorer

Total NEA Surveyor =Y/X * B/A

Total NEA Explorer = Y/X * D/C

These capabilities are not linked to the Mars DRM which represents technologies needed for deep space human capabilities. This information is utilized from industry experts and ultimately the user has the responsibility to back the information out from higher missions.

8.2.2 Feed-Forward Relationships

Now that the input file has been created the feed-forward relationships must be created. If the NEA and Mars Elements are broken down, there are the element specific parts and the overlap as depicted below in Figure 66.

Table 25: Feed-Forward Dependency Requirements

Element	Equation	Dependency
NEA Element	A+B	None
Mars Element	B + C	Mars OR Partial Mars Element
Partial Mars Element	C	Partial Mars Element needs NEA Element

The element that was created in the START input is the entire element. Therefore

$$\text{NEA element} = \text{A} + \text{B}$$

$$\text{Mars Element} = \text{B} + \text{C}$$

The idea of a feed-forward system is that if the NEA element is selected then there is no need to fund the entire Mars element. However, if the NEA element is not chosen then the entire Mars element must be chosen. In order to simulate this, a new Partial Mars element is created. Partial Mars Element = C These three elements can then have dependencies created between them according to Table 25 in order to show a feed-forward analysis.

Including these dependencies with the backed out elements will give a feed forward methodology that automatically takes into account the fact if a technology is executed in an earlier mission than it does not need to be fully funded in the subsequent mission.

8.2.3 NEA Project Design Space Input file

Examining the NEA design space gave three distinct missions, with 14 cases, 18 strategic Areas, and 41 Capabilities. The technologies were kept the same as the capabilities as described above. Three different axes were studies: Dependency, Time Span and Expected Input Percentage. The expected percentage were value changes were applied to the input file and where applied in 25% increments from 25% to 200% of the input values assessed by the user. The dependencies were tested separately as well as compounded.

The 10 different cases showed different dependency types as well as answered

Table 26: Case Studies and the Dependencies they test

Cases	Dependencies Tested
Surveyor	Capabilities
Explorer	Capabilities
NEA Human Mission	Capabilities
Mars Human Mission	Capabilities
Surveyor and Explorer	Mission and Capabilities
Explorer and NEA Mission	Mission and Capabilities
Surveyor and NEA Mission	Mission and Capabilities
NEA Human and Mars Human	Mission, MIMO and Capabilities
NEA Campaign	Mission, MIMO and Capabilities
NEA Campaign	and Mars Mission, MIMO and Capabilities

different NEA campaign specific questions. These 10 cases can be translated into 14 different scenarios when the cases are combined. This information is represented in Table 26 and Table 27. Table 26 correlates the dependencies tested with the possible cases associated with the mission combinations. The time span was tested as described in Table 28 by changing the starting year of the portfolio while keeping the yearly amount constant. The last change was for the expected percentage of the input values. This was done so that the research question number 4 could be investigated. The information would be taken to compare the change in data values, or derivative of the data to see how they compare to the addition of including dependencies.

8.3 Dependencies Tested

Chapter 6 gave the basics of what happens when dependencies are added to a START input file. In order to get real world results capability names and technologies were given to the generic capabilities and utilized in the feed-forward analysis as previously explained. Table 29 and Table 30 give more specific missions and capabilities that will be tested in the series.

The next few tables give specific mission dependencies included in each of the cases given in Table 26. The one rule for applying dependencies here was that an earlier mission cannot be dependent on a later mission.

Table 27: Case Combinations expanded

Case	Scenario
1	Surveyor
2	Explorer
3	NEA
4	Surveyor and Explorer
5	Explorer Needs Surveyor
6	Explorer OR Surveyor
7	NEA and Surveyor
8	NEA and Explorer
9	NEA Needs Explorer
10	NEA Needs Surveyor
11	NEA, Explorer and Surveyor
12	NEA and Explorer OR Surveyor
13	NEA and Explorer NEEDS Surveyor
14	NEA NEEDS Explorer NEEDS Surveyor

Table 28: Case Studies and their time spans

Cases	Time Changes
Surveyor	5
Explorer	9
NEA Human Mission	5
Mars Human Mission	5
Surveyor and Explorer	5
Explorer and NEA Mission	5
Surveyor and NEA Mission	5
NEA Human and Mars Human	5
NEA Campaign	5
NEA Campaign and Mars	5

Table 29: Mission Dependencies

Mission A	Mission B	Operation
NEO Surveyor	NEO Explorer	A Needs B
NEO Surveyor	NEO Explorer	OR
NEO Human	NEO Surveyor	A Needs B
NEO Human	NEO Explorer	A Needs B
NEA Human	NEA Explorer and NEA Surveyor	A Or B

Table 30: Capabilities Dependencies

Capability A	Capability B	Operation
High Power electric Propulsion	Human Exploration Tele-robotics	A Needs B
High Speed Earth re-entry	Precision Landing	A Needs B
In-space Chemical Propulsion	High Power Electrical Propulsion	A Needs B
Cryogenic Fluid Management	Cryogenic Fluid Transfer	A Needs B
Space Radiation Protection	Advanced Thermal Control and Protection Systems	A Needs B
Surface Mobility	Human Robotic Systems	A Needs B
Behavioral Health and Performance	Human Health and Counter measures	A Needs B

Table 31: Surveyor Capability Dependencies

Capability A	Capability B	Operation
High Power Space Electrical Power generation	High space Power Storage	A AND B
In-Space Chemical Propulsion	EVA Technology	B Needs A
High Power Electric Propulsion	In-Space Chemical Propulsion	A Needs B
Solar Observations	Sensor Development for Environmental Characterization	A Needs B

8.3.0.1 Surveyor

8.3.0.2 Explorer

Table 32: Explorer Capability Dependencies

Capability A	Capability B	Operation
High Power Space Electrical Power generation	High space Power Storage	A Needs B B Needs A
Mobility	EVA Technology	A AND B
Sensor Development for EVA	EVA Technology	A Needs B
Surface Mobility	EVA Mobility	A Needs B

Table 33: Explorer Cross Mission Dependencies

Capability A	Capability B	Operation
NEA Surveyor: In-Space chemical Propulsion	In-Space chemical Propulsion	B Needs A
NEA Surveyor: Environmental Characterization	Environmental Characterization	A OR B
NEA Surveyor: Solar Observations	Solar Observations	A OR B

Table 34: Explorer Cross Mission MIMO Dependencies

Capability A	Capability B	Operation
NEA Surveyor: In-Space chemical Propulsion	In-Space chemical Propulsion	MIMO
NEA Surveyor: Environmental Characterization	Environmental Characterization	MIMO
NEA Surveyor: High power electric Propulsion	High Power Electric Propulsion	MIMO
NEA Surveyor: Sensor Development for subsurface access	Sensor Development for subsurface access	MIMO

8.3.0.3 NEA Human

Table 35: NEA Human Mission Capability Dependencies

Capability A	Capability B	Operation
Mobility	EVA Technology	A OR B
Microgravity	Human Health and Counter Measures	B Needs A

Table 36: NEA Human Mission Cross Mission Dependencies

Capability A	Capability B	Operation
NEA Explorer: Precision Landing	NEA Human: Landing	B Needs A
NEA Explorer: Sensor Development for subsurface access	Sensor Development for subsurface access	B Needs A

Table 37: NEA Human Mission MIMO Dependencies

Capability A	Capability B	Operation
NEA Explorer: Rendezvous & Docking	NEA Human: Rendezvous & Docking	MIMO
NEA Surveyor: Solar Observations	Solar Observations	MIMO

The dependencies presented in Table 31 through Table 37 are the dependencies investigated throughout the design space. The results of these mission results will be showcased in Chapters 9-13 to demonstrate the effects of adding dependencies to the technology portfolio process. This information will be subsequently available in upcoming case studies and conference proceedings.

CHAPTER IX

SURVEYOR RESULTS

The purpose of this chapter is to look at the possible portfolios that would give solutions and forecast into the future as to the best decisions to make. A bottom up solution finds the highest possible objective function given current input values. A top down approach determines what the investment solution must be right now in order to get the desired objective function later.

In order to do the methodology the user must have some sort of criteria to discern what is needed. The feed-forward analysis has the advantage of only choosing technologies that will enhance the selection for Deep Space Human technology. Therefore any selection of the objective function is an addition of technology to be added. The user can make a decision to say that there must be some threshold of adding technology to legitimately enhance the objective. For this example, the user will look at the baseline and require a 10% increase in the TOPSIS value to make it a feasible investment option.

9.1 Surveyor: No Partial Funding

The Surveyor baseline is shown in Figure 67 with an objective function of 89.9817.

Just looking at this portfolio shows it is a good approximation of a step function. In this case most of the technologies are 1 and 2 year selections. The solution is also extremely sparse in its population of the technology Portfolio. Looking closer at the actual technology associated with the technology portfolio from Figure 67, gives Table 38. This shows that the technology have roughly the same objective value, but are fully funded, because the partial funding level is at 1 or 100% of what the user input into the system.

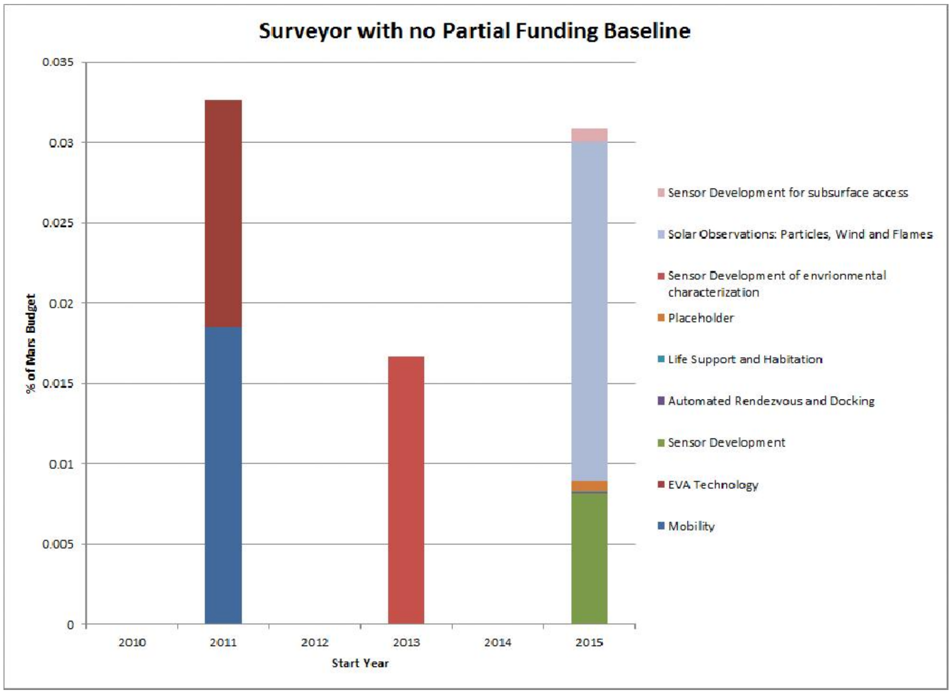


Figure 67: Surveyor Baseline Portfolio with no Partial funding

There were four dependencies integrated into the investment portfolio. Table 38 shows that the In-Space Chemical Propulsion, High Efficiency Space Power Storage and Heavy Lift Propulsion technology were not chosen when the technology had to be all or nothing funded.

Table 38: Surveyor no Partial Funding Baseline Portfolio

Metric	Selection	Partial Funding Scale
Mobility	Selected	1
EVA Technology	Selected	1
Sensor Development	Selected	1
Automated Rendezvous and Docking	Selected	1
Life Support and Habitation	Selected	1
Placeholder	Selected	1
In-Space Chemical Propulsion	NOT Selected	
Sensor Development of environmental characterization	Selected	1
High Power Space Electrical Power Generation	NOT Selected	
High Efficiency Space Power Storage	NOT Selected	
Heavy Lift Propulsion Technology	NOT Selected	
High Power Electric Propulsion	NOT Selected	
Solar Observations: Particles, Wind and Flames	Selected	1
Sensor Development for subsurface access	Selected	1

Four dependencies were added to this process as shown in Table 39. The individual dependencies were added to the process first and then compounded with the others in a combinatorial process which ended in dependency Level 15 which had all of the dependencies associated applied at the same time.

Table 39: Dependency Combinations

Dependency Number	Capability A	Capability B	Operation	Included in dependency level
1	High Power Space Electrical Power generation	High space Power Storage	A Needs B B Needs A	1,5,6,7,11,12,13,15
2	In-Space Chemical Propulsion	EVA Technology	B Needs A	2,5,8.9,11,12,14,15
3	High Power Electric Propulsion	In-Space Chemical Propulsion	A Needs B	3,6,8,10,11,13,14,15
4	Solar Observations	Sensor Development for Environmental Characterization	A Needs B	4,7,9,10,12,13,14,15

Comparing Table 38 and Table 39 shows that Dependency 2 was not upheld, so this portfolio would change the investment decisions based upon the fact that they have not modeled these relationships within the baseline. The other three dependencies were upheld with this portfolio. Adding the other 3 dependencies should not change the baseline portfolio if the portfolio converges to the optimal solution. If the portfolio does not converge to the optimal value, then adding the dependencies could potentially change the portfolio choices.

Freezing the start year at 2010 and varying the expected input percentage as well as adding dependencies gives Figure 68 Performing TOPSIS on this same data gives Figure 69. The interesting phenomena show that as the expected input percentage is increased, the value of the overall portfolio increases. Both graphs give a slightly vertical feels, **meaning that the change in the input percentage is a dominating factor in the change of the portfolio objective value**. The same phenomena is mimicked in the TOPSIS contour plot, but in the TOPSIS case the vertical phenomena is even stronger. The changes are not as abrupt as they are for looking at only the objective value. This translates into the bottom up and top down analysis to show that in this plane of information, the expected input percentage dominates the choices.

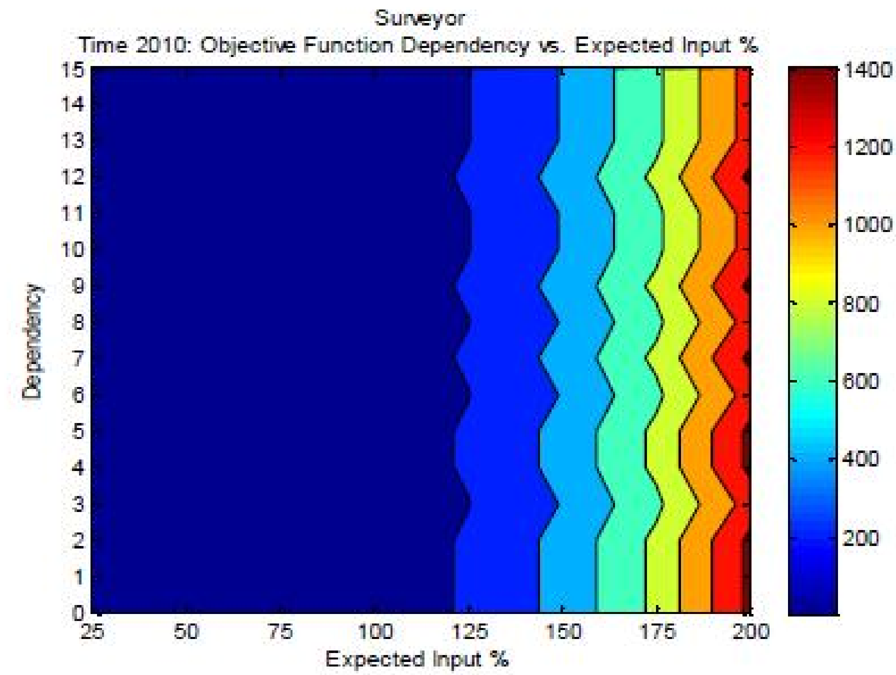


Figure 68: Surveyor Contour with 2010 as the first investment year

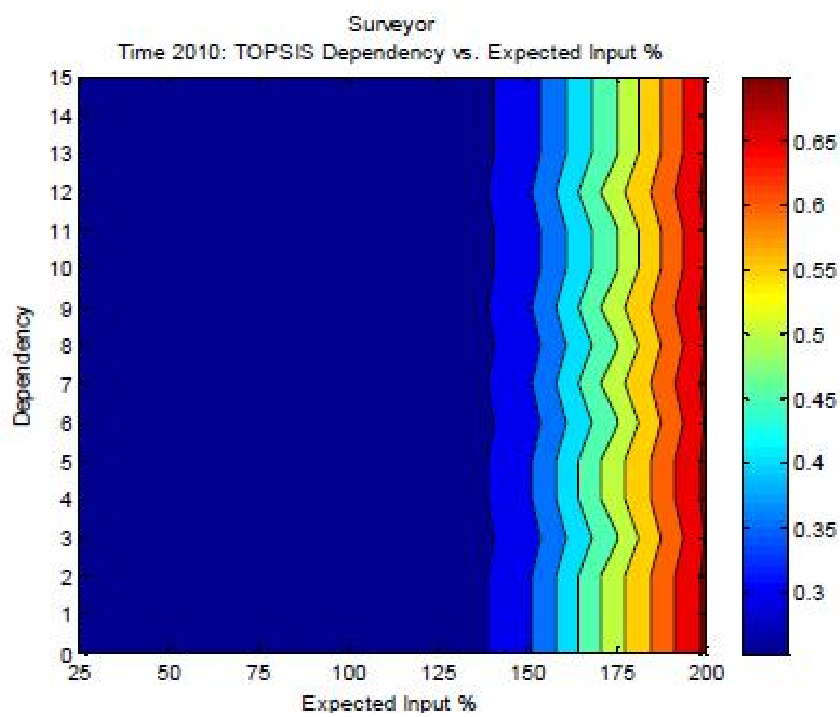


Figure 69: Surveyor starting year 2010,TOPSIS Dependency vs. Expected Input % Contour Plot

Another way to look at this information is to look at the contour plot on just one variable for a given start date. Figure 70 and Figure 71 show the changes in Objective Value and TOPSIS value with respect to a given Dependency level. The first trend that is shown here is that the objective value takes on a polynomial form. The TOPSIS value takes on a similar polynomial form that has been distorted by the relative differences between each portfolio.

Figure 70 shows two distinct lines. The higher one contains the baseline case and Cases 2,4,5,7,9, and 12, while the lower one contains Cases 3,6,8,10,11,13,14, and 15. this refers back to the fact that case 3 and any combination retaining it gives a lower objective value.

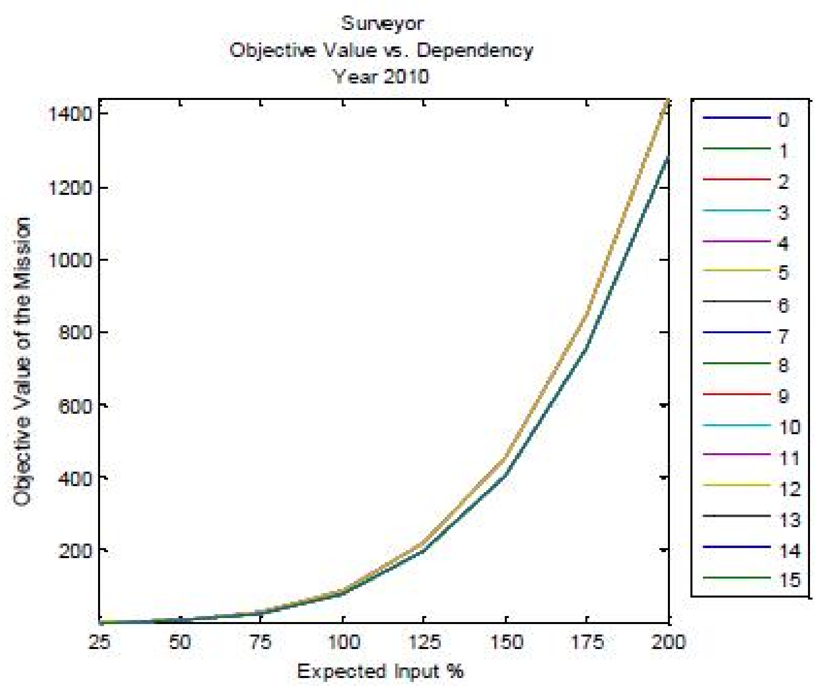


Figure 70: Surveyor Objective Value vs. Dependency

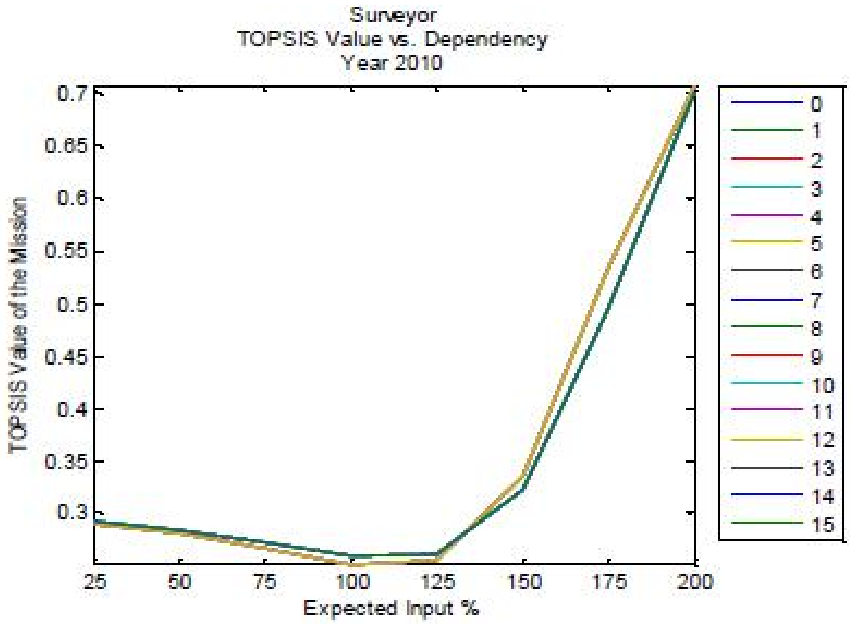


Figure 71: Surveyor TOPSIS vs. Expected Input Percentage

Freezing the start year once again and showing the data at a given expected input percentage shows Figure 72 and Figure 73. The interesting aspects that come from these graphs are the fact that the values are a more linear trend. Literally the portfolios have almost only two settings. Neither of which are larger than the no dependency case. As the expected input value percentage increases the jump in portfolio changes are larger. Looking at Figure 72 shows these jumps as much larger in the 200% expected input change vs. the 150% change.

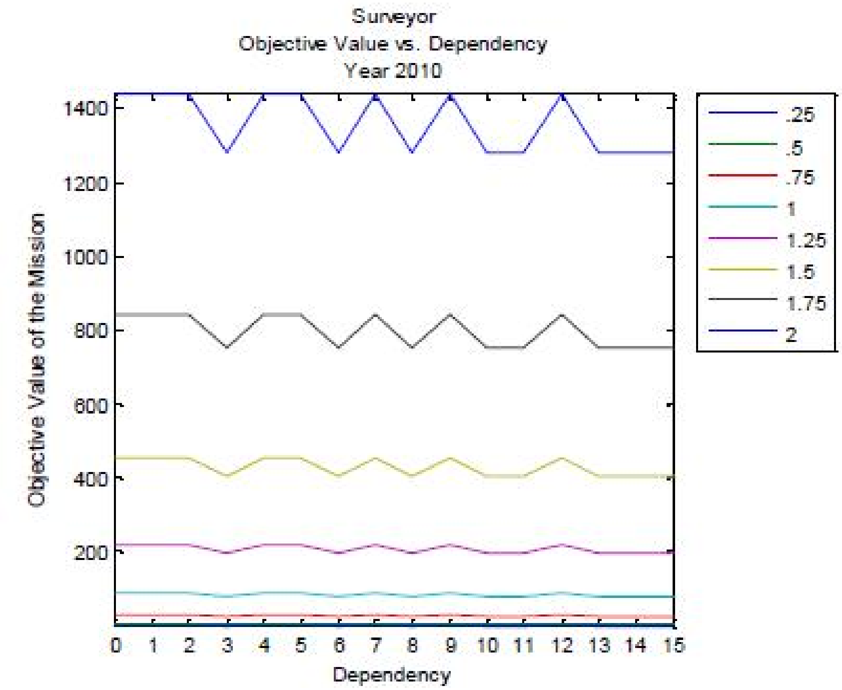


Figure 72: Surveyor for start year 2010, Objective value vs. Dependency

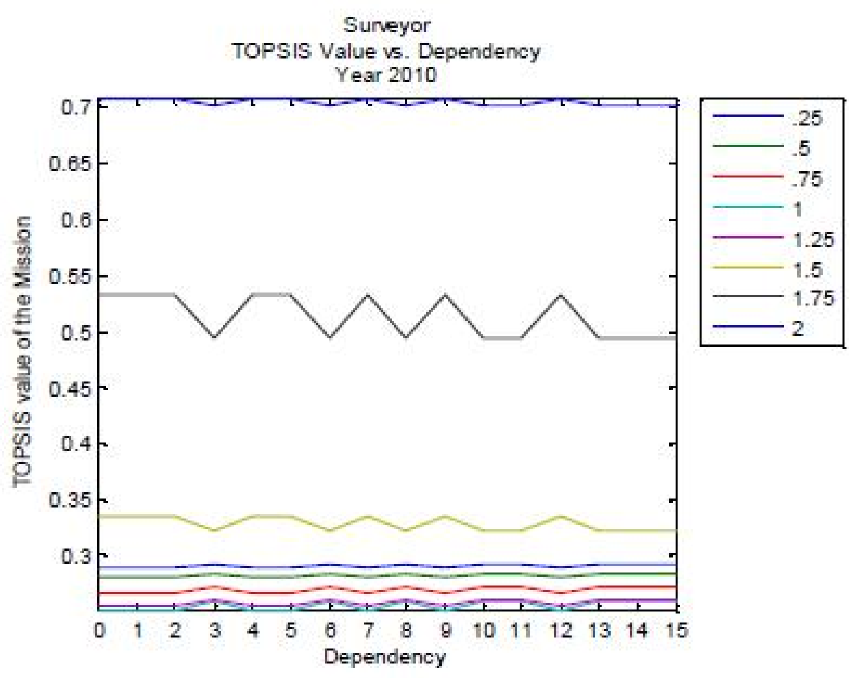


Figure 73: Surveyor TOPSIS Value vs. Dependency

Notice here that the curves are polynomial in appearance compared to Figure 72 and Figure 73 which have horizontal changes. The difference in the four graphs shows that the expected input percentage has a larger affect than the dependency addition. It also suggests that when certain dependency levels are added, the portfolio goes to a second default configuration than the baseline portfolio. Continuing on to the TOPSIS values in Figure 73 shows that the values are linear in nature once again; however, the changes are not defined jumps as with the objective value.

Changing the Expected Input Percentage is a continuous change, but adding dependencies to the process is an incremental change. Looking at the first and second derivatives of the Objective value contour plots can give some insight into the expected input percentage changes, but only arbitrary information about the dependency additions. The dependencies can just as well be added in any order. It so happens that they are ordered according to Table 31, but that does not mean that going from one level to the next has a significance in the way they are changing. Therefore the following graphs give more information on the expected percentage changes than it does the dependency level.

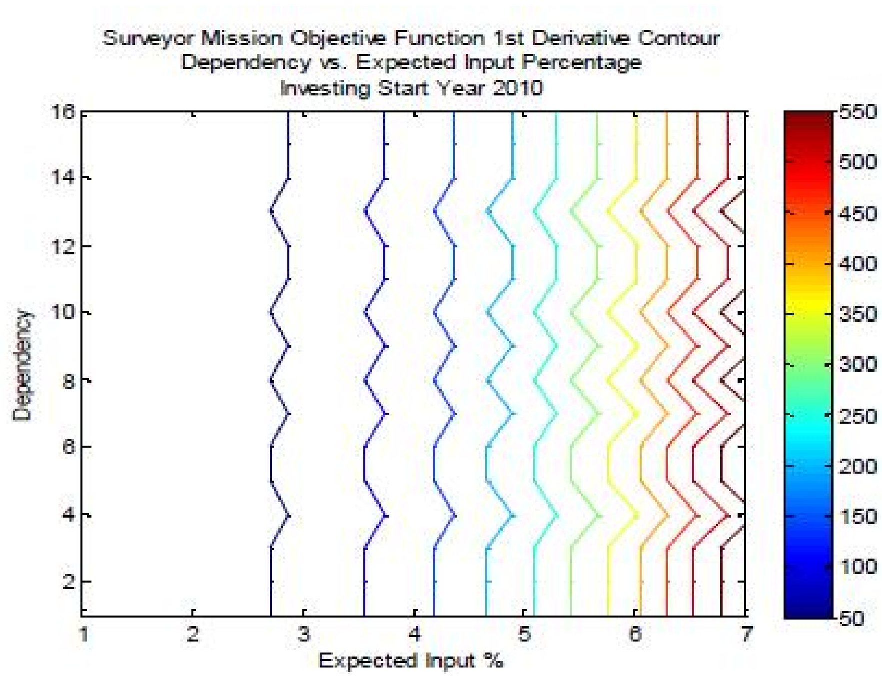


Figure 74: Surveyor 1st derivative of Objective Value Contour Plot for start year 2010

Looking at the derivatives of the objective value contours gives useful information as to how fast the objective value is changing with respect to the given parameters. In Figure 74, the first derivative shows that changes are uniform across the expected input percentage changes as expected from the original contour plot. However, they also show that Dependency level 13, 10, 8,6,5,3,2,1 all give the same universal change. This was eluded to when there was assumed to be some sort of secondary default portfolio that changes it. Once again, if the dependencies were grouped so that all the primary portfolios and secondary portfolios were sequential, the graph would look completely different. However, the point is to show that with the first derivative there is a significant jump from one portfolio to the next.

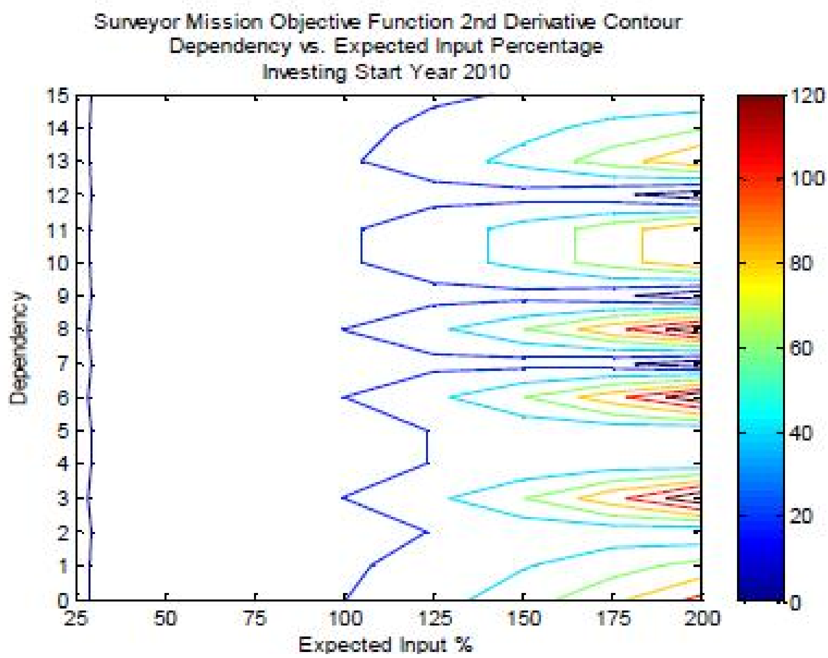


Figure 75: Surveyor Laplacian 2nd derivative for Objective Value Start year 2010

Figure 75 on the other hand gives the Laplacian or 2nd derivative of the objective value contour plot. This graph gives the acceleration of the changes. It confirms that dependency levels 13, 10, 8, 6, 5, 3, 2, 1 have the interesting phenomena; levels 3,6, and 8 give the greatest acceleration jump from one portfolio to the next. It also shows that portfolios 12, 9, and 7 all of which were the primary portfolio has the slowest moving changes.

The derivatives shows that the changes in the expected utility are the dominant feature when the start year is kept constant, but that the derivative levels give larger accelerated change features.

9.2 Surveyor No Partial Funding with Constant Expected Input Percentage

Previously Surveyor's start date was frozen and the changes in the expected input value and dependency levels were examined. This section looks at what happens when the Expected Input Percentage is assumed to be what the user expects, but the start date is changed. There are a few assumptions that go into changing the start year. The first is that the need by date has not been changed. Therefore starting the entire portfolio earlier only shortens the time period in which to develop technology. The second assumption is that there is a constant funding level for each year; effectively shortening the technology development time shortens the total amount of funding associated with the portfolio.

Both assumptions are valid in certain scenarios. Specifically, the first is valid if a new technology portfolio is an afterthought instead of a planned mission that calls for feed-forward technology. This would mean that the scientist or engineers see the opportunity to demonstrate some technology needed for a future mission on an earlier mission after they have already gathered all the technologies necessary for the current mission. The second assumption is valid when there is only a certain budget for feed-forward analysis that is considered an addition and specifically for enhancing technologies. Otherwise, enabling mission critical technologies would accept all the money before enhancing technologies are even considered.

The same analysis can be done by freezing the Surveyor expected input value. This contour plot is given in Figure 76. The first interesting feature of this contour plot is the range of values given by the plot colorbar. The graph goes from 80 to 89 objective

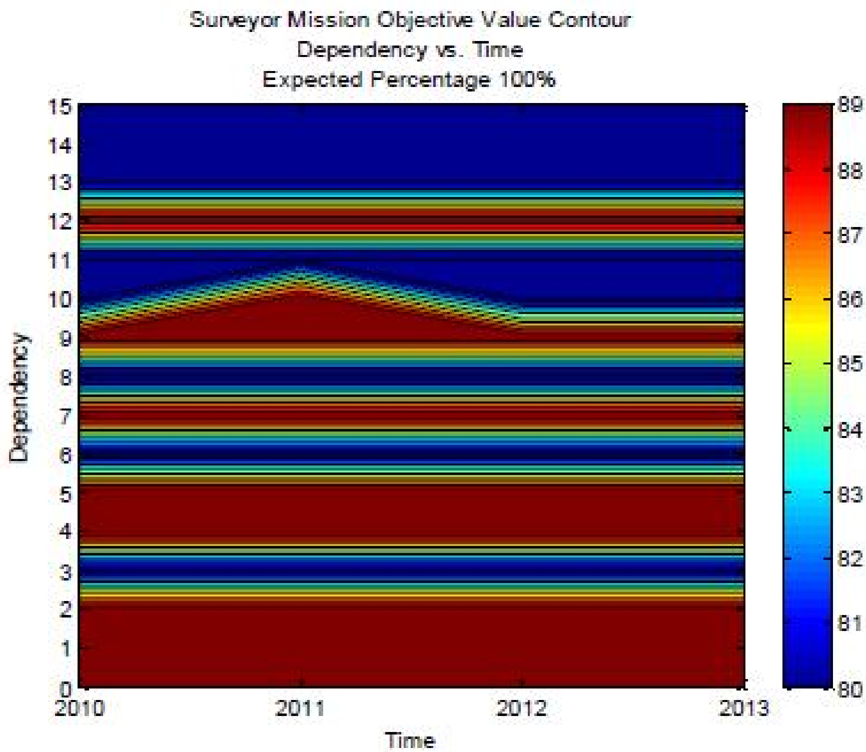


Figure 76: Surveyor No Partial Funding at 100% Expected Input Value Contour Plot

value score. This counts for every single portfolio tested. The next interesting feature is that the portfolio values at the no dependency level is larger than the values found at any other dependency level. It is also noted that the portfolio values are at the most the same as the no dependency level. This is expected. As dependencies are included in the process, the situation changes in the process, but not as it goes through the different start years. When it comes to the portfolio changes, it seems that there is no effect on the objective value. The actual portfolio technologies must be investigated in order to see if the portfolios are the same.

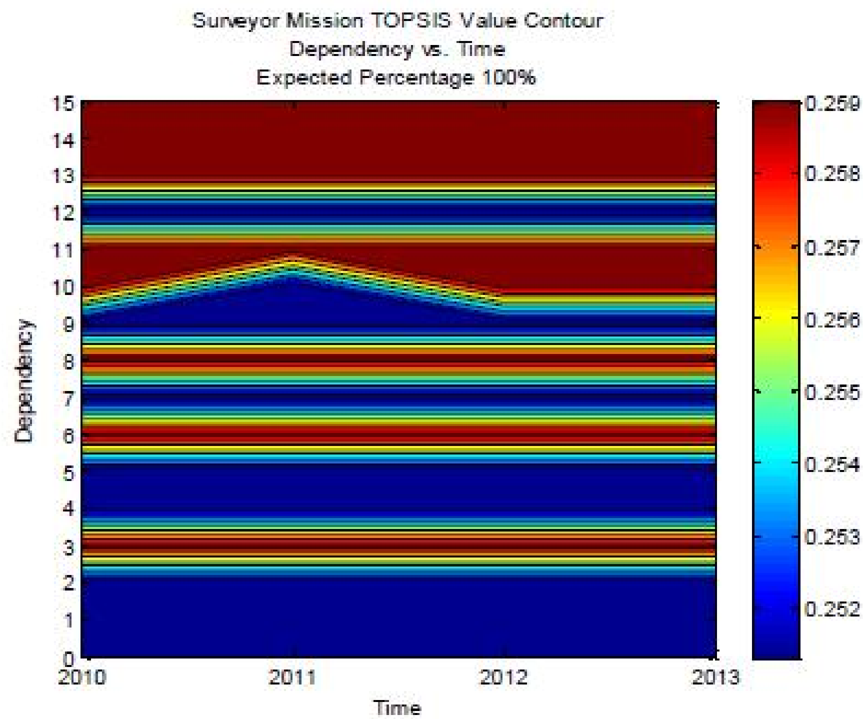


Figure 77: Surveyor No Partial Funding 100% Expected Input Value TOPSIS Contour Plot

Moving onward to Figure 77 shows the changes for the TOPSIS values. In this case the TOPSIS values are literally the inverse values of the objective values. For example, the bulge formed at the Dependency levels 8-11 is red in Figure 76 and the same feature in the TOPSIS value is blue in Figure 77. It is also noteworthy to see that the TOPSIS value is also only a change of 0.007 or 2.7% in TOPSIS. This type of information can be shown in another way by mapping the objective value and TOPSIS value vs. the start year and dependency value in Figure 78 and Figure 79.

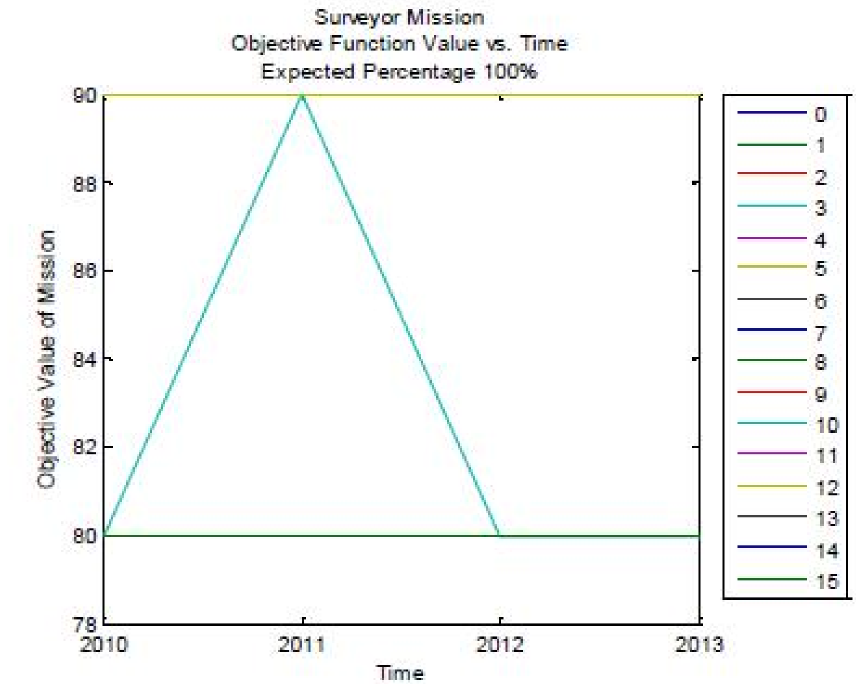


Figure 78: Surveyor Objective Function vs. Time

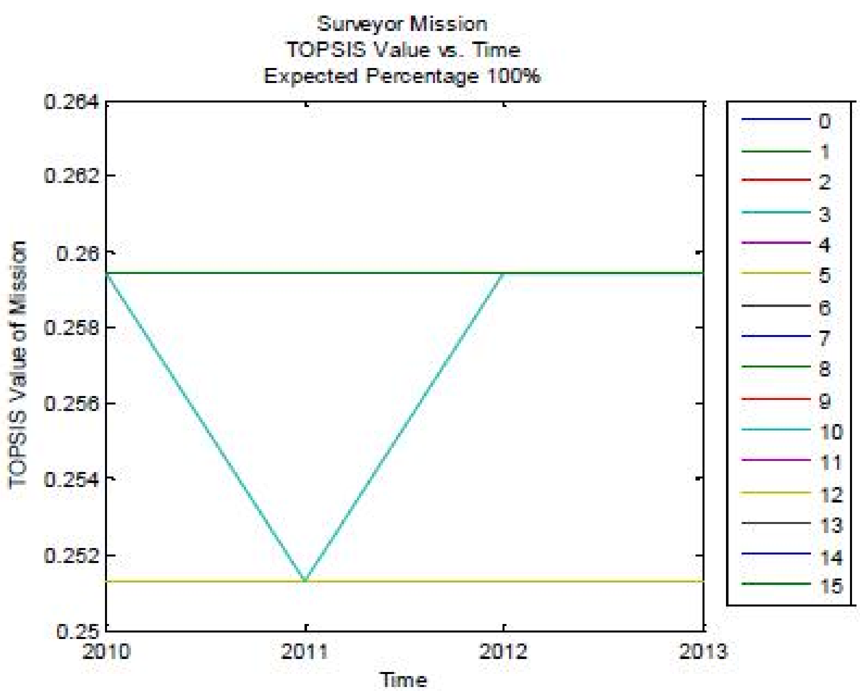


Figure 79: Surveyor TOPSIS value vs. start year for 100% Expected value

Looking at Figure 78 and Figure 79 shows that the interesting pyramid feature in Figure 76 and Figure 77 at the dependency level 10 happens specifically in the start year of 2011. That portfolio must be an interesting phenomena that allows for larger values.

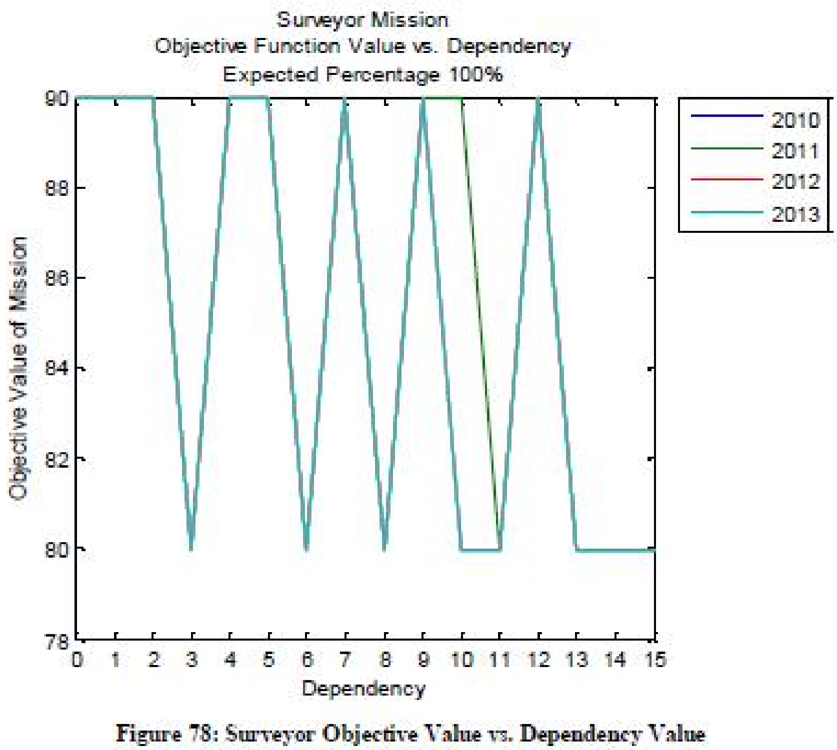


Figure 80: Surveyor Objective Value vs. Dependency Value

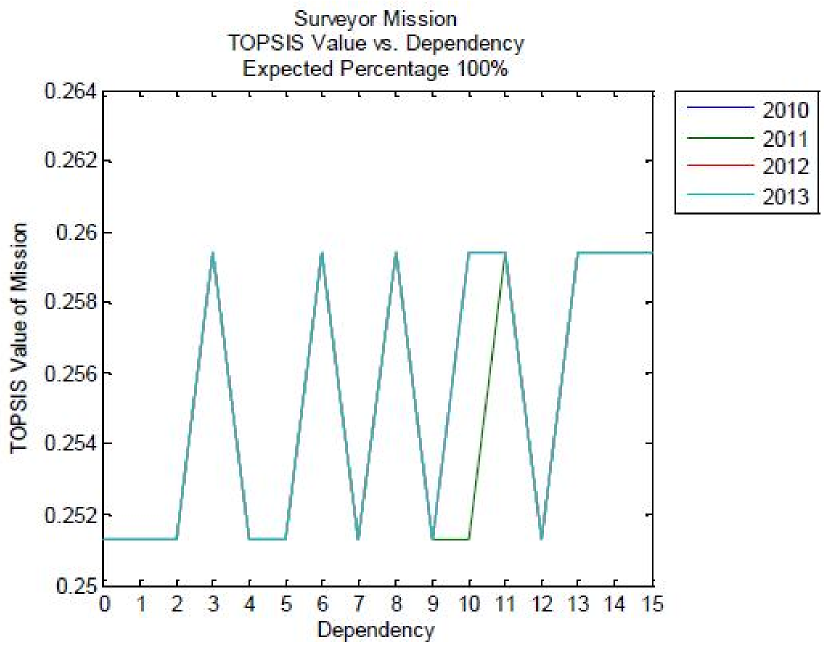


Figure 81: Surveyor TOPSIS va Dependency Value for 100% Expected Value

The same feature happens in the TOPSIS value Figure 79 as well. Looking at these two dependency levels and start years are shown below in Table 40. Comparing the baseline portfolio and the start year of 2011 and dependency level 10 gives the same technologies selection as shown in Table 40. However, they are not the same portfolio. This is shown in Figure 82 and Figure 83.

Table 40: Surveyor Portfolio comparison

Name	Dependency Level 0 Selection	Dependency Level 10 Selection
Mobility	Selected	Selected
EVA Technology	Selected	Selected
Sensor Development	Selected	Selected
Automated Rendezvous and Docking	Selected	Selected
Life Support and Habitation	Selected	Selected
Placeholder	Selected	Selected
In-Space Chemical Propulsion	NOT Selected	NOT Selected
Sensor Development of environmental characterization	Selected	Selected
High Power Space Electrical Power Generation	NOT Selected	NOT Selected
High Efficiency Space Power Storage	NOT Selected	NOT Selected
Heavy Lift Propulsion Technology	NOT Selected	NOT Selected
High Power Electric Propulsion	NOT Selected	NOT Selected
Solar Observations: Particles, Wind and Flames	Selected	Selected
Sensor Development for subsurface access	Selected	Selected

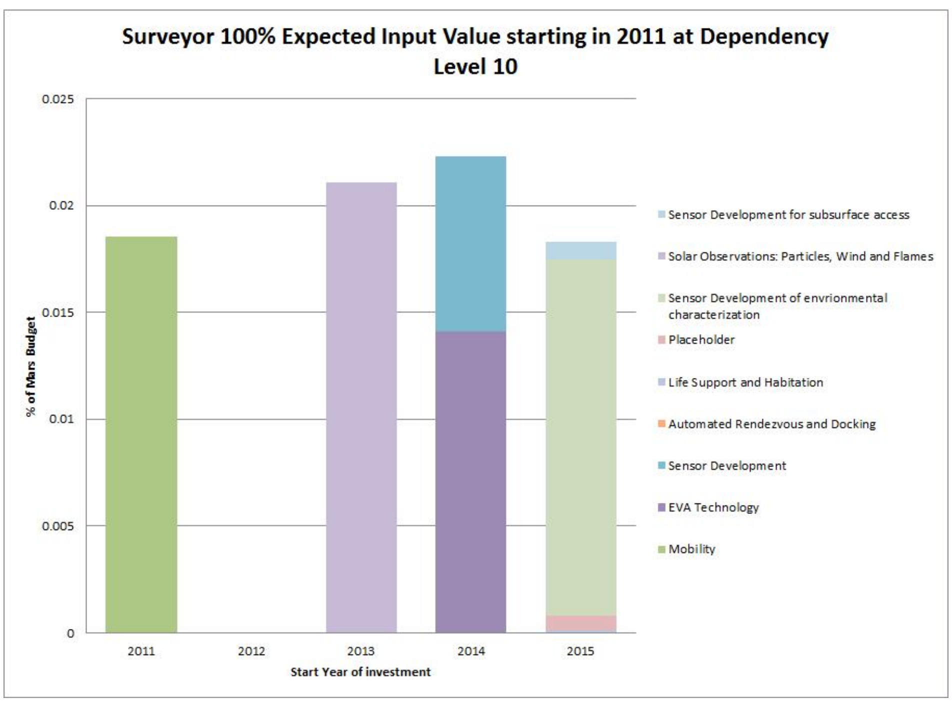


Figure 82: Surveyor Dependency Level 10 and start year in 2011

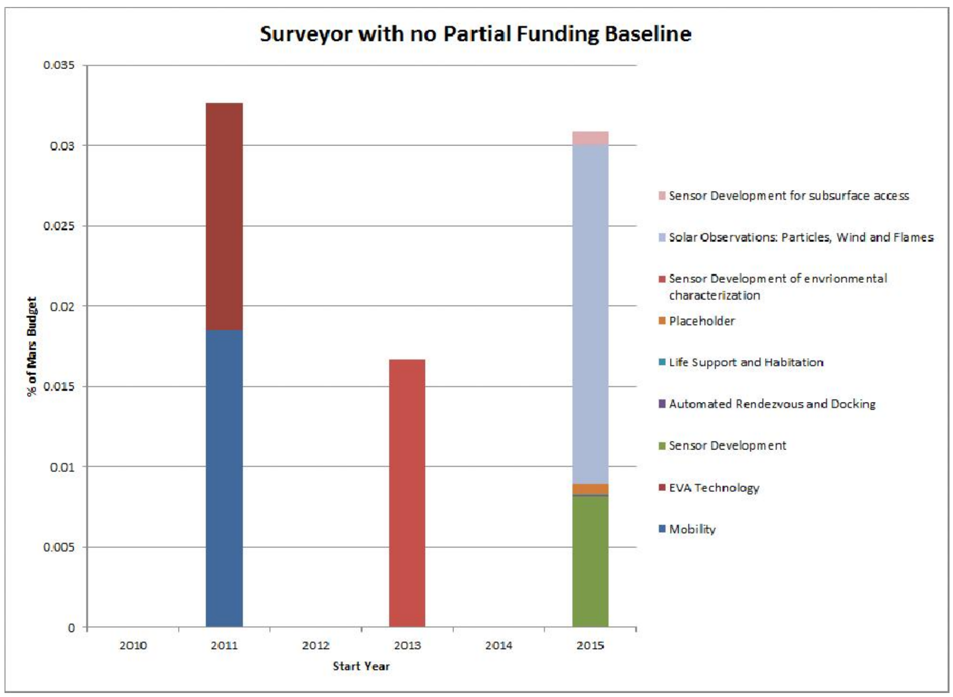


Figure 83: Surveyor Dependency Baseline

Moving towards how fast the changes happens gives the first derivative of the objective value. In this case the changes in dependency levels do not mean as much due to the fact that the dependency levels are not continuous changes. They are incremental changes and do not occur continuously like the time or expected input percentage changes. This can be seen in Figure 84 and Figure 85.

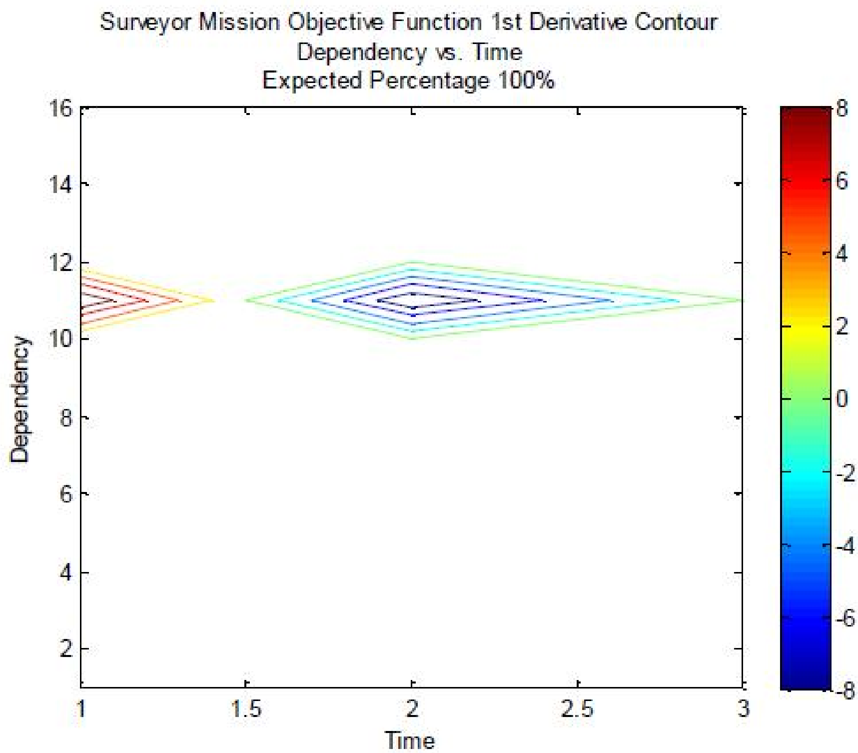


Figure 84: Surveyor 1st Derivative Objective Function for Expected Percentage 100%

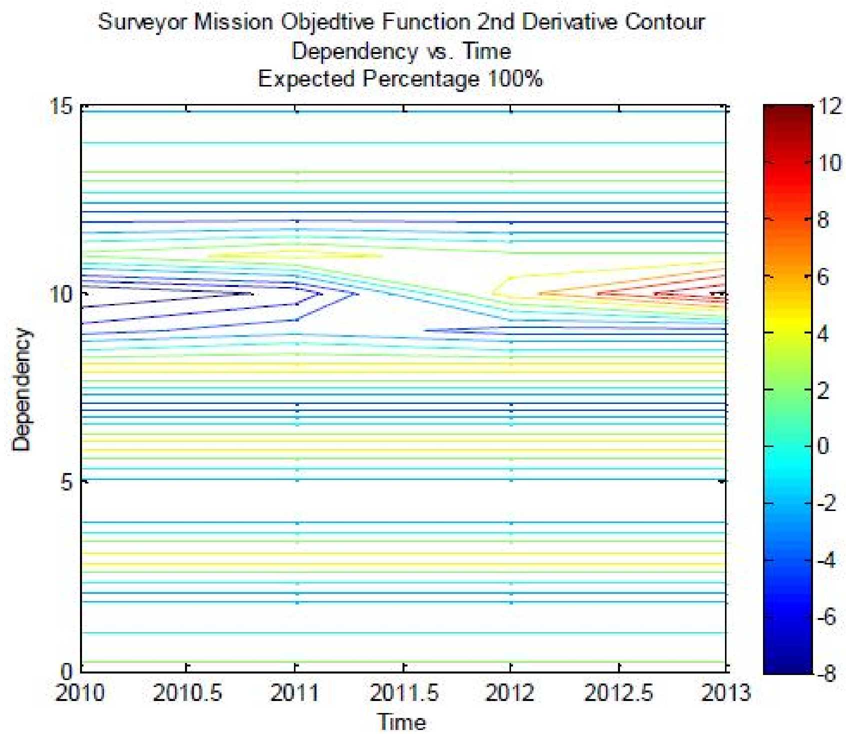


Figure 85: 2nd Derivative for Surveyor with no Partial Funding

9.2.1 No Partial Funding Surveyor Summary

Looking through the no partial funding of the Surveyor shows that there are two step portfolio values that exist. However, the actual portfolio are different due to when they are funded. Adding dependencies, with no partial funding, show that changes occur. The no partial funding aspect shows that there are very low portfolio values given, but more importantly that a baseline value has the largest value possible. This is consistent with the hypothesis that the baseline changes, along with being the largest possible funding scenario.

9.3 Partial Funding Surveyor Case

Changing the focus on Surveyor from no partial funding to partial funding gives drastically different results than scenarios investigated previously. The Partial funding feature of START shows that a portion of the technology can be funded instead of the entire process. Running the partial funding baseline gives Figure 84. This is a completely different view from Figure 67. The baseline value was 109.76. This was a 22% increase in objective function value.

The first differences between the two baselines are that more technologies are funded. Comparing these portfolios side by side as in Table 41 shows the difference in the portfolios. Allowing partial funding of the process funded 12 technologies vs only 9 technologies with no partial funding scenario. Every technology selected in the no partial funding scenario was chosen in the partial funding scenario. However, the four technologies chosen with partial funding technology were not chosen in the no partial funding scenario. There is no guarantee that technologies will follow the trends in this example; however, it does reinforce the concept that there are core technologies associated with a technology portfolio.

Looking at Table 41 shows that the In-Space Chemical Propulsion, High Efficiency Space Power Storage and Heavy Lift Propulsion technology were not chosen when the

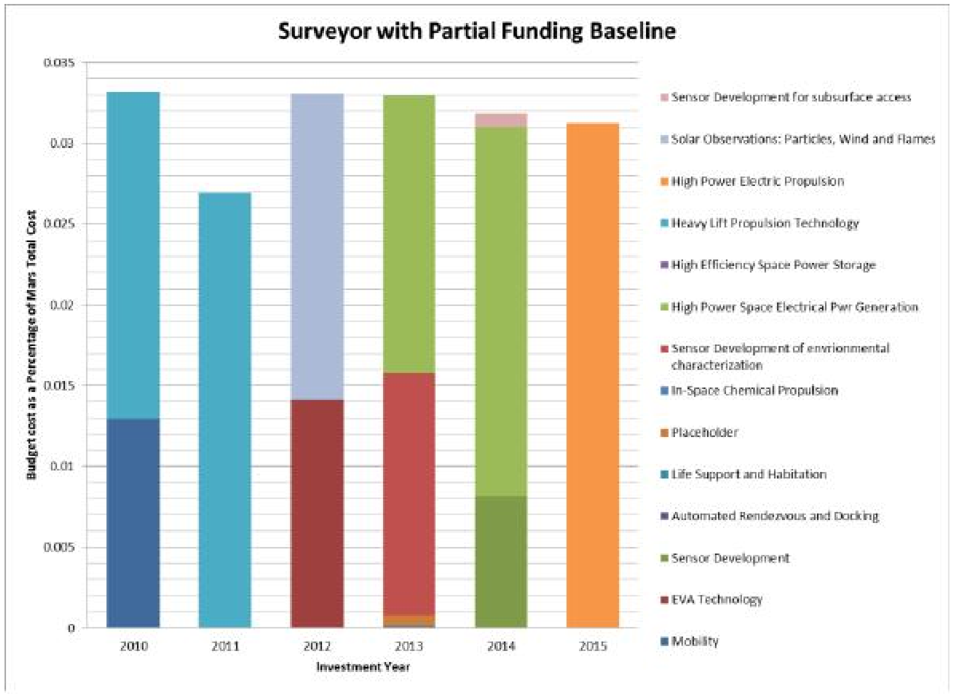


Figure 86: Surveyor with Partial Funding Baseline

Table 41: Surveyor Baseline Portfolio

Technology	Partial Funding Selection	Partial Funding	No Partial Funding Selection	Partial Funding
Mobility	Selected	0.7	Selected	1
EVA Technology	Selected	1	Selected	1
Sensor Development	Selected	1	Selected	1
Automated Rendezvous and Docking	Selected	1	Selected	1
Life Support and Habitation	Selected	1	Selected	1
Placeholder	Selected	1	Selected	1
In-Space Chemical Propulsion	NOT Selected		NOT Selected	
Sensor Development of environmental characterization	Selected	0.9	Selected	1
High Power Space Electrical Power Generation	Selected	0.6	NOT Selected	
High Efficiency Space Power Storage	NOT Selected		NOT Selected	
Heavy Lift Propulsion Technology	Selected	0.4	NOT Selected	
High Power Electric Propulsion	Selected	0.8	NOT Selected	
Solar Observations: Particles, Wind and Flames	Selected	0.9	Selected	1
Sensor Development for subsurface access	Selected	1	Selected	1

technology had to be all of nothing funded. Partial Funding scenarios did not select the In-Space Chemical Propulsion and High Efficiency Space Power Storage. The same four dependencies were added to this process as shown in Table 42. Individual dependencies were added to the process first and then compounded with the others in a combinatorial process which ended in dependency level 15.

Comparing Table 41 and Table 42 shows that Dependency 1 and 3 are not upheld,

Table 42: Dependency Combinations

Dependency Number	Capability A	Capability B	Operation	Included in dependency level
1	High Power Space Electrical Power generation	High space Power Storage	A Needs B B Needs A	1,5,6,7,11,12,13,15
2	In-Space Chemical Propulsion	EVA Technology	B Needs A	2,5,8.9,11,12,14,15
3	High Power Electric Propulsion	In-Space Chemical Propulsion	A Needs B	3,6,8,10,11,13,14,15
4	Solar Observations	Sensor Development for Environmental Characterization	A Needs B	4,7,9,10,12,13,14,15

so this portfolio would change the investment decisions these relationships are not modeled by the baseline. Dependency 2 and 4 are currently upheld, so adding them would have no affect. Applying these dependencies to the solution and looking at the first start year shows Figure 87.

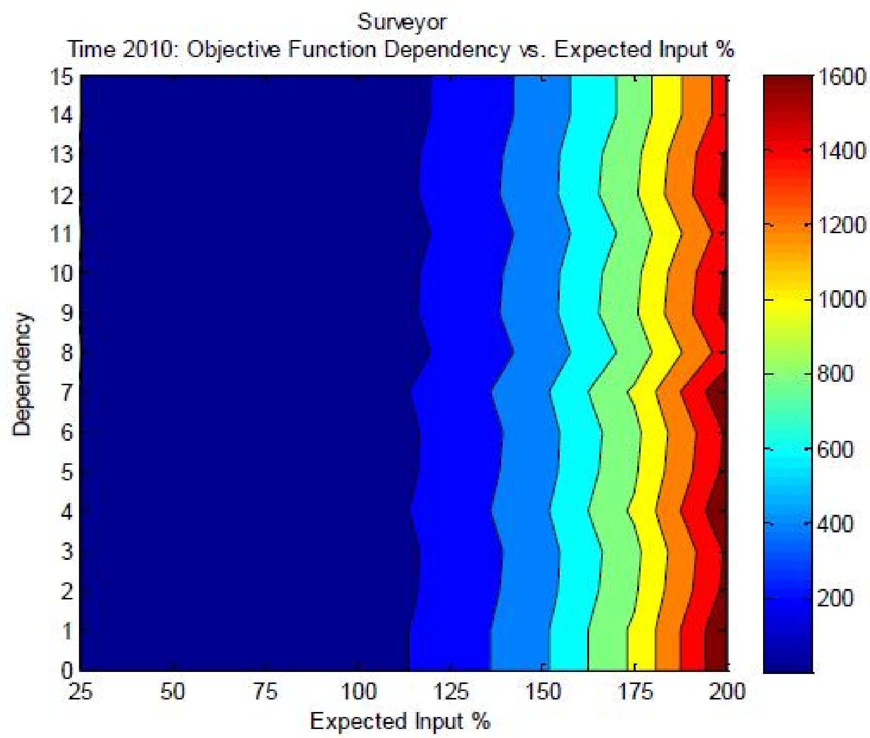


Figure 87: Surveyor Portfolio Contour Plot at starting year 2010

Figure 87 shows the colorbar on the right has a large variation in the portfolio objective value magnitude. While there is phenomena going on in the changing of adding dependency relationships into the process, there is quite a large change in the magnitude of the portfolio objective function.

Adding dependencies to the Surveyor baseline caused the portfolio to decrease.

Changing the expected input percentages increased the value if the expected percentage was over 100% and decreased the portfolio if the input change was below 100%.

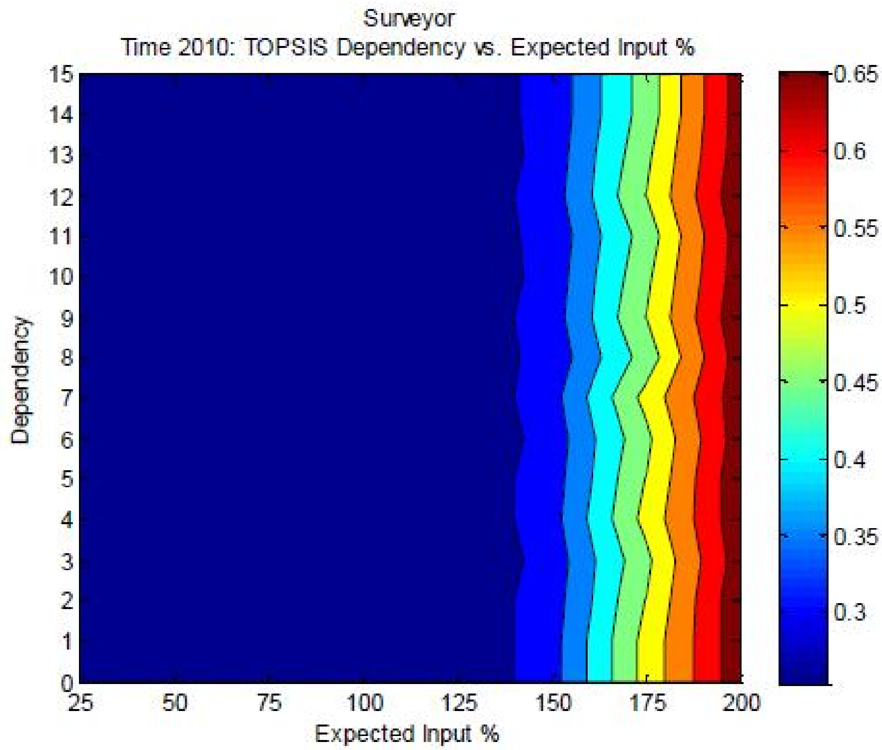


Figure 88: Surveyor Portfolio TOPSIS Value Contour Plot for starting year 2010

The baseline TOPSIS value is 0.35. This takes into account the objective function, cost, start year, partial funding, and flexibility term associated with the technologies being optimized. Therefore looking at the TOPSIS contour plot in Figure 88 shows that there is a change of 0.65 or 65% in the TOPSIS values when viewing the relative change with respect to other portfolios. Comparing the no partial funding and the partial funding case gives the same levels for both the objective value and the TOPSIS value; however, the TOPSIS value gave higher values than the partial funding scenario.

The change of the input file with respect to the overall portfolio, shows in Figure 89 that it is a polynomial change as the expected input percentage increases. This

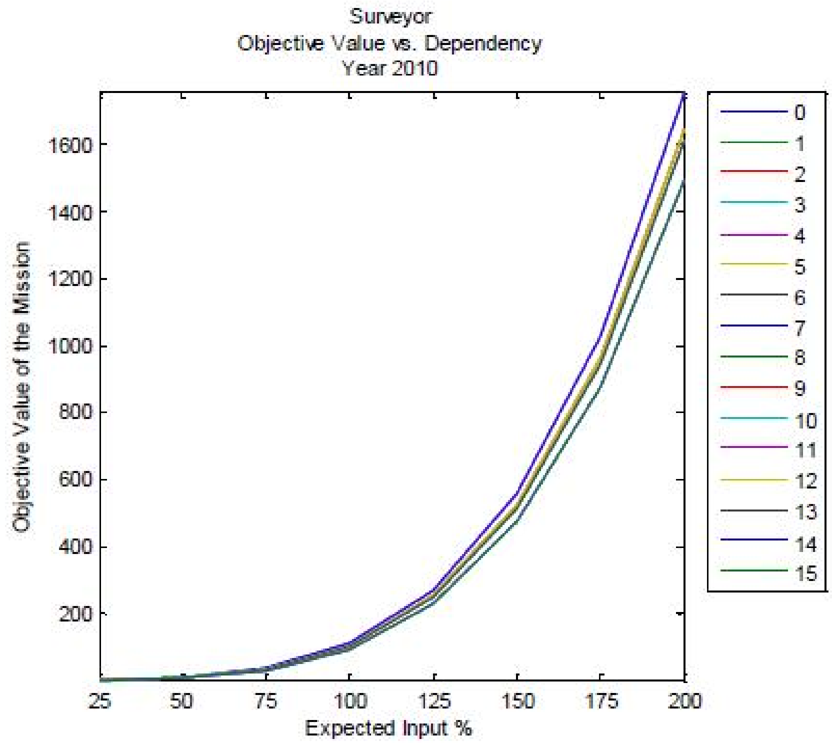


Figure 89: Surveyor vs. Expected Input Percentage Objective Function starting in 2010

is not a perfect polynomial function, but it has that general shape for all of the dependency changes. They are almost unanimous below the 100% mark. They start to vary when the user expect optimistic values for their input file. The same trend was seen in the no partial funding sample.

The TOPSIS values gives a similar feeling of a polynomial fit, but has much more chaotic changes throughout the function. TOPSIS takes into account non-linear changes such as the flexibility and objective value. Since the cost and starting year are uniform for all of this data, the real driving factors are the objective value and flexibility quality of the technologies. The objective value seems to dominate since it gives the same polynomial shape as shown in Figure 89.

Figure 91 shows that as dependencies are added, the portfolio value goes down.

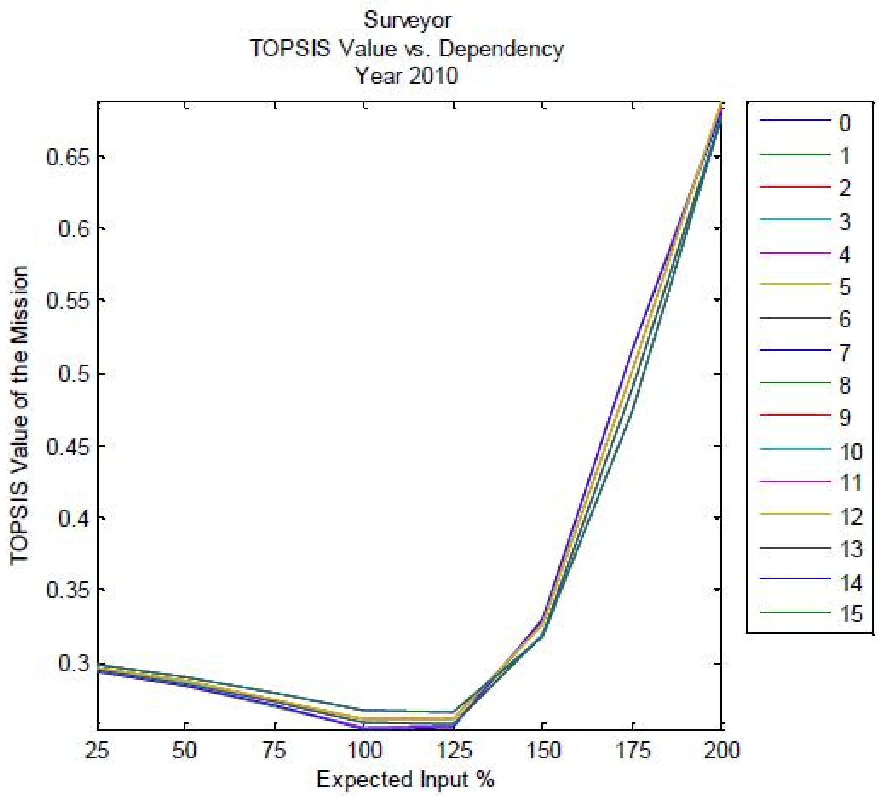


Figure 90: Surveyor vs. Dependency TOPSIS Value stating in 2010

The highest value is always at Dependency = 0 and decreases throughout until Dependency = 15. This trend is similar in nature to the no partial funding scenario. Flashing back to Figure 72 shows a horizontal linear trend, compared to Figure 91 gives a sloped linear look at the changes in dependencies.

Applying TOPSIS to Figure 92 shows how the portfolio changes for each one of the expected input percentages. These values are dominated by the objective value as well, but does not show a pattern as the previous graphs did. The reason is that TOPSIS does gives the relative difference between portfolios, instead of the absolute difference. Adding TOPSIS analysis onto the data makes the data more linear.

Switching over to the derivatives of the changes, the objective values and TOPSIS function are shown Figure 93 and Figure 94. These derivatives give similar pictures of the no partial funding levels. The levels that have the highest levels for partial

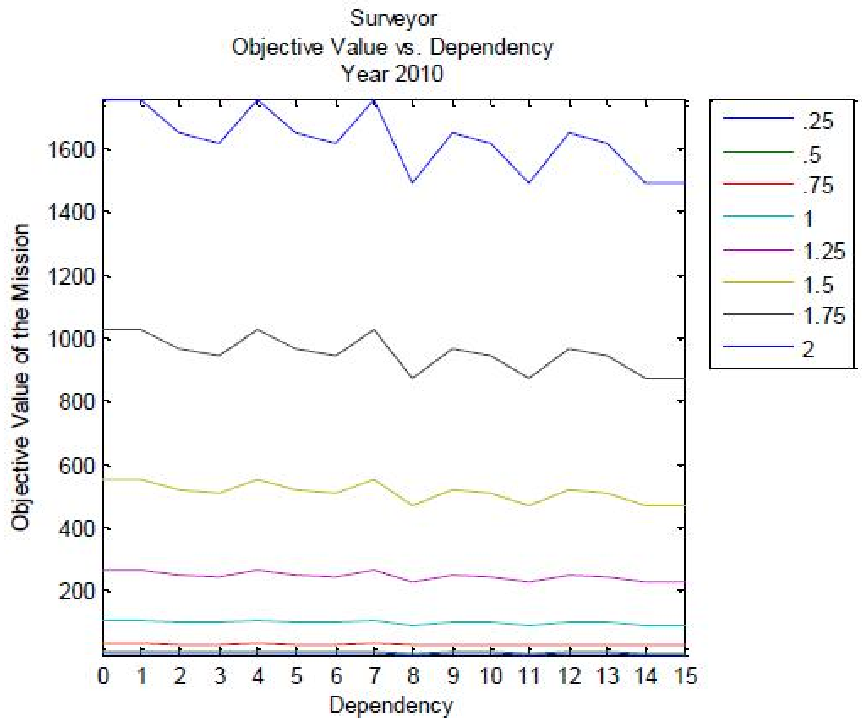


Figure 91: Surveyor Objective value vs. Dependency for 2010

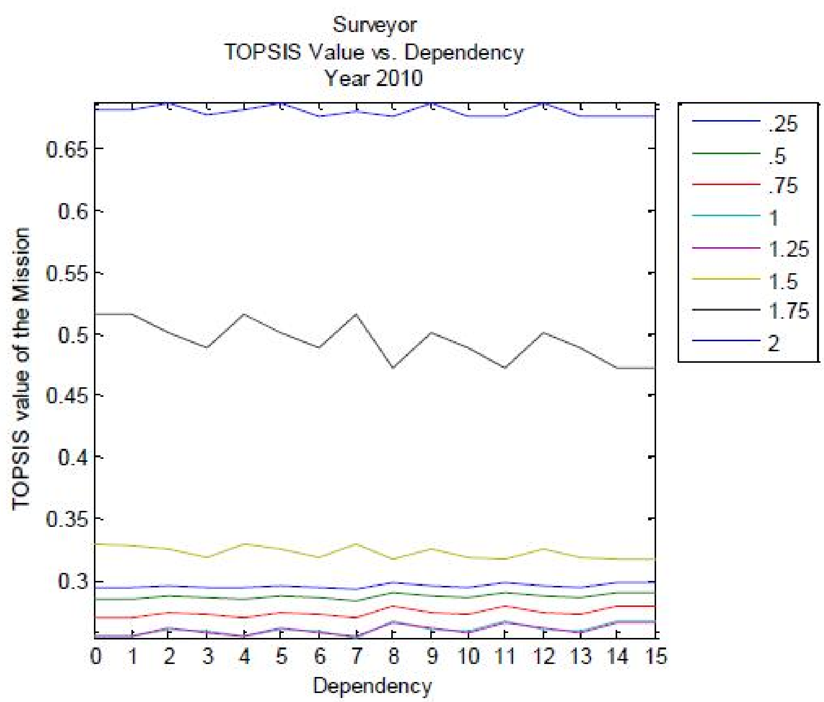


Figure 92: Surveyor TOPSIS value vs. Dependency for 2010

funding (2,3,5,6,8,11,14, and 15) show the changes in the second derivative.

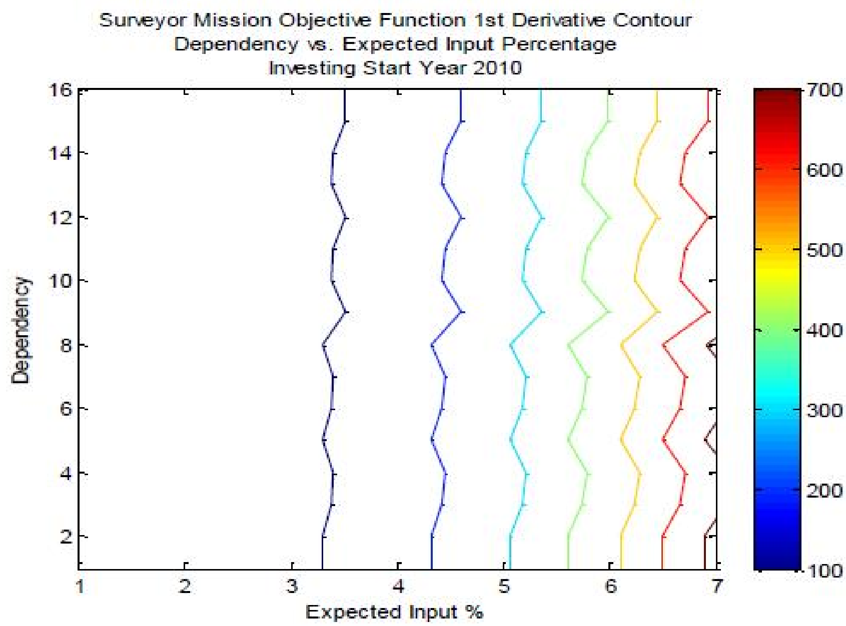


Figure 93: Surveyor partial funding 1st derivative

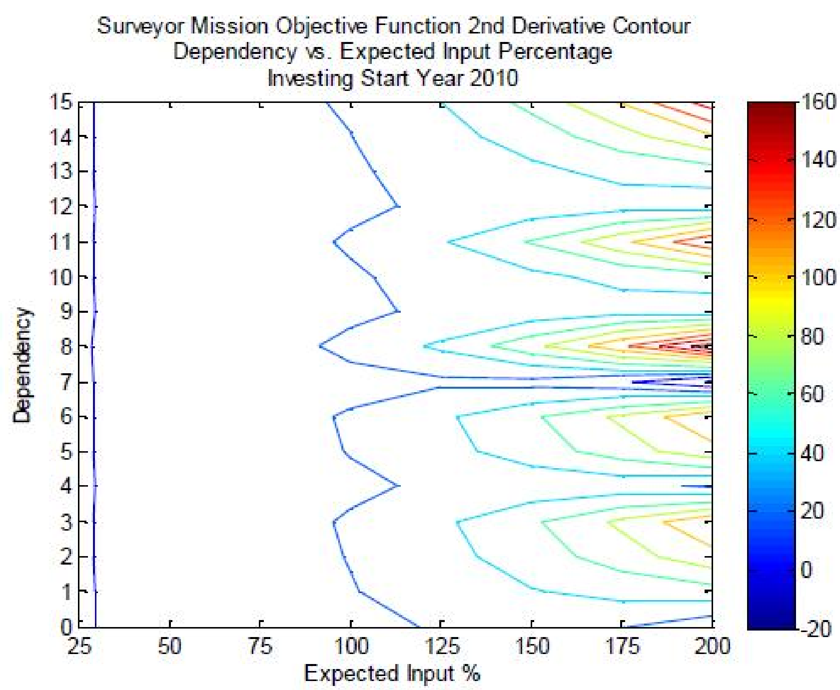


Figure 94: Surveyor Partial Funding 2nd Derivative

9.3.1 Surveyor Summary

The Surveyor partial and no partial funding in the first start year shows that there are similar trends that occur, but give extremely different portfolios.

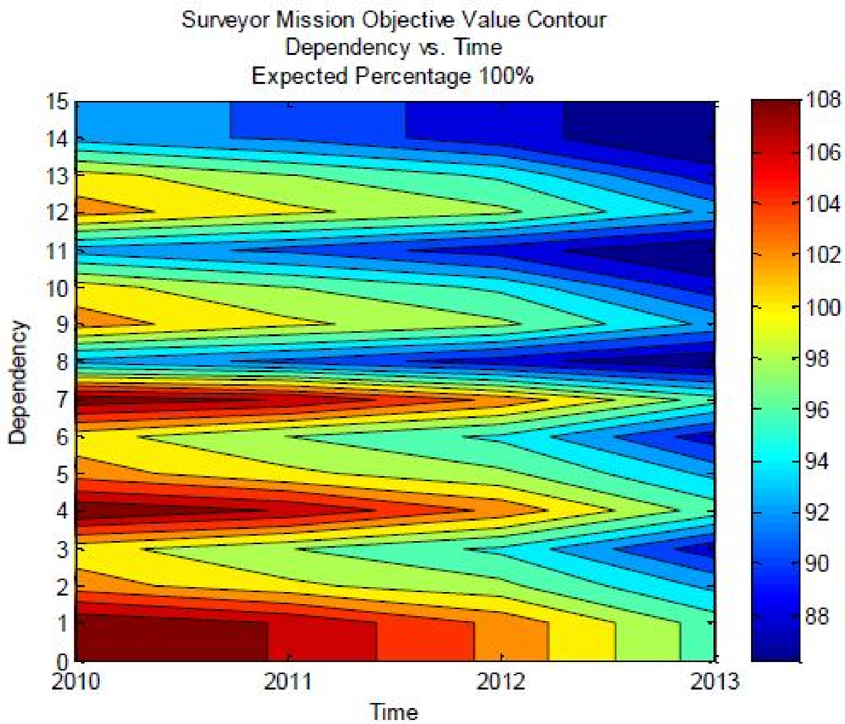


Figure 95: Surveyor Mission Objective Value Contour for Expected Percentage of 100%

In order to do a bottom up analysis, the user must have a certain objective function and TOPSIS change in mind in order to determine what their threshold to invest would be or not. Looking at Figure 95 shows that in order to get a 10% increase in the objective function baseline portfolio of 109.79 to 120.77 is not possible. This is known by looking at the colorbar in Figure 95. However, if the user desires to include all of the dependencies studied in Table 31, then they would have to accept a 85% decrease in objective value. Adding dependencies into the process decreases the objective value if the dependency are Constraint Dependencies. Looking closely at Figure 95 shows that the reddest part is around the Dependency level 0, 1, 4 and 7. The cost of adding fidelity is the reality that the expected value is longer than originally projected. The user must determine if this is acceptable. This fidelity gives the real world value and expectations.

Table 43: Surveyor Bottom Up Table

Dependency	Starting Year	Objective Value
0	2010	109.76
1	2010	109.76
4	2010	109.76
7	2010	109.58

Table 43 shows that given the current input value and expected objective function threshold of 109.29, these scenarios will actually accomplish the goal. Looking at this shows that the user must start during the first year and model three different levels of dependencies. Given what the user expects to be the inputs of their system, they are not able to accomplish their goals as well as model all of the dependencies associated with the process. Specifically they could not get a 10% increase from the baseline case by including the suggested dependencies modeled.

Switching over to the TOPSIS performance of Figure 95 shows a different story than strictly looking at the objective function. The TOPSIS colorbar shows that it is possible to increase 10% for a TOPSIS score. When dealing with TOPSIS the user is only looking for the highest value. In this case the relative difference being viewed takes into account multiple factors. Therefore TOPSIS provides a different type of bottom up and top down analysis based on the best case scenario. In this case leaving the input values the same as the user expects gives the best scenarios shown in Table 44.

Table 44 shows that the best choice comes during the last investment year of 2013. This implies that the start year dominates this choice. The reason is that the cost is lower since the development time is lowest. Cost has a 50% weighting in TOPSIS

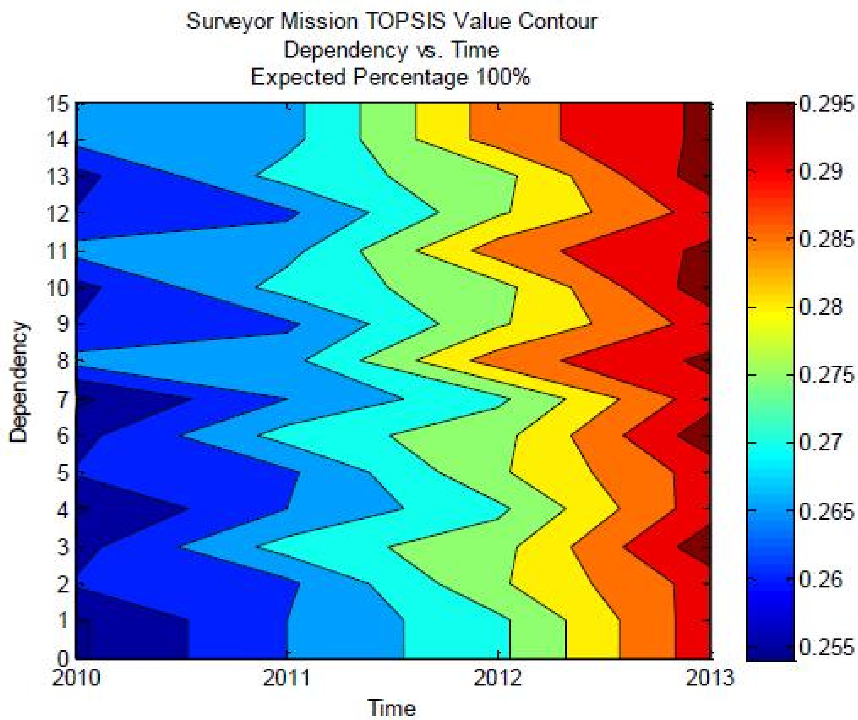


Figure 96: Surveyor Mission TOPSIS Value for Expected Percentage of 100%

Table 44: TOPSIS best values

Dependency	Start Year	TOPSIS Value
3	2013	0.298
6	2013	0.298
8	2013	0.296
10	2013	0.298
11	2013	0.296
13	2013	0.298
14	2013	0.296
15	2013	0.296

and ultimately dominates the decisions given to the user in this case.

Looking at Figure 91 in another light shows Figure 97. Figure 97 shows that

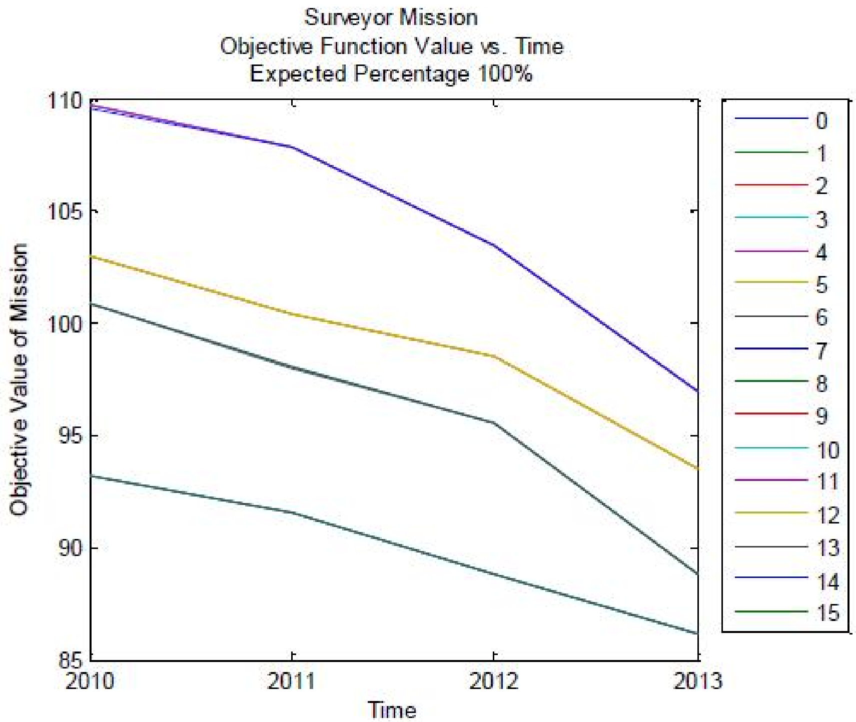


Figure 97: Surveyor Mission Objective Function vs. Time at 100% Expected Input Value

the highest objective values come in 109.76. Once again the line of No Dependency (Dependency = 0) has the highest values due to the fact that the inputs are only Constraint Dependencies. **It is not possible for the portfolio to be higher than the baseline with Constraint Dependencies**. Figure 98 in TOPSIS shows once again that the best scenarios happen in 2013 due to the lower cost that dominates the relative differences between portfolios. Figure 99 shows that as dependencies are added, the portfolio value goes down. The highest value is always at Dependency = 0 and decreases throughout until Dependency = 15.

Comparatively, the TOPSIS values follows the same trends as the Figure 98 where the top scenarios happen in 2013. This is extremely pronounced in Figure 99 where the TOPSIS value has a distinguished gap between 2013 and 2012. The 2010 values are the lowest here, although they show the highest objective values as seen in Figure 98.

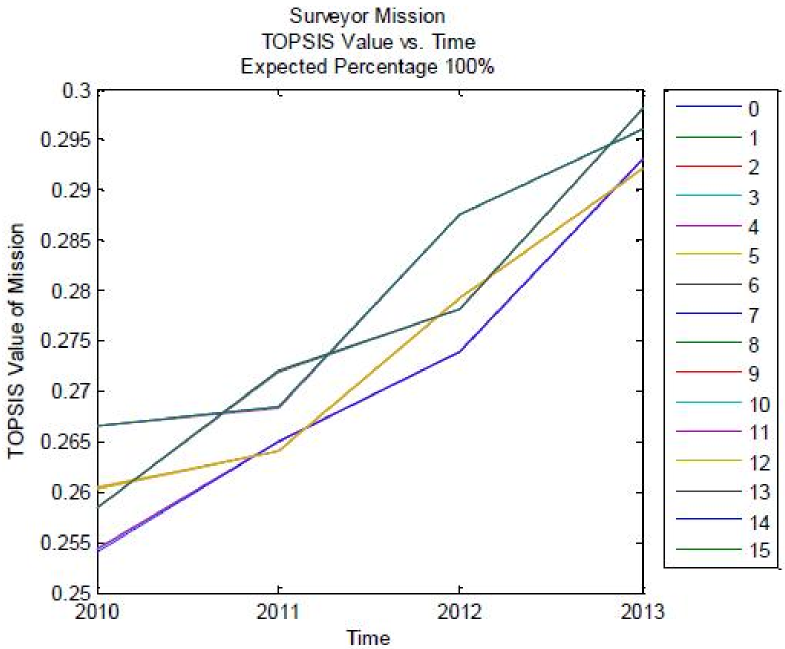


Figure 98: Surveyor TOPSIS Value vs. time for 100% expected percentage input value

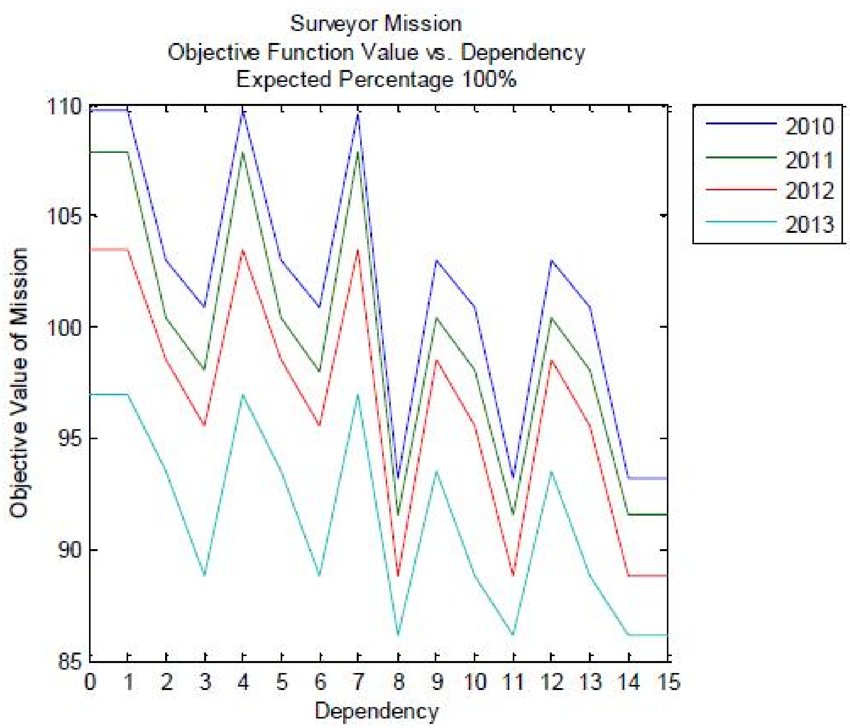


Figure 99: Explorer Mission Objective Function vs. Dependency at 100% Expected Input Value

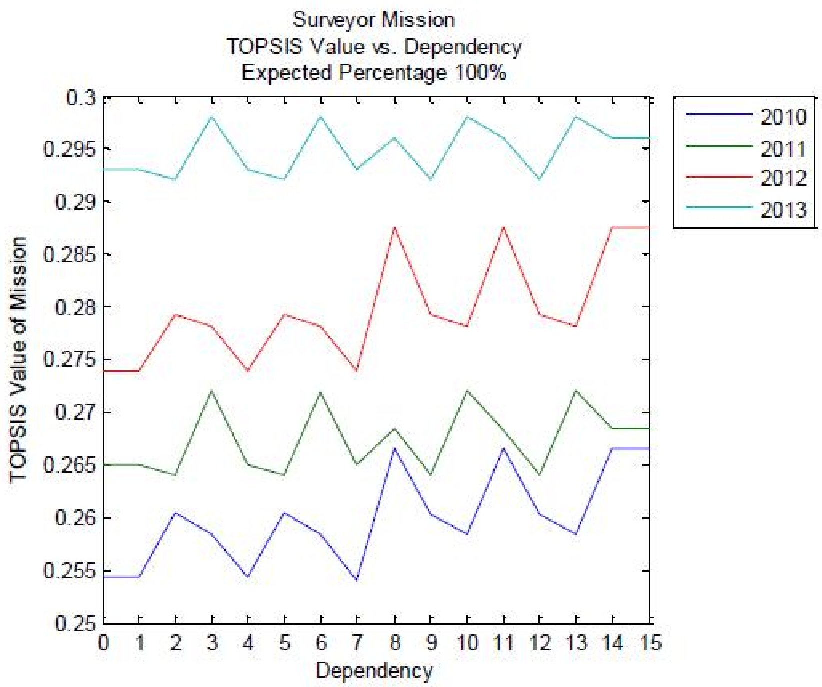


Figure 100: Explorer Mission TOPSIS value vs. Dependency

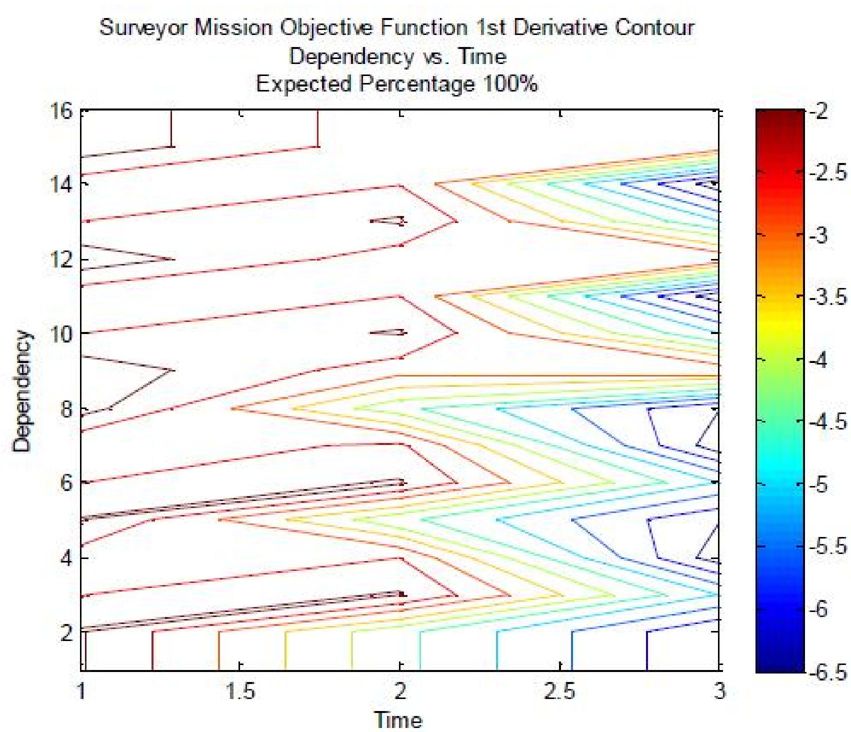


Figure 101: Surveyor Mission 1st derivative objective function contour plot

The first derivative shown in Figure 101 show the changes in the investment year rapidly changes from year to year. The changes in the dependencies are not continuous once again, so the vertical changes in Figure 99 are irrelevant since the dependencies may be implemented in any order. However, it shows that the objective value rapidly changes from year to year in the first derivative. Figure 102 shows the acceleration with the second derivative.

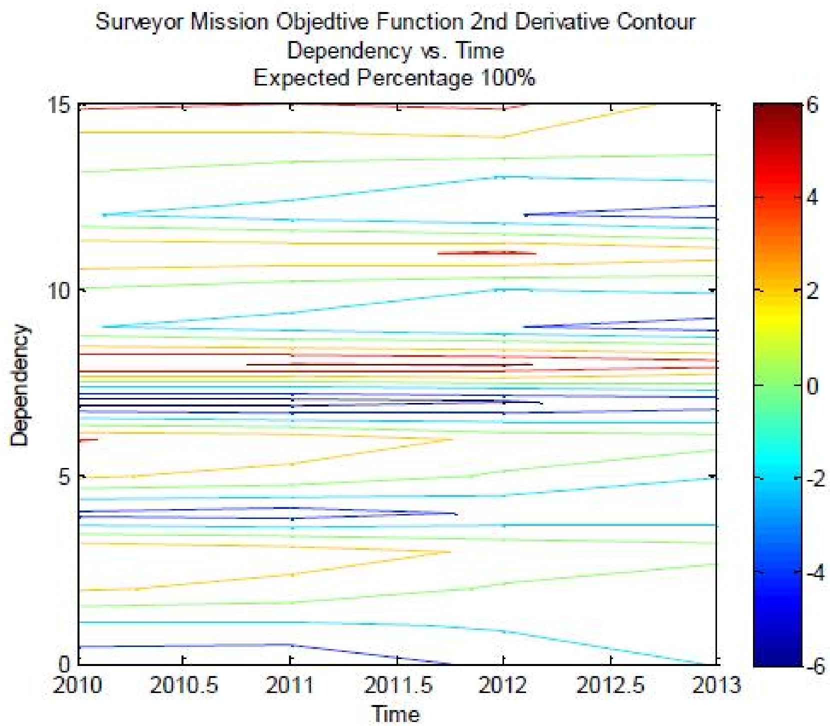


Figure 102: Surveyor Mission 2nd derivative objective function contour plot

The Laplacian in Figure 102 shows the acceleration in the investment time has stark differences for the specific dependency level. This shows that the acceleration changes over time, which has a large effect on the TOPSIS and objective values as shown in Figure 72 and Figure 73.

9.3.2 Surveyor Partial Funding Summary

Using the partial funding scenario showed that the Surveyor showed a 22% increase in the objective value. The dependencies modeled where constraint dependencies.

These dependencies were already in the baseline and mostly already modeled. The last minute additions gave the highest TOPSIS values in the partial funding scenario although it gave the lowest objective value. The most important aspect of this section is the fact that constraint dependencies do not allow the value to go above the baseline value. This is true whether partial funding is used or not.

CHAPTER X

EXPLORER RESULTS

The Explorer mission is set further out than the Surveyor mission. Therefore a 9 year period from 2010 until 2019 was analyzed. If the user starts with the baseline once again and progress through the dependency inclusion they see multiple sets of information. The baseline is different from the Surveyor mission in the sense that there were more technologies studied, more time to develop, and a higher budget than the Surveyor mission. The Explorer mission makes contact with the asteroid rather than doing the remote sensing aspect that Surveyor provides. Therefore, the Explorer has more EVA and hands-on activities associated with the optimization.

10.1 Baseline

Starting with the baseline Explorer mission gives Figure 103 below. Explorer is studied at 6.6% of the total Mars Budget compared to the 3.3% of the Surveyor expected budget. If the user looks at the technologies chosen they will see Table 45 along with the partial funding schedule.

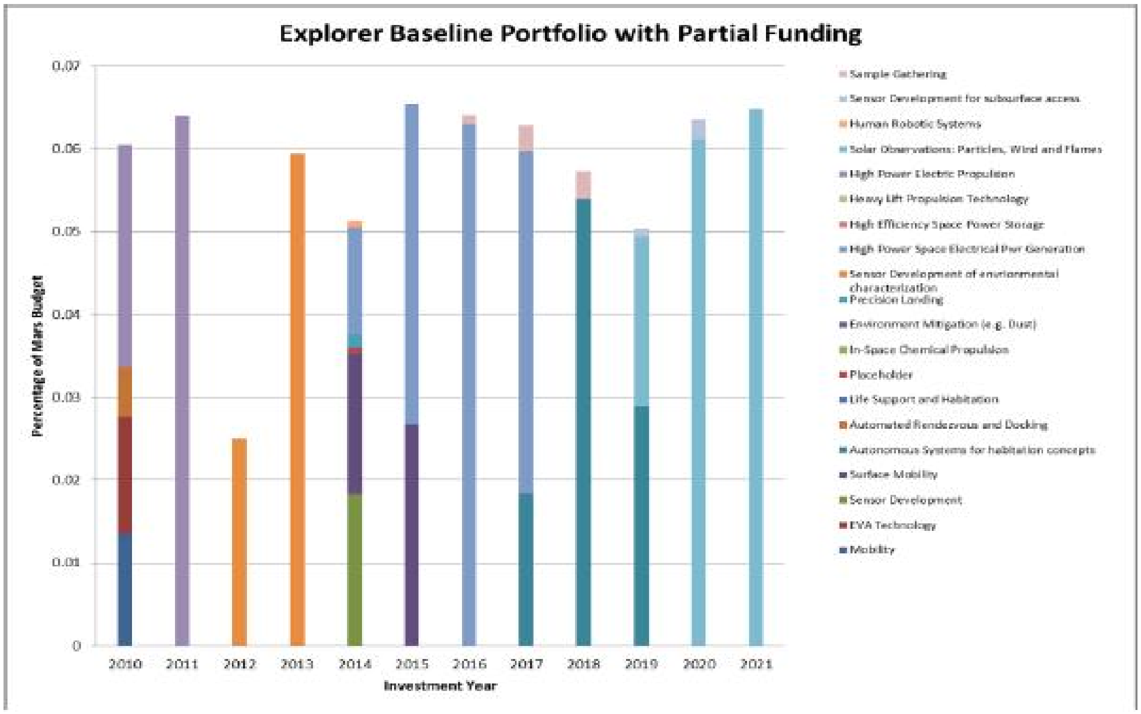


Figure 103: Explorer Baseline

Table 45: Explorer Baseline Portfolio

Capability	Selection	Partial Funding
Mobility	Selected	1
EVA Technology	Selected	1
Sensor Development	Selected	1
Surface Mobility	Selected	0.8
Autonomous Systems for habitation concepts	Selected	0.9
Automated Rendezvous and Docking	Selected	1
Life Support and Habitation	Selected	1
Placeholder	Selected	1
In-Space Chemical Propulsion	NOT Selected	
Environment Mitigation (e.g. Dust)	NOT Selected	
Precision Landing	Selected	1
Sensor Development of environmental characterization	Selected	1
High Power Space Electrical Power Generation	Selected	0.5
High Efficiency Space Power Storage	NOT Selected	
Heavy Lift Propulsion Technology	NOT Selected	
High Power Electric Propulsion	Selected	0.6
Solar Observations: Particles, Wind and Flames	Selected	0.8
Human Robotic Systems	Selected	1
Sensor Development for subsurface access	Selected	1
Sample Gathering	Selected	1

Four Constraint Dependencies were added to this process in the order shown in Table 46. The original baseline would affect the first dependency given. All of the other relationships were actually modeled in the baseline. Therefore adding them into the process should not affect the optimized portfolio.

Table 46: NEA Explorer Dependencies tested

Capability A	Capability B	Operation	Included in dependency level
High Power Space Electrical Power generation	High space Power Storage	A Needs B B Needs A	1,5,6,7,11,12,13,15
Mobility	EVA Technology	A Needs B B Needs A	2,5,8.9,11,12,14,15
Sensor Development for EVA	EVA Technology	A Needs B	3,6,8,10,11,13,14,15
Surface Mobility	EVA Mobility	A Needs B	4,7,9,10,12,13,14,15

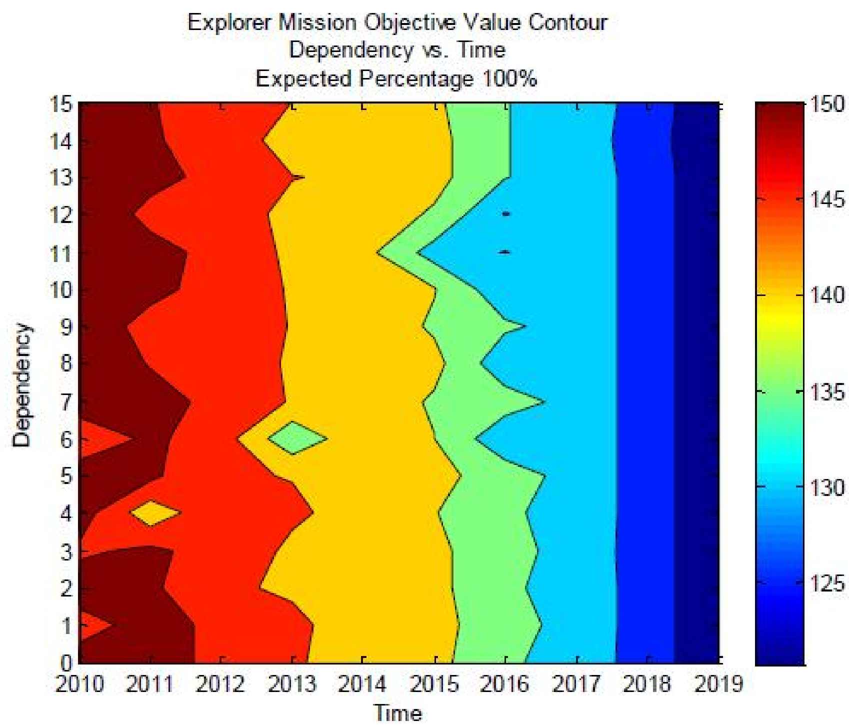


Figure 104: Explorer Design Space Exploration with initial User Inputs

If the user initially trusts their input values and stays with an expected percentage level of 100% of their inputs, they would see Figure 104. Looking at Figure 104 shows that adding dependencies to the process does not give the smooth vertical lines seen in

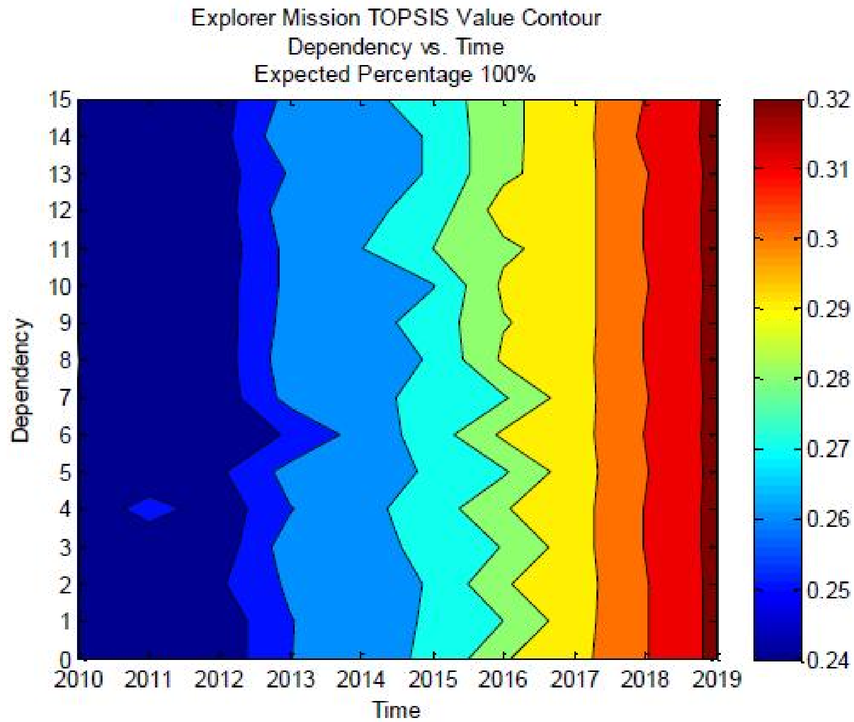


Figure 105: Explorer TOPSIS Value Contour

the Surveyor mission case. Here, the addition of dependencies changes the portfolio values within the same year for years below 2017 as seen in Figure 104. The first dependency in Table 46 is the one that was not chosen in the baseline. Looking at Figure 104, shows that level 1 and 6 have lower values than that of the baseline. The interesting phenomena happens in the TOPSIS contour plot as seen in Figure 105.

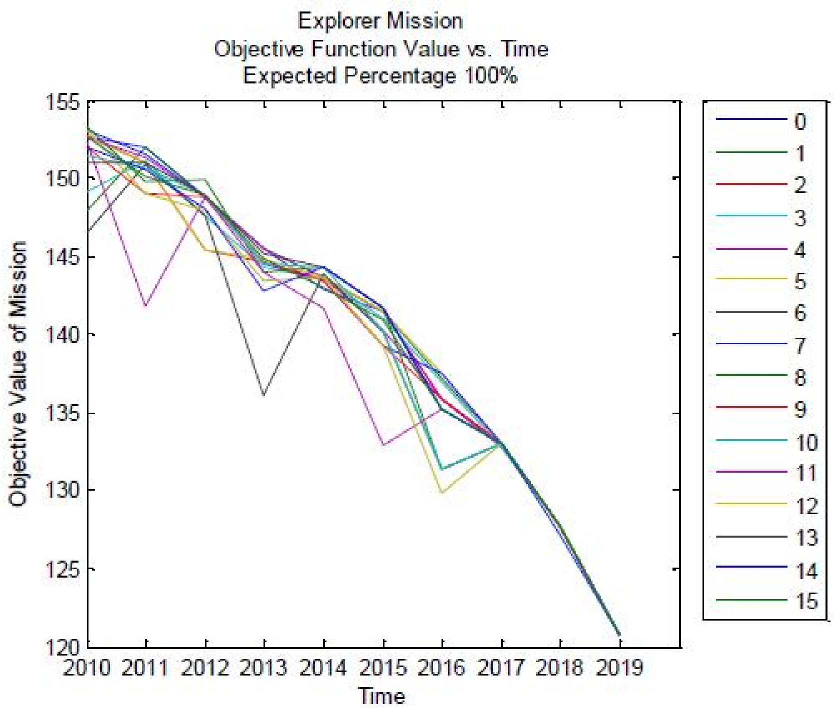


Figure 106: Explorer Mission Objective Function vs. Time at 100% Expected Input Value

Moving to the TOPSIS values of Figure 104 gives Figure 105. Like Surveyor TOPSIS gives slightly inverse values of the objective value. This is because the cost is heavily weighted in TOPSIS and changing the start time effectively reduces the cost. It is important to notice here that the 2010 and 2011 has TOPSIS values that are not shown in the contour plot because they are around the 0.24 minimum value.

Looking at Figure 104 in another light shows Figure 106. Figure 106 shows that the highest objective values come in 2010 for multiple cases around 154 objective value. This even higher than the baseline value of 152.6. The reason for this is that the capability values are slightly different due to the probability values. Therefore the changes that are above the baseline value are within a expected range given the probability capabilities from the START program.

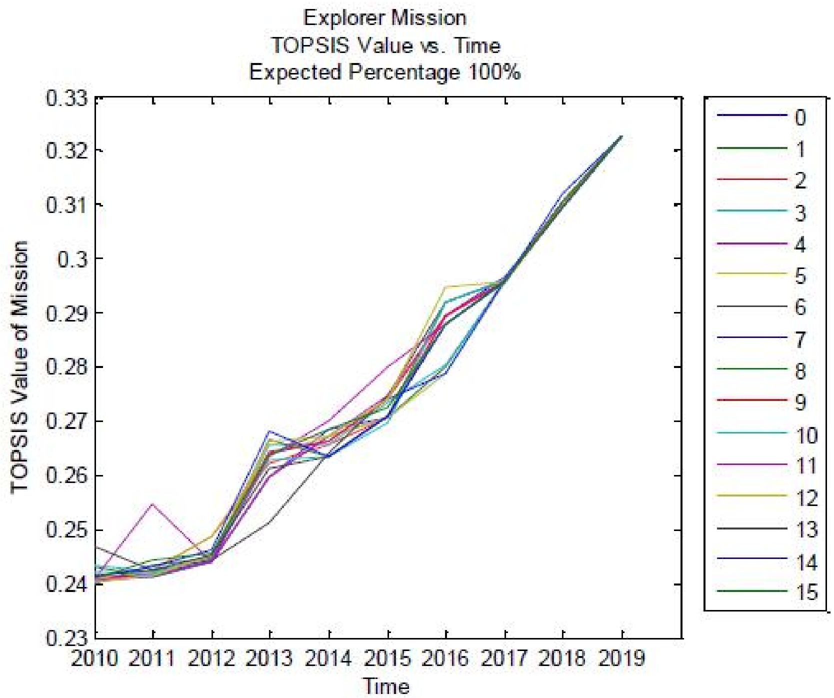


Figure 107: Explorer TOPSIS Value vs. time for 100% expected percentage input value

Figure 107 are dominated by the cost and start year which drive the TOPSIS value up. The Explorer TOPSIS values do not follow along smoothly as the Surveyor lines. They are more volatile than the Surveyor mission technologies having such loose constraints and development time frame as well as partial funding aspects that change the entire dynamic of the optimization process.

Trusting the users input values show that adding dependencies and relationships to the process actually change the objective value. It is possible to change the objective value only negatively utilizing Constraint dependencies. It is extremely important whether or not they are considering partial funding scenarios. It creates more feasible solutions, but it does not fund technologies fully. If this is the case, then that brings into question if the user expects the project to come to fruition at a partial funding level or if the technology has the possibility of getting alternative funding. There may be other circumstances that come into play, but it most certainly changes the dynamic of adding dependencies into the technology selection process.

10.2 Bottom Up and Top Down Analysis

Realizing that adding dependencies is not necessarily a detriment to the objective function allows the user to proceed to determine bottom up and top down analysis with care. In order to do a bottom up analysis, the user must have a certain objective function in mind in order to determine what their threshold to invest would be or not. Figure 106 shows the changes for the entire design space when dependencies and time are changed for having no dependencies added to the process.

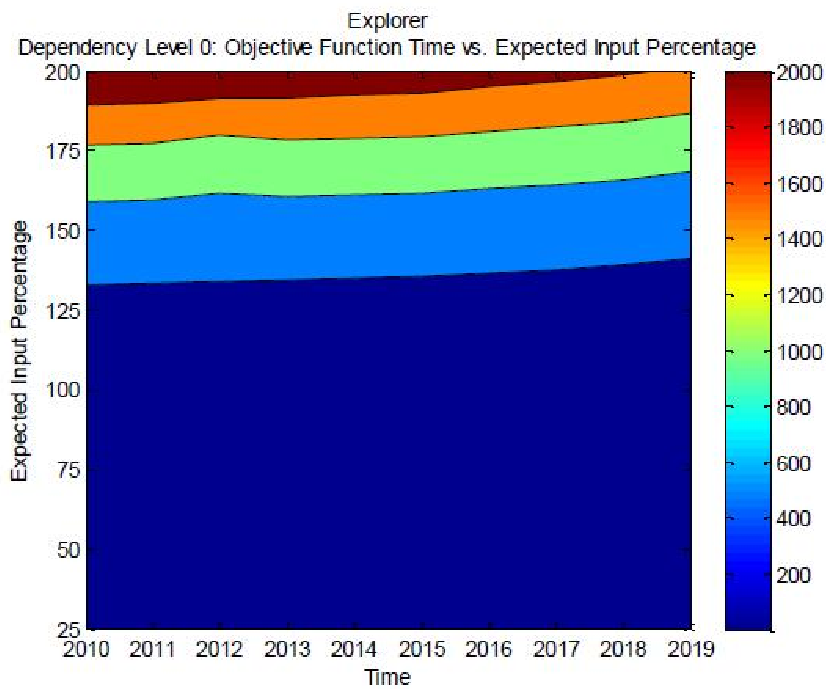


Figure 108: Explorer Mission Objective Value Contour for Expected Percentage of 100%

Figure 108 shows that given the current input value and expected objective function threshold of 150, these scenarios will not accomplish the desired objective value. Looking at this shows that the user can start anytime during this process without having to model the dependency relationships as long as the users input values are increase at least 10% across the board. Given what the user expects to be the inputs of their system, they are not able to accomplish their goals as well as model all of the dependencies associated with the process.

This phenomena changes with respect to TOPSIS as shown in Figure 109. Looking at it globally in Figure 109 shows that it is not the most desirable, but for a threshold of 150 as the corresponding TOPSIS values are around the same value. If the user actually models the dependencies they were considering they will get a different objective value and corresponding TOPSIS contour.

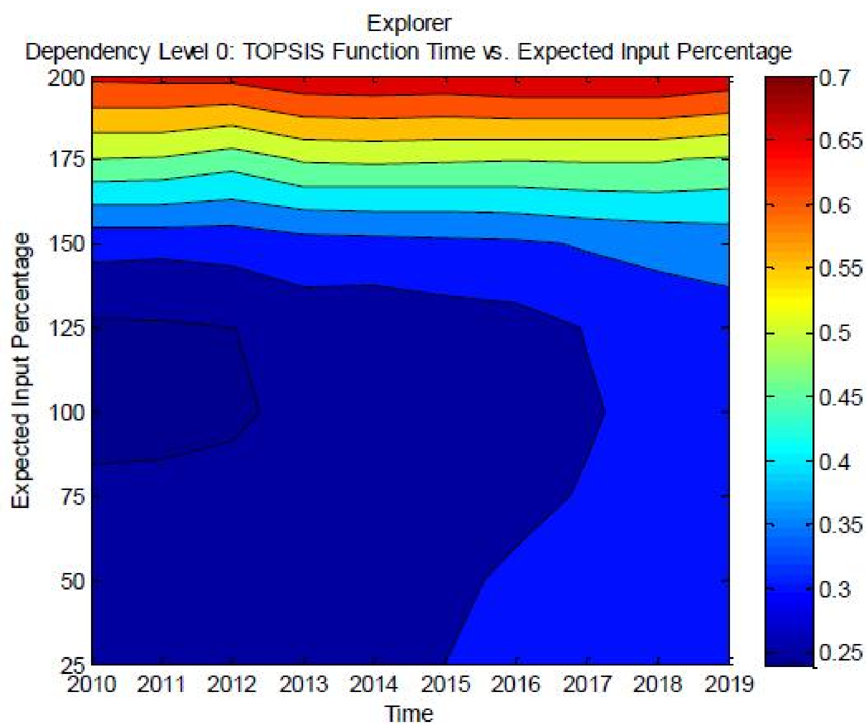


Figure 109: TOPSIS Explorer Mission

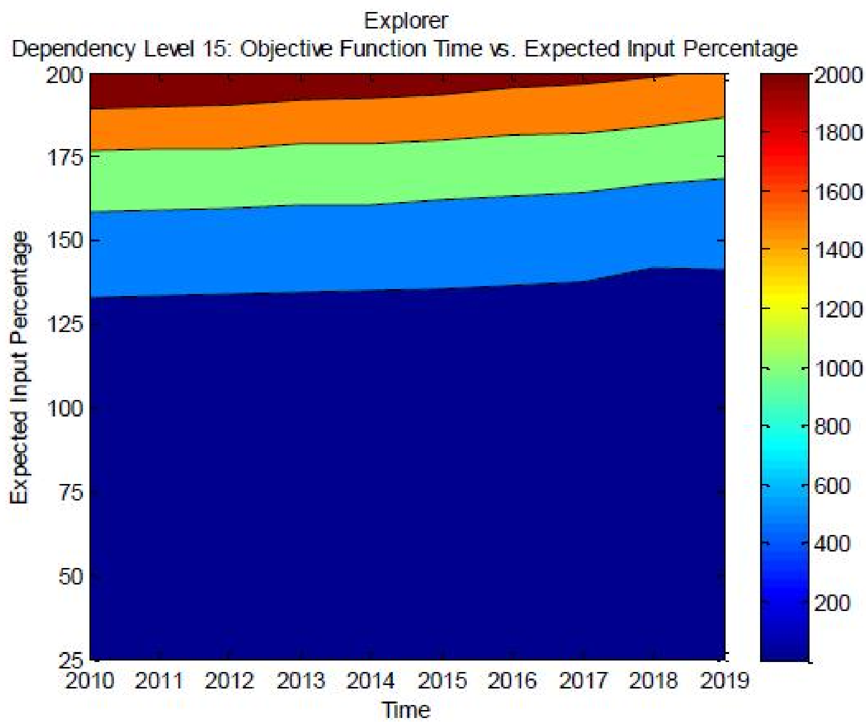


Figure 110: Explorer Full Design face for Dependency Scenario 15

Focusing in on a objective value of 150, and keeping around the user's input values, provides Figure 110. Once again the user must increase the aspects of the inputs from 5%-10%, but the shape is quite different depending on when the changes occur.

Looking at TOPSIS for the entire design space shows that there is interesting phenomena going on for the start year of 2011. In this case there is a clear choice of using the user's input value for the TOPSIS selection. While this TOPSIS will not get the desired 150 objective function, it will however, give the highest relative difference using the user's input values.

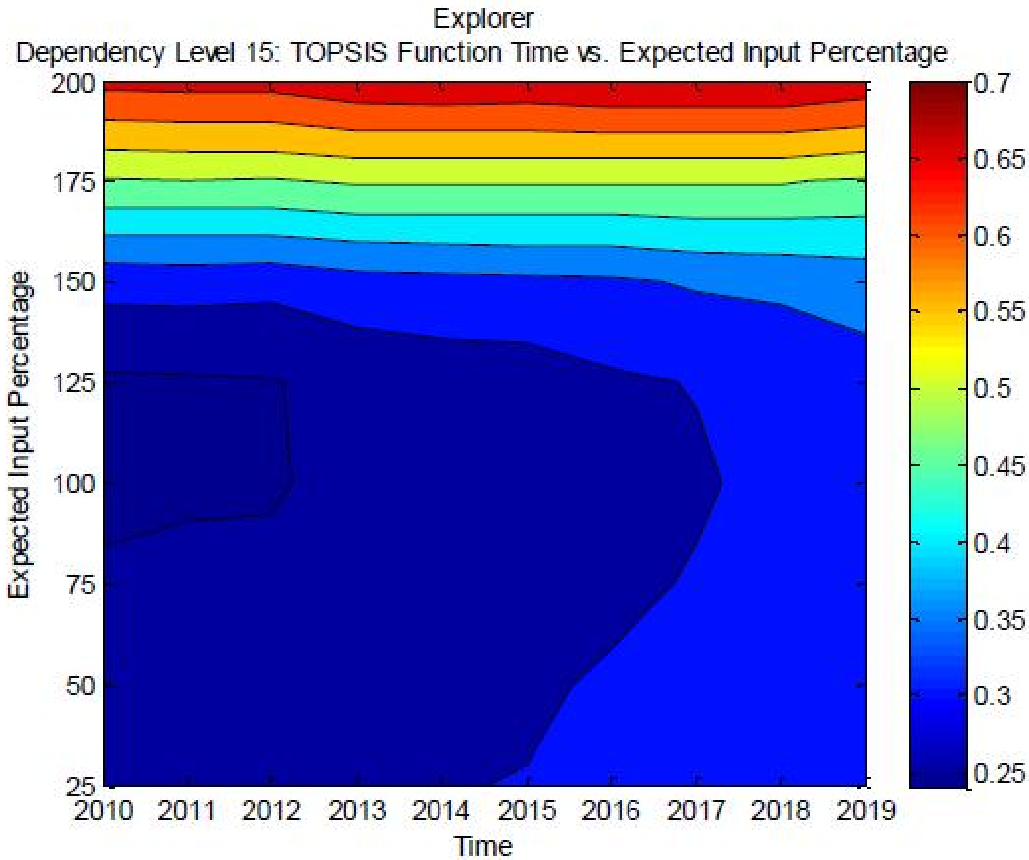


Figure 111: Explorer full design space for Dependency Level 15

10.2.1 Explorer Summary

The Explorer Mission highlights the effect of having a larger design space to investigate. As mission accumulate to later missions, the design space opens up to encompass what the earlier missions may actually contribute. In the case of Explorer the mission is set to actually go to a NEA and take data and information making the way for the NEA human mission. Therefore the technologies focused more on the environmental sensor issues rather than remote sensing issues. These are technologies that take longer to develop with higher cost values. This is all done on only 6.6% of the total Mars budget and 20% of the NEA exploration budget.

10.3 Explorer Temporal

More information can be gathered from the data cube by changing the axes of information. The user can investigate when they should actually invest in the capability. Looking at the changes as the starting year changes will give some insight into this aspect of the technology portfolio selection. START already determines what the best schedule is for the portfolio to be optimized in their software. However, this temporal study adds the aspect of looking at multiple start years and development time frame. The START results show the user when to fund a technology given specific start years and a set investment time frame. This design space exploration effectively changes that investment time frame.

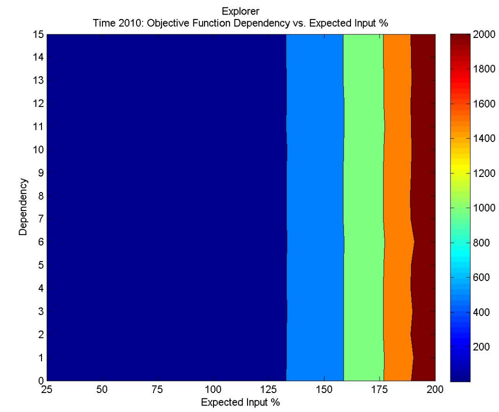
Starting with Explorer, shows that the original objective function was 152.63. Figure 189a shows the colorbar on the right has a large variation in the portfolio volume magnitude. While there is phenomena going on in the changing of adding dependency relationships into the process, there is quite a large change in the magnitude of the portfolio objective function.

The following graphs show the change in the Explorer Mission over the 9 year possible start dates that the design space investigated. Starting in the year 2010, shows that there is a span from 200 to 2000 in the objective value. The second set of graphs show the same information of the objective value vs. the expected input percentage. The same changes can be seen with the polynomial changes of the portfolios as the expected input percentage changes for each year.

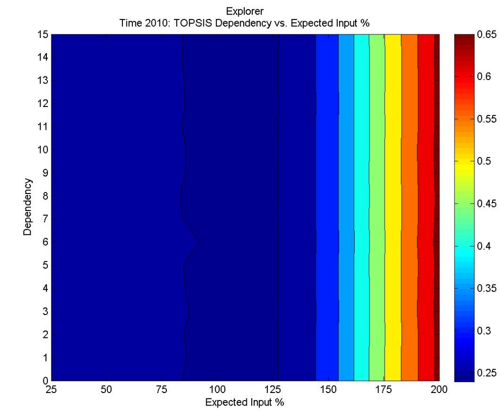
The next interesting feature can be seen with the addition of dependencies vs. the objective value. This is the third set of graphs for each year. The horizontal nature of the dependencies suggest that there is not a large change in the addition of dependencies as they are added to the portfolio for a frozen start year.

The fourth set of graphs show the first and second derivatives of the objective value changes. It must be remembered that the addition of dependencies are not continuous like that of the expected input percentage. The dependency levels may be changed in any order. This would change the output of the derivative graphs. However, they do bring to light how fast the changes in portfolios occur from one portfolio to the next.

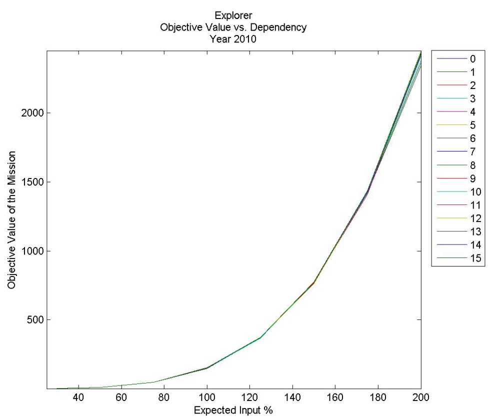
The statistical information given after each graph represents the maximum, minimum, maximum difference between the min and max, standard deviation and how far from the average each maximum and minimum is. The second graph represents the polynomial function that the data takes when the dependency is held constant, but the expected percentage is changed. Figure 112 and Figure 113 as well as Table 47 and Table 48 show the changes for the year of 2010. The information is generalized for the Explorer mission temporal aspect, but a full set of graphs for year 2011-2019 can be found in Appendix A.



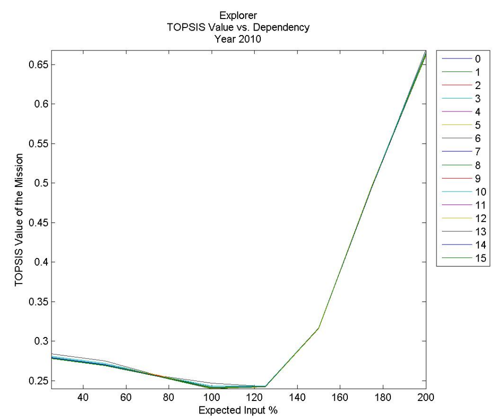
(a) Objective Value Contour



(b) TOPSIS Value Contour

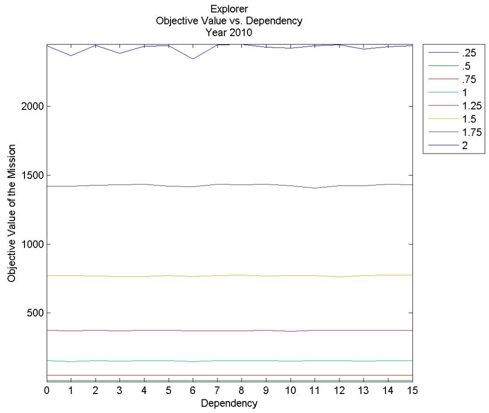


(c) Objective Value vs. Expected Input Percentage

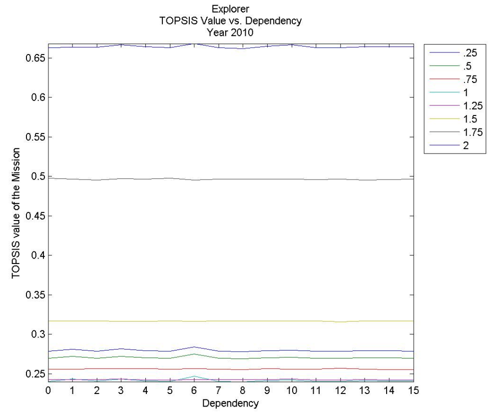


(d) TOPSIS vs. Expected Input Percentage

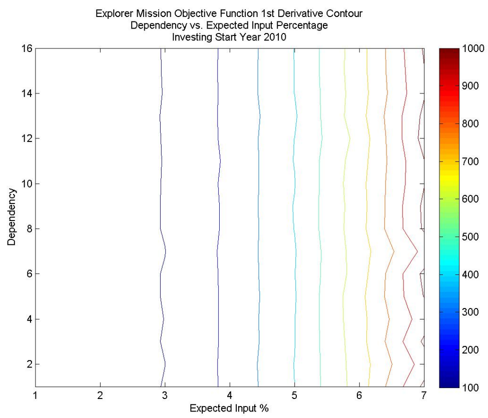
Figure 112: Explorer Start Year of 2010



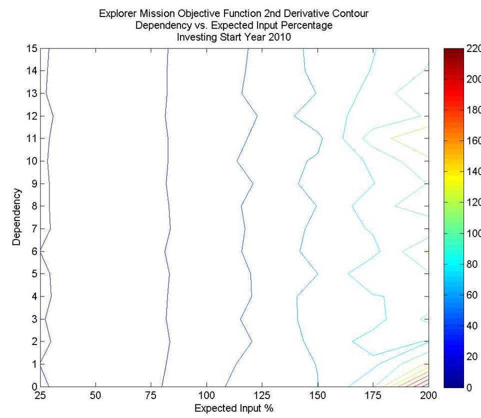
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour



(d) 2nd Derivative of Objective Value Contour

Figure 113: Explorer Start Year of 2010 (con't)

Table 47: 2010 Statistics Information for various Expected Input Percentages

Expected Input Percentages	min	max	Delta	Average	Variation	Max Variation	Min Variation
25%	0.572061	0.598868	0.026808	0.592078	0.045277	0.011468	0.033809
50%	9.152969	9.58189	0.428921	9.473252	0.045277	0.011468	0.033809
75%	47.67758	48.50832	0.83074	48.17678	0.017244	0.006882	0.010362
100%	146.4475	153.3102	6.862744	151.5637	0.04528	0.011524	0.033756
125%	367.8826	374.2926	6.41003	372.106	0.017226	0.005876	0.01135
150%	762.8413	776.1331	13.29184	770.8286	0.017244	0.006882	0.010362
175%	1406.056	1435.336	29.28035	1426.723	0.020523	0.006037	0.014486
200%	2343.16 2	452.964 1	09.8039 2	425.019	0.04528	0.011524	0.033756

Table 48: 2010 Statistics Information for Various Dependency Levels

Dependency	X^3	X^2	X	b
0	0.000695015	-0.10664	6.225169	-107.452
1	0.000607377	-0.08169	4.202767	-65.7886
2	0.000695494	-0.10642	6.174471	-105.899
3	0.000630075	-0.08778	4.651968	-74.3177
4	0.0006856	-0.10333	5.904275	-100.028
5	0.000697184	-0.10731	6.278267	-108.514
6	0.000592178	-0.07789	3.910713	-59.9366
7	0.000692129	-0.10496	6.02932 -	102.518
8	0.000695613	-0.10599	6.120832	-104.547
9	0.000674651	-0.10016	5.652095	-94.9429
10	0.000673228	-0.10014	5.657751	-95.0795
11	0.000706568	-0.11064	6.586342	-115.399
12	0.00070919	-0.11067	6.532393	-113.425
13	0.00066143	-0.09684	5.417434	-90.6237
14	0.00066987	-0.09867	5.537555	-92.7323
15	0.000683343	-0.10261	5.860174	-99.4105

10.4 Explorer Mission:

Stepping through the Explorer data showed that the TOPSIS value has a base value for each year that is dominated by the start year: the later the start year the higher the TOPSIS value. Changing the expected input percentage was a much more significant change than adding the dependencies.

Table 49: Temporal Ranges

Start Year	Objective Value Range	TOPSIS Value Range	Low High	Low High
2010	0.5721	2353.0	0.2399	0.6682
2011	0.5539	2431.3	0.2412	0.6766
2012	0.5679	2397.6	0.2441	0.6702
2013	0.5314	2333.3	0.2510	0.6935
2014	0.5535	2309.2	0.2633	0.6937
2015	0.5192	2266.0	0.2695	0.6937
2016	0.5070	2199.9	0.2788	0.7037
2017	0.5185	2128.7	0.2955	0.7003
2018	0.4990	2041.9	0.3095	0.7041
2019	0.4716	1931.8	0.3226	0.6854

Looking only at the change of the input file, with respect to the overall portfolio, shows that it is a polynomial change as the expected input % increases. This is not a perfect polynomial function, but it has that general shape for all of the dependency changes. They are almost unanimous below the 100% mark. They start to slightly vary when the user expect optimistic values for their input file. This is seen in the (c) and (d) of 112 and the Appendices.

Looking at the TOPSIS values gives a similar feeling of a polynomial fit, but has much more chaotic changes throughout the function. TOPSIS takes into account non-linear changes such as the flexibility and objective value. Since the cost and starting year are uniform for all of this data, the real driving factors are the objective value and flexibility quality of the technologies.

Viewing the data from another viewpoint to see what happens when the objective value is faced with the dependencies shows that as dependencies are added, the portfolio fluctuates with adding the relationships. This information for starting year of 2010 is actually quite uniform throughout the dependency changes. There are a few significant changes that can be seen, but they are within a cushion zone of the baseline portfolio.

Knowing the general trends of the first starting year of 2010, then brings into question what happens when the starting year is changed. The TOPSIS suggests that the relative difference and desirability of an option is driven higher because of the dominance of the costing aspect.

Table 50: Polynomial fit with respect to the start year

Start Year	X^3	X^2	X	Constant
2010	0.000695015	-0.10664	6.225169	-107.452
2011	0.000722	-0.11609	7.032122	-124.431
2012	0.000758	-0.13166	8.541669	-158.693
2013	0.000657	-0.09935	5.689892	-96.4975
2014	0.000628	-0.09239	5.188988	-87.0561
2015	0.000636	-0.09581	5.478248	-92.89
2016	0.000596	-0.08718	4.863521	-81.0677
2017	0.000595	-0.08964	5.132048	-87.226
2018	0.000575	-0.0868	4.978623	-84.6879
2019	0.000543	-0.08204	4.705525	-80.0424

Table 50 shows the change in the polynomial coefficients as the start year changes. The most notable change is that of the constant. The constant goes from a -107.5 to -80.04 from start year 2010 to 2019. Graphing this information gives Figure 114 -Figure 117.

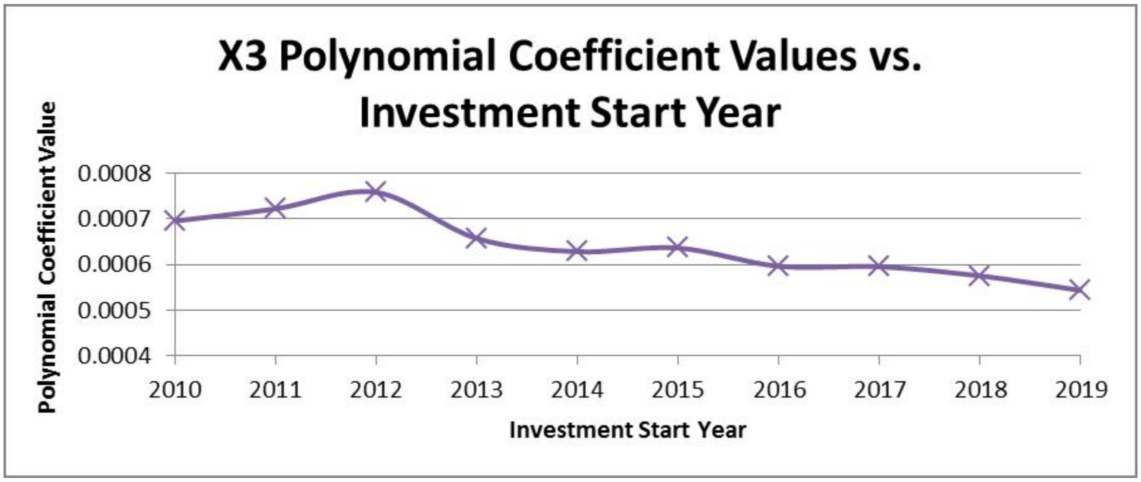


Figure 114: X^3 polynomial coefficient for temporal study

Figure 115: X^2 polynomial coefficient for temporal study

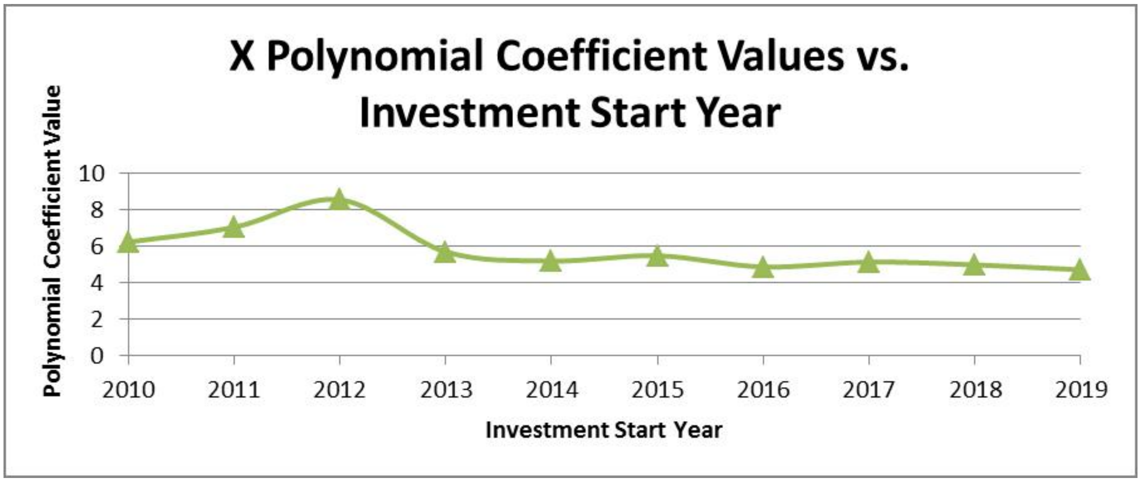


Figure 116: X polynomial coefficient for temporal study

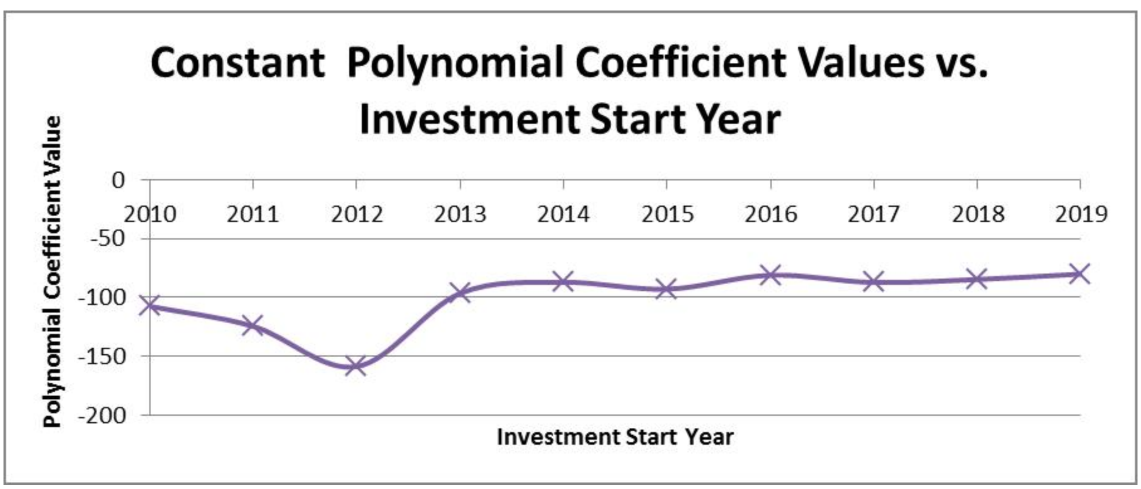


Figure 117: Constant coefficient for temporal study

The coefficients have the most notable change in the constant art of the polynomial. This has the largest effect on the equation. It changes from 107 to 80 going from a start year of 2010 to a start year of 2019. The only other coefficient that changes as drastically is the X^2 value. Looking at the coefficient changes for the investment year, bundles the information contributed in TOPSIS to show the changes from year to year.

10.5 Temporal Summary

Changing the starting year of the portfolio shortens the development time. This has an effect on the portfolio objective value. However, it is not a nicely fitting equation that compiles the temporal effects on the portfolio. Changing the start year and keeping the need by year constant shortens the development time frame. Adding constraints to an optimization problem only allows the objective function to stay the same or go lower. This comes into play when TOPSIS is applied and eventually used as a decision making tool. If the start year is important to the user than TOPSIS takes into account the relative range time period of the portfolio. TOPSIS has a larger impact of changing the portfolio start year than the actual optimizer does when looking at the start year with respect to dependencies.

CHAPTER XI

EXPLORER AND SURVEYOR COMBINED PRECURSOR MISSION RESULTS

The Surveyor and Explorer missions are considered precursor missions to the NEA human mission. Both missions have been introduced, so the next scenario is to run both missions together in the START framework. This round is ran with both missions enabled, no dependencies between missions, but the dependencies between the capabilities. The funding level is the two mission budgets combined.

This scenario will be different than the previous separate missions because the budget is larger per year, and there are no rules or suggestions as to what percentage of the budget goes to which mission. If the funding percentages of the total were decided beforehand, then the results would resemble those of the previous mission scenarios given. As such, this scenario suggests that the mission that provides the most value to the feed-forward capabilities will get more funding, and show a disproportionate amount to the other mission. In this case, it is assumed that more money will go towards funding the Explorer mission and lower amount towards the surveyor mission, since the values are larger for the Explorer mission. The baseline for both missions together is given below in Figure 118. Figure 119 and Figure 120 show the individual baseline portfolios for each mission separately.

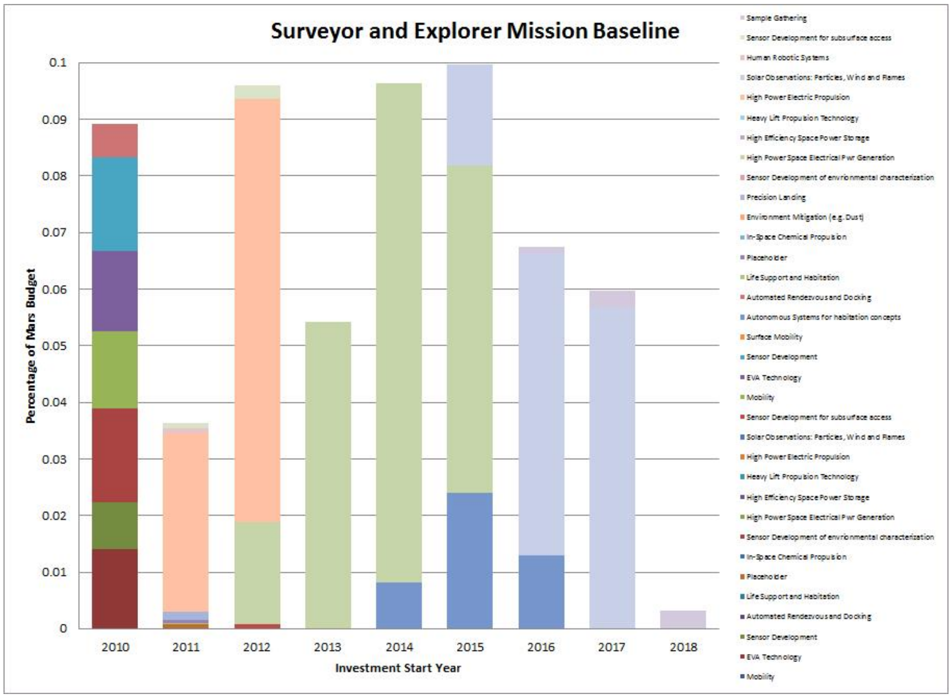


Figure 118: Surveyor and Explorer Mission Baseline

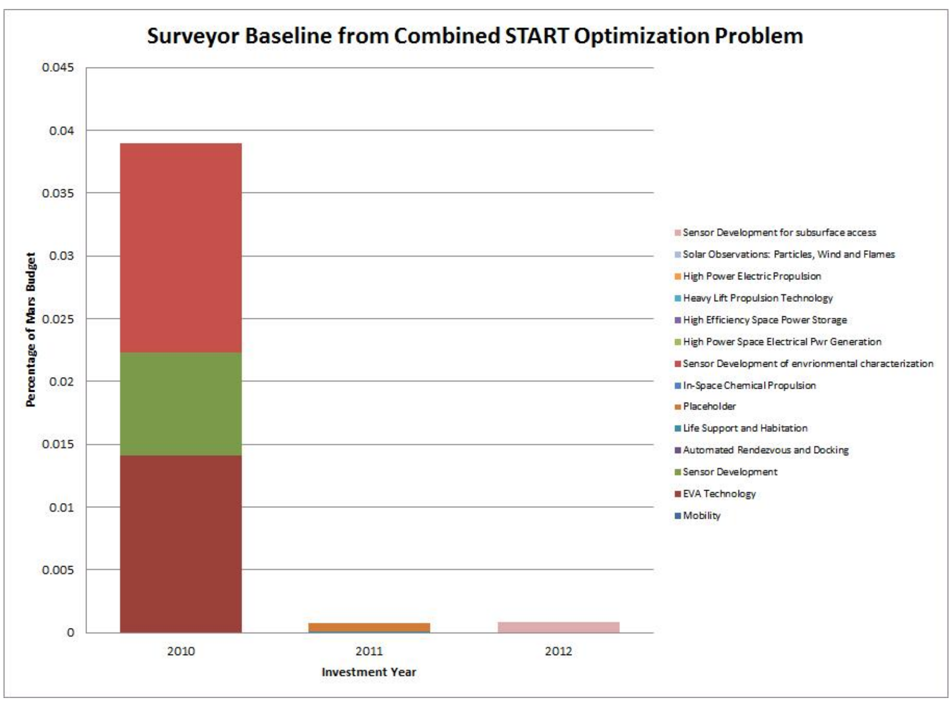


Figure 119: Surveyor Baseline from combined START optimization

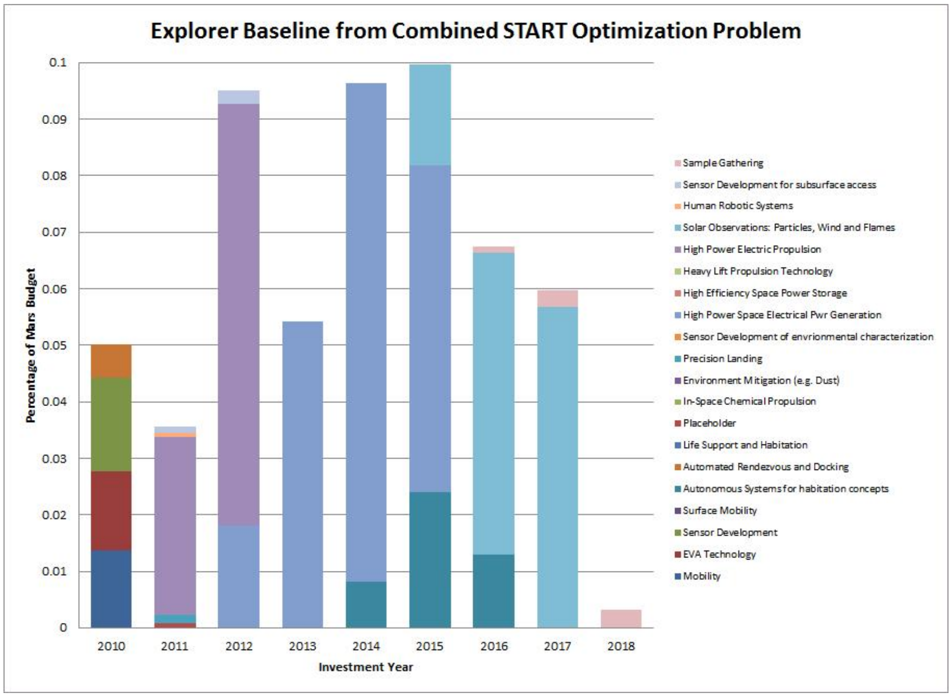


Figure 120: Explorer Baseline from combined START optimization

Table 51 shows the dependencies that were added in addition to the individual Surveyor and Explorer dependencies tested in Chapter 9 and 10. These dependencies represented Explorer's dependence on the Surveyor mission. It was assumed for this study that no technology could be dependent on a future technology. It must be dependent on a past technology such as a technology test bed.

Table 51: Explorer Cross Mission Dependencies

Capability A	Capability B	Operation
NEA Surveyor: In-Space chemical Propulsion	In-Space chemical Propulsion	MIMO
NEA Surveyor: Environmental Characterization	Environmental Characterization	A OR B
NEA Surveyor: Solar Observations	Solar Observations	A OR B
NEA Surveyor: Sensor Development for subsurface access	Sensor Development for subsurface access	MIMO

Table 52 and Table 53 gives a breakdown of the combined portfolios and the original mission baseline values. Focusing on Surveyor shows that the baseline values originally selected 12 technologies, but when combined with the Explorer mission only choose 7 technologies. However, the technologies that were chosen were funded completely. All the technologies funded fully in the Surveyor mission were funded fully in the combined mission. The sensor Development of environmental characterization was originally funded at 90% and was funded at 100% in the combined mission.

The Explorer Mission originally selected 16 technologies and then only selected 14 technologies when combined with the Surveyor mission. However, it must be noted that technologies originally partially funded had a higher partial funding level in the combined mission scenario. An in-depth look at the actual costs associated with each capability would show the specific reasons for each technology decisions. It my be that funding a higher value capability needed the cost of multiple capabilities shown beforehand.

Table 52: Surveyor Comparison Portfolio

Metric	Surveyor Alone Selection	Partial Funding	Combined Missions Selection	Partial Funding
Mobility	Selected	0.7	Not Selected	
EVA Technology	Selected	1	Selected	1
Sensor Development	Selected	1	Selected	1
Automated Rendezvous and Docking	Selected	1	Selected	1
Life Support and Habitation	Selected	1	Selected	1
Placeholder	Selected	1	Selected	1
In-Space Chemical Propulsion	NOT Selected		Not Selected	
Sensor Development of environmental characterization	Selected	0.9	Selected	1
High Power Space Electrical Power Generation	Selected	0.6	Not Selected	
High Efficiency Space Power Storage	NOT Selected		Not Selected	
Heavy Lift Propulsion Technology	Selected	0.4	Not Selected	
High Power Electric Propulsion	Selected	0.8	Not Selected	
Solar Observations: Particles, Wind and Flames	Selected	0.9	Not Selected	
Sensor Development for subsurface access	Selected	1	Selected	1

Comparing Table 52, Table 53, and Table 51 shows an expected change in all three of the inter-mission dependencies tested. Table 52 shows that the In-space Chemical propulsion was not selected for the Surveyor mission, but selected for the Explorer mission. This violates the first dependency tested on the inter-mission level. The Solar Observations and Sensor Development for Environmental Characterization were selected on both missions, but violates the OR inter-mission dependency placed upon it according to Table 51. Had these missions only be analyzed separated as in Chapter 9 and Chapter 10, these inter-mission dependencies or Crosslinked-Dependencies according to Sage would have been missed. Running the missions together gives the

Table 53: Explorer Comparison Portfolio

Metric	Explorer Alone Selection	Partial Funding	Combined Missions Selection	Partial Funding
Mobility	Selected	1	Selected	1
EVA Technology	Selected	1	Selected	1
Sensor Development	Selected	1	Selected	0.9
Surface Mobility	Selected	0.8	NOT Selected	
Autonomous Systems for habitation concepts	Selected	0.9	Selected	0.4
Automated Rendezvous and Docking	Selected	1	Selected	1
Life Support and Habitation	Selected	1	Selected	1
Placeholder	Selected	1	Selected	1
In-Space Chemical Propulsion	NOT Selected		NOT Selected	
Environment Mitigation (e.g. Dust)	NOT Selected		NOT Selected	
Precision Landing	Selected	1	Selected	1
Sensor Development of environmental characterization	Selected	1	NOT Selected	
High Power Space Electrical Power Generation	Selected	0.5	Selected	0.7
High Efficiency Space Power Storage	NOT Selected		NOT Selected	
Heavy Lift Propulsion Technology	NOT Selected		NOT Selected	
High Power Electric Propulsion	Selected	0.6	Selected	0.7
Solar Observations: Particles, Wind and Flames	Selected	0.8	Selected	0.7
Human Robotic Systems	Selected	1	Selected	1
Sensor Development for subsurface access	Selected	1	Selected	1
Sample Gathering	Selected	1	Selected	1

user a sense of the fidelity of the Precursor scenario as well as higher fidelity with dependency links between missions such as the testbed scenario.

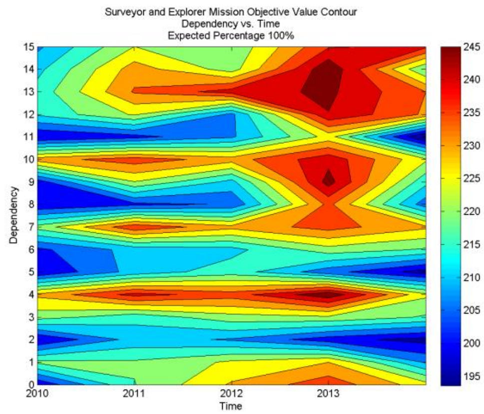
Table 54compares the portfolios objective values and cost values. It shows that running the Precursor mission had a lower objective value and overall cost. Looking at this information gives a Return on Investment (ROI) of 0.222 for the precursor mission vs 0.199 for the separate missions combined.

Table 54: Precursor Mission Baseline Cost Comparison

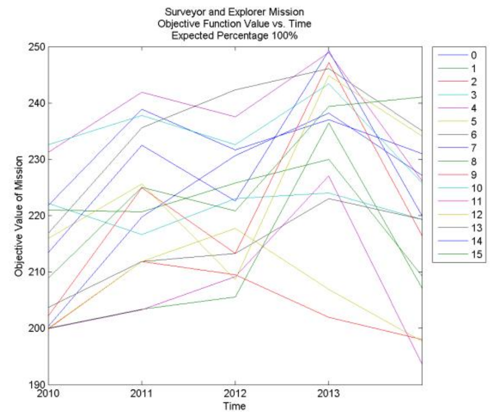
Mission Scenario	Objective Value Precursor Mission	Objective Value Separate Missions
Combined	200.30	262.39
Surveyor	69.98	109.76
Explorer	130.32	152.63
Total Cost	902.75	1315.97

Figure 121-Figure 123 and Table 55 - Table 57 summarize the combined portfolio and separate portfolios for the Surveyor and Explorer precursor missions. The important point is that a lower portfolio objective value was funded at a lower cost in

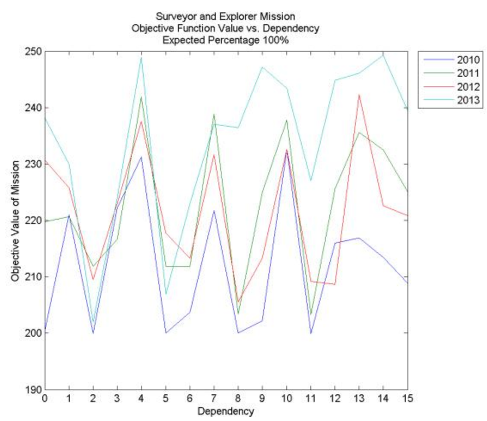
the combined mission compared to the separate mission scenarios.



(a) Objective Value Contour

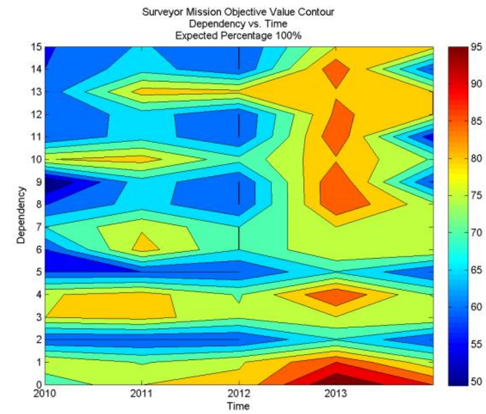


(b) Objective Value vs. Start Year

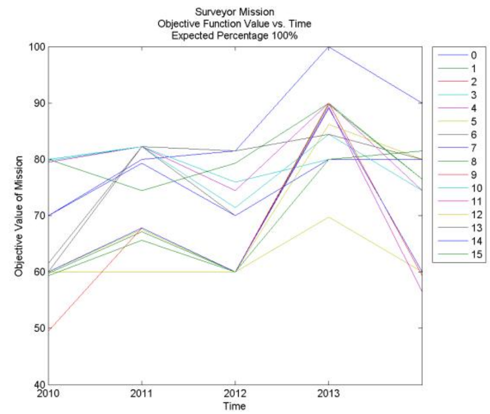


(c) Objective Value vs. Dependency

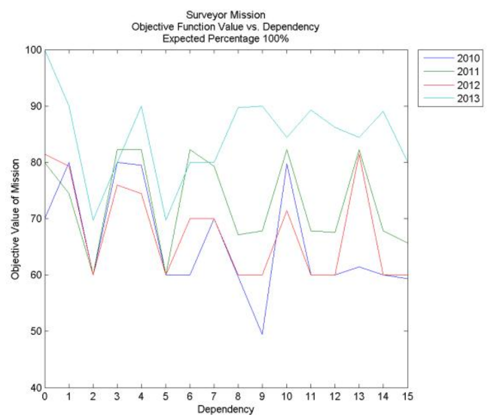
Figure 121: Combined Surveyor and Explorer Portfolio Information



(a) Objective Value Contour

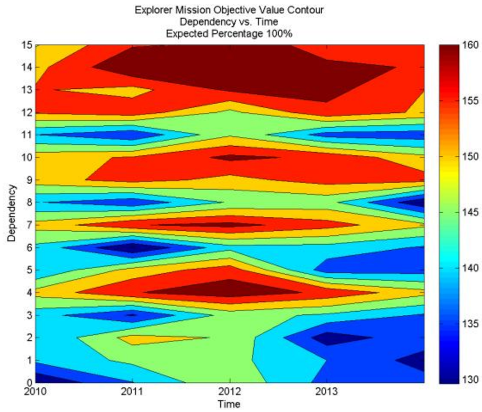


(b) Objective Value vs. Start Year

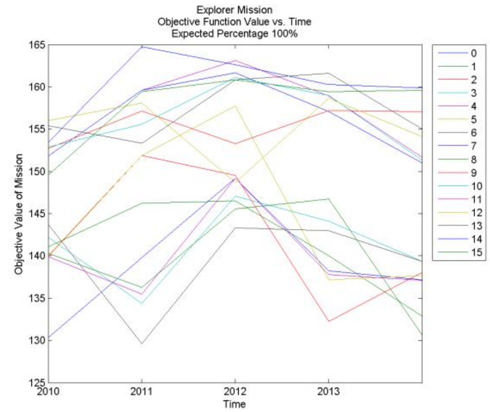


(c) Objective Value vs. Dependency

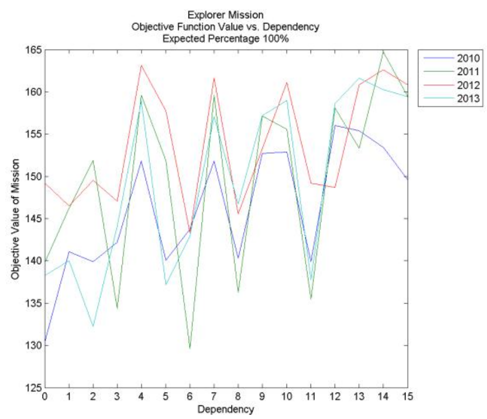
Figure 122: Surveyor Portfolio Information from Precursor Missions



(a) Objective Value Contour



(b) Objective Value vs. Start Year



(c) Objective Value vs. Dependency

Figure 123: Explorer Portfolio Information from Precursor Missions

Table 55: Statistical information for Surveyor and Explorer Combined Portfolio

Year	min	max	Delta	Average	Variation	Max Variation	Min Variation
2010	199.9053	232.5801	32.67486	211.8816	0.154213	0.097689	0.056524
2011	203.2664	241.887	38.62054	222.6093	0.17349	0.086599	0.086892
2012	205.5463	242.3105	36.76423	221.5174	0.165965	0.093867	0.072099
2013	201.939	249.2842	47.34515	233.988	0.20234	0.065371	0.136969
2014	193.5993	241.0297	47.43042	218.7983	0.216777	0.101607	0.11517
Extremes	193.599	249.284					

Table 56: Surveyor Statistical Information from Surveyor and Explorer Mission

Year	min	max	Delta	Average	Variation	Max Variation	Min Variation
2010	49.44683	79.98373	30.53691	65.55956	0.465789	0.220016	0.245772
2011	59.9878	82.26637	22.27857	73.04415	0.305001	0.126255	0.178746
2012	59.9878	81.45574	21.46794	67.74522	0.316892	0.202383	0.114509
2013	69.71426	99.97967	30.2654	84.5261	0.35806	0.182826	0.175234
2014	56.46083	89.9817	33.52087	73.05665	0.458834	0.23167	0.227164
Extremes	49.447	99.980					

Table 57: Explorer Statistical Information

Year	min	max	Delta	Average	Variation	Max Variation	Min Variation
2010	130.3246	155.9982	25.67361	146.322	0.17546	0.066129	0.109331
2011	129.6218	164.7385	35.11672	149.5651	0.234792	0.10145	0.133342
2012	143.3296	163.1222	19.79258	153.7722	0.128714	0.060805	0.067909
2013	132.2248	161.6317	29.40697	149.4619	0.196752	0.081424	0.115328
2014	130.6399	159.8987	29.2588	145.7417	0.200758	0.097138	0.10362
Extremes	129.622	164.739					

11.0.1 Precursor Extreme Statistical Information

Table 55, Table 56, and Table 57 show interesting characteristics about the extreme of the portfolios. The Precursor mission had an extreme minimum in 2014 at dependency level 14, but an extreme maximum in 2013 at dependency level 14. Looking at the separate missions, the Surveyor mission had its extreme minimum in 2010 at dependency level 9, but its maximum in 2013 at the baseline. The Explorer mission had both its extreme maximum and minimum value in 2012 for dependency levels 14 and 6 respectively. Focusing on the extremes gives some insight into the nature of dependencies between the missions. The maximum Precursor mission value was at in 2013 with almost every single dependency implemented at Dependency Level 14.

This seems contrary to the dependency conclusions in Chapter 9 and Chapter 10 where Constraint Dependencies required that the value of the overall portfolio could not be higher than the baseline. In this case, the more constraints added between missions actually gave a higher combined objective value although not the highest individual mission values. In the Precursor mission case, 2013 is the last year where the combined scenario can get the most value out of the Surveyor mission. Since everything is due in 2014, any technology in Surveyor that has a two year development time is not feasible in the 2013 investment year start date of the Precursor scenario. Therefore the highest contribution that Surveyor could possibly give is in 2013.

While 2013 is the last year that the Precursor mission can receive any value from Surveyor, it is also the year where Surveyor's unused funds transfer over to the Explorer mission. This effectively gives Explorer a higher yearly budget for the duration of the Surveyor mission. Without meaning to model it, a time dependency was constructed naturally from the addition of the two mission scenario in addition to the interlinked dependencies applied through the thesis methodology.

11.0.2 MIMO Affects on Individual Technology

This Precursor scenario is the first time that MIMO dependency results are presented. Specifically the MIMO implemented on the Sensor Development for Subsurface access in the Explorer Mission had a large effect on the overall portfolio. By implemented the MIMO dependency, Table 58 shows how the Sensor Development for Subsurface Access jumps 140% in its objective value. Figures 124 and Figure 125 show the actual contour plot of the technology capability with the MIMO implemented. Specifically Figure 125 shows the two objective value levels of the technology with and without MIMO. Having such a high objective value with MIMO almost ensured that this technology would be chosen along with its dependent technology. This interlinked dependency gave fidelity to the process and changed the system engineers decision based on the new information presented in MIMO.

Table 58: Surveyor Statistical Information from Surveyor and Explorer Mission

Mission	Element	Case 0		Case 4		Case 9		Case 14	
Surveyor	Mobility	9.998	1	9.998	1	9.998	1	9.998	1
	EVA Technology	9.998	1	Not Selected		Not Selected		Not Selected	
	Sensor Development	9.998	1	9.998	1	9.998	1	9.998	1
	Automated Rendezvous and Docking	9.998	1	9.998	1	9.998	1	9.998	1
	Life Support and Habitation	9.998	1	9.998	1	9.998	1	9.998	1
	Placeholder	9.998	1	9.998	1	9.998	1	9.998	1
	In-Space Chemical Propulsion	Not Selected		Not Selected		Not Selected		Not Selected	
	Sensor Development of environmental characterization	9.998	1	9.998	1	9.998	1	9.726	0.9
	High Power Space Electrical Power Generation	9.998	1	9.998	1	9.998	1	9.321	0.8
	High Efficiency Space Power Storage	Not Selected		Not Selected		Not Selected		Not Selected	
	Heavy Lift Propulsion Technology	Not Selected		Not Selected		Not Selected		Not Selected	
	High Power Electric Propulsion	Not Selected		Not Selected		Not Selected		Not Selected	
	Solar Observations: Particles, Wind and Flames	9.998	1	9.998	1	9.998	1	9.998	1
	Sensor Development for subsurface access	9.998	1	9.998	1	9.998	1	9.998	1
Explorer	Mobility	9.998	1	9.998	1	9.726	0.9	9.998	1
	EVA Technology	8.717	0.7	9.321	0.8	9.321	0.8	9.998	1
	Sensor Development	9.726	0.9	7.816	0.6	8.717	0.7	9.726	0.9
	Surface Mobility	Not Selected		9.998	1	9.998	1	9.998	1
	Autonomous Systems for habitation concepts	9.998	1	9.998	1	8.717	0.7	8.717	0.7
	Automated Rendezvous and Docking	9.998	1	9.998	1	9.998	1	9.998	1
	Life Support and Habitation	9.998	1	9.998	1	9.998	1	9.998	1
	Placeholder	9.998	1	9.998	1	9.998	1	9.998	1
	In-Space Chemical Propulsion	Not Selected		Not Selected		Not Selected		Not Selected	
	Environment Mitigation (e.g. Dust)	4.465	0.4	Not Selected		1.472	0.3	Not Selected	
	Precision Landing	9.998	1	9.998	1	9.998	1	9.998	1
	Sensor Development of environmental characterization	7.816	0.6	9.998	1	7.816	0.6	9.998	1
	High Power Space Electrical Power Generation	Not Selected		Not Selected		8.717	0.7	8.717	0.7
	High Efficiency Space Power Storage	Not Selected		Not Selected		Not Selected		Not Selected	
	Heavy Lift Propulsion Technology	Not Selected		Not Selected		Not Selected		Not Selected	
	High Power Electric Propulsion	7.816	0.6	8.717	0.7	9.321	0.8	9.726	0.9
	Solar Observations: Particles, Wind and Flames	9.726	0.9	9.726	0.9	Not Selected		Not Selected	
	Human Robotic Systems	9.998	1	9.998	1	9.998	1	9.998	1
	Sensor Development for subsurface access	9.998	1	23.384	1	23.384	1	23.384	1
	Sample Gathering	9.998	1	9.998	1	9.998	1	9.998	1
Total		238.228		248.926		247.16		249.284	

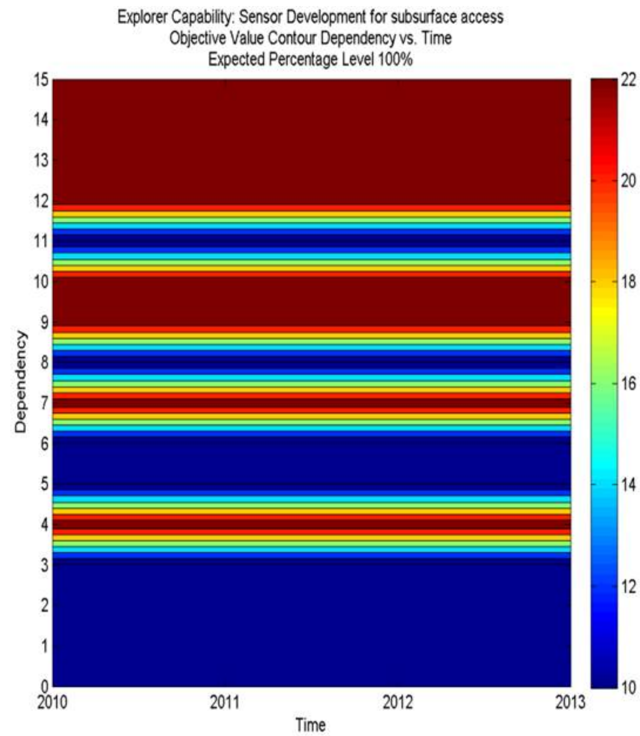


Figure 124: NEA Baseline Portfolio

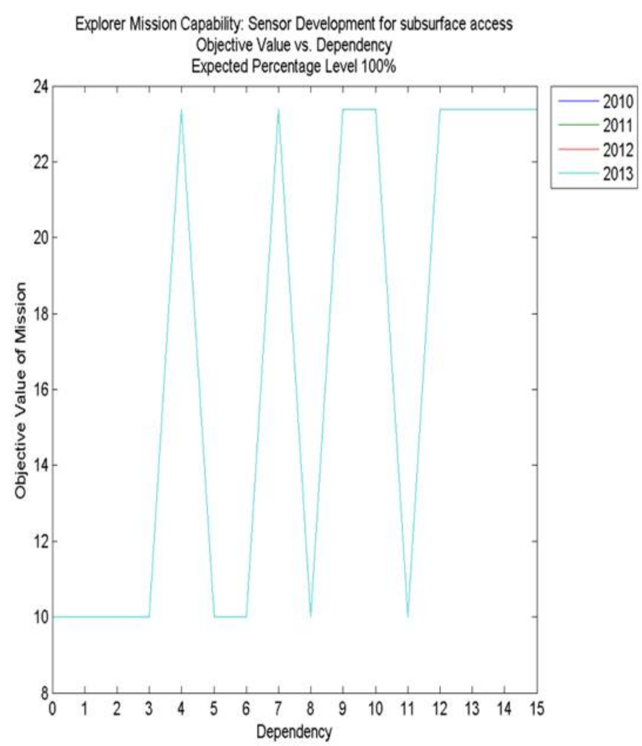


Figure 125: NEA Baseline Portfolio

11.1 Surveyor and Explorer Precursor Mission Summary

In conclusion, combining the Surveyor and Explorer Missions into one Precursor mission gives different results from the separate mission solutions. The technology portfolios are drastically different in the Precursor mission from their separate mission counterparts. This results in technology being seen as a feed-forward result in the interlinked dependencies from mission to mission. Including these crosslinked dependencies gives higher fidelity to the systems engineer to determine the best technology portfolio impact on the overall objective of deep-space human capabilities.

The Precursor mission showed that the Explorer Mission has a larger impact than Surveyor. This is evident in the placement of extremes for the Precursor contour graphs. Combining the missions effectively add the two contour plots for objective value together. This shows that the values of the Precursor mission is lower, but at a lower cost and ultimately higher ROI. This highlights the fact that optimizing the missions independently does not necessarily optimize the missions together. If the systems engineer requires a higher objective value and has the funding to complete this, they are still loosing out on the fidelity of being able to apply dependencies between missions. If the missions were ran separately, the dependency relationships between them would be lost; resulting in reduced fidelity.

In the Precursor mission case, the constraint dependencies acted in accordance to the idea that the constraint must reduce the baseline; however, the addition of the other mission inadvertently introduced a timing dependency. In this case that timing issue was stronger than the dependencies added. Specifically the time to develop Surveyor ran out around 2013, which gave the highest value from Surveyor at the latest date and shifted the available funds to Explorer for the remainder of its development period. This was a timing aspect; not a dependency result.

CHAPTER XII

NEA HUMAN MISSION RESULTS

The NEA mission is the culmination of the Surveyor and Explore Missions. It has the largest single mission design space that will be searched in this thesis. However, it does not have as many dependencies. In the case of NEA, all of the deep space technologies play a critical role in the demonstration of a Mars Mission. The NEA mission is a smaller subset of Mars, with the same amount of risk. Therefore the technologies should stand alone when it comes to being adopted. In this case the technologies were ran with the partial funding scenario and all enhancing capabilities. They would actually be ran as enabling technologies with no dependencies if the mission was not a precursor mission to Mars.

12.0.1 Baseline

Running a baseline scenario with no dependencies or changes to the input file in the earliest year gives Figure 126. The NEA mission is funded at 33% of the Mars budget. This is reflected in the Y-axis of Figure 126 as a Mars Yearly Budget Percentage. The baseline value is 244.35. These choices are represented separately in Table 59.

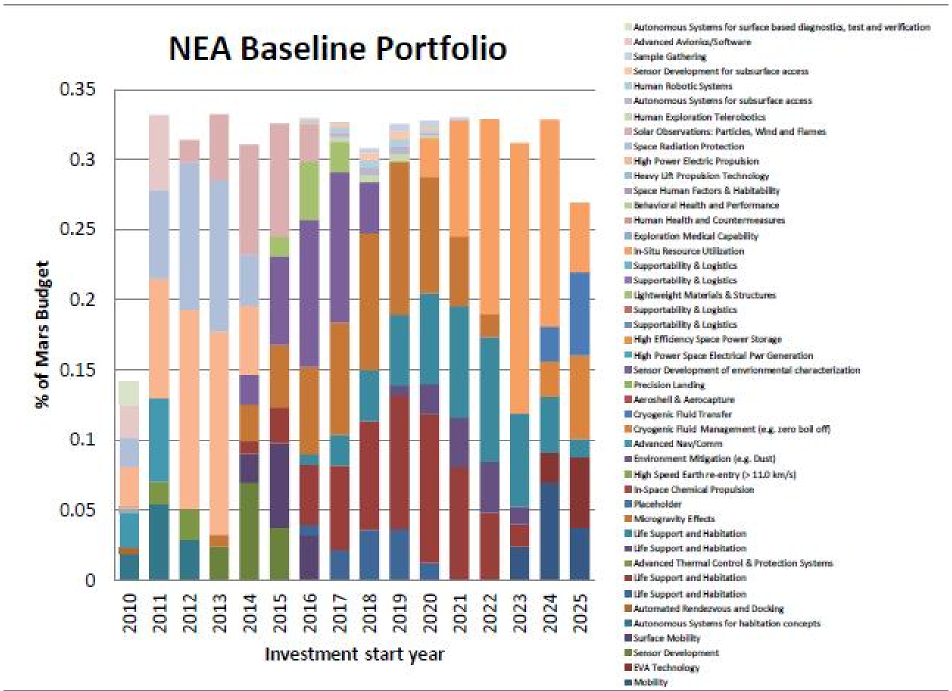


Figure 126: NEA Baseline Portfolio

Table 59: NEA Technology Portfolio

Technology	Selection	Partial Funding Scale
Mobility	Selected	1
EVA Technology	Selected	1
Sensor Development	Selected	1
Surface Mobility	Selected	1
Autonomous Systems for habitation concepts	Selected	0.9
Automated Rendezvous and Docking	Selected	1
Life Support and Habitation	Selected	1
Life Support and Habitation	Selected	0.6
Advanced Thermal Control & Protection Systems	Selected	1
Life Support and Habitation	Selected	1
Life Support and Habitation	Selected	0.5
Microgravity Effects	Selected	0.6
Placeholder	NOT Selected	
In-Space Chemical Propulsion	NOT Selected	
High Speed Earth re-entry (¿ 11.0 km/s)	NOT Selected	
Environment Mitigation (e.g. Dust)	NOT Selected	
Advanced Nav/Comm	Selected	1
Cryogenic Fluid Management (e.g. zero boil off)	Selected	1
Cryogenic Fluid Transfer	Selected	1
Aeroshell & Aerocapture	NOT Selected	
Precision Landing	NOT Selected	
Sensor Development of environmental characterization	Selected	0.8
High Power Space Electrical Power Generation	NOT Selected	
High Efficiency Space Power Storage	NOT Selected	
Supportability & Logistics	NOT Selected	
Supportability & Logistics	NOT Selected	
Lightweight Materials & Structures	Selected	0.8
Supportability & Logistics	Not Selected	
Supportability & Logistics	NOT Selected	
In-Situ Resource Utilization	Selected	0.6
Exploration Medical Capability	Selected	1
Human Health and Countermeasures	Selected	1
Behavioral Health and Performance	Selected	1
Space Human Factors & Habitability	Selected	1
Heavy Lift Propulsion Technology	NOT Selected	
High Power Electric Propulsion	Selected	0.6
Space Radiation Protection	Selected	0.8
Solar Observations: Particles, Wind and Flames	Selected	0.6
Human Exploration Telerobotics	Selected	1
Autonomous Systems for subsurface access	Selected	1
Human Robotic Systems	Selected	1
Sensor Development for subsurface access	Selected	1
Sample Gathering	Selected	1
Advanced Avionics/Software	Selected	0.9
Autonomous Systems for surface based diagnostics, test and verification	Selected	1

Two dependencies were included in the NEA Human Mission as shown in Table 60. Running the design space four dependency levels yields Figure 127. It shows that the highest objective values are found at the baseline in the earliest start year. This is expected given the information presented in the previous two cases. It also shows that the different specific dependencies have a different effect on the portfolio.

Table 60: NEA Human Mission Capability Dependencies

Capability A	Capability B	Operation
Mobility	EVA Technology	A OR B
Microgravity	Human Health and Counter Measures	B Needs A

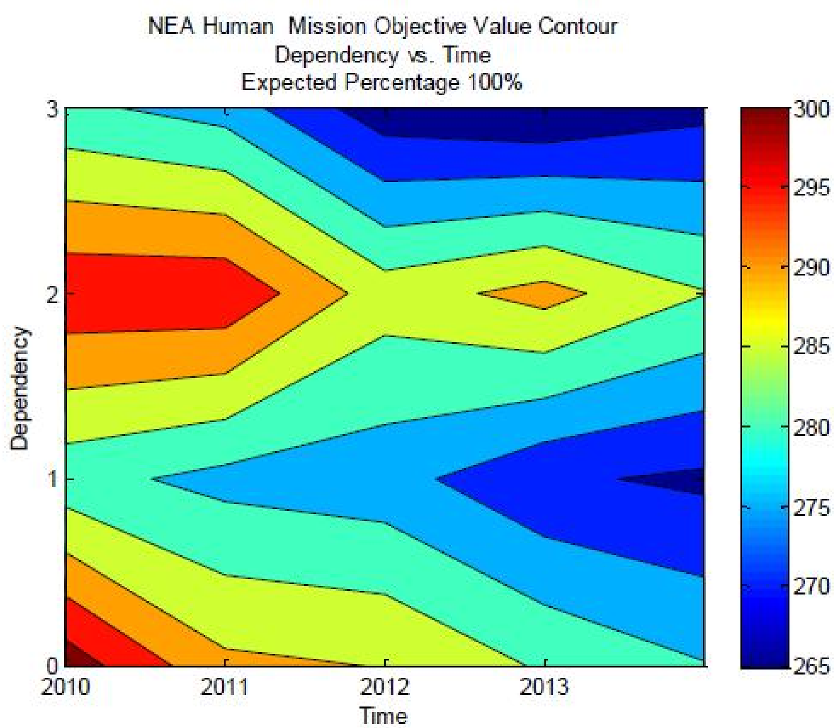


Figure 127: NEA Mission Objective Value Contour for Expected Percentage of 100%

At dependency level 1 in start year 2010, the contour is sea green or about 280, compared to dependency level 2 in start year 2010 which is still red and around 295.

Dependency level 3 in that same year is the same sea green as dependency level 1. This seems to say that adding dependency level 1 dominates the portfolio more than that of adding dependency level 2. Dependency level 3 is a combination of both dependencies in the portfolio ran together, so once again level 1 is dominating the portfolio. Looking across the investment years between level 1 and level 2 shows that level 2 is always higher in values than that of level 1. Level 3 diminishes faster than level 1, but follows the same general rule of decreasing lower than that of level 2.

Looking at this from a bottom up and top down perspective, the user cannot get 10% above the baseline because the dependencies prevent this. However, the user can recognize that the dependency of level 1 has a higher impact on the portfolio than that of level 2. This may be useful information in determining the necessity of the dependency and allow them to reexamine that relationship.

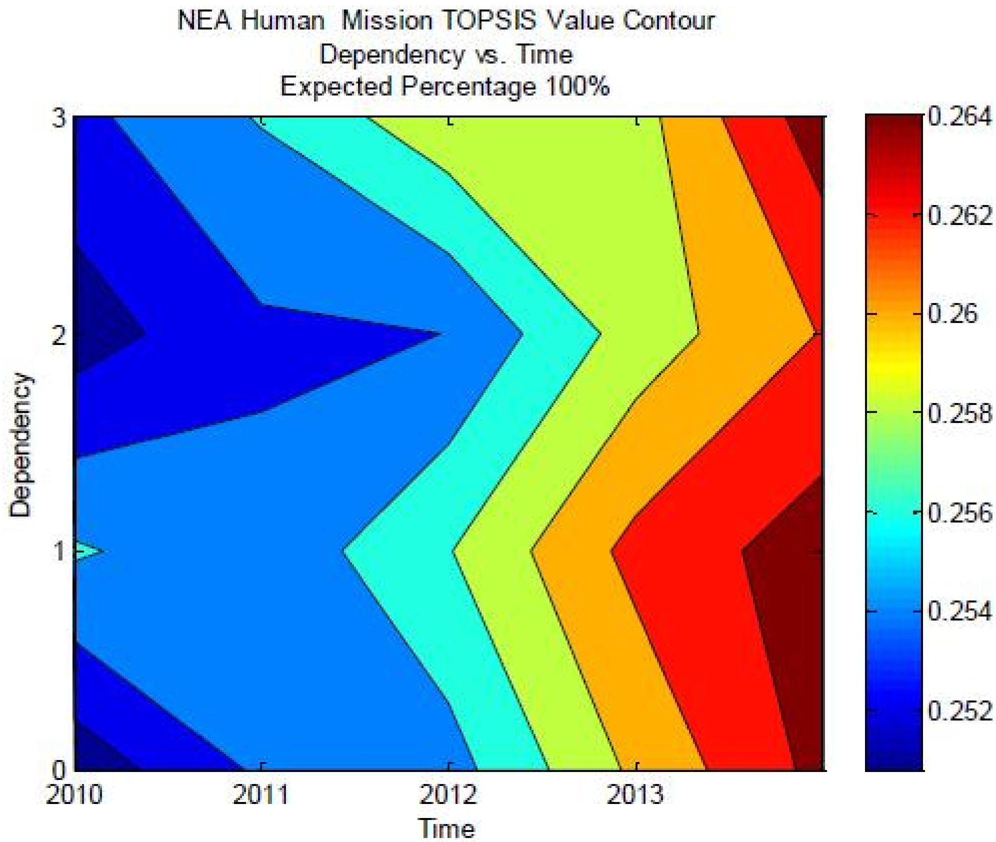


Figure 128: NEA Mission TOPSIS Value for Expected Percentage of 100%

Switching over to the TOPSIS performance of Figure 127 shows a different story than strictly looking at the objective function. In this case, Figure 128 shows the relative difference. It takes into account multiple factors. The case that gave the worst objective function actually gave the highest TOPSIS value which is expected given the nature of TOPSIS and the dominating effect of the starting date and partial funding explained previously in Surveyor. Here the very same dominating dependency level 1 contributes to the highest TOPSIS overall case value.

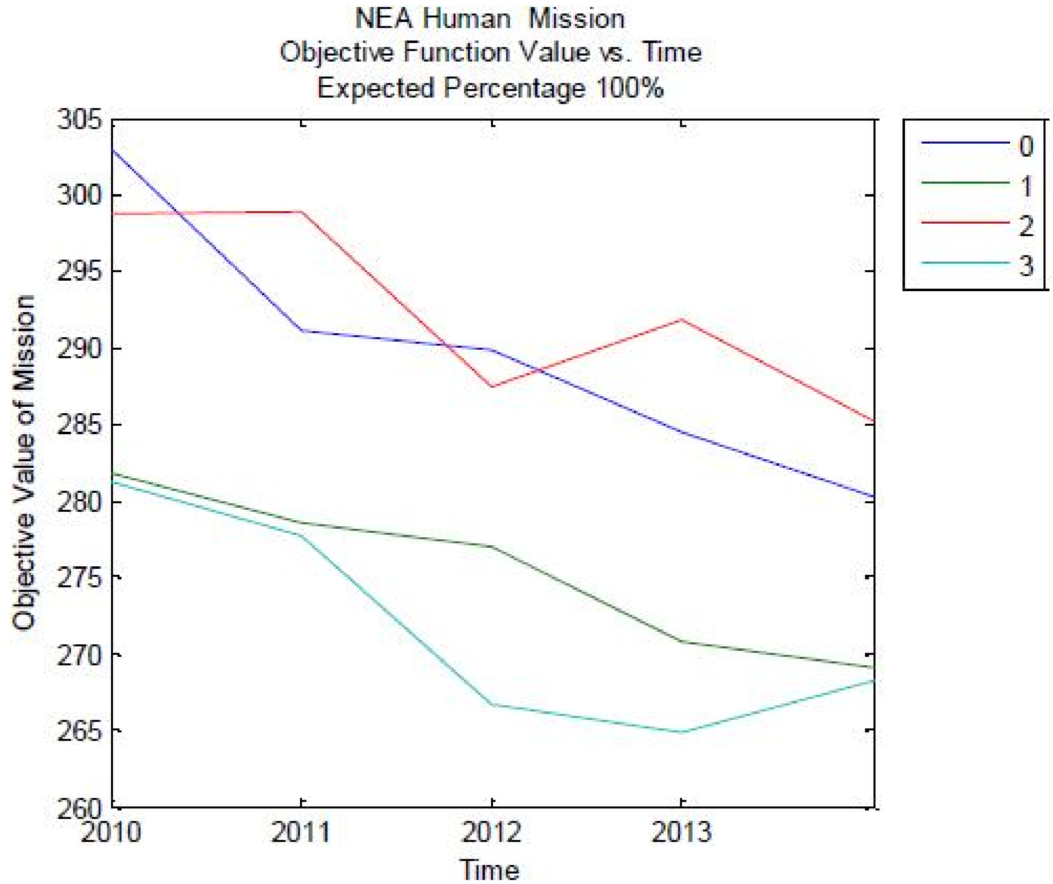


Figure 129: NEA Mission Objective Function vs. Time at 100% Expected Input Value

Figure 129 tells more about the starting time frame of the portfolio. The interesting phenomena happens here when the user looks at dependency level 1,2 and 3. In years 2010 and 2011, dependency level 3 follows along the dependency level 1 line. However in 2012, level 1 does not have a large dip, but level 2 does have a large dip. Therefore it can be concluded that dependency level 2 dominates the change in dependency level 3 in 2012.

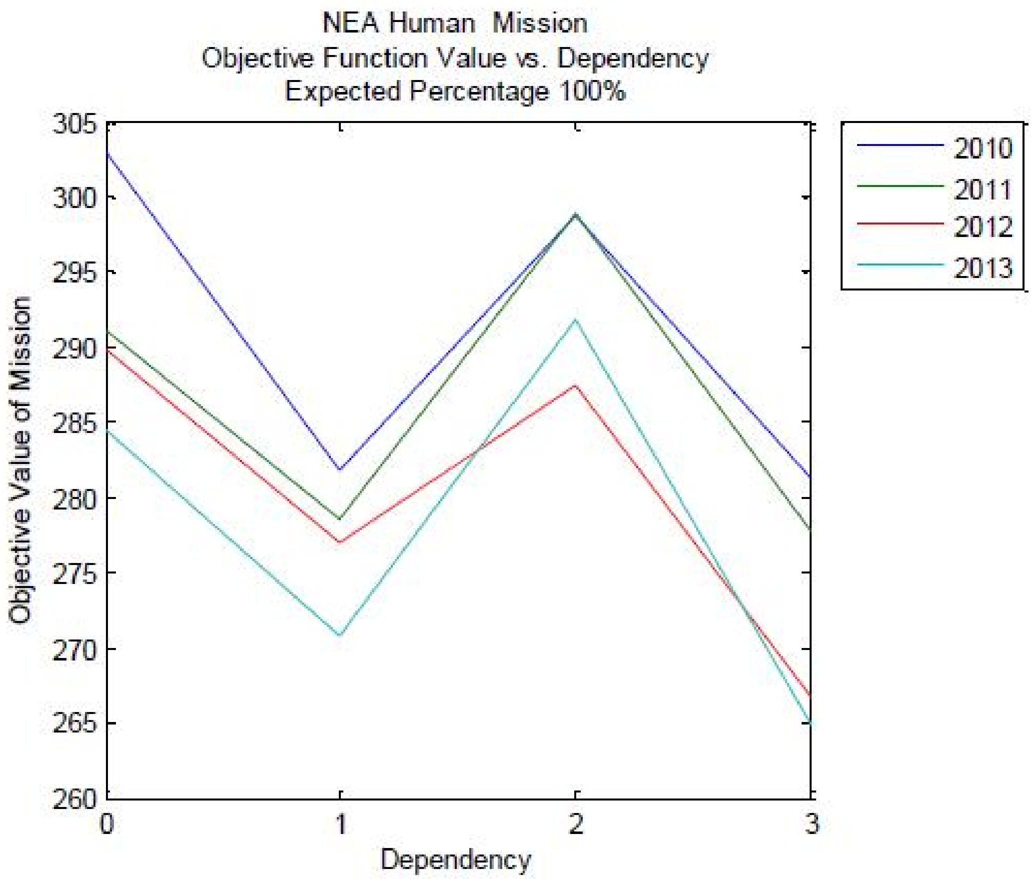


Figure 130: NEA Mission Objective Function vs. Dependency at 100% Expected Input Value

Viewing the data from another viewpoint to see what happens when the objective value is faced with the dependencies Figure 130 shows that as dependencies are added for specific years, the portfolio value goes down. No year always increases by modeling relationships, but there is a general decrease in the objective value from the standpoint of increasing relationships within a specific year.

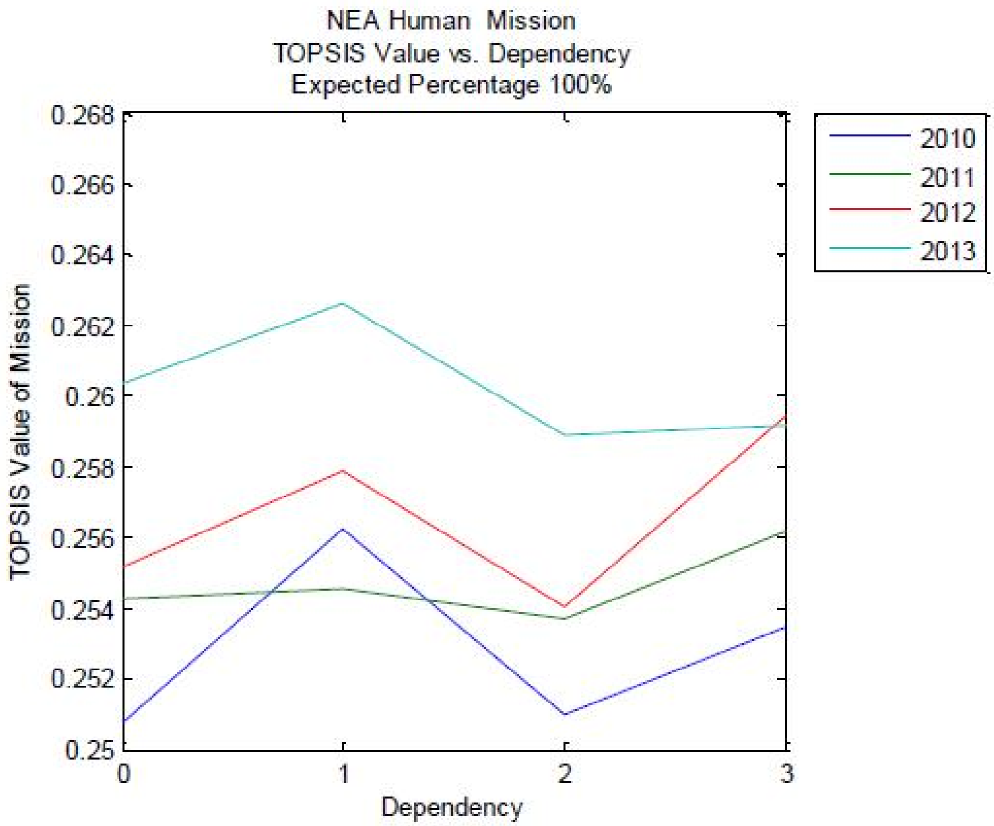


Figure 131: NEA Mission TOPSIS value vs. Dependency

Finally looking at Figure 131, shows that there is an increase in the relative portfolio changes as relationships are added. In this case TOPSIS is still dominated by the decreasing cost values due to later start years.

12.0.2 NEA summary:

The NEA Human mission gave a larger design space to investigate. As stated before, the NEA mission capabilities are more so enabling capabilities rather than enhancing. The NEA human mission must demonstrate human deep space capabilities for a short term mission unlike its long-term Mars counterpart. Thus the mission has fewer dependencies between capabilities and the phenomena is more pronounced. As the NEA mission adopts technologies the bottom up and top down analysis tells the user that the partial funding scenarios will not demonstrate all the needed capabilities.

This strengthens the concept that dependencies only matter during the enhancing capabilities; not the enabling ones. The entire Data cube for the NEA Human Mission Expected Input Percentages can be found in Appendix B.

12.1 NEA Baseline and Research Question 4

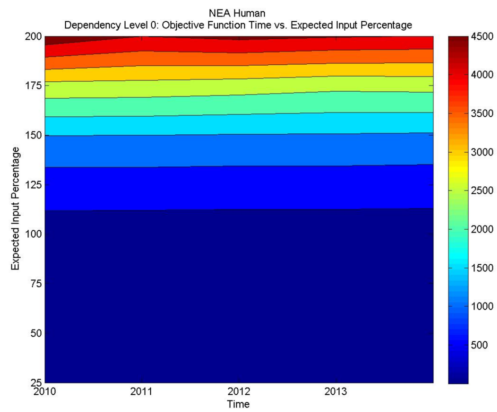
The last question dealt with the impact of changing the input values and adding dependencies. The question focuses on which one has a larger impact on the process. For this question, the NEA portfolio was chosen. Revisiting the NEA Baseline gives Figure 132 and Table 119 using a starting date of 2010, 100% expected input and no dependencies. Changing the expected input gives Figure 145 - Figure 152 as well as the associated statistics.

12.1.1 NEA Expected Input Percentage Conclusions

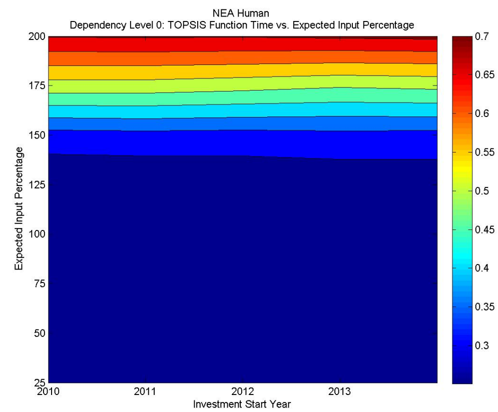
Cycling through the expected input percentages shows that adding dependencies into the process and cycling through the starting year shows that the values have a linear feel to them. The statistics show that as the changes have a negative slope going through the changes of the start investment year. In general the changes are lower than that of the baseline portfolio year. Each change in the dependency level from the baseline or change from the initial start year lowers the dependency portfolio.

12.1.2 NEA Dependency Level Cycle

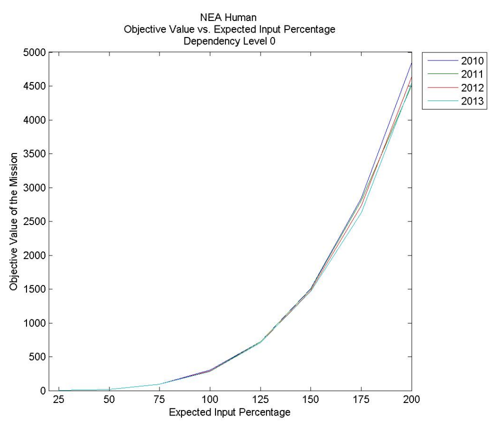
Changing focus to the NEA dependency cycles gives Figure 132 - Figure 139. This cycle gives the data cube in a different light.



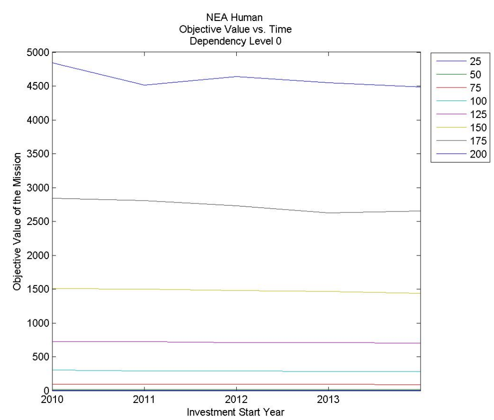
(a) Objective Value Contour



(b) TOPSIS Value Contour

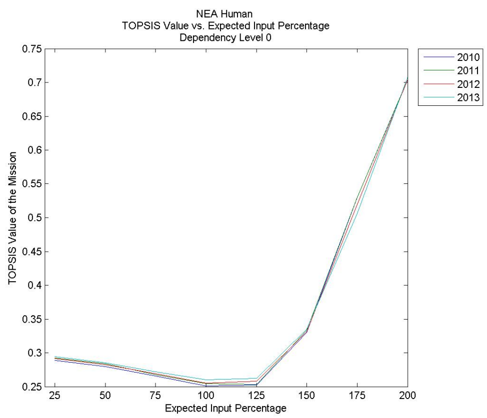


(c) Objective Value vs. Expected Input Percentage

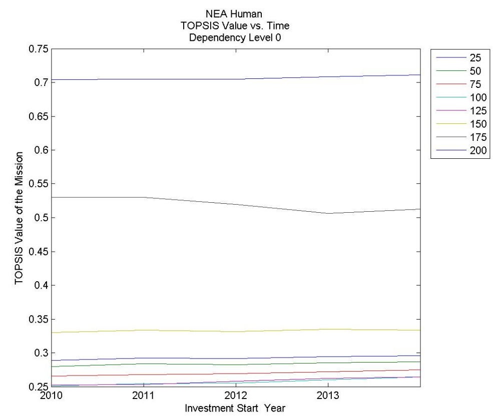


(d) TOPSIS vs. Expected Input Percentage

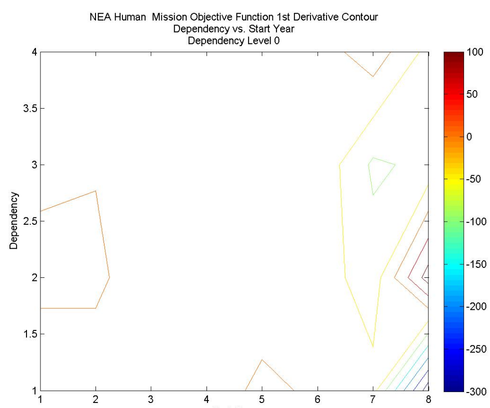
Figure 132: NEA Dependency Level 0 Data Information



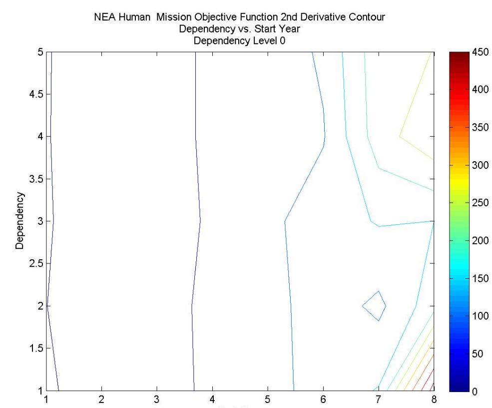
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour



(d) 2nd Derivative of Objective Value Contour

Figure 133: NEA Dependency Level 0 Data Information

Table 61: NEA Technology Portfolio

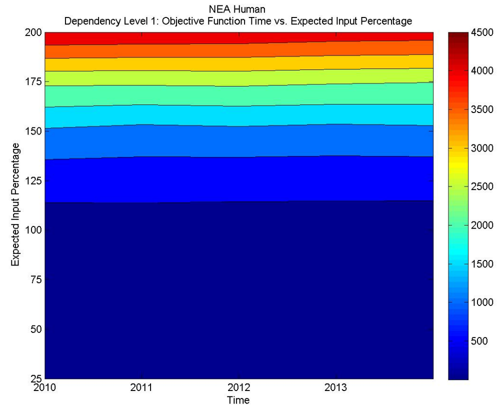
min	max	Delta	Average	Variation	Max Variation	Min Variation
1.094882	1.183298	0.088416	1.124673	0.078615	0.052126	0.026488
17.63147	18.93277	1.301303	18.06089	0.072051	0.048274	0.023777
90.02221	94.18233	4.16012	92.4656	0.044991	0.018566	0.026425
280.2898	302.9243	22.63446	289.7263	0.078124	0.045553	0.03257
700.4789	726.5552	26.07625	714.7374	0.036484	0.016534	0.019949
1440.355	1506.917	66.56192	1479.45	0.044991	0.018566	0.026425
2625.013	2843.286	218.2723	2733.552	0.079849	0.040143	0.039706
4484.638	4846.789	362.1514	4606.661	0.078615	0.052126	0.026488

Table 62: NEA Technology Portfolio

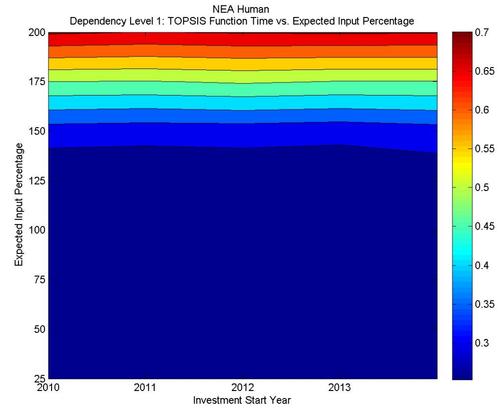
Start Year	x3	x2	x	b
2010	0.001398	-0.21604	12.52154	-213.367
2011	0.001043	-0.11831	4.883513	-60.2018
2012	0.001277	-0.18822	10.578	-177.416
2013	0.001276	-0.19403	11.37725	-198.362

Table 63: NEA Technology Portfolio

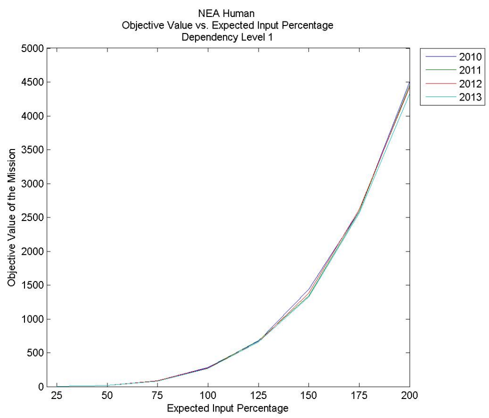
Expected Input Percentage	x	b
25	-0.01678	34.88667
50	-0.2219	464.5214
75	-1.05241	2209.907
100	-5.20075	10753.64
125	-5.59839	11978.71
150	-16.8385	35358.51
175	-56.0022	115410
200	-68.7322	142895.8



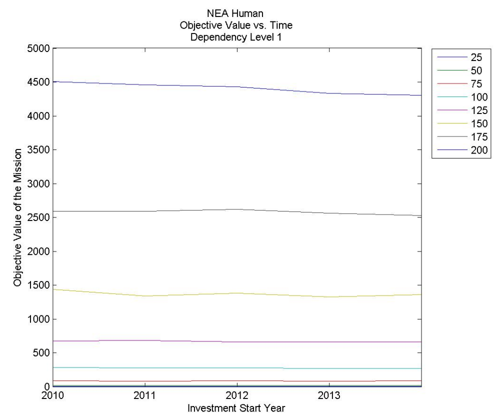
(a) Objective Value Contour



(b) TOPSIS Value Contour

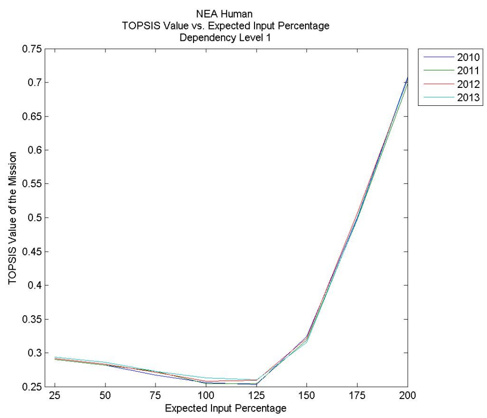


(c) Objective Value vs. Expected Input Percentage

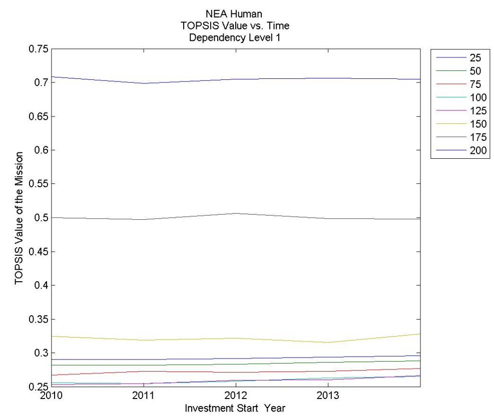


(d) TOPSIS vs. Expected Input Percentage

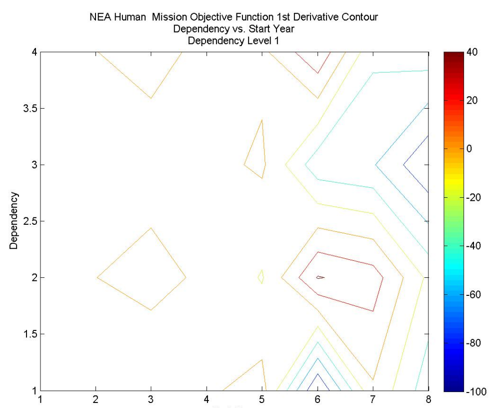
Figure 134: NEA Dependency Level 1 Data Information



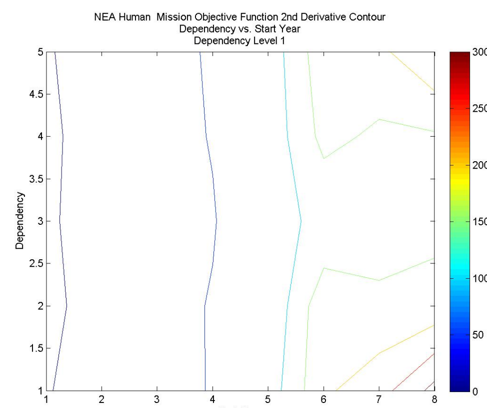
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour



(d) 2nd Derivative of Objective Value Contour

Figure 135: NEA Dependency Level 1 Data Information

Table 64: NEA Technology Portfolio

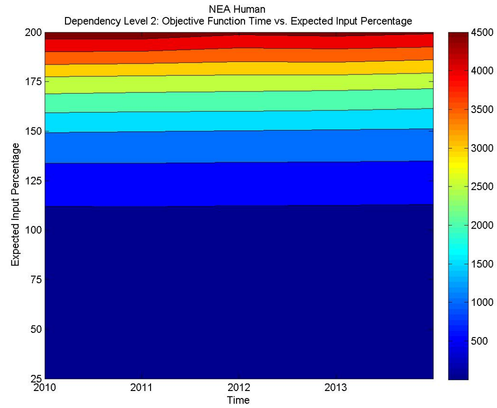
min	max	Delta	Average	Variation	Max Variation	Min Variation
1.050963	1.100663	0.0497	1.075885	0.046194	0.023031	0.023164
16.83238	17.61061	0.77823	17.22342	0.045184	0.022481	0.022703
82.9032	89.9143	7.011103	85.56223	0.081942	0.050864	0.031077
269.0467	281.7698	12.72318	275.4266	0.046194	0.023031	0.023164
659.2895	682.5056	23.21607	668.1556	0.034747	0.021477	0.013269
1326.451	1438.629	112.1777	1368.996	0.081942	0.050864	0.031077
2524.657	2618.668	94.01082	2576.912	0.036482	0.016204	0.020278
4304.746	4508.317	203.5709	4406.825	0.046194	0.023031	0.023164

Table 65: NEA Technology Portfolio

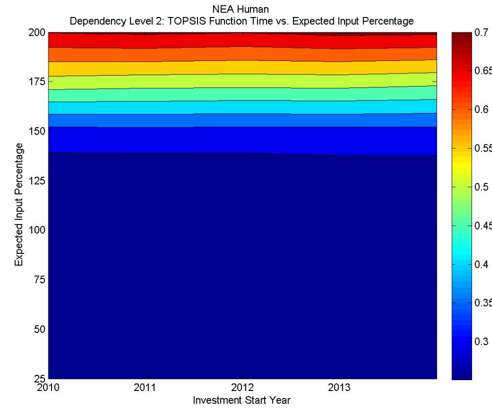
Start Year	x3	x2	x	b
2010	0.001303	-0.20337	12.03362	-209.501
2011	0.001348	-0.21931	13.32937	-235.557
2012	0.001261	-0.19165	10.92344	-183.621
2013	0.001248	-0.19261	11.17937	-191.053

Table 66: NEA Technology Portfolio

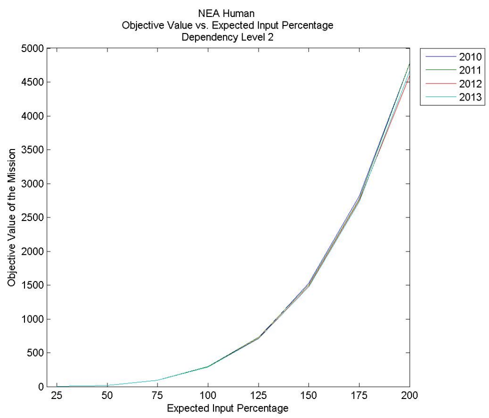
Expected Input Percentage	x	b
25	-0.01294	27.11463
50	-0.20367	427.0151
75	-1.00739	2112.426
100	-3.31308	6941.346
125	-4.87608	10478.83
150	-16.1182	33798.82
175	-16.2918	35356.08
200	-53.0093	111061.5



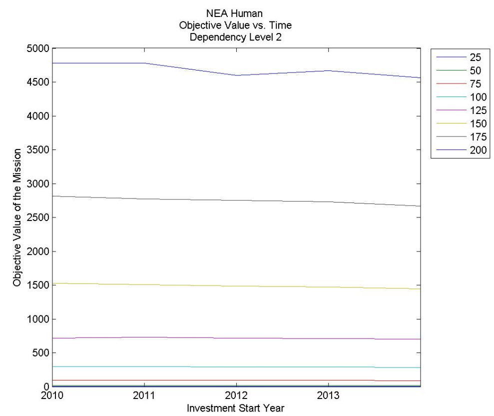
(a) Objective Value Contour



(b) TOPSIS Value Contour

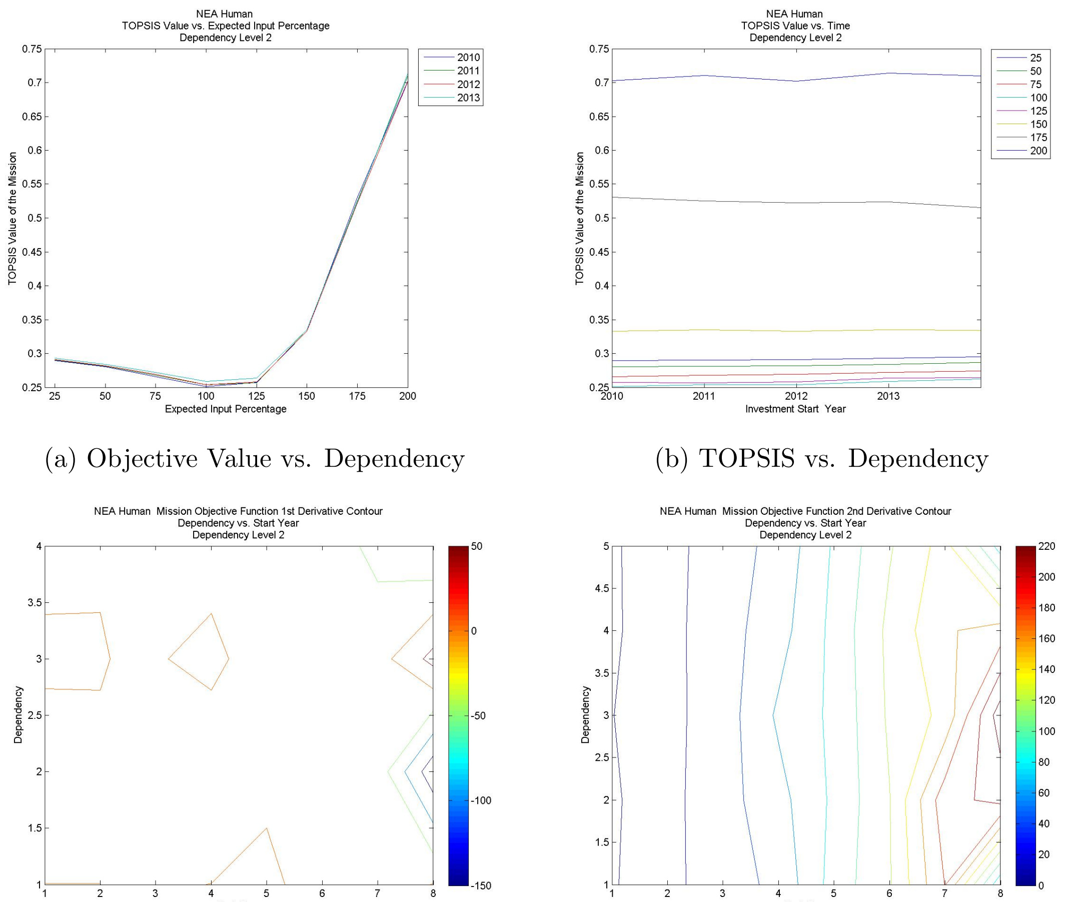


(c) Objective Value vs. Expected Input Percentage



(d) TOPSIS vs. Expected Input Percentage

Figure 136: NEA Dependency Level 2 Data Information



(a) Objective Value vs. Dependency

(b) TOPSIS vs. Dependency

(c) 1st Derivative of Objective Value Contour

(d) 2nd Derivative of Objective Value Contour

Figure 137: NEA Dependency Level 2 Data Information

Table 67: NEA Technology Portfolio

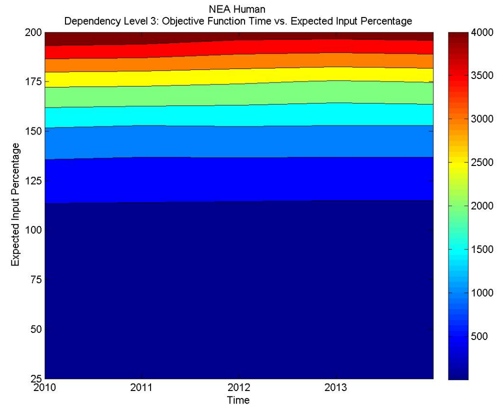
min	max	Delta	Average	Variation	Max Variation	Min Variation
1.114162	1.167706	0.053544	1.142087	0.046883	0.022432	0.024451
17.84328	18.68329	0.840015	18.28012	0.045952	0.022055	0.023897
90.32767	95.45798	5.130305	93.01133	0.055158	0.026305	0.028853
285.2254	298.9327	13.70731	292.4285	0.046874	0.022242	0.024632
701.7311	728.0082	26.27712	715.6568	0.036717	0.017259	0.019459
1445.243	1527.328	82.08488	1488.181	0.055158	0.026305	0.028853
2667.664	2814.422	146.7574	2747.551	0.053414	0.024338	0.029076
4563.606	4782.923	219.3169	4677.988	0.046883	0.022432	0.024451

Table 68: NEA Technology Portfolio

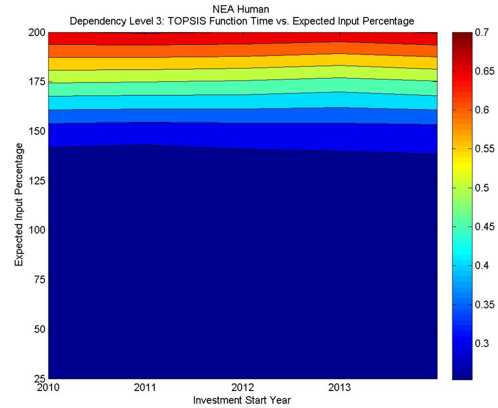
Start Year	x3	x2	x	b
2010	0.001324	-0.19595	10.99478	-183.526
2011	0.001375	-0.21318	12.54489	-217.637
2012	0.001202	-0.16594	8.76685	-140.289
2013	0.001323	-0.20157	11.64368	-198.954

Table 69: NEA Technology Portfolio

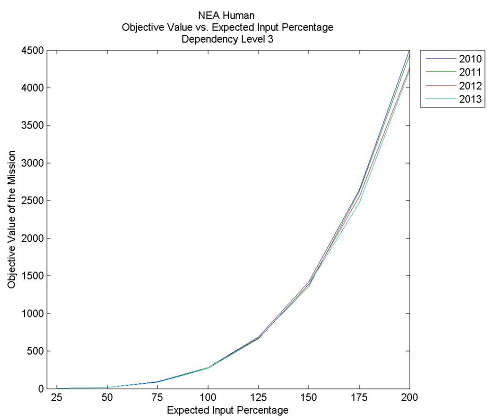
Expected Input Percentage	x	b
25	-0.01349	28.28325
50	-0.2108	442.4079
75	-1.28276	2673.924
100	-3.4262	7185.94
125	-5.31936	11418.2
150	-20.5242	42782.78
175	-33.539	70228.03
200	-55.2536	115848.2



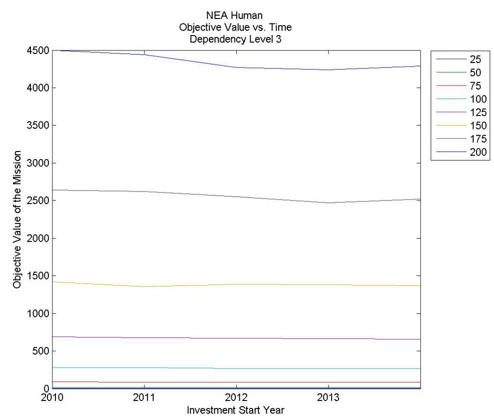
(a) Objective Value Contour



(b) TOPSIS Value Contour

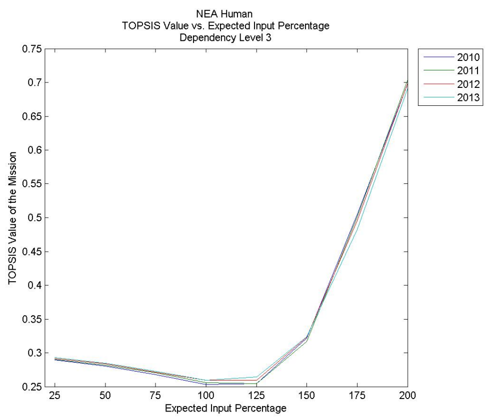


(c) Objective Value vs. Expected Input Percentage

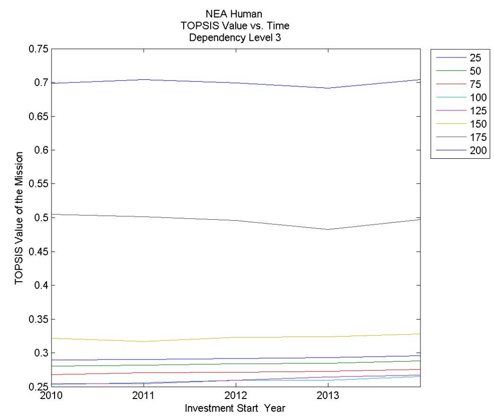


(d) TOPSIS vs. Expected Input Percentage

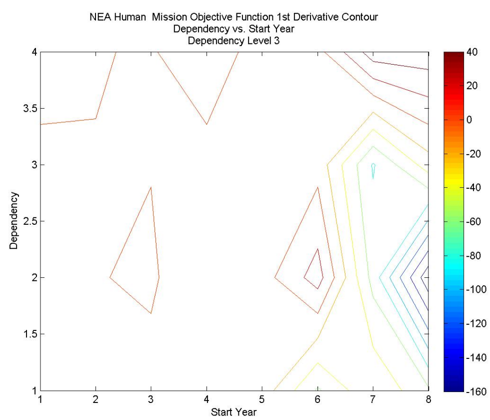
Figure 138: NEA Dependency Level 3 Data Information



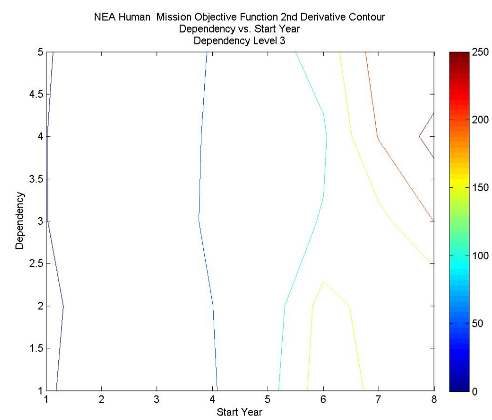
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour



(d) 2nd Derivative of Objective Value Contour

Figure 139: NEA Dependency Level 3 Data Information

Table 70: NEA Technology Portfolio

Expected Input Percentage	min	max	Delta	Average	Variation	Max Variation	Min Variation
25	1.034861	1.098411	0.06355	1.061571	0.059864	0.034703	0.025161
50	16.55777	17.57457	1.016804	17.00777	0.059785	0.033326	0.026459
75	84.77781	88.69699	3.919184	86.39339	0.045364	0.026664	0.0187
100	264.9243	281.1932	16.26887	271.7621	0.059864	0.034703	0.025161
125	659.2336	686.1356	26.90202	670.4361	0.040126	0.023417	0.016709
150	1356.445	1419.152	62.70694	1382.294	0.045364	0.026664	0.0187
175	2468.977	2641.083	172.106	2559.914	0.067231	0.031708	0.035524
200	4238.789	4499.091	260.3019	4348.194	0.059864	0.034703	0.025161

Table 71: NEA Technology Portfolio

Start Year	x3	x2	x	b
2010	0.001267	-0.19147	10.9767	-186.518
2011	0.001291	-0.201	11.74583	-201.598
2012	0.001115	-0.15394	8.146547	-130.652
2013	0.001158	-0.17048	9.694298	-165.271

Table 72: NEA Technology Portfolio

Expected Input Percentage	x	b
25	-0.0151	31.44889
50	-0.23163	483.0407
75	-0.45696	1005.805
100	-3.86638	8050.916
125	-6.89625	14545.69
150	-7.31143	16092.88
175	-39.1256	81280.6
200	-61.8621	128814.7

12.1.3 NEA Dependency Data Cube Cycle

Cycling through the NEA dependency data cube shows that the changes in the expected input percentage take on a polynomial form while the time changes take on a linear form. The expected input percentages has a faster change in objective function than the time changes. Adding dependencies is not a continuous change and depends on the class of dependency being input into the system.

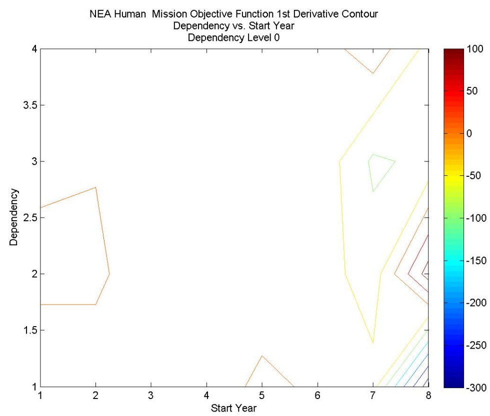
12.1.3.1 NEA Dependency 1st and 2nd Derivatives Analysis

Focusing on the first and second derivatives given in Figures 132 - Figure 139 shows the derivatives for the expected input percentage and the start investment year. This is the only case of the data cubes that give continuous derivatives. Every other derivative graph had the Dependency as an axis which corresponds into discrete steps in the derivative, not continuous ones. This information is reproduced below in Figure 140 and Figure 141.

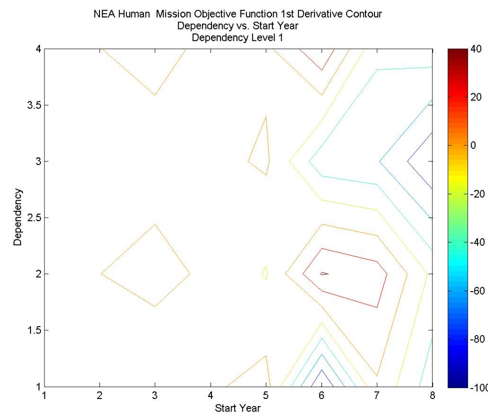
Looking closer at the derivative information provides some insight as to how fast the expected input percentage and start investment year changes with respect to a given dependency level. Starting at Dependency Level 0 shows that it has the largest variation in values looking at the colorbar in Figure 140a and Figure 140b. This shows the first derivative varying from -300 to 100 and the second derivative varying from 0 to 450. As dependencies are added, this design space reduces for each of the constraining dependencies implemented. The same concept happens in the Laplacian where the baseline has the highest acceleration at 450 and it goes down to 220 in Dependency Level 2.

The NEA Human Mission only has Constraint Dependencies included. As constraint dependencies are added to the design space, the first and second derivatives are reduced accordingly. The Dependency Level 3 is a combination of both Dependency 1 and Dependency 2 constraints. Looking strictly at the colorbars on the sides show that Dependency Level 3 is always between Dependency Level 1 and Level 2's colorbar extreme graph; however, all changes in dependency levels are an extreme decrease from Dependency Level 0's colorbar design space constraints.

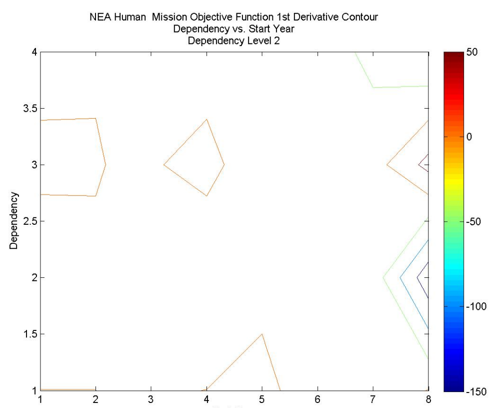
Shifting focus to the actual shapes of the derivatives show that the changes in expected input percentage are not as disruptive as the change in the investment start year. This is shown in the fact that the second derivative as seen in Figure 141 has horizontal lines coming from the expected percentage changes. The real changes in the second derivative comes from the higher investment start years where the acceleration increases quite substantially; especially in the higher expected input percentage ranges. This is consistent with the fact that there is a polynomial function associated with the expected input percentage. Therefore, the derivative of the polynomial function will have a higher impact in the higher values of the expected input percentage than that of the lower expected input percentage. The investment start year has larger disruptive effect on the portfolio objective value. This is seen in the first derivatives shown in Figure 140.



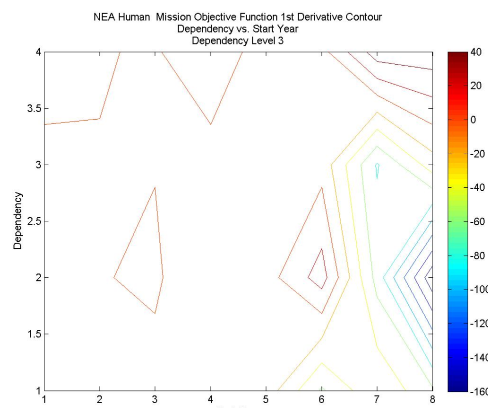
(a) Dependency Level 0



(b) Dependency Level 1

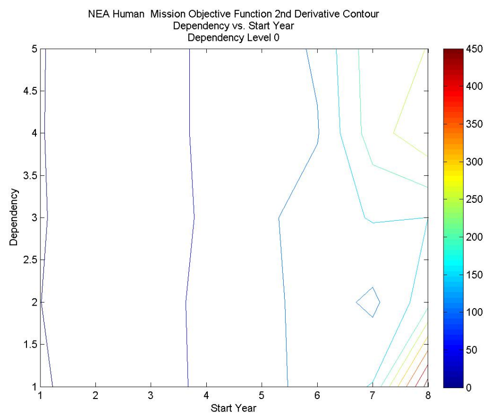


(c) Dependency Level 2

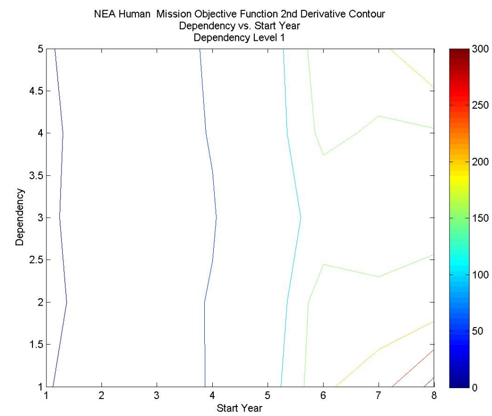


(d) Dependency Level 3

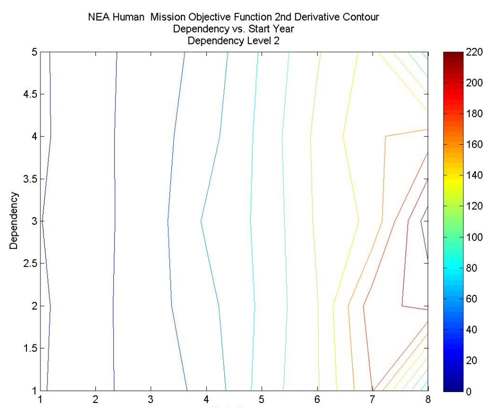
Figure 140: NEA 1st Derivative Comparison



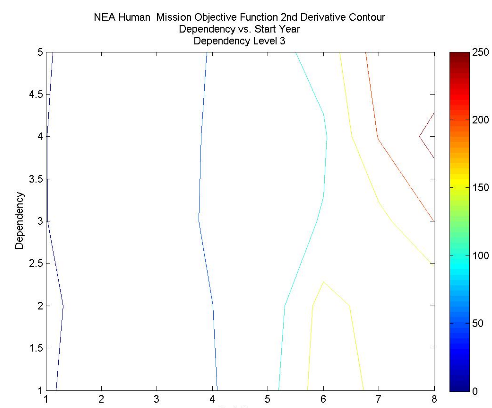
(a) Dependency Level 0



(b) Dependency Level 1



(c) Dependency Level 2



(d) Dependency Level 3

Figure 141: NEA 2nd Derivative Comparison

12.1.4 NEA Dependency and Expected Input Conclusions

The conclusions for this research question focuses that adding dependency constraints give higher fidelity by constraining the problem. Changing the expected input percentages changes the portfolio by a polynomial fit in this case. Both changes occur due to the constraint changes in the START equations. There is no answer to which is more important, but changing the expected input percentages has a larger net positive impact on the portfolio while the dependency inclusions have a local negative impact. The reason is that a dependency inclusion is a new constraint. Including this constraint means that it may be adhered to or not. If it is not met, then it is possible that both elements are not chosen. Therefore the portfolio cannot be greater than that of the baseline if it is a Constraint dependency.

If the dependency is a Value dependency then the dependency is changing the objective function rather than the constraints found in START. In this case, the dependency inclusion can increase the objective function, but that is not going to be a polynomial fit with the way the expected input percentage is changed in this case. In this case the comparison is between MIMO and the polynomial fit of the capability. Assuming it is always chosen, that is the potential change in the portfolio objective value.

When it comes to the addition of Constraint dependencies, the expected percentage will have a larger effect on the portfolio. The addition of Value dependencies by themselves, has the possibility to be higher than that of the expected input percentage. This is heavily influenced by the partial funding concept input into START as well. If there is no partial funding, and the capability is chosen then it depends on if the Value has a higher change in the objective value than that of the polynomial change from the expected input percentage. If there is partial funding, then the potential change is still dependent upon the difference between the two, but in the case of Value the dependency may change the portfolio objective value.

CHAPTER XIII

NEA CAMPAIGN RESULTS

The NEA Campaign consists of the three missions together: Surveyor, Explorer and NEA Human. The Surveyor and Explorer Missions done together showed the precursor mission scenario for the feed-forward network. The NEA Campaign shows all three missions and how their technology adoption plays into the feed-forward capability for human deep space exploration capabilities.

Figure 142- Figure 145 show the technology portfolio for the combined NEA Campaign mission as well as the three missions that make up the Campaign. Once again the NEA Human mission dominates the NEA Campaign, because it has the largest contribution to the objective value possible. The Explorer Mission has more technologies funded than the Surveyor mission. It must be noted here, that the capabilities were not constrained between the missions. If mobility was chosen on Surveyor, Explorer and NEA Human, then there would be the appropriate Mars contributions as backed out from the Feed-Forward analysis explain in Chapter 8. A paper is coming out on the feed-forward aspects of constraining the capabilities to be appropriated between missions.

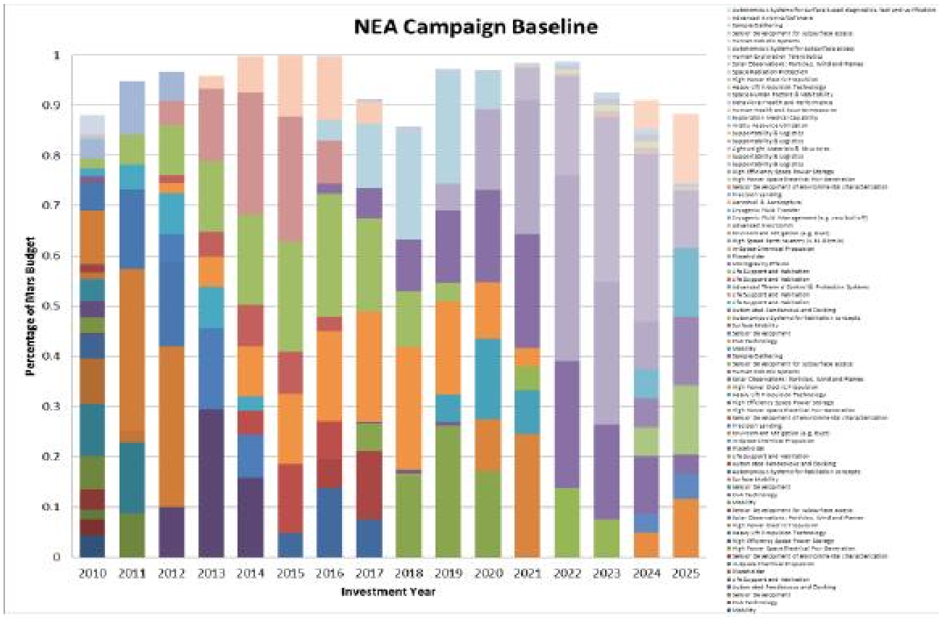


Figure 142: NEA Campaign Baseline

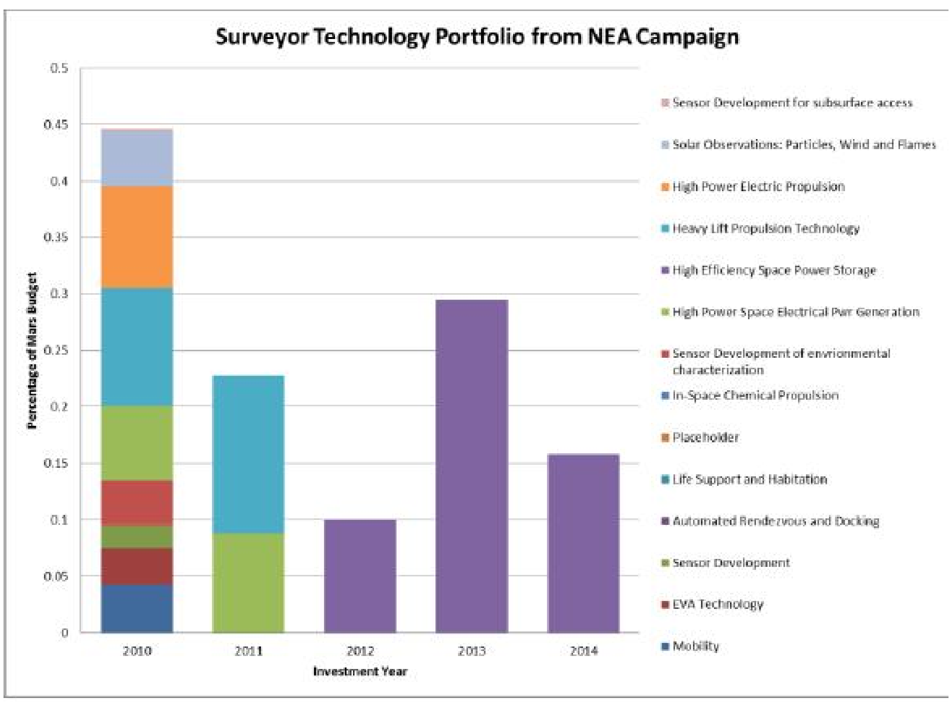


Figure 143: Surveyor Mission from NEA Campaign Baseline

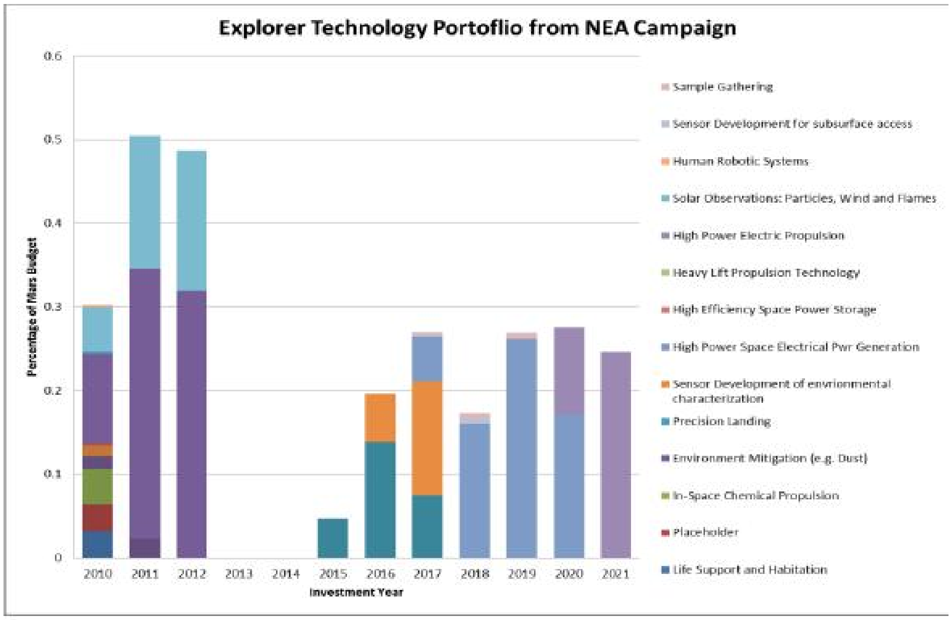


Figure 144: Explorer Mission from NEA Campaign Baseline

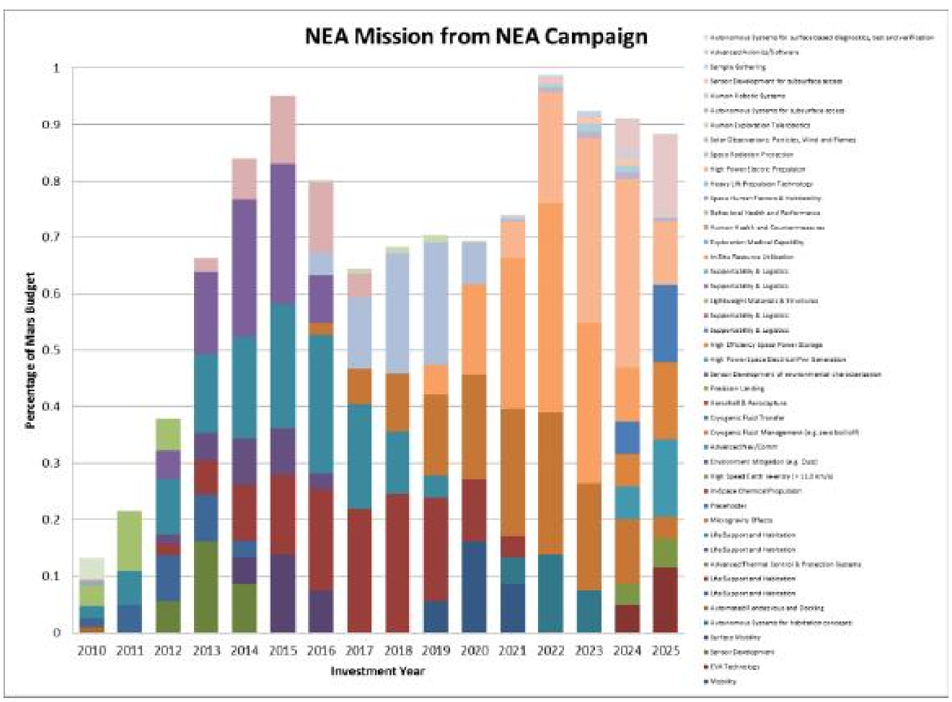


Figure 145: NEA Mission from NEA Campaign Baseline

Table 73-Table 75 give the specific portfolios compared to their separate mission baseline counterparts. The NEA Campaign gave different results than the Precursor Mission scenario. The NEA Campaign actually produced higher objective value than optimizing the three missions separately. The NEA Campaign came in with an objective value of 579.89 compared to a value of 565.38 when optimized separately. The Precursor mission gave opposite results with a combined objective value of 200.31, but a separate value of 262.39. Both are combined scenarios, with completely different results. Figure 146-Figure 147 give the data design space for each mission of the NEA Campaign and the combined mission. Figure 146 and Figure 147 is a summation of Figure 148-Figure 149. The information presented is different from that of the Explorer and Surveyor missions given separately and previously.

Table 73: Surveyor Comparison Portfolio

Metric	Surveyor Alone Selection	Partial Funding	Surveyor Combined Selection	Partial Funding
Mobility	Selected	0.7	Selected	1
EVA Technology	Selected	1	Selected	1
Sensor Development	Selected	1	Selected	1
Automated Rendezvous and Docking	Selected	1	Selected	1
Life Support and Habitation	Selected	1	Selected	1
Placeholder	Selected	1	Selected	1
In-Space Chemical Propulsion	NOT Selected		NOT Selected	
Sensor Development of environmental characterization	Selected	0.9	Selected	1
High Power Space Electrical Power Generation	Selected	0.6	Selected	1
High Efficiency Space Power Storage	NOT Selected		Selected	0.9
Heavy Lift Propulsion Technology	Selected	0.4	Selected	0.9
High Power Electric Propulsion	Selected	0.8	Selected	1
Solar Observations: Particles, Wind and Flames	Selected	0.9	Selected	1
Sensor Development for subsurface access	Selected	1	Selected	1

Table 74: Explorer Comparison Portfolio

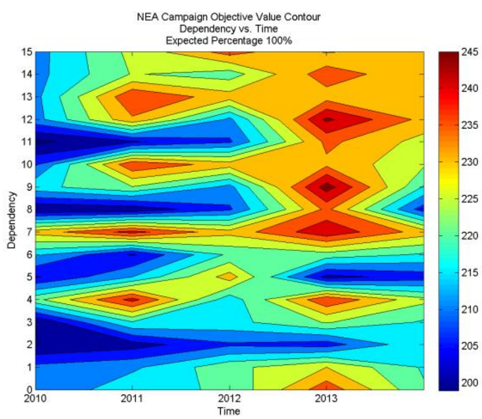
Metric	Explorer Alone Selection	Partial Funding	Combined Missions Selection	Partial Funding
Mobility	Selected	1	Selected	1
EVA Technology	Selected	1	Selected	1
Sensor Development	Selected	1	Selected	1
Surface Mobility	Selected	0.8	Selected	0.3
Autonomous Systems for habitation concepts	Selected	0.9	Selected	1
Automated Rendezvous and Docking	Selected	1	Selected	1
Life Support and Habitation	Selected	1	Selected	1
Placeholder	Selected	1	Selected	1
In-Space Chemical Propulsion	NOT Selected		NOT Selected	
Environment Mitigation (e.g. Dust)	NOT Selected		Selected	0.8
Precision Landing	Selected	1	Selected	1
Sensor Development of environmental characterization	Selected	1	Selected	1
High Power Space Electrical Power Generation	Selected	0.5	Selected	0.9
High Efficiency Space Power Storage	NOT Selected		NOT Selected	
Heavy Lift Propulsion Technology	NOT Selected		NOT Selected	
High Power Electric Propulsion	Selected	0.6	Selected	1
Solar Observations: Particles, Wind and Flames	Selected	0.8	Selected	0.9
Human Robotic Systems	Selected	1	Selected	1
Sensor Development for subsurface access	Selected	1	Selected	1
Sample Gathering	Selected	1	Selected	1

Table 75: NEA Comparison Portfolio

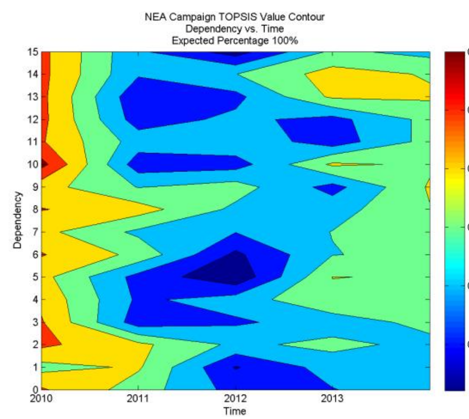
Technology	NEA Human Baseline Selection	Partial Funding	NEA Campaign Baseline Selection	Partial Funding
Mobility	Selected	1	Selected	1
EVA Technology	Selected	1	Selected	1
Sensor Development	Selected	1	Selected	1
Surface Mobility	Selected	1	Selected	1
Autonomous Systems for habitation concepts	Selected	0.9	Selected	1
Automated Rendezvous and Docking	Selected	1	Selected	1
Life Support and Habitation	Selected	1	Selected	1
Life Support and Habitation	Selected	0.6	Selected	0.6
Advanced Thermal Control & Protection Systems	Selected	1	Selected	1
Life Support and Habitation	Selected	1	Selected	1
Life Support and Habitation	Selected	0.5	Selected	0.6
Microgravity Effects	Selected	0.6	Selected	0.6
Placeholder	NOT Selected		NOT Selected	
In-Space Chemical Propulsion	NOT Selected		NOT Selected	
High Speed Earth re-entry (> 11.0 km/s)	NOT Selected		NOT Selected	
Environment Mitigation (e.g. Dust)	NOT Selected		NOT Selected	
Advanced Nav/Comm	Selected	1	Selected	1
Cryogenic Fluid Management (e.g. zero boil off)	Selected	1	Selected	1
Cryogenic Fluid Transfer	Selected	1	Selected	1
Aeroshell & Aerocapture	NOT Selected		NOT Selected	
Precision Landing	NOT Selected		NOT Selected	
Sensor Development of environmental characterization	Selected	0.8	Selected	0.8
High Power Space Electrical Power Generation	NOT Selected		NOT Selected	
High Efficiency Space Power Storage	NOT Selected		NOT Selected	
Supportability & Logistics	NOT Selected		NOT Selected	
Lightweight Materials & Structures	Selected	0.8	Not Selected	
Supportability & Logistics	NOT Selected		NOT Selected	
In-Situ Resource Utilization	Selected	0.6	Selected	0.5
Exploration Medical Capability	Selected	1	Selected	1
Human Health and Countermeasures	Selected	1	Selected	1
Behavioral Health and Performance	Selected	1	Selected	1
Space Human Factors & Habitability	Selected	1	Selected	1
High Power Electric Propulsion	Selected	0.6	Selected	0.6
Space Radiation Protection	Selected	0.8	Selected	0.7
Solar Observations: Particles, Wind and Flames	Selected	0.6	Selected	0.4
Human Exploration Telerobotics	Selected	1	Selected	1
Autonomous Systems for subsurface access	Selected	1	Selected	1
Human Robotic Systems	Selected	1	Selected	1
Sensor Development for subsurface access	Selected	1	Selected	1
Sample Gathering	Selected	1	Selected	1
Advanced Avionics/Software	Selected	0.9	Selected	1
Autonomous Systems for surface based diagnostics, test and verification	Selected	1	Selected	1

13.1 *NEA Campaign Summary*

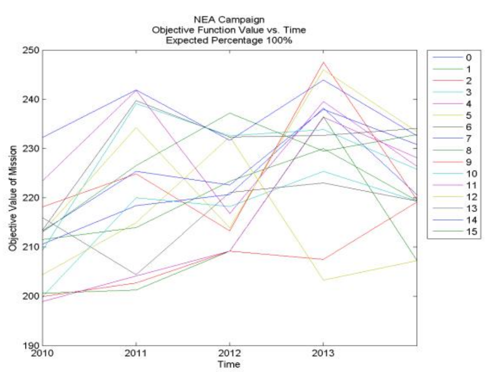
The NEA Campaign consisted of the three nominal missions of the Surveyor, Explorer and NEA Human mission. Four dependencies were included to run 15 different cases as shown in Figures 146-153. Each mission responded differently to the campaign scenario than it did in the separate missions or even the precursor mission. The campaign was funded at 33% of the total Mars campaign.



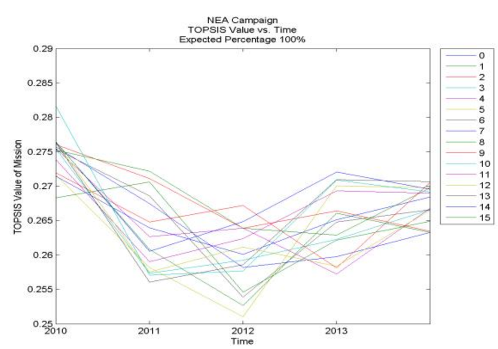
(a) Objective Value Contour



(b) TOPSIS Value Contour

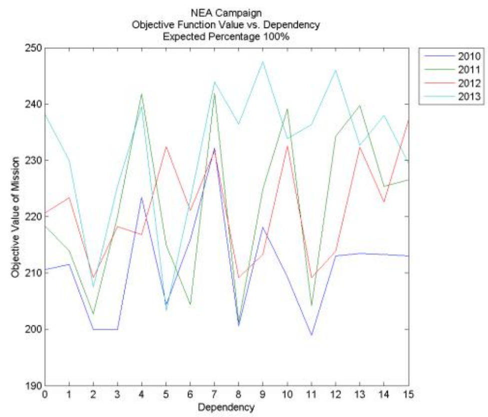


(c) Objective Value vs. Expected Input Percentage

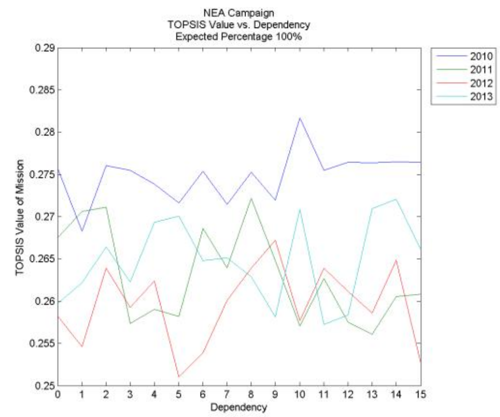


(d) TOPSIS vs. Expected Input Percentage

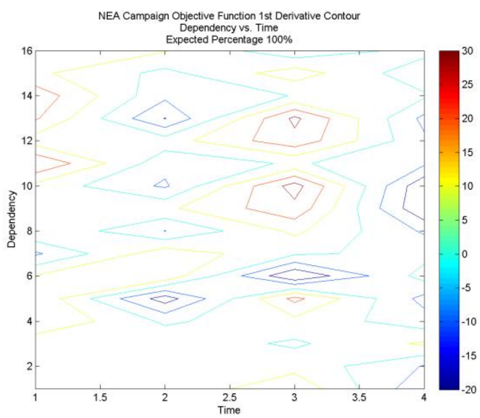
Figure 146: NEA Campaign Portfolio Information



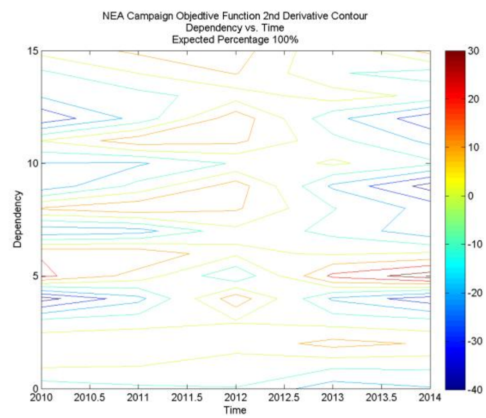
(a) Objective Value vs. Start Year



(b) TOPSIS vs. Start Year



(c) Objective Value 1st Derivative



(d) Objective Value 2nd Derivative

Figure 147: NEA Campaign Portfolio Information (con't)

Table 76: NEA Campaign Statistical Information

Year	min	max	Delta	Average	Variation	Max Variation	Min Variation
2010	198.9572	232.2008	33.24355	211.098	0.157479	0.099967	0.057512
2011	201.2625	241.887	40.62448	222.0845	0.182924	0.089167	0.093757
2012	209.2005	237.2401	28.03961	221.4824	0.1266	0.071147	0.055453
2013	203.2623	247.539	44.2767	231.9509	0.190888	0.067204	0.123684
2014	207.2561	234.12	26.8639	223.4934	0.1202	0.047548	0.072652
Extremes	198.957	247.539					

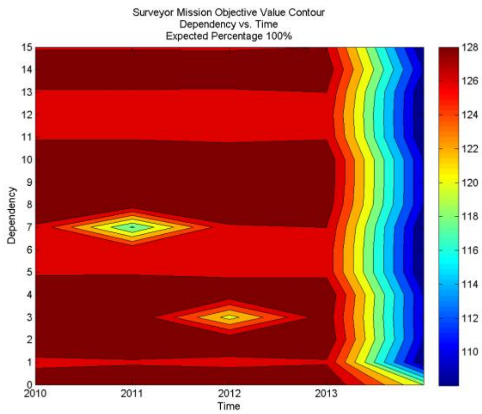
13.1.1 Surveyor

The nominal Surveyor mission selected every technology except for the In-Space Chemical propulsion system in the baseline as seen in Table 73. Looking at the contour in Figure 148a shows that the Surveyor gave values up to 128, but seemed red throughout the process until 2014 when the technology was due and there was no longer reason to fund the mission. The subsequent TOPSIS value in Figure 148b gave similar results with high TOPSIS values looking at the Surveyor from the Campaign point of view. The interesting features here are Dependency Levels 3 and 7 as seen in Figure 149a. Looking at Figure 149a shows a similar step function pattern to the No Partial funding section on Surveyor in Chapter 9. This results suggests that because Surveyor is at such a lower cost, the money not spent on higher priced items in Explorer and NEA Human Mission is expended here and most of the Surveyor technologies are funded in full according to the dependencies implemented.

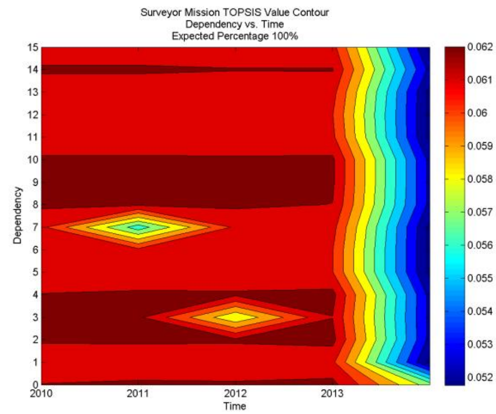
Looking at Dependency 3 shows that it does not directly involve the Surveyor mission. Dependency level 3 involves the Automated Rendezvous and Docking between the NEA human and Explorer missions. Level 3 does not involve Surveyor, but gives an outlier result in investment year 2010. Comparatively Dependency level 7 includes multiple dependencies, but among them MIMO dependencies between the NEA Human mission and Surveyor missions for remote sensing data that pertains to human EVA. While Level 7 does directly involve the Surveyor mission, it only gives an outlier result to the Surveyor portfolio in investment start year 2011.

Table 77: Surveyor Mission from NEA Campaign Statistical Information

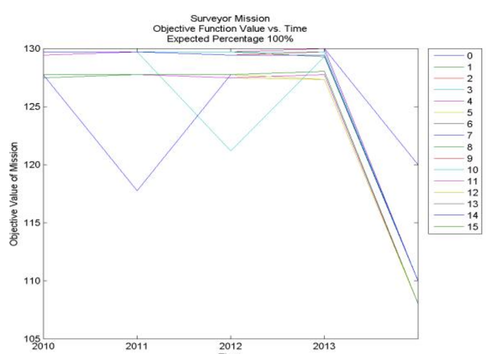
Year	min	max	Delta	Average	Variation	Max Variation	Min Variation
2010	127.4686	129.7021	2.233455	128.6702	0.017358	0.00802	0.009338
2011	117.7421	129.7021	11.95992	128.0962	0.093367	0.012536	0.08083
2012	121.1761	129.7021	8.52596	128.1034	0.066555	0.01248	0.054076
2013	127.3351	129.9736	2.638491	128.6794	0.020504	0.010058	0.010447
2014	108.0157	119.9756	11.95992	109.6215	0.109102	0.094453	0.014649
Extremes	108.016	129.974					



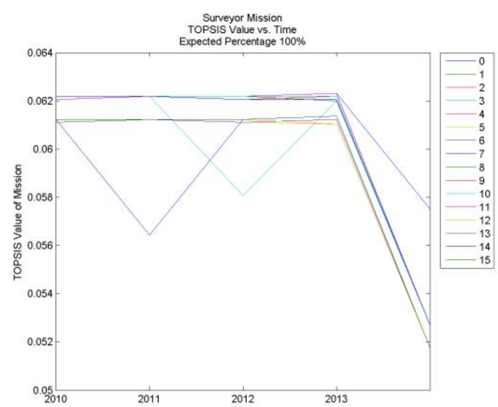
(a) Objective Value Contour



(b) TOPSIS Value Contour

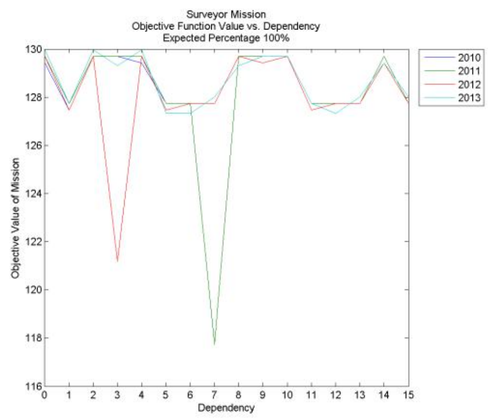


(c) Objective Value vs. Expected Input Percentage

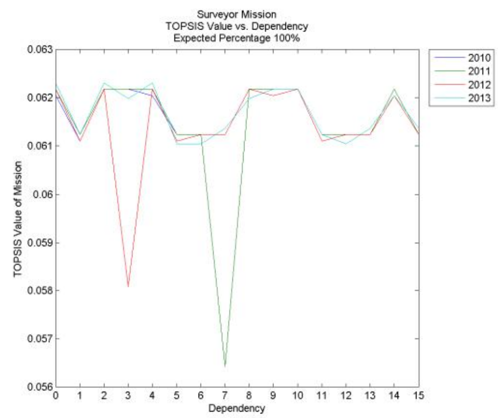


(d) TOPSIS vs. Expected Input Percentage

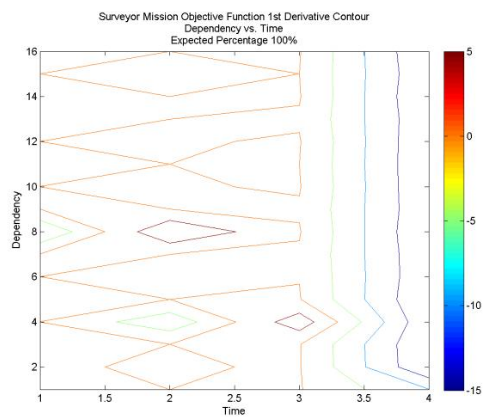
Figure 148: Surveyor Mission from NEA Campaign Information



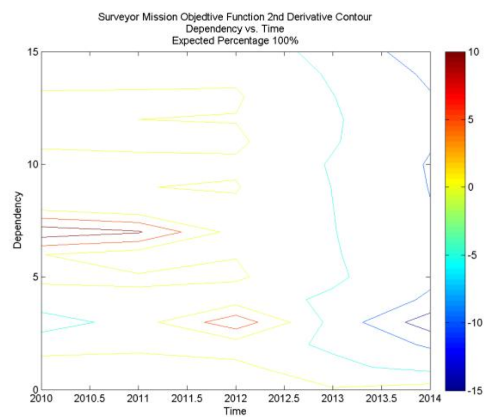
(a) Objective Value vs. Start Year



(b) TOPSIS vs. Start Year



(c) Objective Value 1st Derivative



(d) Objective Value 2nd Derivative

Figure 149: Surveyor Mission from NEA Campaign Information (cont'd)

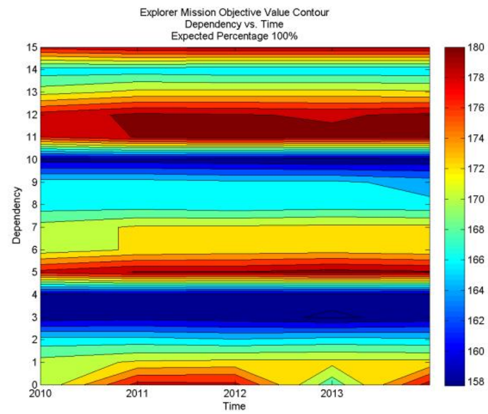
13.1.2 Explorer

The nominal Explorer mission selected most technologies with the separate mission except for it included the Environmental dust mitigation and decreased the funding for the Surface mobility in the baseline as seen in Table 74. The Explorer mission objective value contours and TOPSIS contours in Figure 150a and Figure 150b show that the benefit is not as unanimous for all of the dependency inclusions as the Surveyor Mission was. The highest objective value that the Explorer mission gives is 180 according to the colorbar on Figure 150a. Comparatively Explorer gave a high value of 150 when it was ran by itself in Chapter 10 and 160 when ran in the Precursor Mission Scenario in Chapter 11. Running it in a larger design space with dependencies gave a larger range in possible value which is consistent with running Explorer in the Precursor space in Chapter 11.

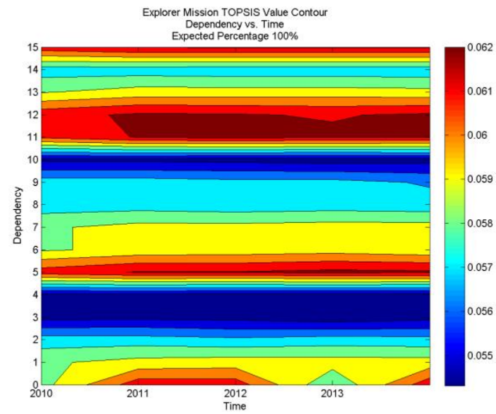
Specifically Dependency Levels 5, 11 and 12 gave much higher results for investment start years after 2011. Conversely Levels 3,4, and 10 gave the lowest values for the Explorer mission as seen in Figure 151a. Level 5, 11 and 12 all include dependencies 1 and 2, while Level 10 includes levels 3 and 4. This suggests that the combination of Dependency level 1 and 2 gives higher results, while the combination of levels 3 and 4 gives lower results.

Table 78: Explorer Mission from NEA Campaign Statistical Information

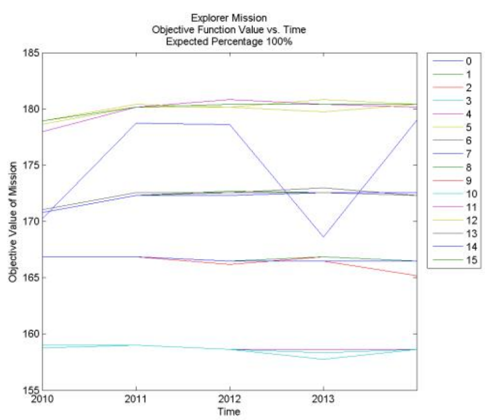
Year	min	max	Delta	Average	Variation	Max Variation	Min Variation
2010	158.7479	178.919	20.17109	169.525	0.118986	0.055414	0.063572
2011	159.0194	180.4247	21.40526	170.853	0.125285	0.056023	0.069262
2012	158.6144	180.8297	22.21533	170.7351	0.130116	0.059125	0.070991
2013	157.7386	180.8297	23.09107	170.1317	0.135725	0.062881	0.072844
2014	158.6144	180.4247	21.81029	170.6636	0.127797	0.057195	0.070602
Extremes	157.739	180.830					



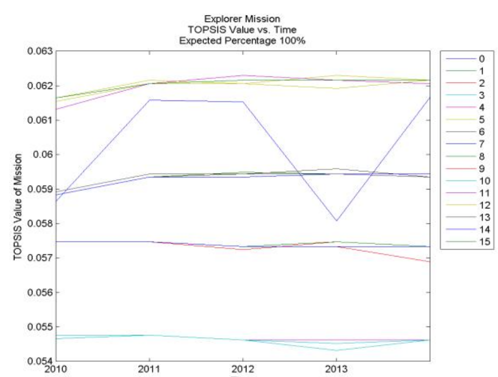
(a) Objective Value Contour



(b) TOPSIS Value Contour

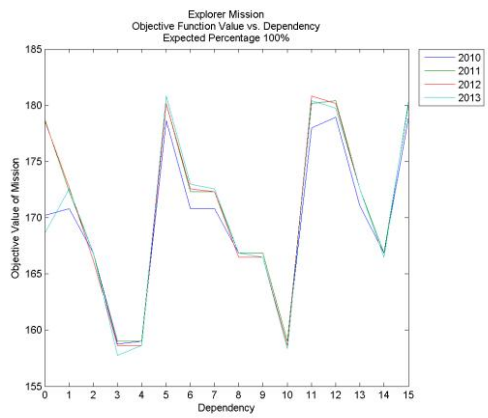


(c) Objective Value vs. Expected Input Percentage

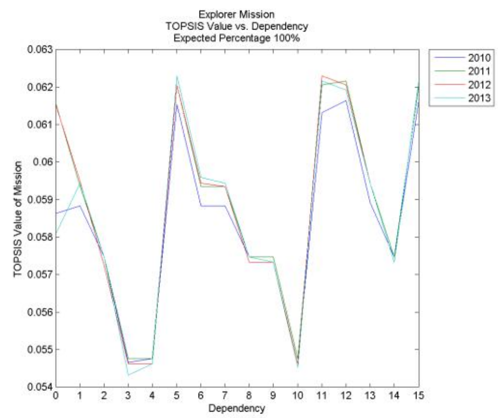


(d) TOPSIS vs. Expected Input Percentage

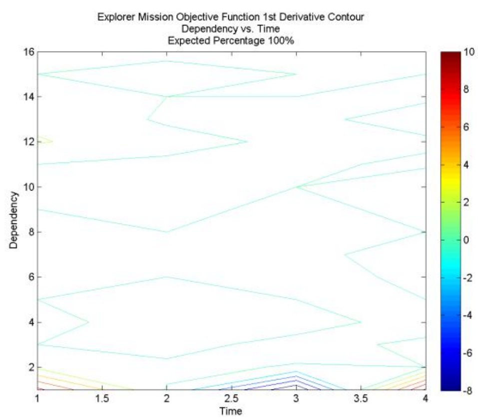
Figure 150: Explorer Mission from NEA Campaign Information



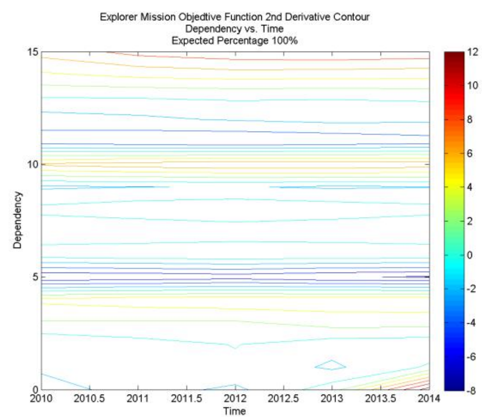
(a) Objective Value vs. Start Year



(b) TOPSIS vs. Start Year



(c) Objective Value 1st Derivative

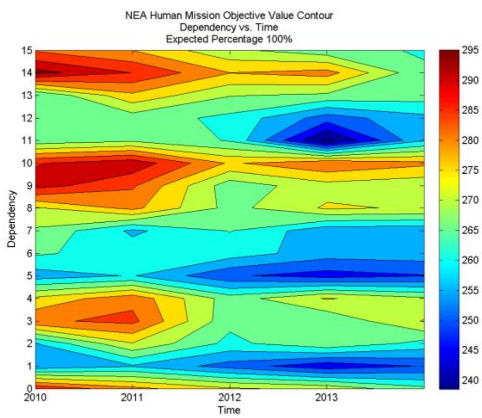


(d) Objective Value 2nd Derivative

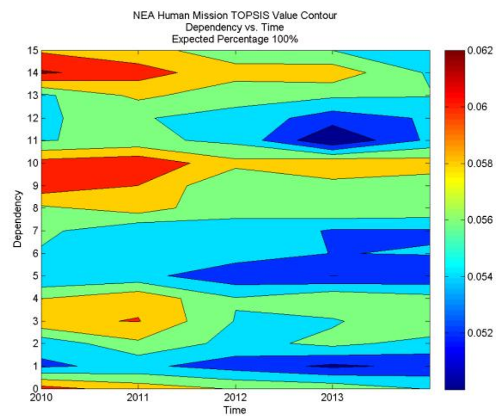
Figure 151: Explorer Mission from NEA Campaign Information(cont'd)

13.1.3 NEA Human

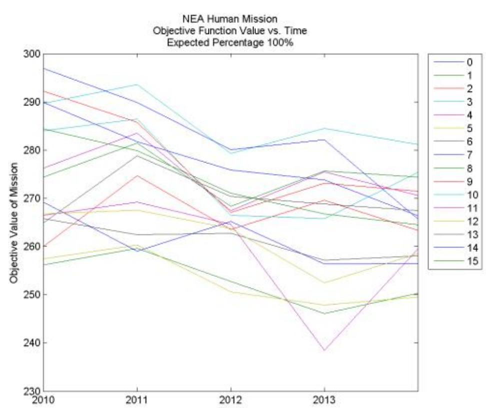
The nominal NEA Human mission selected the same technologies in the baseline for the separate mission as well as the campaign as seen in Table 75. The difference is that most of the campaign selections were funded at a higher amount except for three. The largest portfolio value possible was 295 compared to 300 in the original NEA Human Mission objective value contour given in Chapter 12. The NEA Human mission is the only mission that went down in the extreme objective value in the campaign.



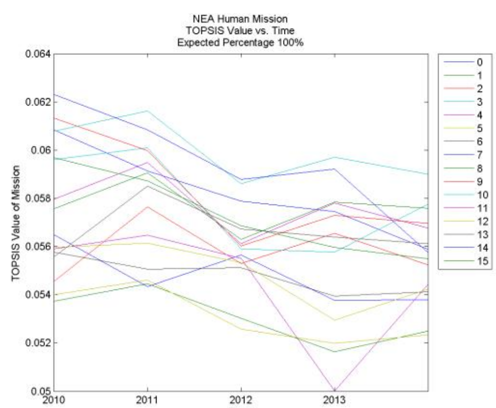
(a) Objective Value Contour



(b) TOPSIS Value Contour

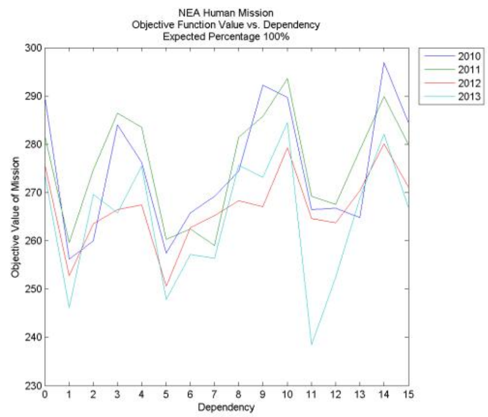


(c) Objective Value vs. Expected Input Percentage

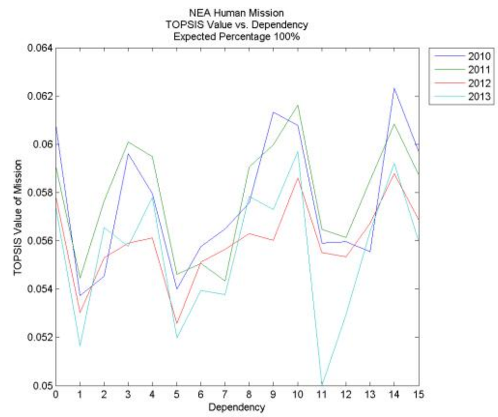


(d) TOPSIS vs. Expected Input Percentage

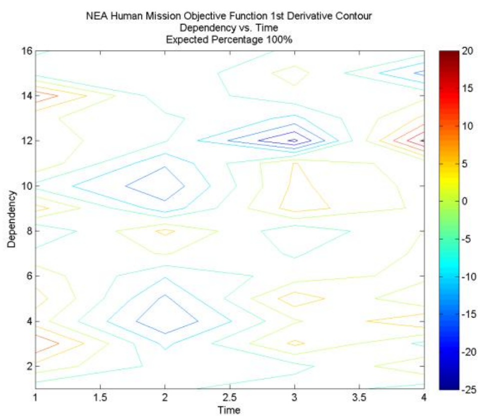
Figure 152: NEA Human Mission from NEA Campaign Information



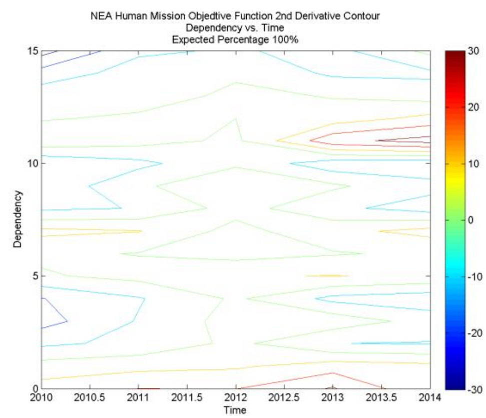
(a) Objective Value vs. Start Year



(b) TOPSIS vs. Start Year



(c) Objective Value 1st Derivative



(d) Objective Value 2nd Derivative

Figure 153: NEA Human Mission from NEA Campaign Information(cont'd)

Table 79: NEA Human Mission statistical Information

Year	min	max	Delta	Average	Variation	Max Variation	Min Variation
2010	256.1498	296.9225	40.77275	274.6467	0.148455	0.081107	0.067348
2011	259.0506	293.619	34.56841	275.8704	0.125307	0.064337	0.06097
2012	250.6564	280.0928	29.43643	266.8209	0.110323	0.049741	0.060582
2013	238.4531	284.481	46.02788	264.6245	0.173937	0.075037	0.0989
2014	249.5467	281.2018	31.65508	264.548	0.119657	0.062952	0.056706
Extremes	238.453	296.923					

13.2 NEA Campaign and MIMO

The NEA Campaign focused on the use of MIMO. MIMO changed the objective function coefficient and ultimately changed the problem being solved. The use of MIMO and other coefficient changes have different effects on the problem. Figure 154 and Figure 155 compare the use of multiple methods to change the objective function coefficient. Comparing the MIMO to the change in expected input polynomial shows that MIMO and an exponential value dominates in values below the 100% mark, but that the input value changes dominates over the 100% mark.

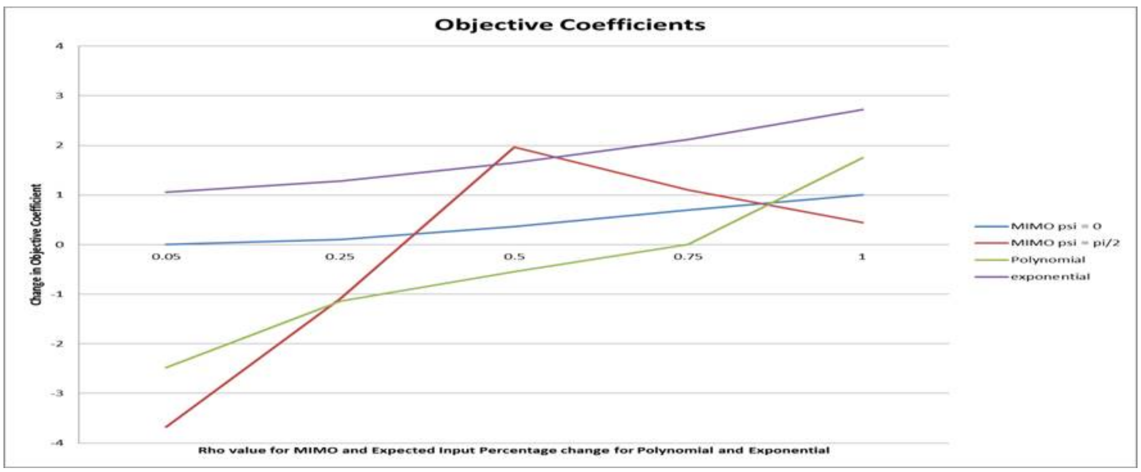


Figure 154: Change in Objective Function Coefficients

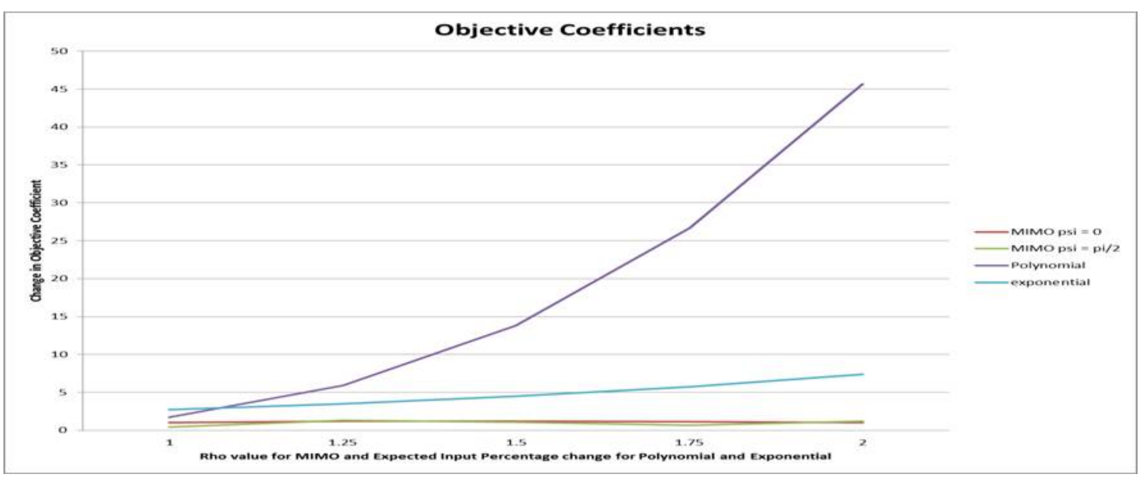


Figure 155: Change in Objective Function Coefficients (cont'd)

When it comes to the actual value of the portfolio the input values dominate this aspect. However, that does not tell the whole story. Changing the input values can not tell the user anything about the fidelity the way dependencies can. Utilizing MIMO gives the dependency change a sense of direction as well as the source of the dependency. It ties the two together. The system engineer could just as easily change the coefficient statically in the input variables as shown in Chapter 4, but this would not dynamically check the dependency source. There are multiple schemes to include dependencies into the technology portfolio process; however, MIMO takes into account more variables and ultimately adds fidelity into the process in addition to giving the user more flexibility.

13.3 NEA Campaign Conclusions

As the missions move onward from Surveyor to the NEA Human Mission, each mission has lower benefit distribution in the objective value contours. This suggests that the earlier missions can extract the most value towards the ultimate goal of deep space human capabilities due to their lower cost, earlier technology demonstration, and ultimate cost distribution compared to later missions. However, this does not suggest that they give the highest value. They simply have the highest TOPSIS value relative to other scenarios earlier on in the campaign analysis.

The campaign has dependencies to the point that the actual individual effects of the dependencies are compounded through multiple dependency interactions between missions. The user can not pinpoint the exact global effect of each dependency combined, but knows that the optimizer has dynamically chosen the correct solution.

The NEA Human Mission suggests that since it the largest mission and last to be implemented, that the technology demonstration has been done in earlier mission and at a cheaper cost. This concept of using earlier missions as a testbed fits right in the aspect of using dependencies to model a feed-forward approach and it is expected that the human mission would have a slightly lower objective value compared to its separate mission analysis.

CHAPTER XIV

RESEARCH CONCLUSIONS AND FUTURE WORK

The methodology presented in this thesis is used to model multiple real world scenarios. It sought to explain the variations in Portfolio recommendations as a function of adding dependencies. This chapter will conclude with the answers to the four research questions it sought to answer by undertaking these changes into the START methodology. It will continue to give future work associated with this research and possible avenues for sample problems to investigate.

14.1 Research Question 1: What are the dependencies associated with technology portfolio investment?

When considering the technology portfolio involved with a program there are five aspects that come into the picture with respect to the elements involved: Value, Cost, Schedule, Risk and Uncertainty. This is regardless of the type of program for which the user is trying to generate content. In order to adequately evaluate the technology element, these five aspects must come into consideration.

Introducing dependencies into this decision criterion must bring the relationships between the elements. Therefore the five aspects of evaluating a technology are the possible relationships that come into play when creating a technology portfolio investment process that has higher fidelity. This thesis looked to connect the Value of the technology elements as well as take into consideration their place within the scheme of the portfolio.

Adding fidelity into the process requires that real world examples can have Value relationships modeled with them in order to take new information into account. This thesis introduced 5 different real world scenarios that could be combined in multiple perspectives to show the levels and elements involved in a space mission as shown in Table 80.

Table 80: Dependency Types Demonstrated

Dependency
Mission A Needs Mission B
Mission A OR Mission B
Technology A Needs Technology B
Technology A OR Technology B
Technology A is enhanced by Technology B

The five dependencies demonstrated allow for integer coefficient changes as well as non-integer coefficient changes. The technologies may be connected to other technologies associated with other missions or within their own mission. Since the elements are interconnected to each other due to their relationship with their perspective mission, the non-integer coefficients must take this into account as well.

14.2 Research Question 2: What is the effect of adding dependencies on the Technology Portfolio Selection Process?

The original hypothesis was that adding dependencies would change the technology portfolio selection process. Adding dependencies as constraints will either allow the objective function to remain the same or make the objective function go lower. This thesis was proven true. Adding both Constraint and Value Dependencies changed the portfolio choices in significant ways. There are four different factors explored in this thesis that must be addressed with respect to how the portfolio objective values changes: Level, Problem Type, Funding Level, and Scenario. The Level refers to the mission or capability dependency type. The Problem type refers to Constraint

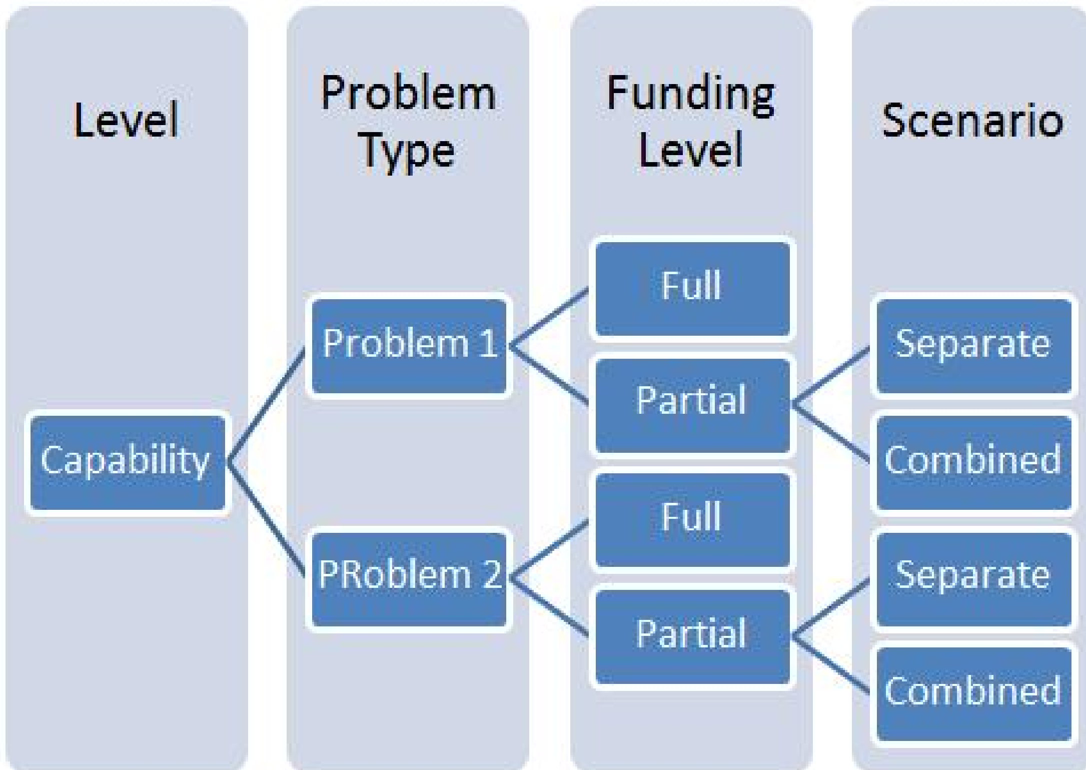


Figure 156: Dependency Portfolio Change Possibilities

or Value. The Funding Level refers to if the problem is fully funded or partially funded. The Scenario refers to the START scenario of combing the missions or running them separately. The levels and options are represented below in Figure 156 for the capability branch. The Mission branch will be addressed first and then the focus will shift to the capability branches in Figure 156.

Starting with the Dependency Level gives two observations for the Mission and Capability possibilities as presented in Table . For the Mission dependencies only Constraint dependencies are possible. Therefore, as seen in Chapter 6 in the sample problems, including a Mission dependency changes the portfolio on a larger scale. If a particular mission is excluded because it was competing with another mission, then it would give the optimal portfolio, but may be lower than the baseline which included both missions. The Mission dependency can include or exclude an entire mission which must by definition DECREASE the objective value from the baseline value. The reason is that if the baseline portfolio is truly the optimal result, then it upheld the mission dependency already or it included the new dependency. If the result was already included the baseline is the optimal. If the dependency was included, a new constraint was enabled in the solution and decreased the solution from the original value.

Focusing solely on Constraint in the capability dependencies, we saw in Chapter 9 that adding integer dependencies, such as Element A Needs Element B and Element A Or Element B are constraints in the START optimization process which translates into the portfolio cannot be larger than the original portfolio with no dependencies. The same reasoning applies from the Mission constraints. The problem is being constrained even further meaning the added dependency may be active or not active. If it is an active constraint the dependency changes the portfolio and changes the investment decisions. If it not active, then the baseline portfolio is within the constraints and the optimal portfolio has already been found. The user has not changed the data information in any way, therefore the elemental values associated with the portfolio has not changed. The portfolio has no choice, but to be the same or have a lower value.

In the case the portfolio was the same value as with no dependencies means that either the dependencies were already activated with no need to model them in the process or the elements were not chosen in the original optimized portfolio and there was no need for the relationship to be modeled. In the case of the portfolio having a lower value means that the dependency changed the decision of where to invest. The dependency prescribed a relationship that the baseline portfolio did not model and had to adhere to a lower constraint. By complying with the constraint, either the original technology chosen was chosen along with the technology it was dependent upon or the competing technology was originally chosen and was not the case of two elements being an "OR" dependency. The other option is that the original technology chosen was not chosen because it was the competition technology that did not win the competition of being most valuable to the portfolio or there was no way to include the other element it had a relationship with in the portfolio.

Knowing this information for Constraint Capability Dependencies, the user may now move on to the Funding levels of the problem. Fully funded levels means that a capability is funded all or nothing. Now the baseline case for the fully funded levels will actually be lower than that of the partially funded levels. The reason is that, by adding partial funding capabilities, START can effectively fill in the gaps of "unused" money per year with partial projects. This was seen in Chapter 9 with the use of Surveyor being fully funded and partially funded. Now the same concept applies when comparing partial funded dependency levels with the partial funded baseline. The dependencies will have a lower value compared to the baseline; although other technologies may have their funding moved upwards.

The next phase of the portfolio changes happen when the START scenario is taken into account. When each mission is done separately, they actually have a higher objective value in the case of the precursor missions saw in Chapter 11 compared with the combined case. The combined case had an objective value of 200.3 while the cases separately had a value of 262.39. This seemed counter intuitive at first, until the cost is taken into account. The combined portfolio cost 902.75 while the separate portfolios cost 1315.97 over the same amount of years. This is due to the scheduling aspect .

Going back to Figure 156 and moving to the Problem Type, the Value Dependencies has completely different effect and results on the technology portfolio. In the case of non-integer dependencies such as Element A enhances Element B using MIMO changes the portfolio in a different aspect than the integer coefficient changes. In the case of the non-integer coefficients the individual objective coefficient is changed. Increasing or decreasing these inputs are literally changing the portfolio being optimized. Problem 2 is not a pure constraint application. It changes the objective coefficient as well, and thus can cause the portfolio to be higher than that of the baseline. In this case it is up to the optimizer to determine if the portfolio is higher or lower. There is no rule for how the Problem 2 will work, because the optimizer is solving a completely different problem by changing the objective function.

There are a few commonalities to changing the actual objective value of the specific element. If the element's objective value is made higher and it was originally chosen in the portfolio with no dependencies then it will be chosen again. The reason is that the element was optimized previously and the only change is the objective function. If the value was large enough originally to be chosen, increasing it will only give cause to choose it again. Therefore the entire portfolio's value will increase, although the actual elements remained the same. In this case the user's investment decision has not been changed in making a different element choice, rather it has been reinforced by increasing the value of the already chosen elements. Now if the objective value was increase and it was not chosen originally in the portfolio then it is left up to the optimizer to decide whether or not it is valuable enough and meets the cost and time constraints to include it. Once again the optimizer is left to create a new optimized portfolio.

On the converse side if the element has been decreased in value then the model sees a different portfolio to optimize and will change the decisions accordingly. There must be some threshold where the element's objective value decreases enough that it is not chosen in the portfolio as it was initially chosen. Conversely there is some threshold value that will make an element chosen in a portfolio. None the less the process allows for different decision making strategies for the user to investigate.

Adding dependencies to the technology portfolio selection process models the relationships that are important to the user as well as allows them to evaluate the pending decisions as well. This research question shows that the inclusion of this fidelity does change the portfolio process.

14.3 Research Question 3: How does changing the investment time frame affect the technology portfolio?

This question must be broken down into two parts. The first is when should the user model the dependencies they are interested in? This question was originally asked in the aspect of should a user add the dependency in the initial year vs. subsequent years. Changing the starting year of the portfolio shortens the development time. This decreases the portfolio objective value, but increases its TOPSIS fidelity value. The reason is that changing the start year changes the development time frame associated with the project, which is essentially changing the constraints on the project. Adding constraints to an optimization problem only allows the objective function to stay the same or go lower.

This comes into play when TOPSIS is applied and eventually used as a decision making tool. If the start year is important to the user than TOPSIS takes into account the relative range time period of the portfolio. TOPSIS has a larger impact of changing the portfolio start year than the actual optimizer does when looking at the start year with respect to dependencies. This can be in Chapter 10 throughout the Explorer data information for the specific start year. It shows that the portfolio value is almost constant when adding dependencies for a given expected percentage input value.

The second question that comes into play with respect to changing the timing of dependencies is a more iterative approach to changing the time. This was going to be a field of study, but it is reserved for a future work. The reason is that if a portfolio is optimized for the first year and the elements are frozen, and then re-optimized for the second year with the remaining elements, that is not an optimized portfolio anymore. This type of optimization is not within the bounds of START and would require an entirely new approach to portfolio selection. This is suggested as another time to be investigated. The methodology would require a new approach to turning dependencies and fidelity on and off. Whether this is a good approach or not is not known at this time.

14.4 Research Question 4: Which has a larger impact on the Portfolio Selection: Input values vs. Dependencies?

Dependencies provide fidelity into the technology portfolio selection process. The dependencies operate under the assumption that the user is making decisions based upon the accurate estimated input values. These estimations come from industry and expert advice as to what the ultimate value, cost, and schedule of an element. This last research question addresses the fact that the input values while researched are still estimations. It looks at what happens when those estimates are wrong.

The hypothesis presented in Chapter 2, was that dependencies are dominant to the technology portfolio selection and analysis. This research question looked at this hypothesis and tested it multiple times; specifically in the MIMO cases. This is looked at specifically in Chapter 13 when the MIMO vs. input values are investigated. This hypothesis is conditional. MIMO dominate in a few cases since there can be a drastic change in coefficient value; however, in general the user input has a much greater effect on the portfolios.

Within the START framework, the expected utility value is linear change with respect to the input values associated with the value aspect of the element; however, it is polynomial change with respect to the objective value. These two values are important aspects of the START optimization process, but represent two different aspects of the portfolio. This can be seen in Chapter 12. Changing the expected input percentage follows a polynomial change. This was shown in Chapter 12 when the MIMO changes were graphed against the polynomial changes of the expected input percentages.

Adding non-integer dependencies utilizing the MIMO methodology is a equation. There is no rule as to which is always a faster convergence. There may not be a physical relationship that an element has on another element so there is no dependency to model. However, a technologist can make aggressive goals that will make the technology more valuable. The user may even change their input values to see where the element aspects lie in the portfolio. This follows along with the Bottom-Up and Top-Down approach.

This research question really revealed the sensitivity of the input values which are subjective to expert opinion and experience. Users are creating methodologies in order to quantify their decisions on subjective material. What happens if they are wrong? What is the magnitude to which they are changing the optimization process? The real answer depends on the process.

START utilizes a linear branch and bound solution because the software is commercially available and deals with as many as 10,000 elements at one time. These are all excellent points to create a methodology that will quantify decisions. However, there is a chance that the real optimizer is non-linear requiring more computing power and non-converging algorithms.

Utilizing START and its linear properties give a beginning to quantifying these types of decisions. The sensitivity of the change of the input values will change with the algorithm utilized to optimize the system. These can be non-linear and will thus have a larger error.

Adding Constraint Dependencies do not affect the sensitivity of the portfolio's objective value because it is modeling relationships that affect individual elements. Adding Value Dependencies do affect the sensitivity of the portfolio objective value because it is changing the value in the same way as the MIMO equation.

Therefore the answer is that it depends upon the optimizers algorithm, the elements' value thresholds, and the users required objective value. In the case of START the input values had a larger effect on the portfolio's value, but the dependencies had a larger effect on the portfolio elemental selection. Changing all of the elements expected input percentage would give the same portfolio with a larger value. Changing just one dependency MIMO or otherwise would change the entire portfolio. Looking at the user decision making prospects, the input values are important when comparing the portfolio to the baseline, but the dependency changes are more important when looking only at fidelity.

14.5 Future Work

This thesis went through an entire methodology, assumed certain scenarios, and ultimately gave a scope of where this methodology is useful. However, there were a few areas of inquiry that it did not investigate. If this work were to continue the following six areas could be investigated for a larger scope.

14.5.1 Dependencies between Cost, Schedule, Risk and Uncertainty

There were five areas that defined an element in a technology portfolio: Value, Cost, Schedule, Risk and Uncertainty. This thesis only looked at Value due to scoping the problem to a reasonable one within a graduate students' time frame. The other four elemental definitions are equally as important and could very well include dependencies between them. A few examples are given below.

- Technology A utilizes the same testbed as Technology B so there is cost sharing between the two
- Technology A must be developed after Technology B is developed
- Technology A must be developed in conjunction with Technology B

- If Technology A is developed and fails, then Technology B is a greater risk and has a lower objective function
- There is a higher uncertainty probably distribution for Technology A that is dependent on Technology B's distribution thresholds

Some of the examples given are not linear relationships and thus cannot be modeled in START. Some of the possibilities may be completely irrelevant cases. However, there is research to be done to further the understanding of relationships between technology portfolio elements.

14.5.2 Grouping Scenarios for Problem 2

Originally I planned to program both new groups as well as separate MIMO entities into START. I have since realized that the problem with grouping is how to redefine two technologies together. This was briefly stated in Chapter 6, it goes back to the assumption about element C is composed of element A and element B - does $C = A+B$, $C < A + B$ or $C > A + B$? For this thesis the assumption was that $C = A+B$. However, it is possible to have the other two scenarios. The question is

- Where does this occur?
- How does it affect the investment decisions?
- Is it really possible to model that with a Constraint scenario where it's just a simple AND case?

As discussed in Chapter 6 any one of the assumptions loses fidelity in the START model because it cannot dynamically optimize the starting time of the technologies in these scenarios. While it was not appropriate for this methodology, there may be times where it is appropriate to look at how to combine technologies in an optimization in put file. This would be an interesting process to show the possibility of getting higher fidelity into the input process.

Table 81: Element A and Element B must be allowed logic table

A	B	1	2	3	4	5	7	8
		Currently	A Needs B	B Needs A	A AND B	A OR B	Not A	Not B
1	1	Y	Y	Y	Y	Y	Y	Y
1	0	Y	N	Y	N	Y	N	Y
0	1	Y	Y	N	N	Y	Y	N
0	0	Y	Y	Y	Y	N	N	N

14.5.3 Non-linear solutions for Problem 2

START is a linear branch and bound program. This was a perfect platform to show the changes in investment decisions by adding dependencies; however, it brings into question how the process would change if a non-linear program had been demonstrated instead. This thesis did not look at non-linear programming models; however, a future investigation may be to make a non-linear optimization the basis of their work to see how the higher computing power of non-linear model affects the decision making process. It is not normally utilized due to the complications of getting the optimizer to converge, but if a user could get beyond the convergence issue it would be interesting to see how the process changes with respect to higher levels of dependencies.

14.5.4 Scenarios in logic gates where you MUST choose a solution

As studied in Chapter 6, the logic gate scenarios used gave the solution that neither element may be chosen. However, there is the entire lineage of logic gates that states that element A and B must be chosen. This would amount to the following logic gate.

Table 81 is the opposite logic gate of Table 8 given in Chapter 6. This logic gate was not used, because the problem was not trying to make sure both element are allowed to be chosen at the same time, the problem was to make sure that if neither element was not chosen, the system would accept this as a solution. The questions associated with this process are: Is this a relevant logic gate? Where would these types of scenarios exist? Is this applicable? Also discussed in Chapter 6 is the concept

Table 82: Example 1: A, B, C are competing (A or B or C or neither). Can be implemented using (not both A and B) and (not both A and C) and (not both B and C). So in this example the 3-element dependency can be implemented using 2-way dependencies.

A	B	C	A needs (B&C)	A needs B	A needs C
0	0	0	Y	Y	Y
0	0	1	Y	Y	Y
0	1	0	Y	Y	Y
0	1	1	Y	Y	Y
1	0	0	N	N	N
1	0	1	N	N	Y
1	1	0	N	Y	N
1	1	1	Y	Y	Y

of reserved, enabling and enhancing. The logic is that dependencies may only work on enhancing elements since reserved and enabling require that the associated elements are funded for either the portfolio and/or the mission. START only utilizes enabling and enhancing in the current version, but it is possible to enable the reserved quality within the process. However, once again the question is this possible or necessary?

14.5.5 2-element interactions to N-Element Interactions

This thesis demonstrated 2-element interactions. The next logical step is to investigate N-element interactions. The interactions are not additive functions. 2-element interactions are powerful and can model some of the N-element interactions. However, there is not a general formula to model these, and it must be investigated further to find out how to use this information. Just to start here are a few examples.

These scenarios were not modeled, but it is clear that it is possible to model some of the 3-element interactions. The procedure is not an additive process, but it would be an extremely powerful tool to have N-element interactions added to a technology

Table 83: Example 2: All A, B, and C are needed. Can be implemented using (both A and B or neither) and (both B and C or neither). So in this example the 3-element dependency can be implemented using 2-way dependencies. Note only 2 of the 2-way dependencies are n

A	B	C	A, B, C are competing alternatives	not both A and B	not both A and C	not both B and C
0	0	0	Y	Y	Y	Y
0	0	1	Y	Y	Y	Y
0	1	0	Y	Y	Y	Y
0	1	1	N	Y	Y	N
1	0	0	Y	Y	Y	Y
1	0	1	N	Y	N	Y
1	1	0	N	N	Y	Y
1	1	1	N	N	N	N

Table 84: Example 3: A needs (B and C). The best way to look at this is using the allowed table: see attached Excel.

A	B	C	all A,B,C or none	both A and B or neither	both B and C or neither
0	0	0	Y	Y	Y
0	0	1	N	Y	N
0	1	0	N	N	N
0	1	1	N	N	Y
1	0	0	N	N	Y
1	0	1	N	N	N
1	1	0	N	Y	N
1	1	1	Y	Y	Y

Table 85: Example 4: A needs (B or C). I do not think this can be done using 2-element capability dependencies. It can be done more complicated by using capability and mission dependencies together. If mission 1 is made that has capabilities B or C. And mission

B	C	A needs (B or C)	Doesn't seem possible to do with 2-element constraints
0	0	Y	Y
0	1	Y	Y
1	0	Y	Y
1	1	Y	Y
0	0	N	N
0	1	Y	N
1	0	Y	N
1	1	Y	Y

selection process. This would increase the value of the fidelity term modeled and ultimately give the user greater flexibility in modeling case scenarios.

14.5.6 Other Problem 2 connections other than MIMO

I utilized the multiple-input-multiple-output method for changing value dependencies. There may be other possibilities to change other aspects of an element. These include the time, cost, risk and uncertainty aspects. In order to narrow the scope, the thesis only focused on value dependencies, which is modeled as the expected utility function. There may be other possibilities to model this. These could be linear, polynomial, exponential, or trigonometric. The point is that there are other possibilities out there to change the elemental aspects. An investigation into other types of theories would expand the design space.

14.5.7 When does a problem change its N value in scalability?

Scalability was brought up in Chapter 6. This concept dealt with the fact that there are different scales of problems dealing with the phase of the project. This may be an expert opinion problem and if so a historic analysis that could show when a problem changes its N-value. The N-value should be associated with the phases of the project since the more information that is known the more elements that must be optimized within the process.

14.5.8 Is there a new approach to technology portfolio selection by optimizing the temporally the elements and then progressing each year instead of fitting building blocks into a set time period with set dependencies?

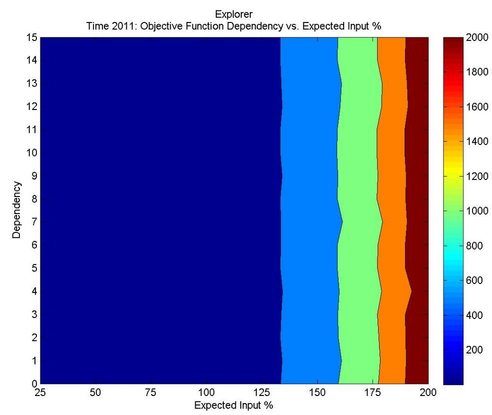
This was a spinoff of research question 3 that asked when in the process should dependencies be added to the process. The current process assumes that the dependencies are active throughout the entire constraint process and thus applies them in the beginning. Taking an iterative approach where dependencies are turned on and off as the optimizer steps through the time space would be a new avenue to create a different optimizing methodology.

14.6 Conclusion

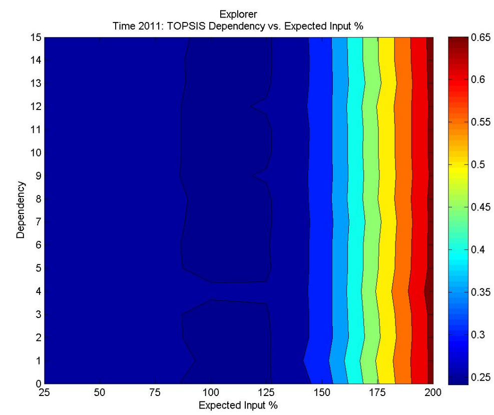
Dependency inclusion in the technology portfolio selection process opens the door to multiple scenarios that the user may investigate. This fidelity into the process by creating relationships within the optimization process that was previously not included. This is a possible without making the process non-linear and loosing flexibility with respect to when technologies are funded. The end result is an optimization process that quantifies the users' decisions with a higher level of fidelity that can now be applied to feed-forward analysis and processes across multiple industries. Adding fidelity to the technology selection process is a process that has great potential for complex engineering problems. Research in this field will only allow for bigger and better solutions to user needed complexity.

APPENDIX A

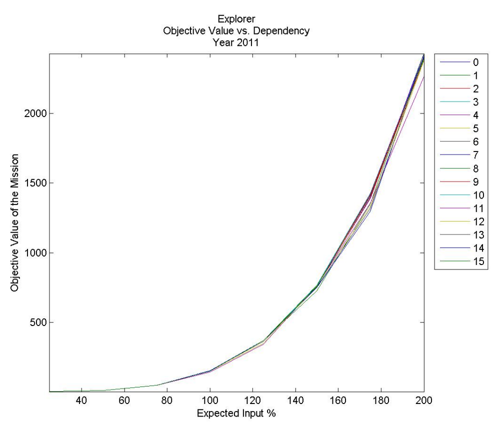
EXPLORER DATA CUBE



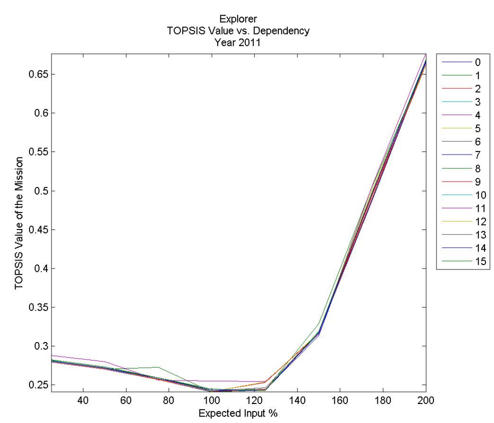
(a) Objective Value Contour



(b) TOPSIS Value Contour

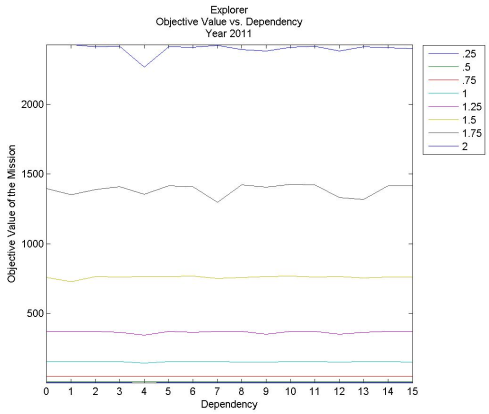


(c) Objective Value vs. Expected Input Percentage

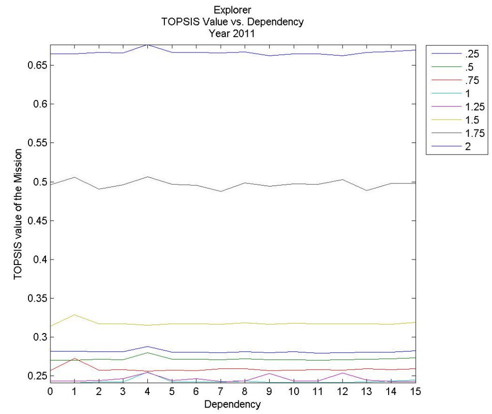


(d) TOPSIS vs. Expected Input Percentage

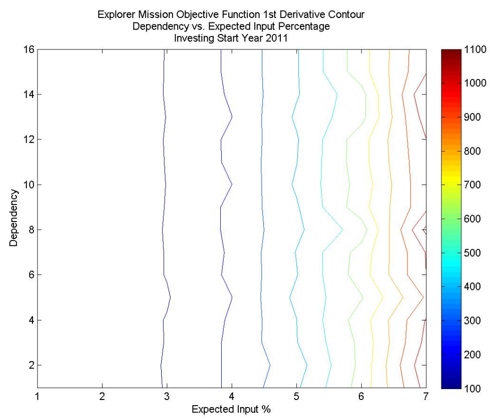
Figure 157: Explorer Start Year of 2011



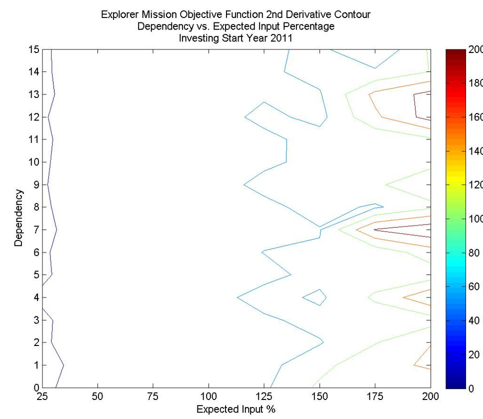
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour

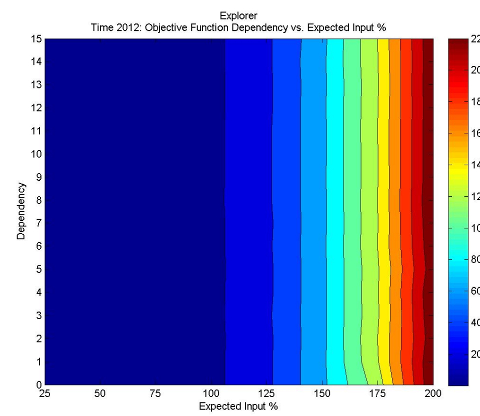


(d) 2nd Derivative of Objective Value Contour

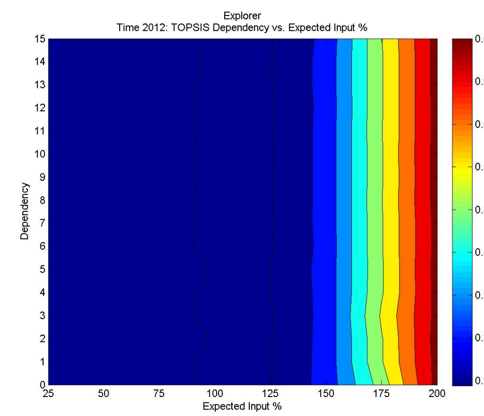
Figure 158: Explorer Start Year of 2011 (con't)

Table 86: 2011 Statistics Information for various Expected Input Percentages

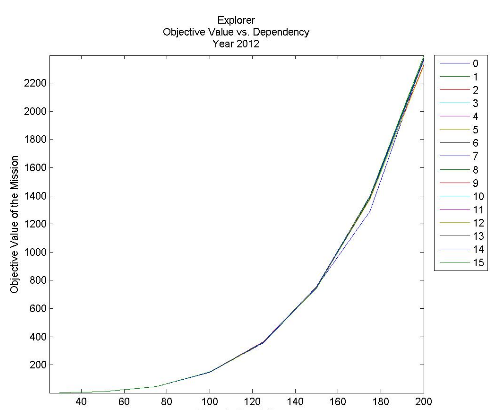
Expected Input Percentages	min	max	Delta	Average	Variation	Max Variation	Min Variation
25%	0.553932	0.592001	0.038068	0.583549	0.065236	0.014483	0.050753
50%	8.862913	9.497323	0.63441	9.385704	0.067593	0.011892	0.055701
75%	45.3147	47.97492	2.660218	47.44341	0.056071	0.011203	0.044868
100%	141.8066	151.9572	10.15055	150.1713	0.067593	0.011892	0.055701
125%	341.0397	370.9892	29.9495	363.9132	0.082298	0.019444	0.062854
150%	725.0352	767.5987	42.56349	759.0946	0.056071	0.011203	0.044868
175%	1298.563	1427.738	129.1755	1386.684	0.093154	0.029606	0.063548
200%	2268.906	2431.315	162.4089	2402.74	0.067593	0.011892	0.055701



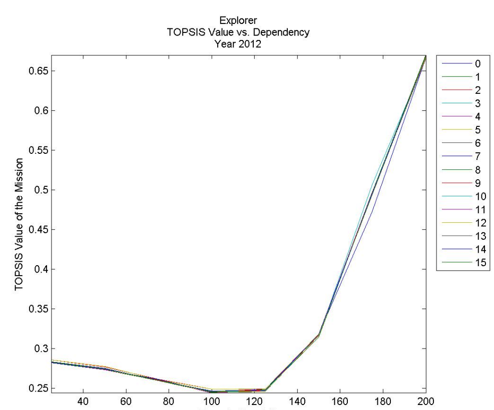
(a) Objective Value Contour



(b) TOPSIS Value Contour



(c) Objective Value vs. Expected Input Percentage

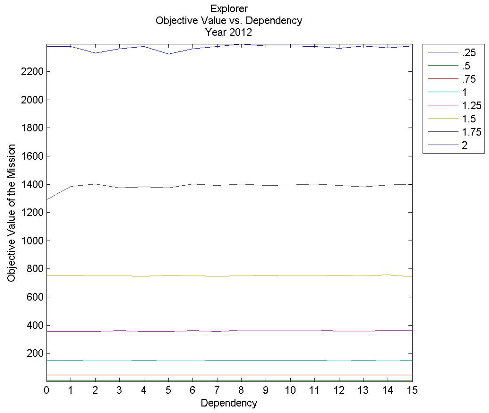


(d) TOPSIS vs. Expected Input Percentage

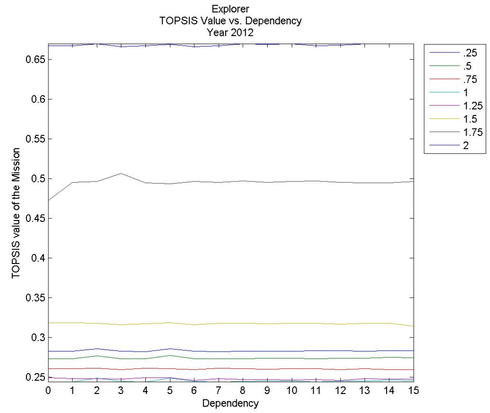
Figure 159: Explorer Start Year of 2012

Table 87: 2011 Statistics Information for various Expected Input Percentages

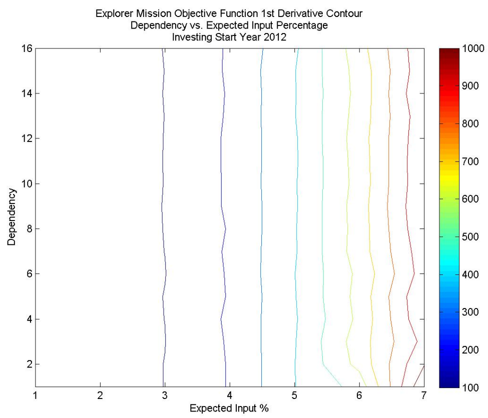
Dependency	X^3	X^2	X	b
0	0.000722	-0.11609	7.032122	-124.431
1	0.000792	-0.13902	9.004694	-166.388
2	0.000702	-0.11048	6.604489	-116.062
3	0.000692	-0.10628	6.167472	-105.552
4	0.00057	-0.07423	3.657631	-54.7466
5	0.00068	-0.10261	5.885742	-100.118
6	0.000682	-0.10357	5.962952	-101.622
7	0.000797	-0.14269	9.477026	-178.823
8	0.000658	-0.09627	5.364244	-89.2631
9	0.000661	-0.09761	5.422909	-89.3897
10	0.000662	-0.09711	5.408821	-89.874
11	0.000682	-0.1032	5.921603	-100.694
12	0.000721	-0.1186	7.3325 -1	31.618
13	0.000772	-0.13428	8.705125	-161.531
14	0.000673	-0.10086	5.744242	-97.1928
15	0.000667	-0.099	5.594241	-94.1189



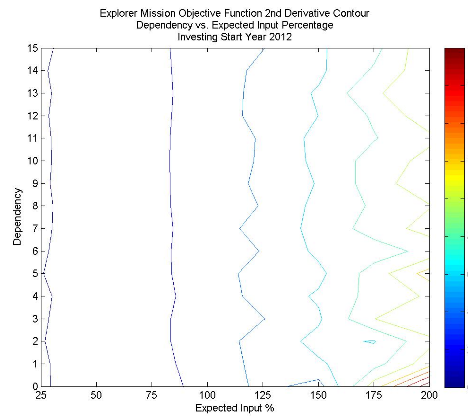
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour



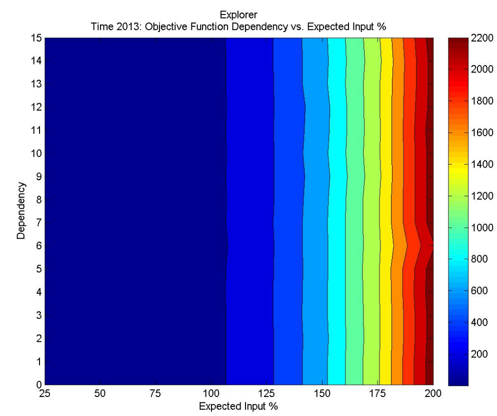
(d) 2nd Derivative of Objective Value Contour

Figure 160: Explorer Start Year of 2012 (con't)

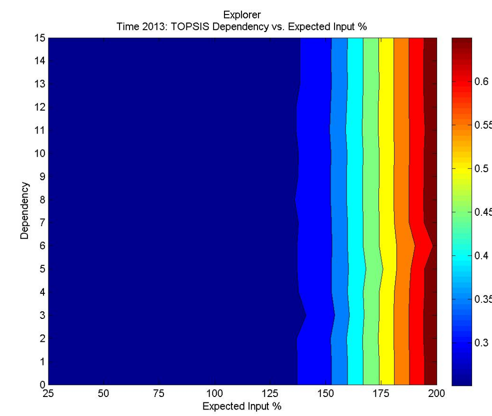
Table 88: 2012 Statistics Information for various Expected Input Percentages

Expected Input Percentages	min	max	Delta	Average	Variation	Max Variation	Min Variation
25%	0.567893	0.58182	0.013927	0.578004	0.024095	0.006601	0.017494
50%	9.086285	9.365454	0.279169	9.263792	0.030135	0.010974	0.019161
75%	46.45764	47.26421	0.806571	46.93436	0.017185	0.007028	0.010157
100%	145.3806	149.8473	4.466703	148.2037	0.030139	0.01109	0.019049
125%	354.933	364.3001	9.367144	359.2895	0.026071	0.013946	0.012125
150%	743.3222	756.2273	12.90514	750.9497	0.017185	0.007028	0.010157
175%	1290.567	1401.004	110.4379	1385.382	0.079717	0.011277	0.06844
200%	2326.089	2397.556	71.46724	2371.531	0.030135	0.010974	0.019161

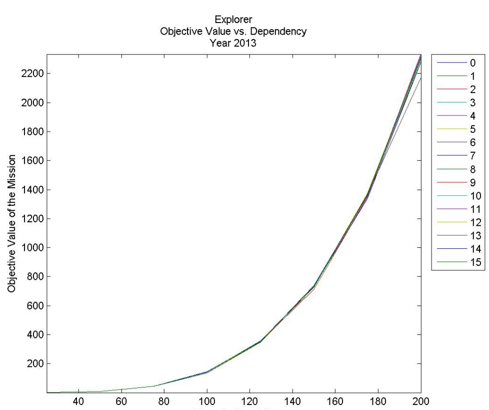
?



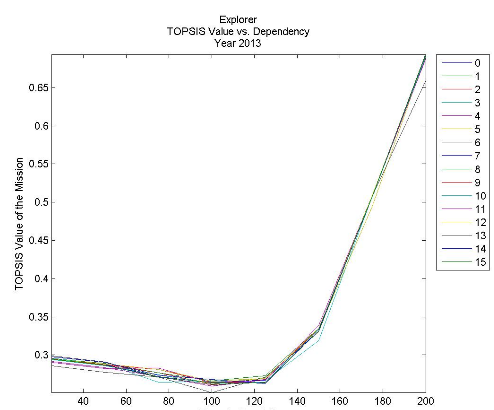
(a) Objective Value Contour



(b) TOPSIS Value Contour



(c) Objective Value vs. Expected Input Percentage

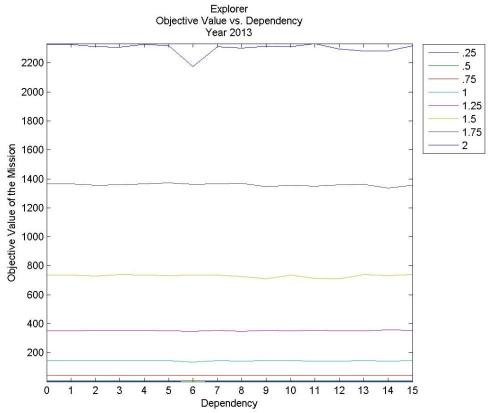


(d) TOPSIS vs. Expected Input Percentage

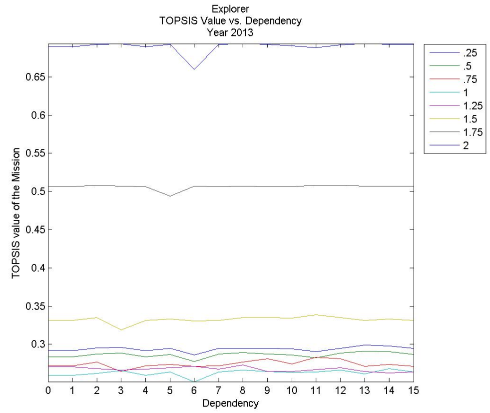
Figure 161: Explorer Start Year of 2013

Table 89: 2012 Statistics Information for various Expected Input Percentages

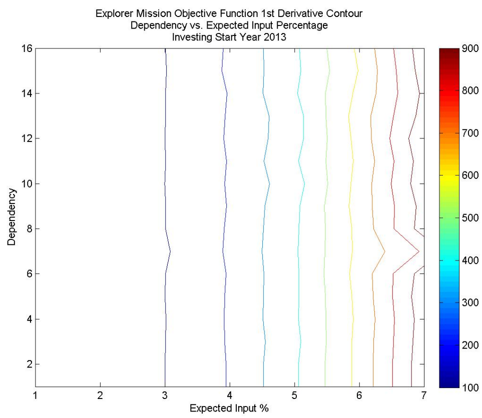
Dependency	X^3	X^2	X	b
0	0.000758	-0.13166	8.541669	-158.693
1	0.000683	-0.10522	6.125807	-105.111
2	0.000614	-0.08512	4.456608	-70.1739
3	0.000666	-0.10132	5.880132	-101.194
4	0.000689	-0.10729	6.292878	-108.489
5	0.000626	-0.0896	4.894005	-80.2377
6	0.000646	-0.09407	5.208003	-86.1316
7	0.00068	-0.10429	6.022049	-102.532
8	0.000687	-0.10593	6.170757	-105.998
9	0.000672	-0.10201	5.879806	-100.408
10	0.000675	-0.10289	5.949924	-101.825
11	0.000667	-0.10028	5.716804	-96.7198
12	0.000657	-0.0975	5.494203	-92.0871
13	0.000688	-0.1069	6.270791	-108.235
14	0.000654	-0.09662	5.431309	-90.9921
15	0.00068	-0.10388	5.990645	-101.975



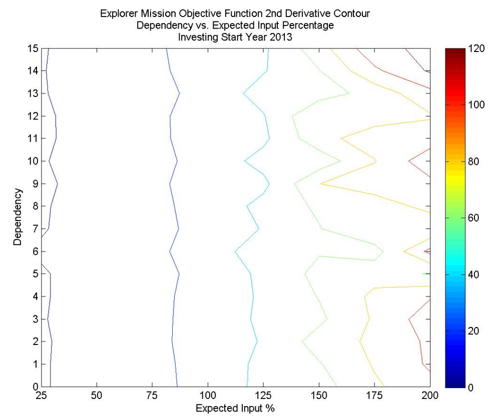
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour



(d) 2nd Derivative of Objective Value Contour

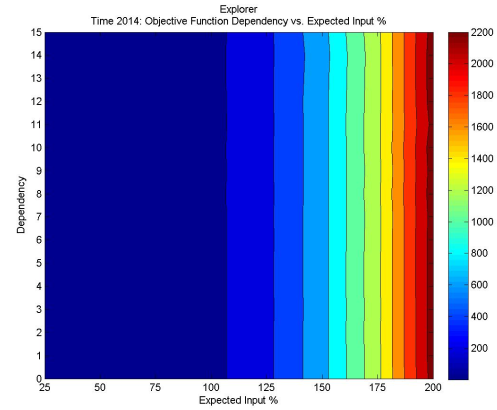
Figure 162: Explorer Start Year of 2013 (con't)

Table 90: 2013 Statistics Information for various Expected Input Percentages

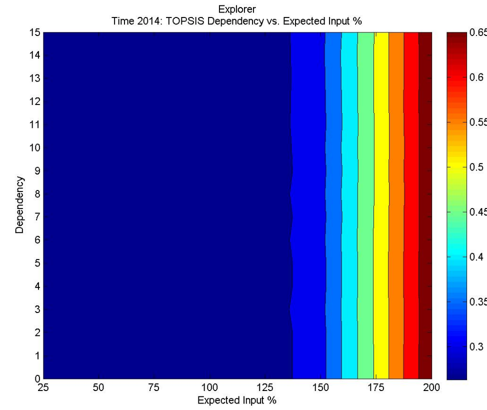
Expected Input Percentages	min	max	Delta	Average	Variation	Max Variation	Min Variation
25%	0.531365	0.569648	0.038284	0.562898	0.068012	0.011991	0.05602
50%	8.501834	9.114373	0.612539	8.998121	0.068074	0.01292	0.055154
75%	44.4128	46.16089	1.748086	45.67952	0.038268	0.010538	0.027731
100%	136.0293	145.5585	9.52912	144.0235	0.066164	0.010658	0.055506
125%	349.0508	356.8426	7.791709	352.2273	0.022121	0.013103	0.009018
150%	710.6049	738.5742	27.96938	730.8724	0.038268	0.010538	0.027731
175%	1337.767	1371.724	33.95691	1359.06	0.024986	0.009318	0.015667
200%	2176.47	2333.28	156.81	2303.519	0.068074	0.01292	0.055154

Table 91: 2014 Statistics Information for various Expected Input Percentages

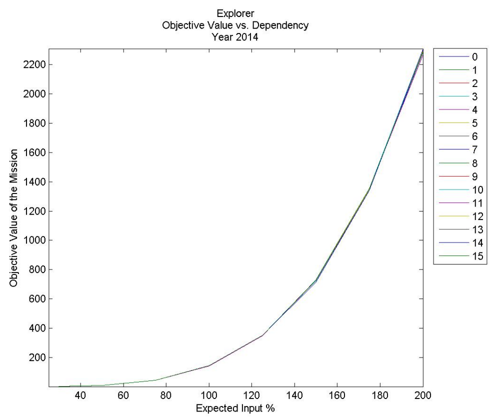
Dependency	X^3	X^2	X	b
0	0.000657	-0.09935	5.689892	-96.4975
1	0.000657	-0.09935	5.689892	-96.4975
2	0.000652	-0.09871	5.670498	-96.5149
3	0.000634	-0.0933	5.233199	-87.6355
4	0.000656	-0.09909	5.680175	-96.4975
5	0.000643	-0.09503	5.318111	-88.4936
6	0.000484	-0.05048	1.767748	-16.2987
7	0.000637	-0.09378	5.253318	-87.7733
8	0.000636	-0.09365	5.209631	-86.2223
9	0.000683	-0.10861	6.494597	-113.64
10	0.000647	-0.09692	5.511826	-93.0413
11	0.000696	-0.11172	6.716054	-117.828
12	0.000651	-0.09871	5.645818	-95.4639
13	0.000606	-0.08522	4.581233	-74.1504
14	0.000627	-0.09265	5.262927	-89.4376
15	0.000647	-0.09712	5.558745	-94.6082



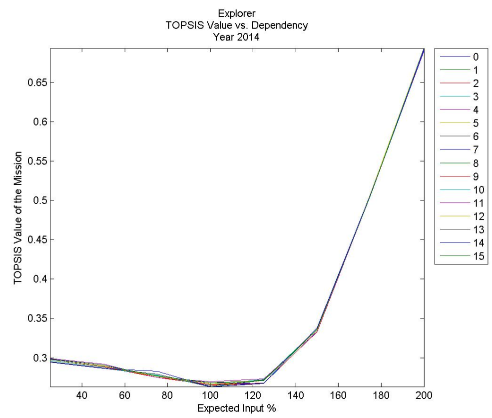
(a) Objective Value Contour



(b) TOPSIS Value Contour



(c) Objective Value vs. Expected Input Percentage

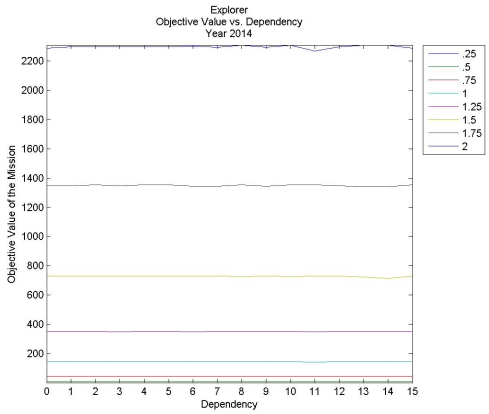


(d) TOPSIS vs. Expected Input Percentage

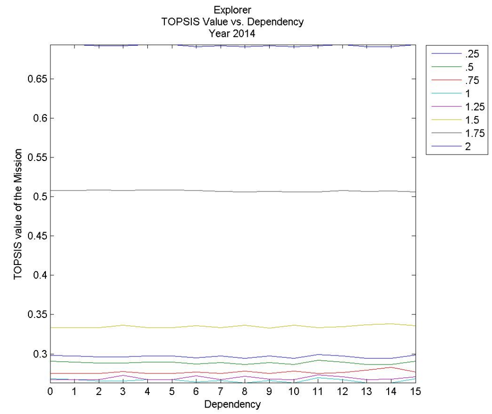
Figure 163: Explorer Start Year of 2014

Table 92: 2014 Statistics Information for various Expected Input Percentages

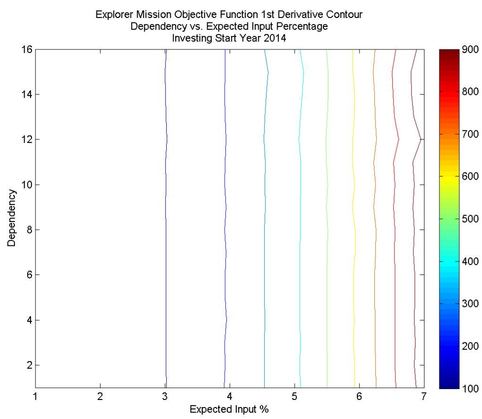
Expected Input Percentages	min	max	Delta	Average	Variation	Max Variation	Min Variation
25%	0.553511	0.563767	0.010256	0.560987	0.018282	0.004956	0.013326
50%	8.856172	9.020269	0.164097	8.974208	0.018285	0.005133	0.013153
75%	44.65444	45.66511	1.010675	45.51892	0.022203	0.003212	0.018992
100%	141.6988	144.3243	2.625548	143.5873	0.018285	0.005133	0.013153
125%	347.7454	352.3543	4.608822	351.0707	0.013128	0.003656	0.009472
150%	714.471	730.6418	16.17081	728.3027	0.022203	0.003212	0.018992
175%	1340.314	1353.604	13.2905	1348.562	0.009855	0.003739	0.006117
200%	2267.18	2309.189	42.00877	2297.397	0.018285	0.005133	0.013153



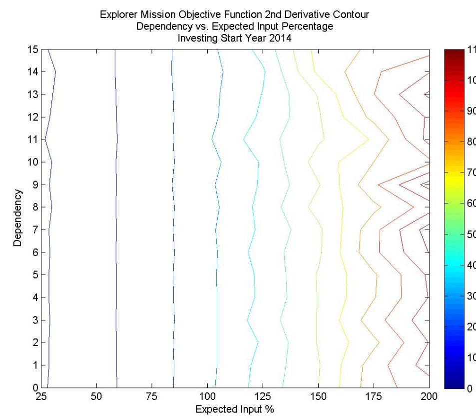
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour



(d) 2nd Derivative of Objective Value Contour

Figure 164: Explorer Start Year of 2014 (con't)

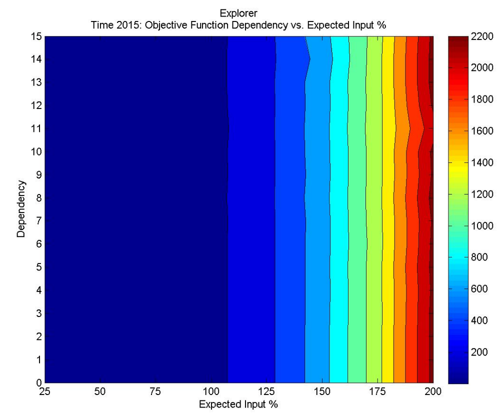
?

Table 93: 2015 Statistics Information for various Expected Input Percentages

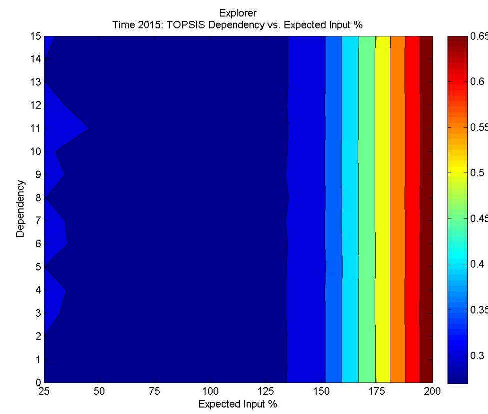
Dependency	X^3	X^2	X	b
0	0.000628	-0.09239	5.188988	-87.0561
1	0.000643	-0.09642	5.507049	-93.4571
2	0.000636	-0.09435	5.329065	-89.6657
3	0.000644	-0.09693	5.535888	-93.8054
4	0.000638	-0.09479	5.359743	-90.2188
5	0.000638	-0.09479	5.359743	-90.2188
6	0.00065	-0.09865	5.685066	-97.0456
7	0.000642	-0.09633	5.510936	-93.6974
8	0.000654	-0.09952	5.731301	-97.6425
9	0.000643	-0.09671	5.542506	-94.349
10	0.000654	-0.0994	5.726845	-97.6425
11	0.000603	-0.08477	4.534702	-72.9841
12	0.000644	-0.09684	5.54158	-94.1689
13	0.000668	-0.10416	6.153601	-106.989
14	0.000677	-0.10702	6.374699	-111.305
15	0.000625	-0.09125	5.069319	-84.1728

Table 94: 2015 Statistics Information for various Expected Input Percentages

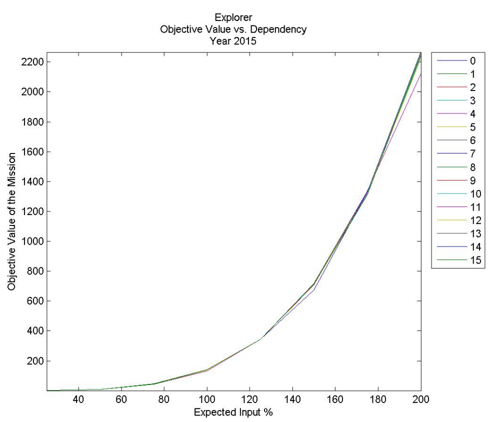
Expected Input Percentages	min	max	Delta	Average	Variation	Max Variation	Min Variation
25%	0.519177	0.553228	0.034052	0.547821	0.062158	0.009872	0.052287
50%	8.30683	8.851654	0.544824	8.765783	0.062153	0.009796	0.052357
75%	42.05333	44.93965	2.886328	44.48393	0.064885	0.010245	0.05464
100%	132.9093	141.6265	8.717185	140.2525	0.062153	0.009796	0.052357
125%	343.1742	345.3749	2.200742	344.4287	0.00639	0.002747	0.003642
150%	672.8532	719.0345	46.18124	711.7429	0.064885	0.010245	0.05464
175%	1311.033	1335.221	24.18753	1325.309	0.01825	0.007479	0.010772
200%	2126.548	2266.023	139.475	2244.041	0.062153	0.009796	0.052357



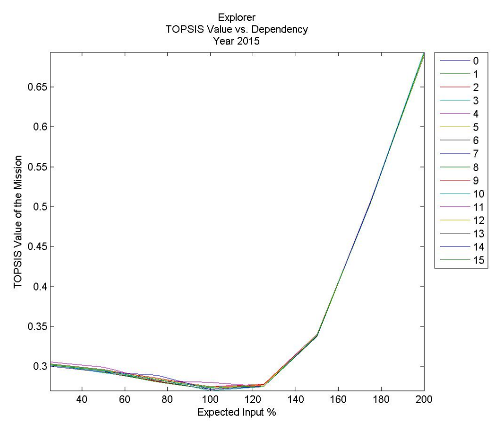
(a) Objective Value Contour



(b) TOPSIS Value Contour

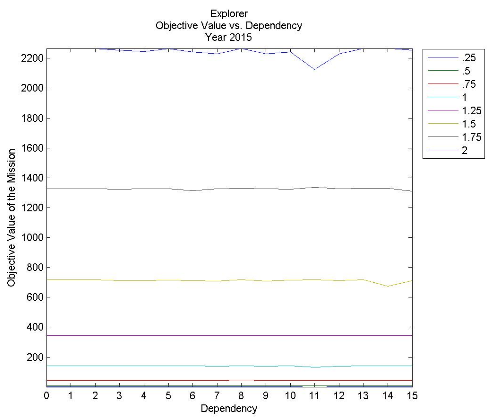


(c) Objective Value vs. Expected Input Percentage

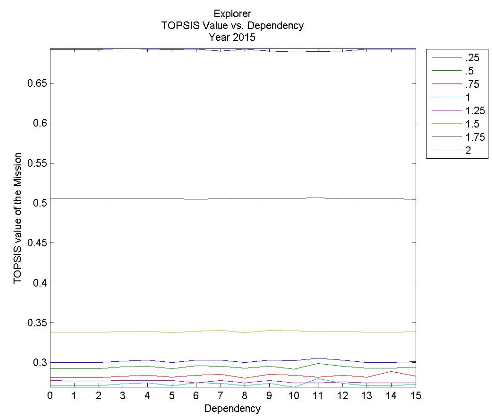


(d) TOPSIS vs. Expected Input Percentage

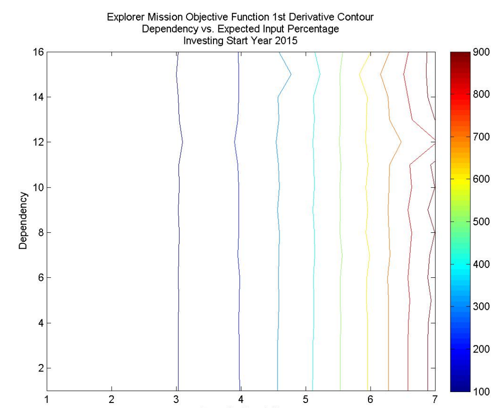
Figure 165: Explorer Start Year of 2015



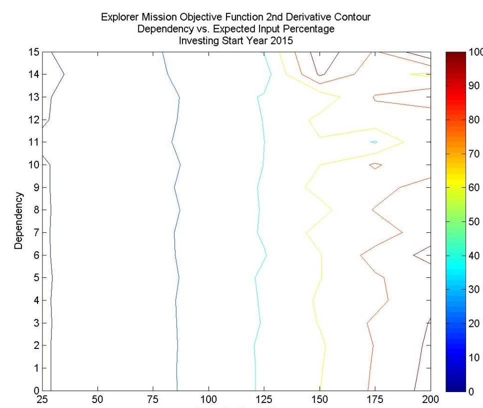
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour



(d) 2nd Derivative of Objective Value Contour

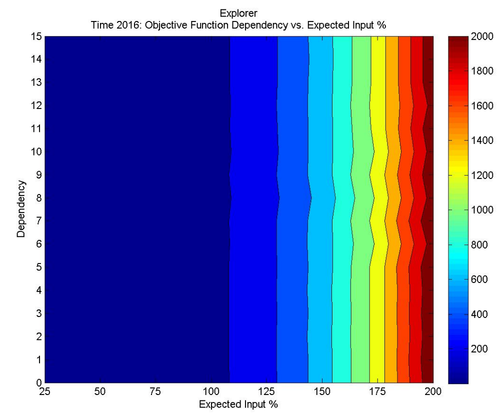
Figure 166: Explorer Start Year of 2015 (con't)

Table 95: 2016 Statistics Information for various Expected Input Percentages

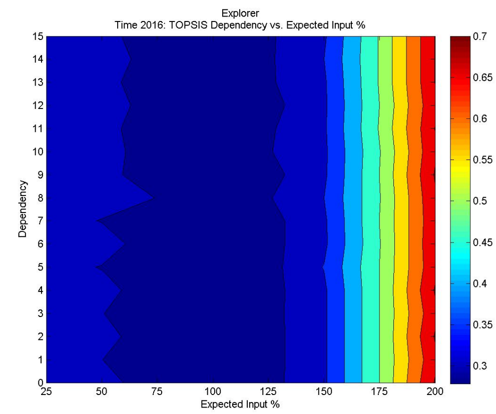
Dependency	X^3	X^2	X	b
0	0.000636	-0.09581	5.478248	-92.89
1	0.000635	-0.0957	5.473882	-92.89
2	0.000635	-0.09572	5.474779	-92.89
3	0.000635	-0.09605	5.514006	-93.8475
4	0.00062	-0.09163	5.140896	-85.914
5	0.00064	-0.09711	5.58137 -	94.9464
6	0.000627	-0.09409	5.388176	-91.7303
7	0.000604	-0.08718	4.782604	-78.5597
8	0.000635	-0.0956	5.468124	-92.8289
9	0.000604	-0.08718	4.782604	-78.5597
10	0.000619	-0.09149	5.153985	-86.5964
11	0.000469	-0.04811	1.633669	-14.1351
12	0.000601	-0.08618	4.711448	-77.297
13	0.000638	-0.09628	5.521373	-93.8908
14	0.000684	-0.11073	6.667387	-116.744
15	0.000642	-0.09848	5.750566	-99.2985

Table 96: 2016 Statistics Information for various Expected Input Percentages

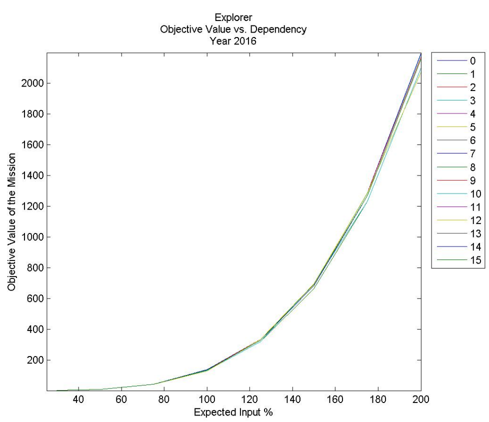
Expected Input Percentages	min	max	Delta	Average	Variation	Max Variation	Min Variation
25%	0.506982	0.536841	0.029859	0.526322	0.056732	0.019985	0.036746
50%	8.111709	8.593281	0.481572	8.426125	0.057152	0.019838	0.037314
75%	41.57692	43.50349	1.926562	43.16981	0.044628	0.007729	0.036898
100%	129.7873	137.4925	7.705157	134.818	0.057152	0.019838	0.037314
125%	320.8096	335.675	14.86545	332.2463	0.044742	0.01032	0.034422
150%	665.2308	696.0558	30.82499	690.717	0.044628	0.007729	0.036898
175%	1232.422	1286.91	54.48774	1271.709	0.042846	0.011953	0.030893
200%	2076.597	2199.88	123.2825	2157.088	0.057152	0.019838	0.037314



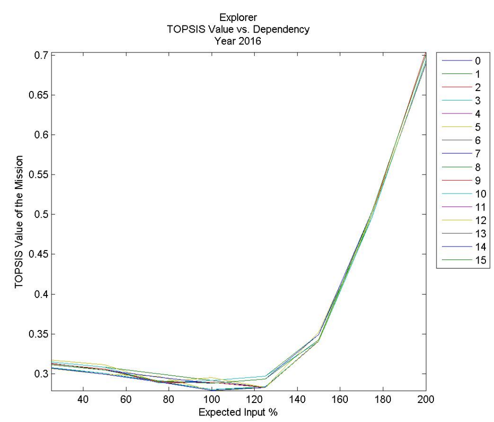
(a) Objective Value Contour



(b) TOPSIS Value Contour

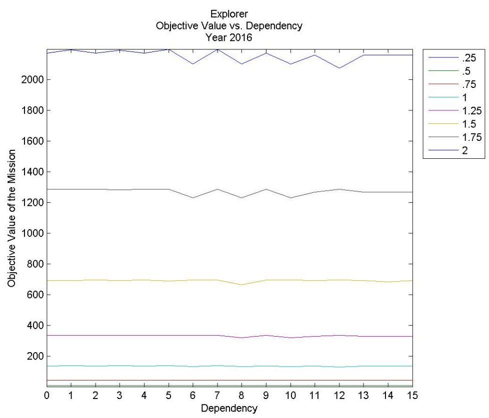


(c) Objective Value vs. Expected Input Percentage

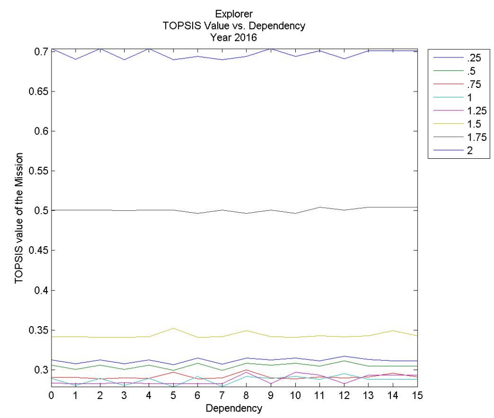


(d) TOPSIS vs. Expected Input Percentage

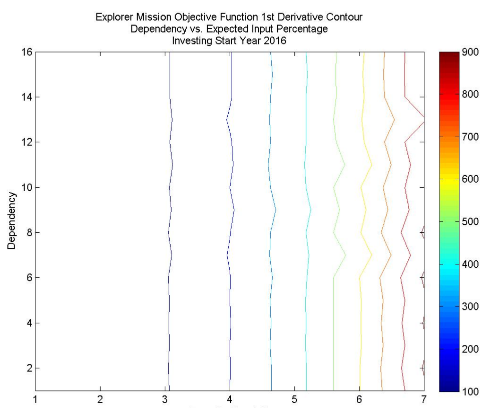
Figure 167: Explorer Start Year of 2016



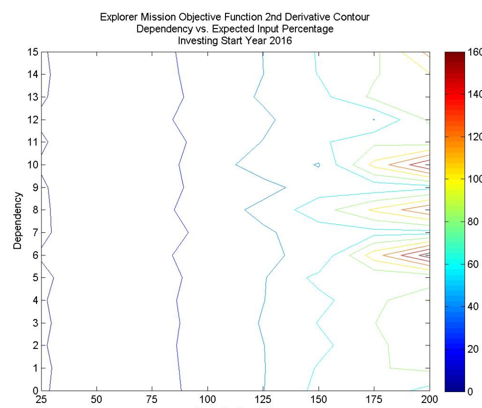
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour



(d) 2nd Derivative of Objective Value Contour

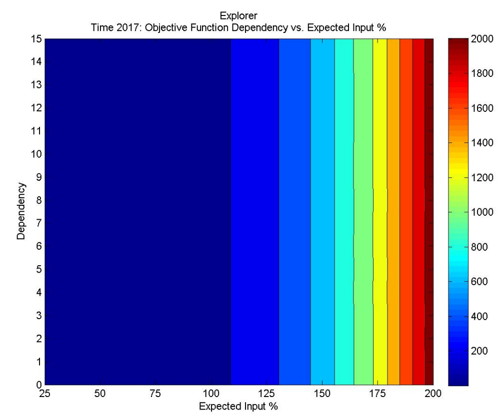
Figure 168: Explorer Start Year of 2016 (con’t)

Table 97: 2010 Statistics Information for various Expected Input Percentages

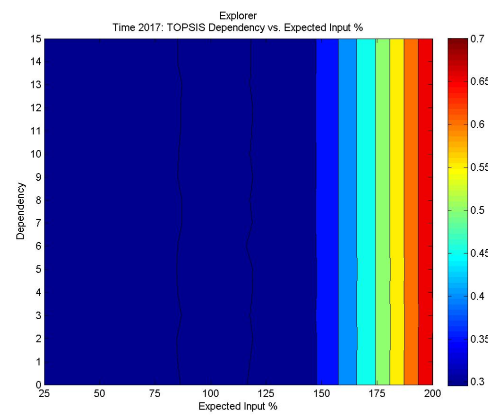
Dependency	X^3	X^2	X	b
0	0.000596	-0.08718	4.863521	-81.0677
1	0.000621	-0.09444	5.447019	-93.0068
2	0.00059	-0.08541	4.723754	-78.2937
3	0.000618	-0.09369	5.38439	-91.6573
4	0.00059	-0.08551	4.731805	-78.4542
5	0.00063	-0.09691	5.644387	-97.0023
6	0.000551	-0.07746	4.255099	-71.1506
7	0.000622	-0.09462	5.46276	-93.3796
8	0.000591	-0.08928	5.121221	-87.1136
9	0.000591	-0.08588	4.760471	-79.0259
10	0.000559	-0.07919	4.320739	-71.1506
11	0.000601	-0.08955	5.08676	-85.9977
12	0.000479	-0.05403	2.203983	-26.7759
13	0.000601	-0.08956	5.086961	-86.0051
14	0.000608	-0.09183	5.267443	-89.6008
15	0.000601	-0.08955	5.08676	-85.9977

Table 98: 2017 Statistics Information for various Expected Input Percentages

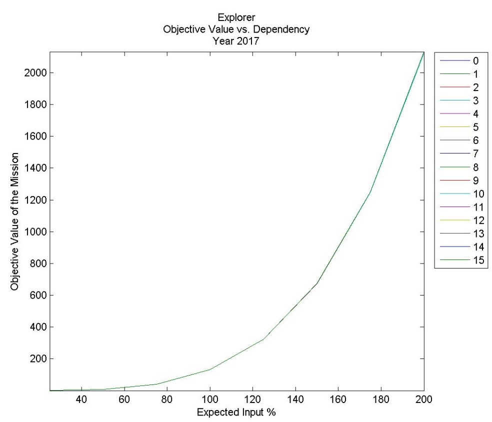
Expected Input Percentages	min	max	Delta	Average	Variation	Max Variation	Min Variation
25%	0.518548	0.519709	0.001161	0.519066	0.002236	0.001239	0.000998
50%	8.296774	8.315347	0.018574	8.311376	0.002235	0.000478	0.001757
75%	42.00242	42.09645	0.09403	42.05612	0.002236	0.000959	0.001277
100%	132.7484	133.0456	0.297181	132.982	0.002235	0.000478	0.001757
125%	323.0103	324.8183	1.80792	324.5857	0.00557	0.000716	0.004854
150%	672.0387	673.5431	1.504481	672.898	0.002236	0.000959	0.001277
175%	1245.035	1247.822	2.787236	1246.732	0.002236	0.000874	0.001361
200%	2123.974	2128.729	4.754902	2127.712	0.002235	0.000478	0.001757



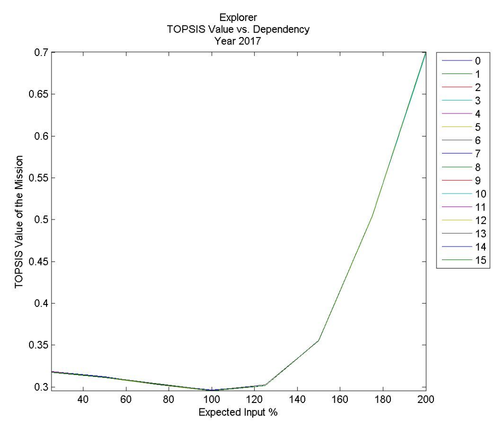
(a) Objective Value Contour



(b) TOPSIS Value Contour

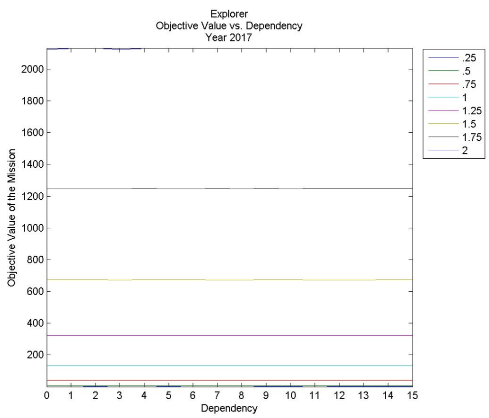


(c) Objective Value vs. Expected Input Percentage

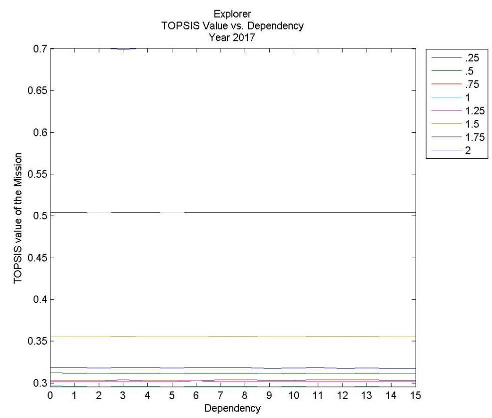


(d) TOPSIS vs. Expected Input Percentage

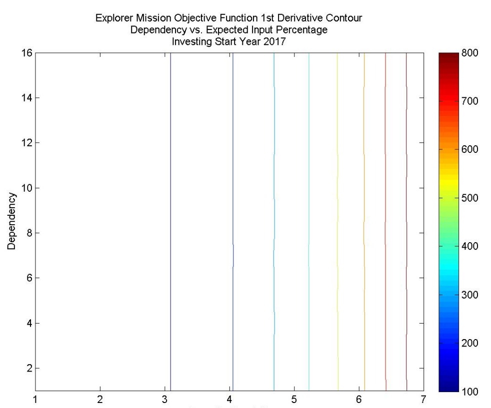
Figure 169: Explorer Start Year of 2017



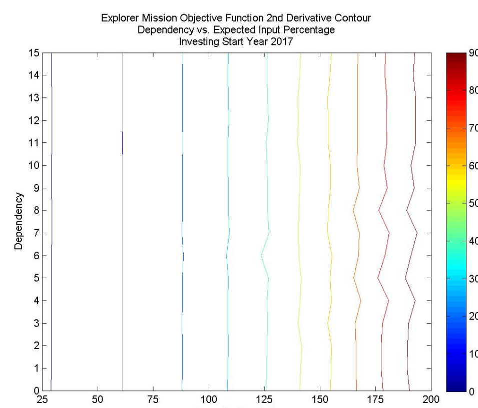
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour



(d) 2nd Derivative of Objective Value Contour

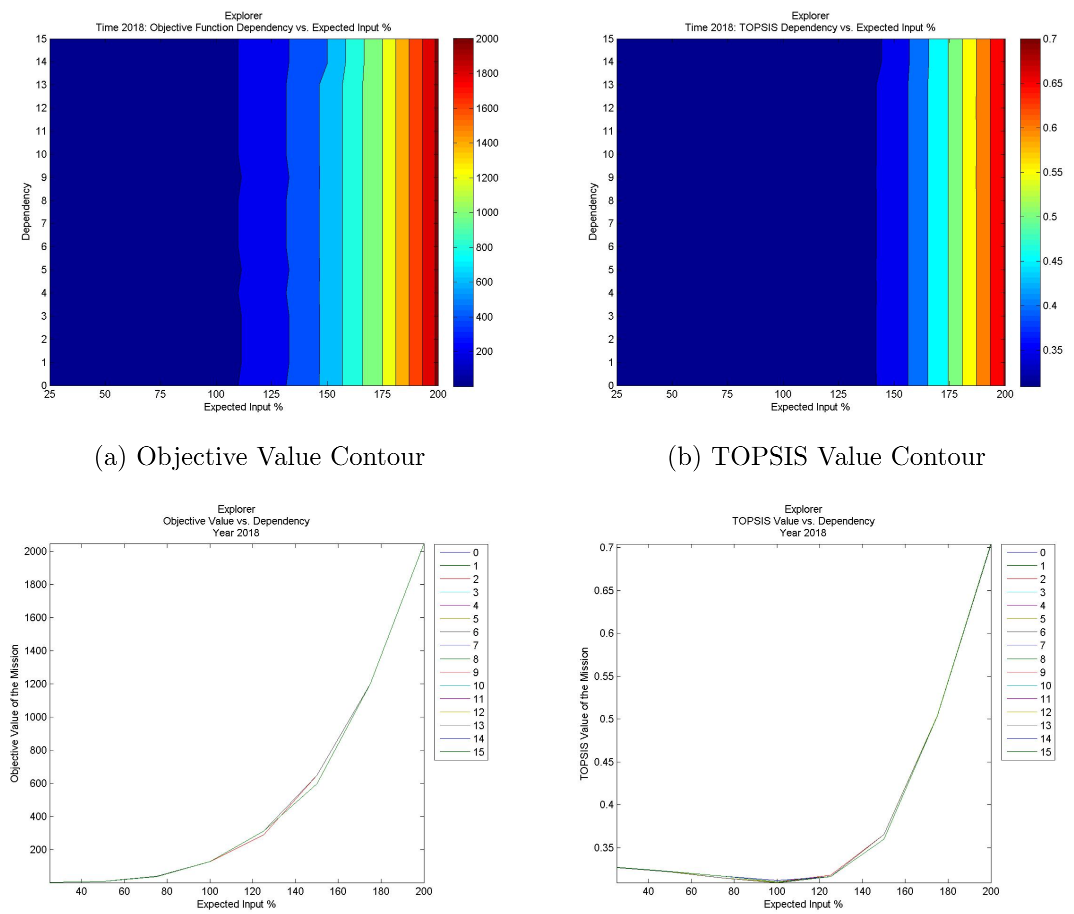
Figure 170: Explorer Start Year of 2017 (con't)

Table 99: 2018 Statistics Information for various Expected Input Percentages

Dependency	X^3	X^2	X	b
0	0.000595	-0.08964	5.132048	-87.226
1	0.0006	-0.09087	5.230724	-89.2435
2	0.000601	-0.09112	5.250853	-89.6577
3	0.000598	-0.09027	5.181982	-88.2272
4	0.000598	-0.09006	5.158202	-87.6508
5	0.000601	-0.09112	5.250853	-89.6577
6	0.000599	-0.09044	5.181651	-87.9937
7	0.000599	-0.09056	5.197271	-88.4299
8	0.000601	-0.09136	5.269794	-90.0226
9	0.000599	-0.09051	5.193625	-88.3698
10	0.000598	-0.09036	5.182719	-88.1597
11	0.000601	-0.09097	5.227273	-88.9835
12	0.0006	-0.09089	5.224282	-88.9812
13	0.000599	-0.09057	5.197911	-88.4281
14	0.000599	-0.09052	5.194315	-88.3698
15	0.000599	-0.09052	5.194315	-88.3698

Table 100: 2018 Statistics Information for various Expected Input Percentages

Expected Input Percentages	min	max	Delta	Average	Variation	Max Variation	Min Variation
25%	0.499003	0.499003	0	0.499003	0	-2.2E-16	2.22E-16
50%	7.984046	7.984046	0	7.984046	0	-2.2E-16	2.22E-16
75%	37.25582	40.41923	3.163419	40.02381	0.079038	0.00988	0.069159
100%	127.1405	127.7447	0.604242	127.707	0.004731	0.000296	0.004436
125%	287.4677	311.8768	24.4091	304.249	0.080227	0.025071	0.055156
150%	596.093	646.7078	50.61471	640.3809	0.079038	0.00988	0.069159
175%	1198.106	1198.106	0	1198.106	0	0	0
200%	2043.916	2043.916	0	2043.916	0	-2.2E-16	2.22E-16



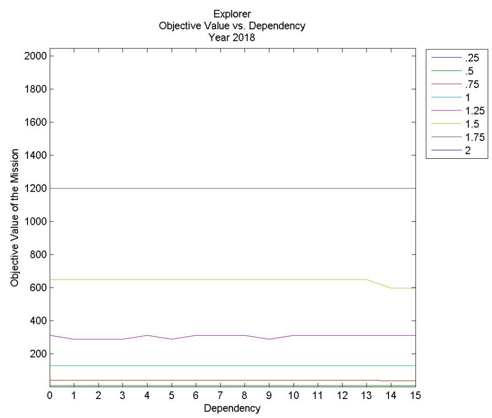
(a) Objective Value Contour

(b) TOPSIS Value Contour

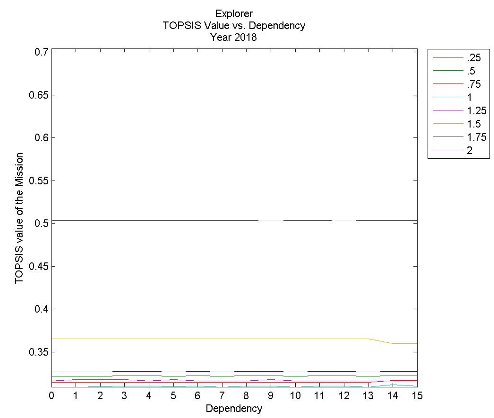
(c) Objective Value vs. Expected Input Percentage

(d) TOPSIS vs. Expected Input Percentage

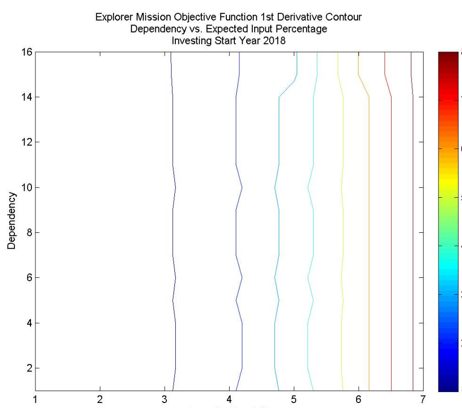
Figure 171: Explorer Start Year of 2018



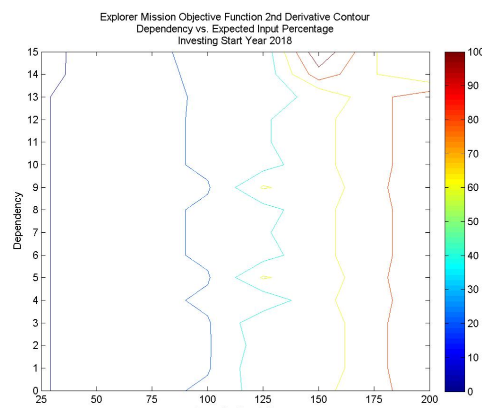
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour



(d) 2nd Derivative of Objective Value Contour

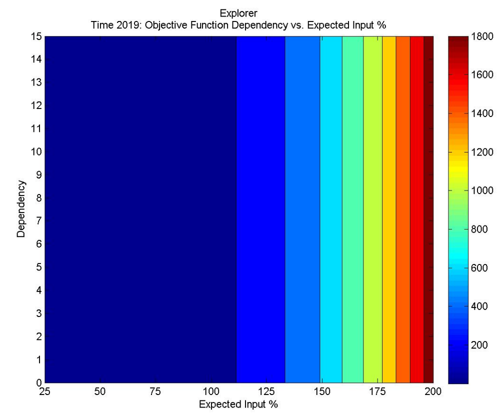
Figure 172: Explorer Start Year of 2018 (con't)

Table 101: 2019 Statistics Information for various Expected Input Percentages

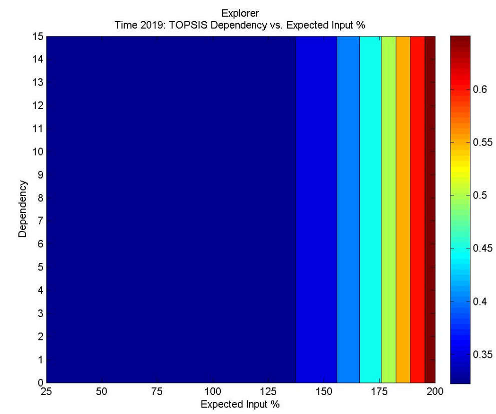
Dependency	X^3	X^2	X	b
0	0.000575	-0.0868	4.978623	-84.6879
1	0.000587	-0.08963	5.086404	-84.6879
2	0.000587	-0.08963	5.086404	-84.6879
3	0.000587	-0.08963	5.086404	-84.6879
4	0.000575	-0.0868	4.978623	-84.6879
5	0.000587	-0.08963	5.086404	-84.6879
6	0.000575	-0.0868	4.978623	-84.6879
7	0.000575	-0.0868	4.978623	-84.6879
8	0.000575	-0.0868	4.978623	-84.6879
9	0.000587	-0.08963	5.086404	-84.6879
10	0.000575	-0.0868	4.978623	-84.6879
11	0.000575	-0.0868	4.978623	-84.6879
12	0.000575	-0.0868	4.978623	-84.6879
13	0.000575	-0.0868	4.978623	-84.6879
14	0.000628	-0.10325	6.277401	-110.511
15	0.000629	-0.10338	6.293017	-110.899

Table 102: 2019 Statistics Information for various Expected Input Percentages

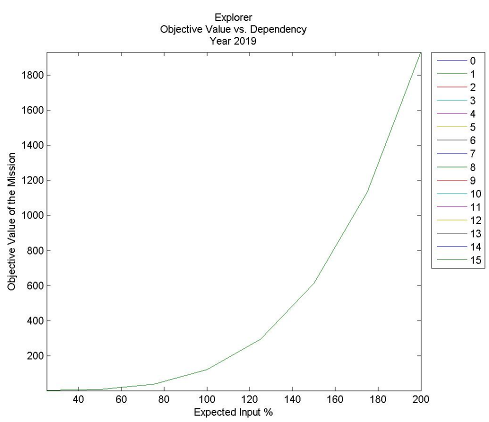
Expected Input Percentages	min	max	Delta	Average	Variation	Max Variation	Min Variation
25%	0.471631	0.471631	0	0.471631	0	0	0
50%	7.546089	7.546089	0	7.546089	0	0	0
75%	38.20207	38.20207	0	38.20207	0	-1.9E-16	1.86E-16
100%	120.7374	120.7374	0	120.7374	0	0	0
125%	294.7691	294.7691	0	294.7691	0	0	0
150%	611.2332	611.2332	0	611.2332	0	-1.9E-16	1.86E-16
175%	1132.385	1132.385	0	1132.385	0	4.02E-16	-4E-16
200%	1931.799	1931.799	0	1931.799	0	0	0



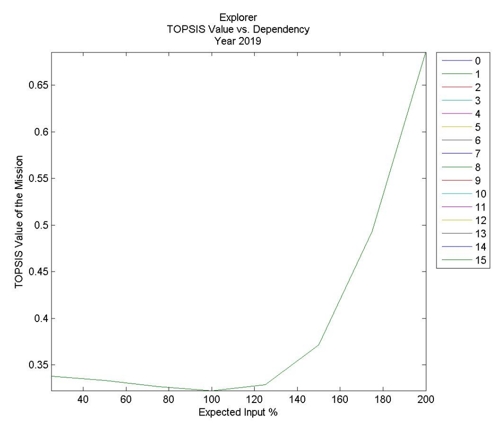
(a) Objective Value Contour



(b) TOPSIS Value Contour

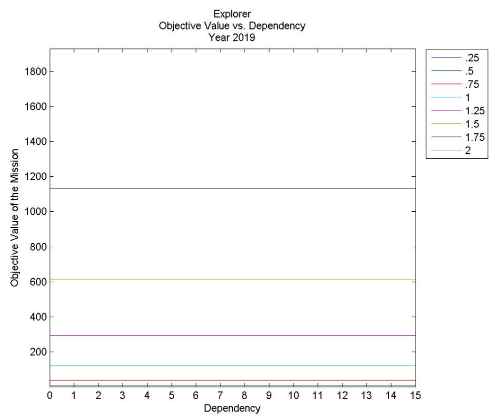


(c) Objective Value vs. Expected Input Percentage

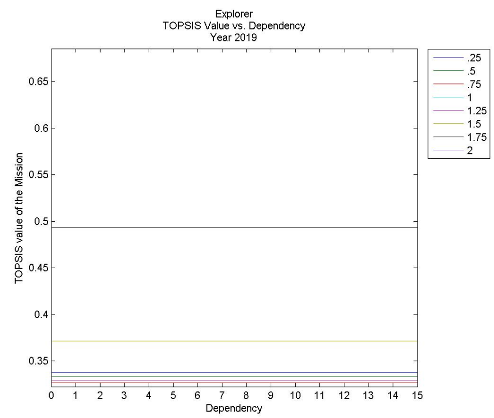


(d) TOPSIS vs. Expected Input Percentage

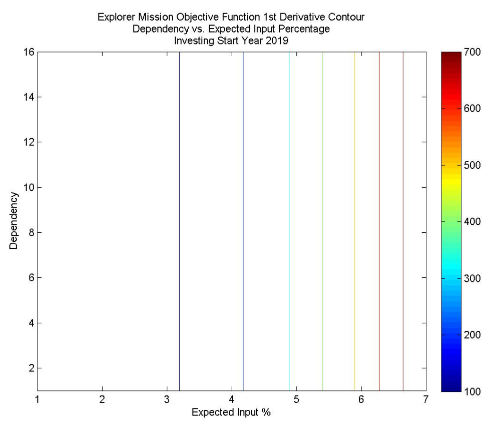
Figure 173: Explorer Start Year of 2019



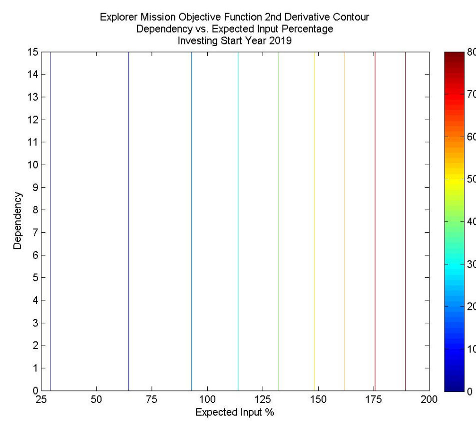
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour



(d) 2nd Derivative of Objective Value Contour

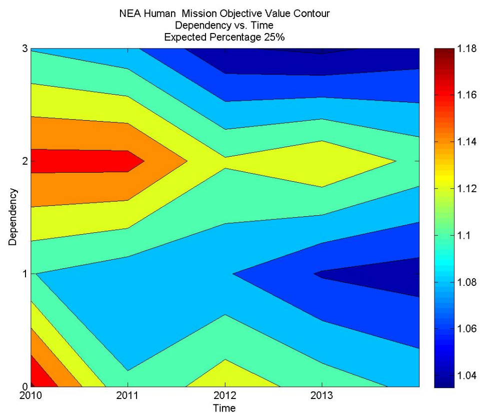
Figure 174: Explorer Start Year of 2019 (con't)

Table 103: 2019 Statistics Information for Various Dependency Levels

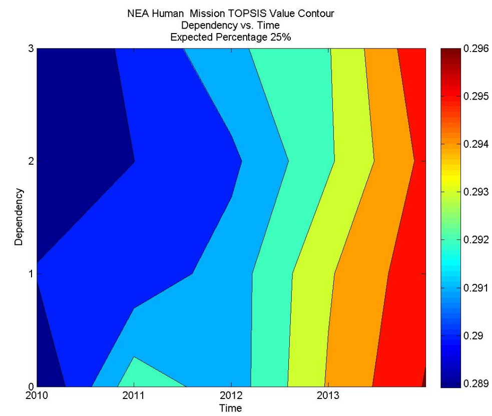
Dependency	X^3	X^2	X	b
0	0.000543	-0.08204	4.705525	-80.0424
1	0.000543	-0.08204	4.705525	-80.0424
2	0.000543	-0.08204	4.705525	-80.0424
3	0.000543	-0.08204	4.705525	-80.0424
4	0.000543	-0.08204	4.705525	-80.0424
5	0.000543	-0.08204	4.705525	-80.0424
6	0.000543	-0.08204	4.705525	-80.0424
7	0.000543	-0.08204	4.705525	-80.0424
8	0.000543	-0.08204	4.705525	-80.0424
9	0.000543	-0.08204	4.705525	-80.0424
10	0.000543	-0.08204	4.705525	-80.0424
11	0.000543	-0.08204	4.705525	-80.0424
12	0.000543	-0.08204	4.705525	-80.0424
13	0.000543	-0.08204	4.705525	-80.0424
14	0.000543	-0.08204	4.705525	-80.0424
15	0.000543	-0.08204	4.705525	-80.0424

APPENDIX B

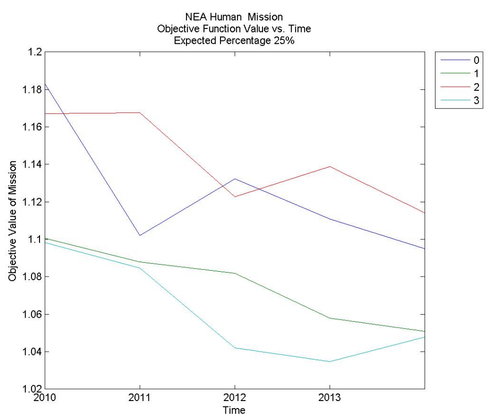
NEA EXPECTED INPUT PERCENTAGE DATA CUBE



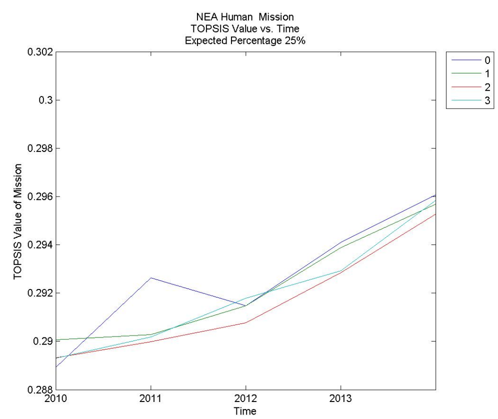
(a) Objective Value Contour



(b) TOPSIS Value Contour

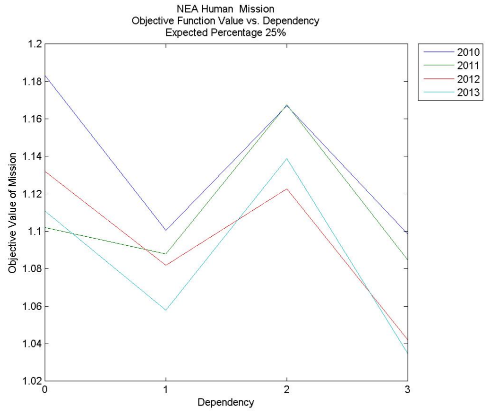


(c) Objective Value vs. Expected Input Percentage

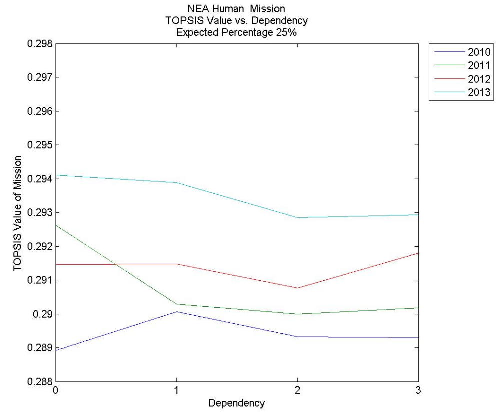


(d) TOPSIS vs. Expected Input Percentage

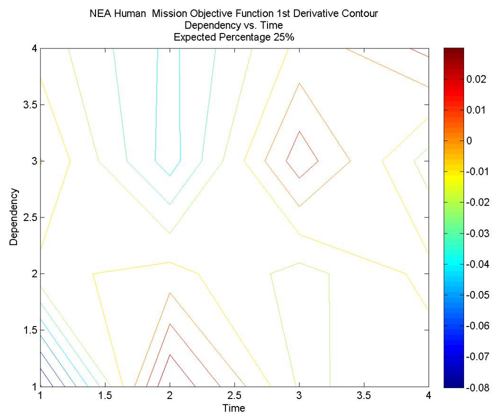
Figure 175: NEA 25% Expected Input Percentage Data Information



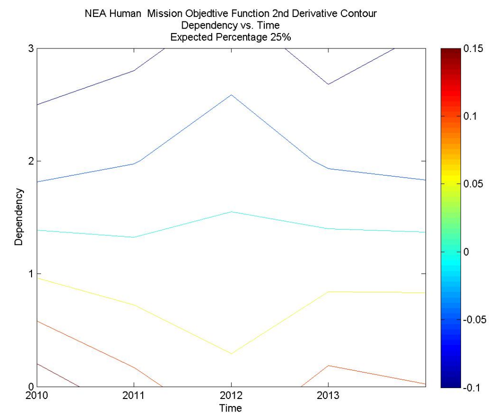
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour



(d) 2nd Derivative of Objective Value Contour

Figure 176: NEA 25% Expected Input Percentage Data Information

Table 104: NEA Technology Portfolio

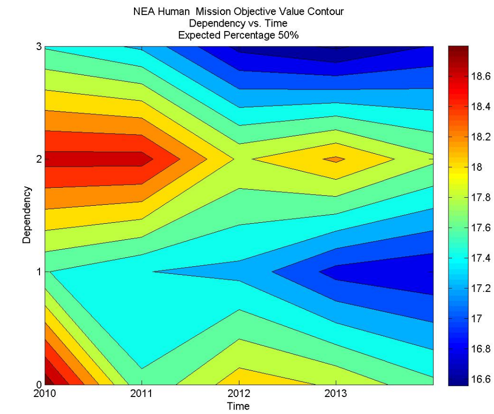
min	max	Delta	Average	Variation	Max Variation	Min Variation
1.098411	1.183298	0.084887	1.137377	0.074634	0.040375	0.03426
1.084733	1.167706	0.082973	1.110593	0.074711	0.051426	0.023285
1.042018	1.132223	0.090205	1.094699	0.082402	0.034278	0.048124
1.034861	1.138758	0.103897	1.085641	0.095701	0.048927	0.046774
1.047832	1.114162	0.06633	1.07696	0.06159	0.034543	0.027047

Table 105: NEA Technology Portfolio

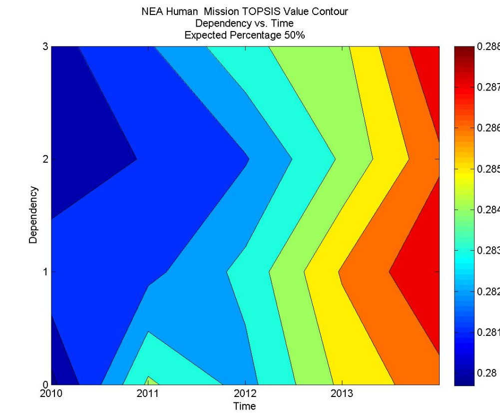
Start Year	x	b
2010	-0.01882	1.165605
2011	0.002804	1.106387
2012	-0.02298	1.129173
2013	-0.01476	1.10778

Table 106: NEA Technology Portfolio

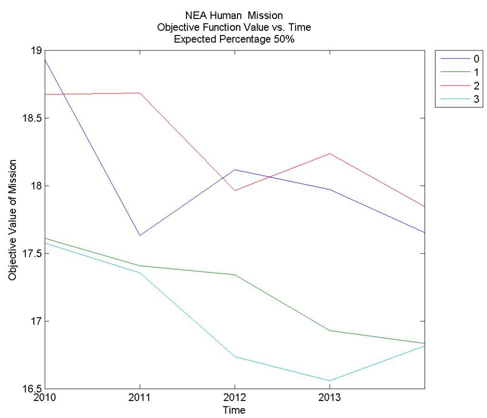
min	max	Delta	Average Variation	Max Variation	Min Variation	
17.57457	18.93277	1.358196	18.19803	0.074634	0.040375	0.03426
17.35572	18.68329	1.327572	17.76948	0.074711	0.051426	0.023285
16.73537	18.11557	1.380203	17.53829	0.078697	0.032916	0.045781
16.55777	18.23709	1.679324	17.42334	0.096384	0.046705	0.049679
16.81542	17.84328	1.027863	17.28611	0.059462	0.032232	0.027229



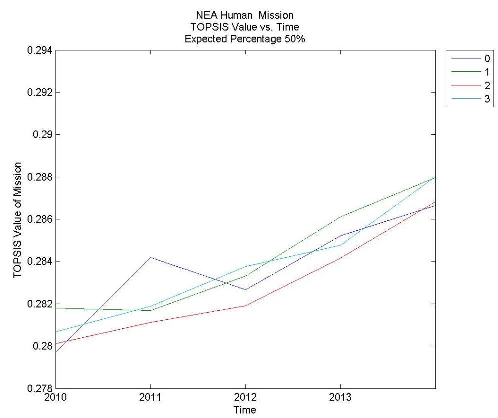
(a) Objective Value Contour



(b) TOPSIS Value Contour



(c) Objective Value vs. Expected Input Percentage

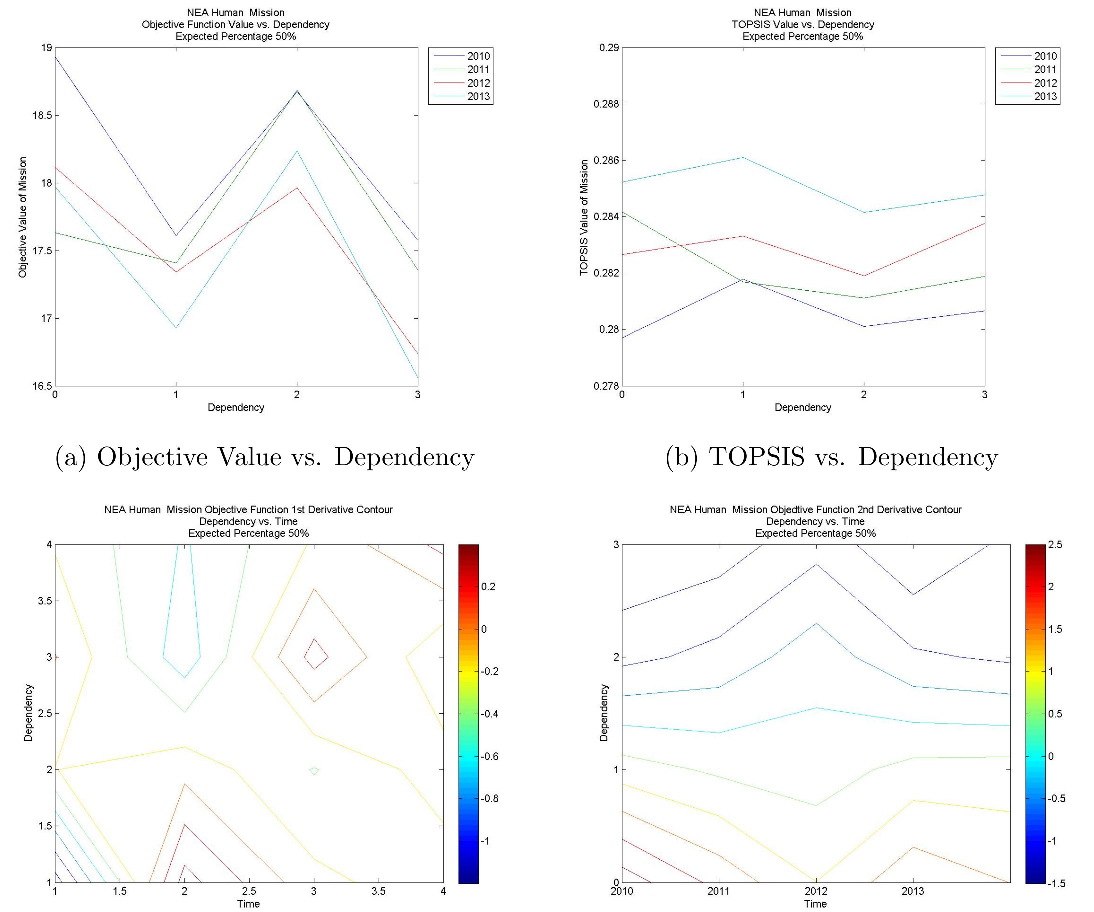


(d) TOPSIS vs. Expected Input Percentage

Figure 177: NEA 50% Expected Input Percentage Data Information

Table 107: NEA Technology Portfolio

Start Year	x	b
2010	-0.3011	18.64969
2011	0.04486	17.70219
2012	-0.35173	18.06588
2013	-0.29307	17.86295



(a) Objective Value vs. Dependency

(b) TOPSIS vs. Dependency

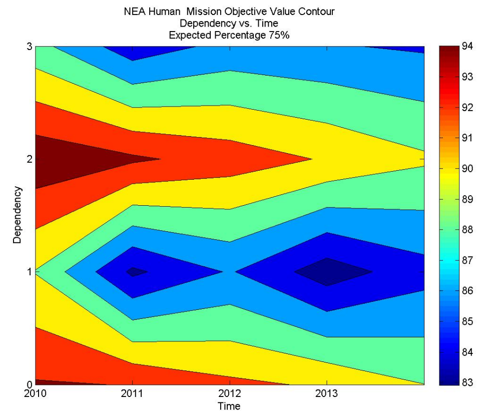
(c) 1st Derivative of Objective Value Contour

(d) 2nd Derivative of Objective Value Contour

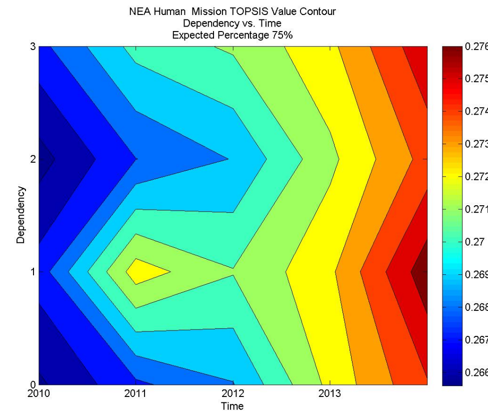
Figure 178: NEA 50% Expected Input Percentage Data Information

Table 108: NEA Technology Portfolio

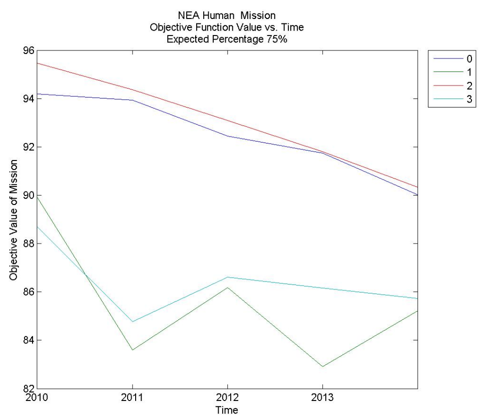
min	max	Delta	Average Variation	Max Variation	Min Variation	
88.69699	95.45798	6.76099	92.0629	0.073439	0.036878	0.036561
83.59262	94.37274	10.78013	89.1717	7 0.120892	0.058325	0.062566
86.17898	93.09247	6.913494	89.5807	0.077176	0.039202	0.037974
82.9032	91.80576	8.902559	88.15186	0.100991	0.04145	0.059541
85.22207	90.32767	5.105602	87.82346	0.058135	0.028514	0.029621



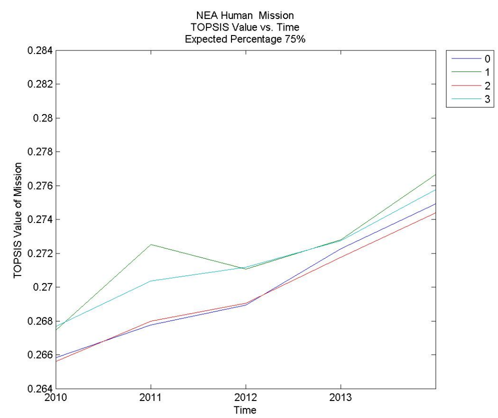
(a) Objective Value Contour



(b) TOPSIS Value Contour



(c) Objective Value vs. Expected Input Percentage

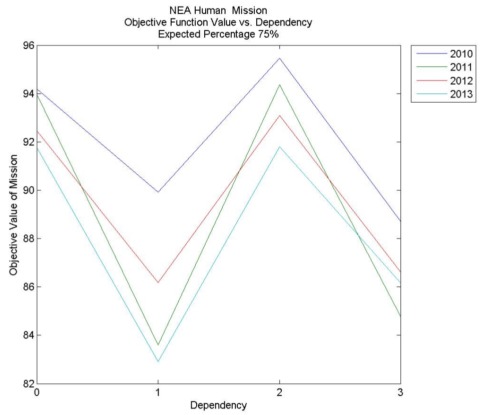


(d) TOPSIS vs. Expected Input Percentage

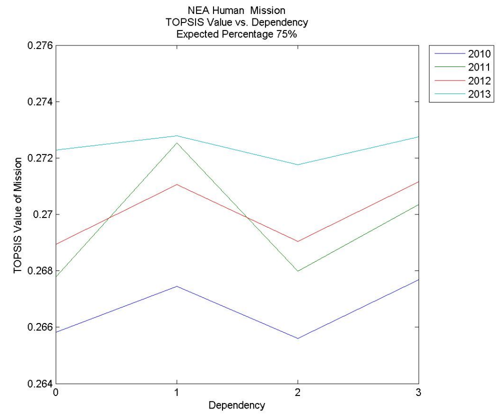
Figure 179: NEA 75% Expected Input Percentage Data Information

Table 109: NEA Technology Portfolio

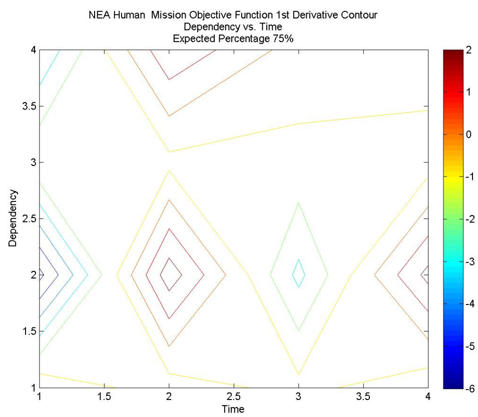
Start Year	x	b
2010	-1.09124	93.69975
2011	-1.67182	91.6795
2012	-1.05692	91.16608
2013	-0.78426	89.32825



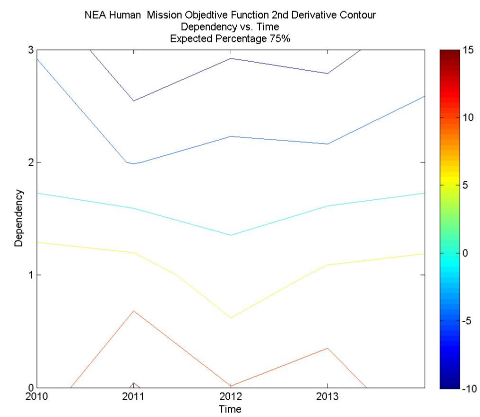
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour

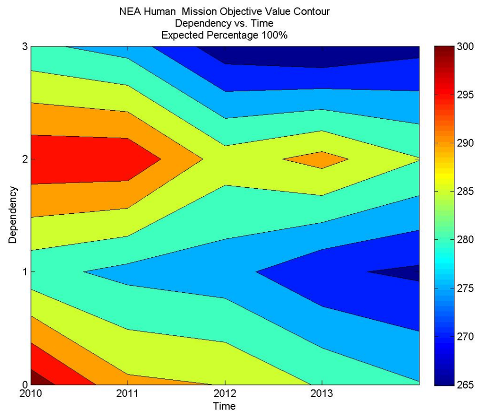


(d) 2nd Derivative of Objective Value Contour

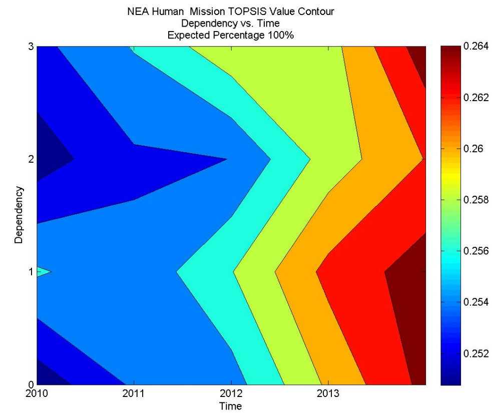
Figure 180: NEA 75% Expected Input Percentage Data Information

Table 110: NEA Technology Portfolio

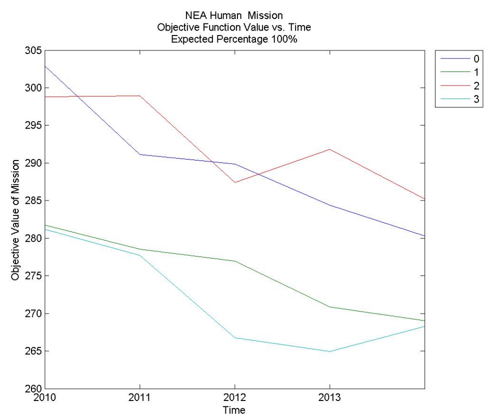
min	max	Delta	Average Variation	Max Variation	Min Variation	
281.1932	302.9243	21.73114	291.1685	0.074634	0.040375	0.03426
277.6915	298.9327	21.24116	286.5742	0.074121	0.043125	0.030996
266.7566	289.8491	23.09252	280.243	0.082402	0.034278	0.048124
264.9243	291.7935	26.86918	277.9919	0.096655	0.049648	0.047007
268.2449	285.2254	16.9805	275.7017	0.06159	0.034543	0.027047



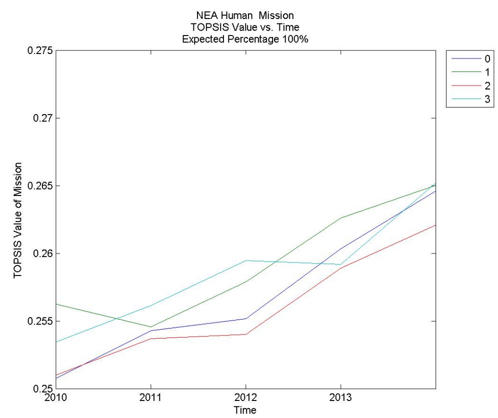
(a) Objective Value Contour



(b) TOPSIS Value Contour



(c) Objective Value vs. Expected Input Percentage

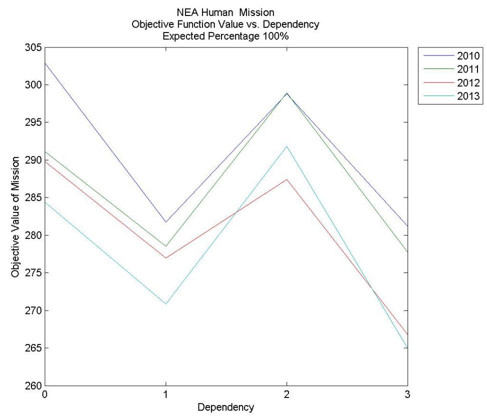


(d) TOPSIS vs. Expected Input Percentage

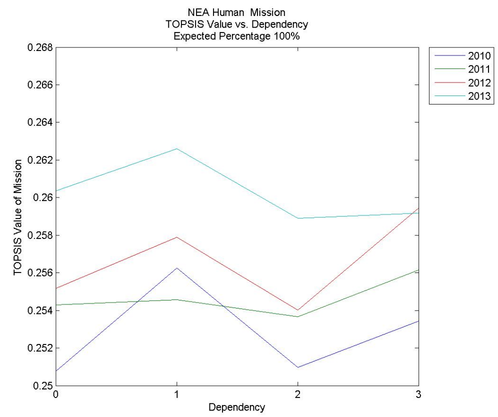
Figure 181: NEA 100% Expected Input Percentage Data Information

Table 111: NEA Technology Portfolio

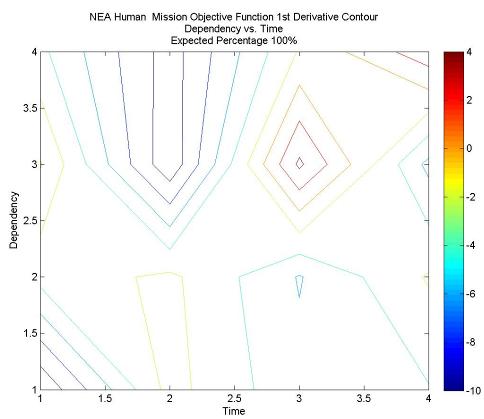
Start Year	x	b
2010	-4.81765	298.395
2011	-1.99722	289.5701
2012	-5.88354	289.0683
2013	-3.75128	283.6188



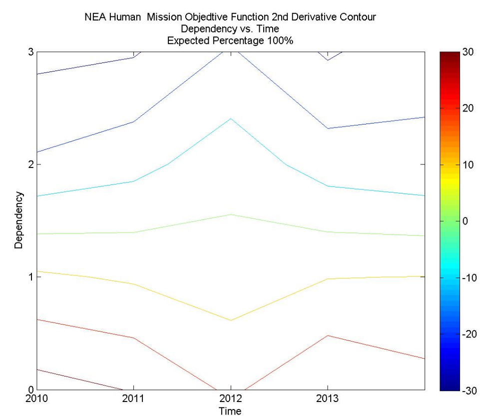
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour

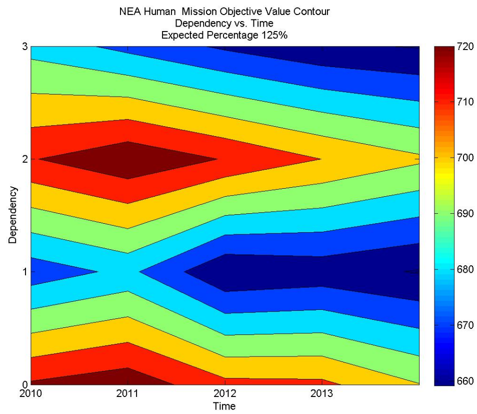


(d) 2nd Derivative of Objective Value Contour

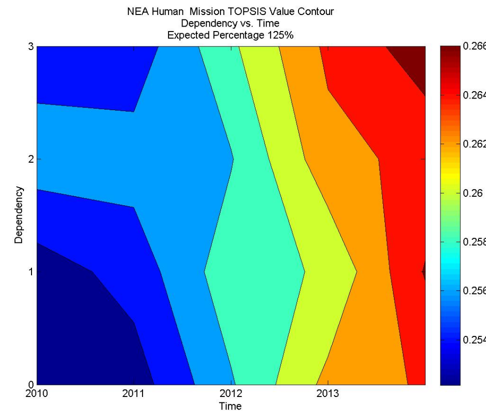
Figure 182: NEA 25% Expected Input Percentage Data Information

Table 112: NEA Technology Portfolio

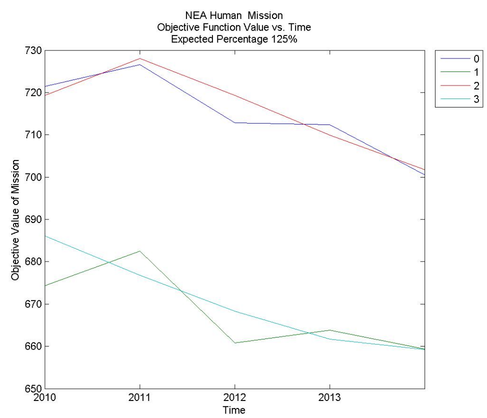
min	max	Delta	Average Variation	Max Variation	Min Variation	
674.3238	721.3865	47.06269	700.2827	0.067205	0.030136	0.037069
676.8519	728.0082	51.15629	703.4802	0.072719	0.034867	0.037852
660.8455	719.3374	58.49188	690.3322	0.08473	0.042016	0.042714
661.6935	712.3864	50.69293	686.9539	0.073794	0.037022	0.036772
659.2336	701.7311	42.49756	680.1833	0.06248	0.031679	0.0308



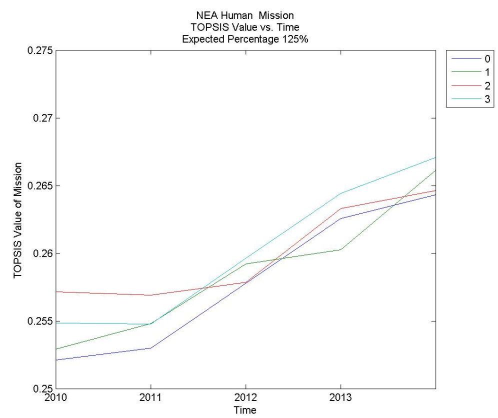
(a) Objective Value Contour



(b) TOPSIS Value Contour



(c) Objective Value vs. Expected Input Percentage

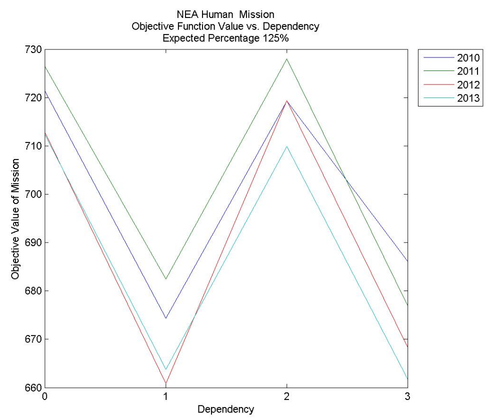


(d) TOPSIS vs. Expected Input Percentage

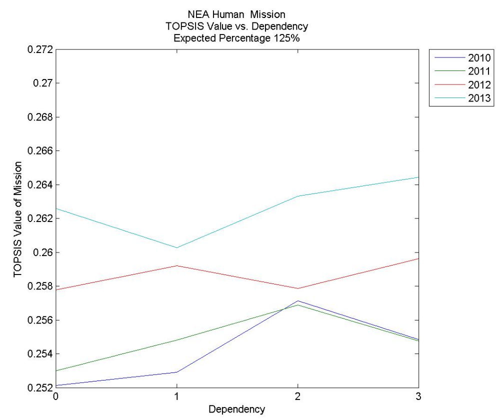
Figure 183: NEA 125% Expected Input Percentage Data Information

Table 113: NEA Technology Portfolio

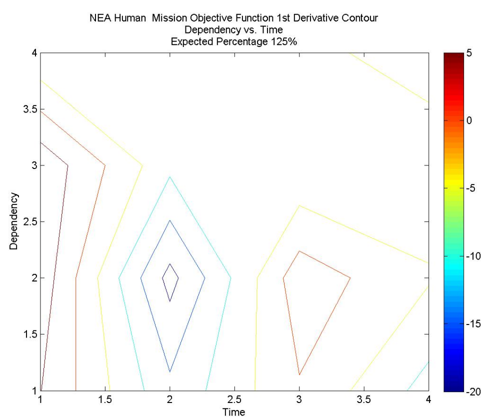
Start Year	x	b
2010	-6.07917	709.4015
2011	-10.3607	719.0213
2012	-7.53502	701.6347
2013	-10.597	702.8494



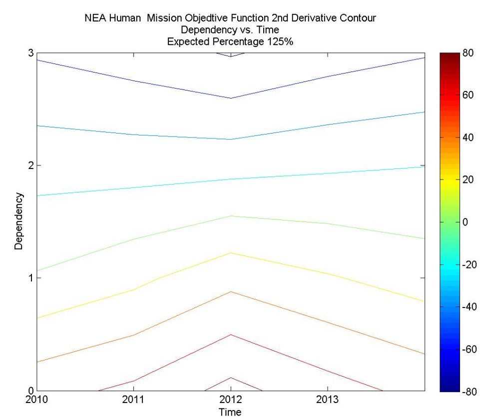
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour

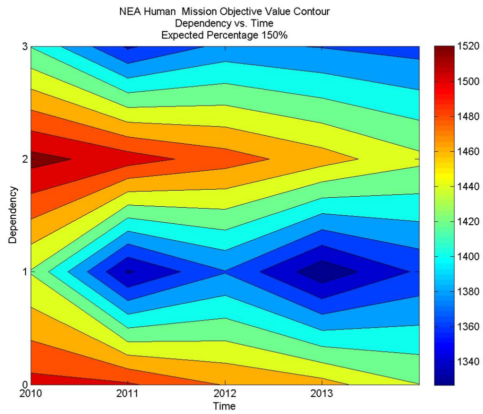


(d) 2nd Derivative of Objective Value Contour

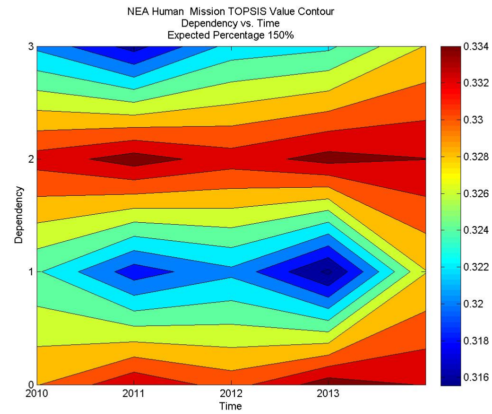
Figure 184: NEA 100% Expected Input Percentage Data Information

Table 114: NEA Technology Portfolio

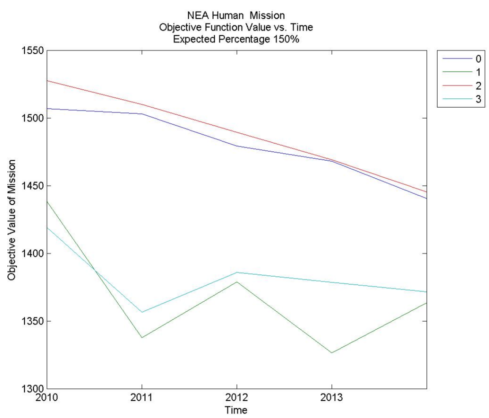
min	max	Delta	Average Variation	Max Variation	Min Variation	
1419.152	1527.328	108.1758	1473.006	0.073439	0.036878	0.036561
1337.482	1509.964	172.482	1426.748	0.120892	0.058325	0.062566
1378.864	1489.48	110.6159	1433.291	0.077176	0.039202	0.037974
1326.451	1468.892	142.4409	1410.43	0.100991	0.04145	0.059541
1363.553	1445.243	81.68963	1405.175	0.058135	0.028514	0.029621



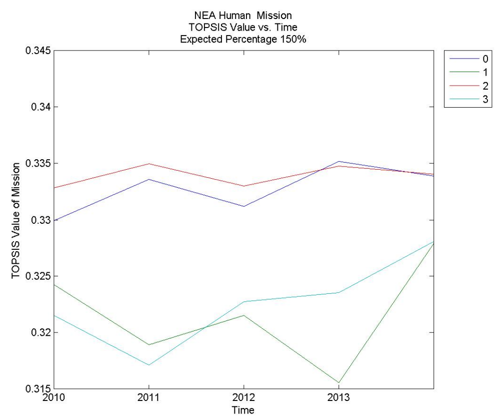
(a) Objective Value Contour



(b) TOPSIS Value Contour



(c) Objective Value vs. Expected Input Percentage

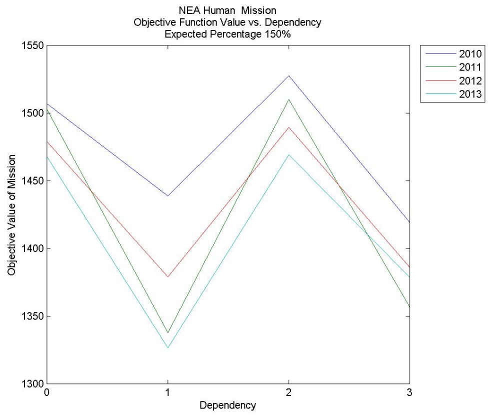


(d) TOPSIS vs. Expected Input Percentage

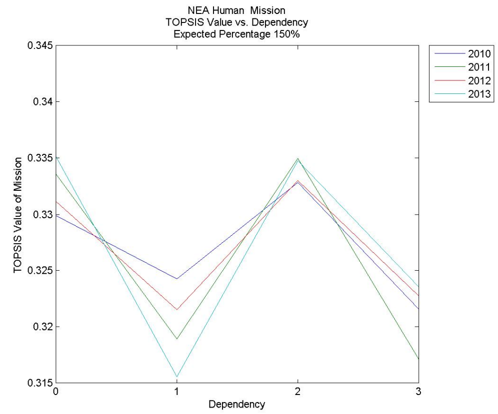
Figure 185: NEA 150% Expected Input Percentage Data Information

Table 115: NEA Technology Portfolio

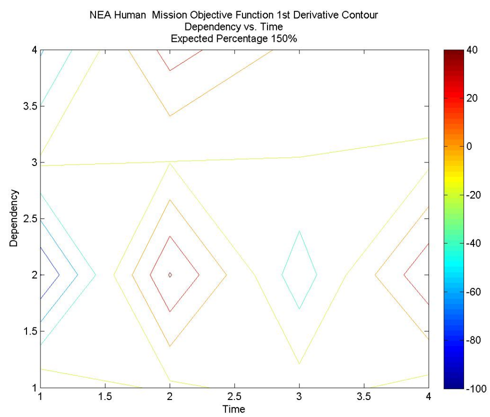
Start Year	x	b
2010	-17.4598	1499.196
2011	-26.7491	1466.872
2012	-16.9107	1458.657
2013	-12.5482	1429.252



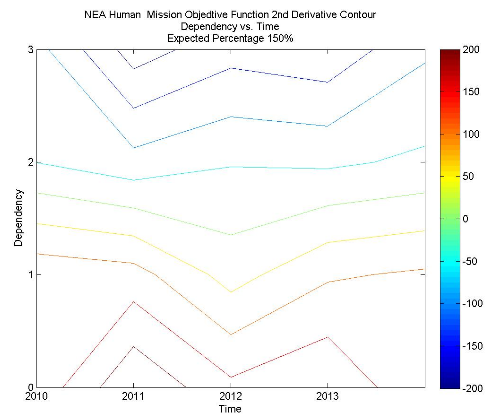
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour

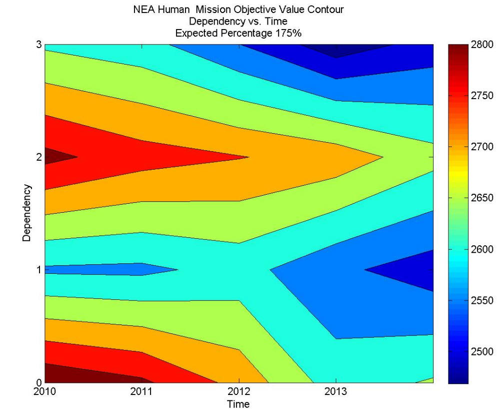


(d) 2nd Derivative of Objective Value Contour

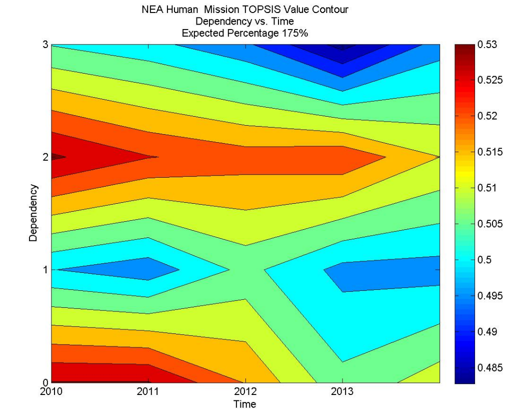
Figure 186: NEA 150% Expected Input Percentage Data Information

Table 116: NEA Technology Portfolio

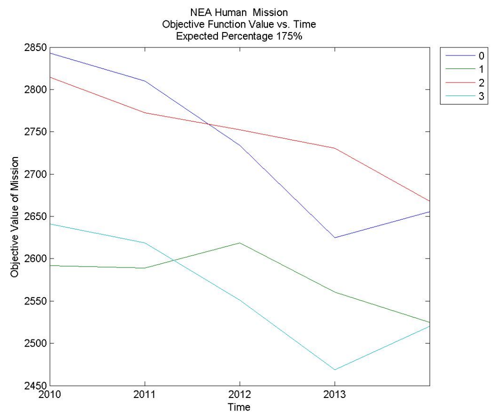
min	max	Delta	Average Variation	Max Variation	Min Variation	
2591.934	2843.286	251.3516	2722.681	0.092318	0.044296	0.048021
2588.831	2810.007	221.1756	2697.515	0.081992	0.041702	0.04029
2550.716	2752.204	201.488	2663.818	0.075639	0.03318	0.042459
2468.977	2730.796	261.8186	2596.313	0.100842	0.051797	0.049045
2520.243	2667.664	147.4217	2592.084	0.056874	0.029158	0.027716



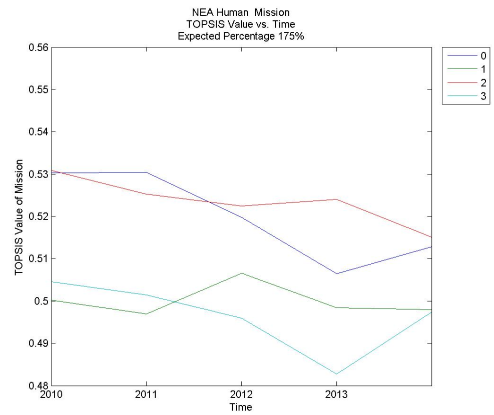
(a) Objective Value Contour



(b) TOPSIS Value Contour



(c) Objective Value vs. Expected Input Percentage

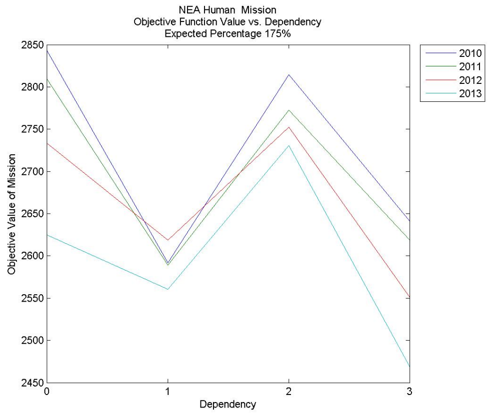


(d) TOPSIS vs. Expected Input Percentage

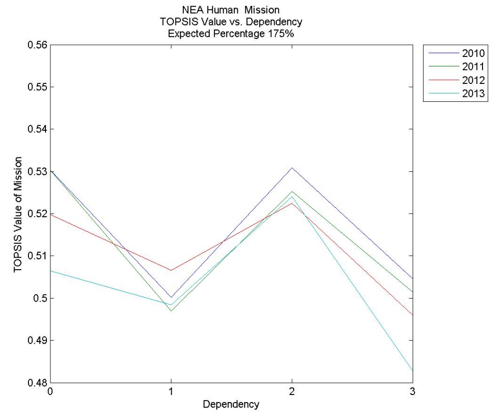
Figure 187: NEA 175% Expected Input Percentage Data Information

Table 117: NEA Technology Portfolio

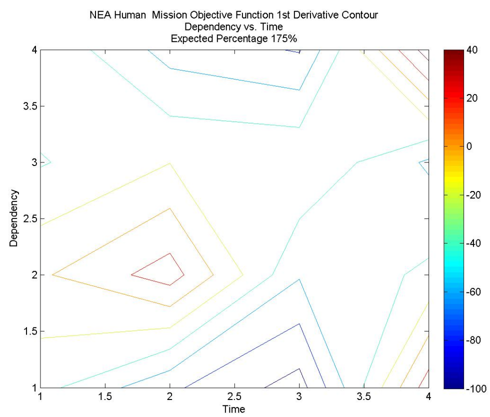
Start Year	x	b
2010	-38.4121	2780.299
2011	-39.0525	2756.094
2012	-41.5367	2726.123
2013	-29.778	2640.98



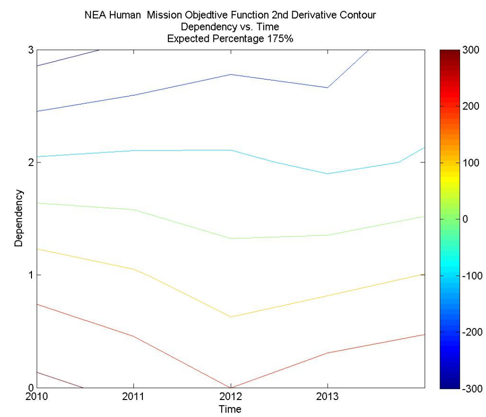
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour

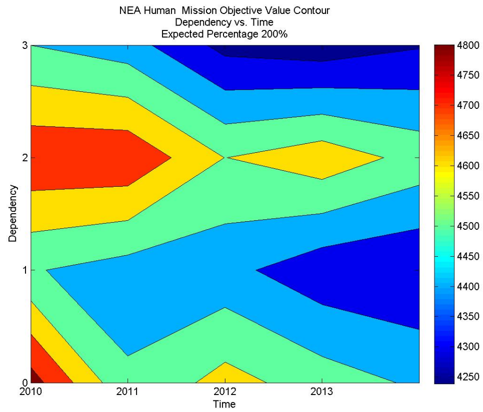


(d) 2nd Derivative of Objective Value Contour

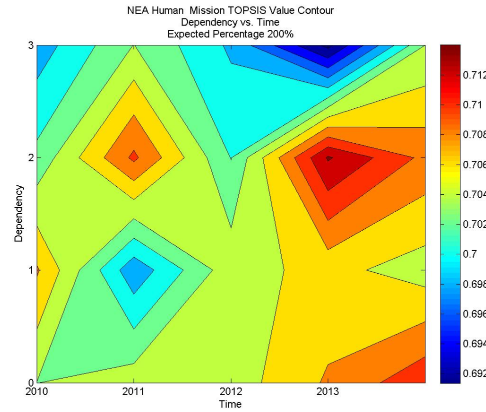
Figure 188: NEA 175% Expected Input Percentage Data Information

Table 118: NEA Technology Portfolio

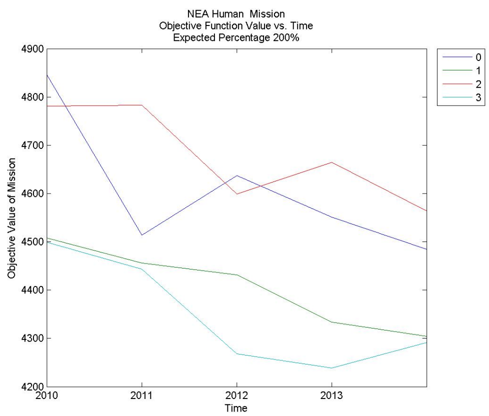
min	max	Delta	Average Variation	Max Variation	Min Variation	
4499.091	4846.789	347.6982	4658.696	0.074634	0.040375	0.03426
4443.065	4782.923	339.8585	4548.988	0.074711	0.051426	0.023285
4268.105	4637.586	369.4804	4483.888	0.082402	0.034278	0.048124
4238.789	4664.352	425.5628	4446.784	0.095701	0.048927	0.046774
4291.918	4563.606	271.688	4411.227	0.06159	0.034543	0.027047



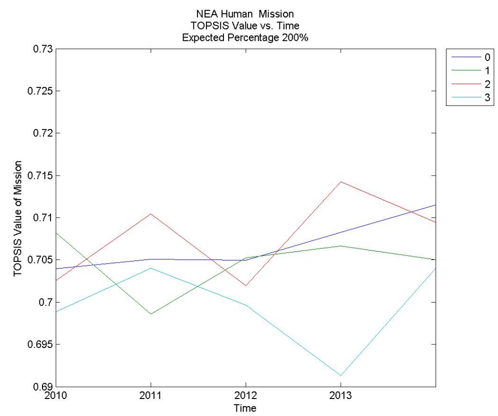
(a) Objective Value Contour



(b) TOPSIS Value Contour



(c) Objective Value vs. Expected Input Percentage

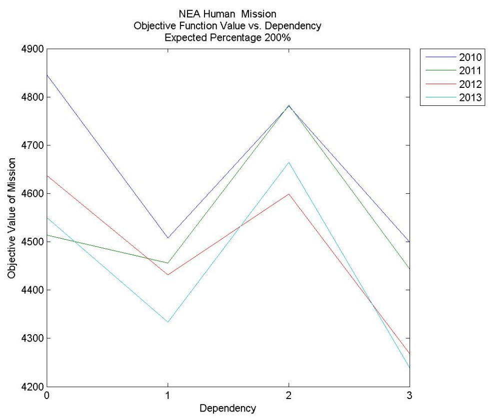


(d) TOPSIS vs. Expected Input Percentage

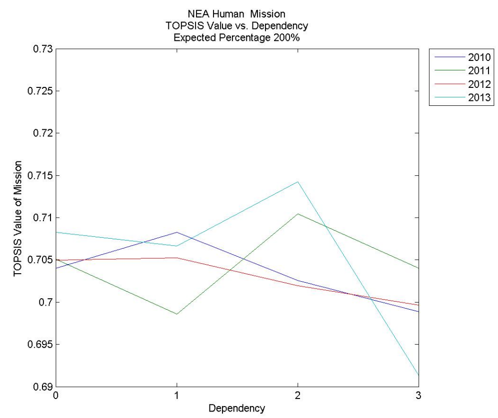
Figure 189: NEA 200% Expected Input Percentage Data Information

Table 119: NEA Technology Portfolio

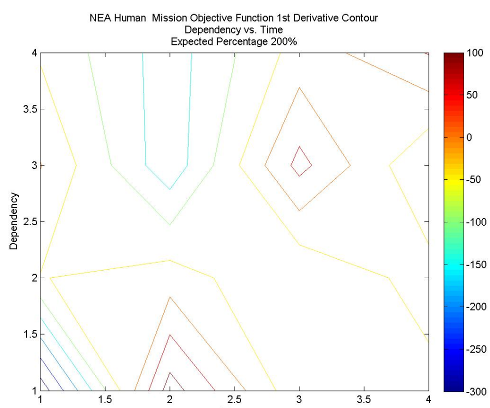
Start Year	x	b
2010	-77.0824	4774.32
2011	11.48418	4531.762
2012	-94.1366	4625.093
2013	-60.4549	4537.466



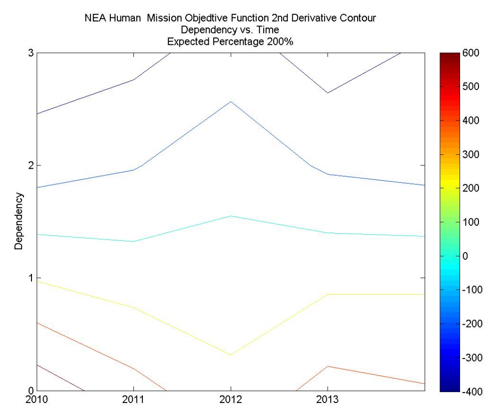
(a) Objective Value vs. Dependency



(b) TOPSIS vs. Dependency



(c) 1st Derivative of Objective Value Contour



(d) 2nd Derivative of Objective Value Contour

Figure 190: NEA 200% Expected Input Percentage Data Information

REFERENCES

[1] "Summary report of the review of u.s. human space flight plans committee."

[2] "Exploration systems architecture study - final report," November 2005.

[3] "Nasa awards thermal protection contract for orion spacecraft," September 2006.

[4] "Nasa selects orion crew exploration vehicle prime contractor," August 2006.

[5] Abell, P., "Roadmap for human exploration of small bodies," January 2011.

[6] Adumitroaie, V. and Weisbin, C., "Technology infusion challenges from a decision support perspective," in *Aerospace conference, 2009 IEEE*, pp. 1–13, IEEE.

[7] Akao, Y., "Development history of quality function deployment," *The Customer Driven Approach to Quality Planning and Deployment*, 1994.

[8] Atallah, M. and Blanton, M., *Algorithms and Theory of Computation Handbook: Special topics and techniques*, vol. 2. Chapman & Hall/CRC, 2010.

[9] Atkinson, N., "Augustine comission: Current funding won't get nasa out of low earth orbit," September 2009.

[10] August, J., "Optimization by the fixed-point method," *Optimization*, 2007.

[11] Bailey, R., *Design of comparative experiments*, vol. 25. Cambridge Univ Pr, 2008.

[12] banke, J., "Orion vs. apollo: Nasa's 21st century moonshot," September 2008.

[13] Bapat, R., *Linear algebra and linear models*. Springer Verlag, 2000.

[14] Berger, B., "Nasa studies early moon shot for new space capsul," January 2007.

[15] Bergin, C., "Orion landings to be splashdowns - ksc buildings to be demolished," August 2007.

[16] Borenstein, S., "Return-to-moon plan gets boost on capitol hill," September 2009.

[17] Braunecker, B., Aebishcer, B., and Appius, R., "Method for determing the position and rotational position of an object," December 2003.

[18] Chimes, A., "Expert group says nasa budget too small for big space plans," August 2009.

[19] Chvátal, V., *Linear programming.* WH Freeman, 1983.

[20] Connolly, J., "Constellation program overview," *NASA Presentation*, 2006.

[21] Connolly, J. F., "Constellation program overview," October 2006.

[22] Coppinger, R., "Nasa quietly sets up budget for ares iv lunar crew launch vehicle with 2017 test flight target," February 2007.

[23] Cormen, T., *Introduction to algorithms.* The MIT press, 2001.

[24] Courant, R., "Variational methods for the solution of problems of equilibrium and vibrations," *LECTURE NOTES IN PURE AND APPLIED MATHEMATICS*, pp. 1–1, 1994.

[25] Dantzig, G., *Linear programming and extensions.* Princeton Univ Pr, 1998.

[26] de Berg, M., van Kreveld, M., Overmars, M., and Schwarzkopf, O., "Computational geometry algorithms and applications. 1998. isbn: 3-540-65620-0."

[27] Denney, R., *Succeeding with use cases: working smart to deliver quality.* Pearson Education India, 2005.

[28] Dickmanns, E. and Schell, F., "Visual autonomous automatic landing of airplanes," AGARD Symp. on Advances in Guidance and Control of Precision Guided Weapons, 1994.

[29] Ditteon, R., *Modern Geometrical Optics.* John Wiley & Sons, Inc.

[30] Drake, B. and Center, L., *Reference Mission Version 3.0: Addendum to the Human Exploration of Mars: The Reference Mission of the NASA Mars Exploration Study Team.* National Aeronautics and Space Administration, Lyndon B. Johnson Space Center, 1998.

[31] Elfes, A., Lincoln, W., Rodriguez, G., Weisbin, C., and Wertz, J., "Risk-based technology portfolio optimization for early space mission design," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 42, no. 1, pp. 22–36, 2006.

[32] Engineering, N. C. S. and Office, I., eds., *Constellation Architecture Requirements Document (CARD)*, NASA: Constellation Systems Engineering and Integration Office, 2008.

[33] Ferguson, G., "Statistical analysis in psychology and education.," 1959.

[34] Feron, E. and Paduano, J., "A passive sensor for position and attitude estimation using and interferometric target," (Atlantis, Bahamas), 43rd IEEE Conference on Decision and Control, December 2004.

[35] Garey, M. and Johnson, D., *Computers and Intractability: A Guide to the Theory of NP-completeness*. WH Freeman & Co., 1979.

[36] Gass, S., "Ifors'operational research hall of fame george b. dantzig," *International Transactions in Operational Research*, vol. 10, no. 2, pp. 191–193, 2003.

[37] Gass, S., "Ifors'operational research hall of fame albert william tucker," *International Transactions in Operational Research*, vol. 11, no. 2, pp. 239–242, 2004.

[38] Goodliff, K., Cirillo, W., Earle, K., Reeves, J., Shyface, H., Andraschko, M., Merrill, R., Stromgren, C., and Cirillo, C., "Lunar exploration architecture level key drivers and sensitivities," in *Aerospace conference, 2009 IEEE*, pp. 1–15, IEEE.

[39] Hong, J., Kim, J., Park, J. H., Min, S. W., and Lee, B., "Viewing-angle-enhanced three-dimensional integral imaging using the combination of a lenticular lens sheet and a two-dimensional lens array," *Information Optics and Photonis Technology*, 2005.

[40] Howard, A., Lincoln, W., Manvi, R., Rodriguez, G., Weisbin, C., and Drummond, M., "A methodology to determine impact of robotic technologies on space exploration missions," in *Automation Congress, 2004. Proceedings. World*, vol. 15, pp. 229–234, IEEE, 2004.

[41] Ilanko, S., "Asymptotic modelling theorems for the static analysis of linear elastic structures," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, vol. 461, no. 2063, p. 3525, 2005.

[42] Ilanko, S., "Introducing the use of positive and negative inertial functions in asymptotic modelling," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, vol. 461, no. 2060, p. 2545, 2005.

[43] Jequier, B., "Theory and practice of lenticular sheets," *Hyperspace*.

[44] Johnson, R. B., "Improvments in lenticular lens arrays design and fabrication," Internation Optical Design Conference, 2006.

[45] Kempthorne, O., "The design and analysis of experiments.," 1952.

[46] Lamia, W., "Integrating qfd with object oriented software design methodologies," in *Transactions from the Seventh Symposium on Quality Function Deployment*, pp. 417–434, Citeseer.

[47] Lee, S. W., Koo, J. S., and Kim, E. S., “Holographic lenticular sheet for an optical direction modulator,” *Optical Pattern Recognition*, vol. XIV.

[48] Lincoln, W. and Weisbin, C., “Temporal investment strategy to enable jpl future space missions,” 2006.

[49] Lindman, H., *Analysis of variance in complex experimental designs.* WH Freeman & Co, 1974.

[50] Loyka, S. and Tsoulos, G., “Estimating mimo system performance using the correlation matrix approach,” *Communications Letters, IEEE*, vol. 6, no. 1, pp. 19–21, 2002.

[51] Manvi, R., Zimmerman, W., Weisbin, C., and Rodriguez, G., “Technology portfolio options for nasa missions using decision trees,” in *Aerospace Conference Proceedings, 2002. IEEE*, vol. 1, pp. 1–115, IEEE, 2002.

[52] Marcuse, D., *Light Transmission Optics Second Edition.* Van Nostrand Reinhold Company, Copyright 1982 by Bell Telephone Laboratories, Inc.

[53] Matoušek, J. and G
”artner, B., *Understanding and using linear programming.* Springer Verlag, 2007.

[54] Mavris, D. and Kirby, M., “Technology identification, evaluation, and selection for commercial transport aircraft,” in *58th Annual Conference Of Society of Allied Weight Engineers, San Jose, California*, pp. 24–26, Citeseer, 1999.

[55] Minium, E., King, B., and Bear, G., *Statistical reasoning in psychology and education.* Wiley New York:, 1970.

[56] Montgomery, D., *Design and analysis of experiments.* John Wiley & Sons Inc, 2008.

[57] Morgan, S., “A comparison of simplex method algorithms,” 1997.

[58] Morrison, D., “News article: Nasa shifts course toward neas,” May 2010.

[59] Nering, E. and Tucker, A., *Linear programs and related problems*, vol. 1. Academic Pr, 1993.

[60] on Appropriations, U. S. C. H. C., *Consolidated Appropriations Act, 2008, Book 1 of 2–Divisions AF, January 2008, 110-1 House of Representatives.* sn, 2008.

[61] Preparata, F. and Shamos, M., *Computational geometry: an introduction.* Springer, 1985.

[62] Sage, A. and Rouse, W., *Handbook of systems engineering and management.* Wiley-Interscience, 2009.

[63] Salz, J. and Winters, J., "Effect of fading correlation on adaptive arrays in digital mobile radio," *Vehicular Technology, IEEE Transactions on*, vol. 43, no. 4, pp. 1049–1057, 1994.

[64] Schell, F. and Dickmanns, E., "Autonomous landing of airplaces by dynamics machine vision," *Machine Vision and Applications*, vol. 7, no. 2, pp. 127–134, 1994.

[65] Schrijver, A., *Theory of linear and integer programming.* John Wiley & Sons Inc, 1998.

[66] Shakernia, O., Vidal, R., Sharp, C., Ma, Y., and Sastry, S., "Multiple view motion estimation and control for landing an unmanned aerial vehicle," (Washington, DC), Proc. IEE Int. Conf. Robotics and Automation, May 2002.

[67] Sharp, C., Shakernia, A., and Sastry, S., "A vision system for landing unmanned vehicles," (Seoul, Korea), Proc, IEEE Int. Conf. Robotics and Automation, May 2001.

[68] Smith, J., Elfes, A., Hua, H., Mrozinski, J., Shelton, K., Lincoln, W., Adumitroaie, V., and Weisbin, C., "Mobility productivity impacts on selection of lunar exploration architectures," in *Aerospace conference, 2009 IEEE*, pp. 1–9, IEEE.

[69] Some, R., Weisbin, C., Neff, J., and Witz, J., "Xml based tools for assessing potential impact of advanced technology," in *Aerospace Conference, 2004. Proceedings. 2004 IEEE*, vol. 1, IEEE, 2004.

[70] Steve Creech, P. S., "Ares v. refining a new heavy lift capability."

[71] Stuart, J., *Observational constraints on the number, albedos, sizes, and impact hazards of the near-Earth asteroids.* PhD thesis, Massachusetts Institute of Technology, 2003.

[72] Tanigawa, H., Tanji, Y., Taketomi, Y., and Kubota, T., "Deep image hologram recorded by using a lenticular lens sheet," (San Jose, California), IS&R/SPIE Conference on Practical Holography XIII, January 1999.

[73] to Review Near-Earth Object Surveys, C. and Strategies, H. M., *Near-Earth Object Surveys and Hazard Mitigation Strategies: Interim Report.* Natl Academy Pr, 2009.

[74] Todd, M., "The many facets of linear programming," *Mathematical Programming*, vol. 91, no. 3, pp. 417–436, 2002.

[75] Tournier, P., Valenti, M., How, J., and Feron, E., "Estimation and control of a quadrotor vehicle using monocular vision and moire patterns," *America Institute of Aeronautics.*

[76] Vazirani, V., *Approximation algorithms.* Springer Verlag, 2001.

[77] Webb, D. and Smith, E., "Constellation program mission operations project office, status and support philosophy," in *Aerospace Conference, 2008 IEEE*, pp. 1–7, IEEE.

[78] Weisbin, C., Mrozinski, J., Hua, H., Shelton, K., Smith, J., Elfes, A., Lincoln, W., Adumitroaie, V., and Silberg, R., "Human-robot lunar exploration: Pressurized vs. unpressurized rovers," in *Systems Engineering, 2008. ICSENG'08. 19th International Conference on*, pp. 8–12, IEEE, 2008.

[79] Weisbin, C., "Rover and telerobotics technology program," *NASA*, no. 19980201211, 1998.

[80] Weisbin, C. and Lavery, D., "Nasa rover and telerobotics technology program," *Robotics & Automation Magazine, IEEE*, vol. 1, no. 4, pp. 14–21, 1994.

[81] Weisbin, C., Lincoln, W., Mrozinski, J., Hua, H., Merida, S., Shelton, K., Adumitroaie, V., Derleth, J., and Silberg, R., "Technology assessment in support of the presidential vision for space exploration," 2006.

[82] Weisbin, C., Moeller, R., Zimmerman, W., Smythe, W., Van Houten, T., Lincoln, W., Smith, J., Elfes, A., and Adumitroaie, V., "Analysis of concepts for large-scale robotic lunar precursor missions," in *System of Systems Engineering, 2007. SoSE'07. IEEE International Conference on*, pp. 1–6, IEEE.

[83] Weisbin, C. and Rodriguez, G., "Nasa robotics research for planetary surface exploration," *Robotics & Automation Magazine, IEEE*, vol. 7, no. 4, pp. 25–34, 2000.

[84] Weisbin, C., Rodriguez, G., Elfes, A., and Smith, J., "Toward a systematic approach for selection of nasa technology portfolios," *Systems engineering*, vol. 7, no. 4, pp. 285–302, 2004.

[85] Weisbin, C., Smith, J., Van Houten, T., Moeller, R., Zimmerman, W., Smythe, W., Lincoln, W., Elfes, A., Adumitroaie, V., and Silberg, R., "Technical feasibility and relative productivity of alternate nasa robotic missions to a lunar dark crater," *Systems Journal, IEEE*, vol. 2, no. 1, pp. 120–128, 2008.

[86] Yu, O., *Technology portfolio planning and management: practical concepts and tools.* Springer Verlag, 2006.

VITA

Christianna Taylor was born and raised in Chicago, IL. She went to Boston University to receive her Bachelors in Aerospace Engineering before coming to the Georgia Institute of Technology to receive her Masters and PhD.